GEOINFO 2016

XVII Brazilian Symposium on GeoInformatics

November 27th to 30th, Campos do Jordão, SP, Brazil

Proceedings

Claudio Campelo and Laercio Namikawa (Eds.)

Dados Internacionais de Catalogação na Publicação

SI57a Simpósio Brasileiro de Geoinformática (11.: 2016: Campos do Jordão, SP)

Anais do 17º. Simpósio Brasileiro de Geoinformática, Campos do Jordão, SP, 27 de novembro a 30 de novembro de 2016. / editado por Claudio Campelo (UFCG), Laercio Namikawa (INPE) – São José dos Campos, SP: MCTIC/INPE, 2016.
Pendrive + On-line
ISSN 2179-4820

1. Geoinformação. 2.Bancos de dados espaciais. 3.Análise Espacial. 4. Sistemas de Informação Geográfica (SIG). 5.Dados espaço-temporais. I. Campelo, C. II.Namikawa, L.M. III. Título.

CDU:681.3.06

Preface

The Brazilian Symposium on GeoInformatics (GEOINFO) is the main Brazilian meeting where researchers in geographic information science and related fields come together to present their latest research and exchange ideas. In 2016, we are organizing the 17th edition of GEOINFO and expect it to continue the tradition of fostering new research ideas through the close proximity provided by our already traditional venue in Campos do Jordão.

We also continue our tradition of bringing keynote speakers that are among the world's best in geographic information science. This year speakers are: Daniel G. Brown, a PhD. in Geography with the School of Natural Resources and Environment at the University of Michigan, USA, who is working to link ecological an social processes; and Prof. Dr. Alexander Zipf from Heidelberg University, Germany, who is working with non-traditional ways of gathering geographic information, such as Volunteered Geographic Information (VGI) and Crowdsourcing.

The number of paper submissions for this year edition reached 67 with most of them of very high quality, which were throughly reviewed by the members of the Program Committee. Unfortunately, only 26 of them could be fitted into GEOINFO proceedings. The effort by the Program Committee is always welcome and without it GEOINFO would not be possible.

In addition, GEOINFO is always a great pleasure to organize, specially thanks to the GEOINFO steering committee and the organizing committee, which have answers to all the questions. Special thanks goes to Thales Sehn Korting who has brought his fresh experience as last year general chair, and Lúbia Vinhas for the support not only as the Head of Image Processing Division at INPE, but also to the proceedings.

GEOINFO supporters of this year edition are the Society of Latin American Remote Sensing Specialists (SELPER), the Coordination for the Improvement of Higher Education Personnel (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES), the São Paulo Research Foundation (Fundação de Amparo a Pesquisa do Estado de São Paulo - FAPESP), and Boeing Research & Technology.

We also thank our institutions, the Systems and Computing Department (DSC) from the Federal University of Campina Grande (UFCG) and the Image Processing Division (DPI) from the Brazilian Institute for Space Research (INPE).

Special thanks goes to Daniela Seki and Janete da Cunha from INPE for the valuable roles in the organization.

Campina Grande and São José dos Campos, Brazil, November, 2016.

Claudio Campelo	Laercio Namikawa
Program Chair	**General Chair**

Conference Committee

General Chair

Laércio M. Namikawa
National Institute for Space Research, INPE

Program Chair

Cláudio Campelo
Federal University of Campina Grande, UFCG

Local Organization

Daniela Seki
National Institute for Space Research, INPE

Janete da Cunha
National Institute for Space Research, INPE

Support

SELPER - Sociedade Latino Americana de Especialistas em Sensoriamento Remoto
CAPES - Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
FAPESP - Fundação de Amparo a Pesquisa do Estado de São Paulo
BOEING - Boeing Research & Technology

Program Committee

Alessandra Raffaeta, Università Ca'Foscari Venezia, Italy
Andrea Iabrudi Tavares, Cadence Design Systems, Brazil
Antônio Miguel V. Monteiro, INPE, Brazil
Armanda Rodrigues, NOVA LINCS, Portugal
Carla Macario, Embrapa, Brazil
Carlos Felgueiras, INPE, Brazil
Carolina Pinho, UFABC, Brazil
Chiara Renso, ISTI/CNR, Italy
Cláudio Baptista, UFCG, Brazil
Claudio Silvestri, Università Ca'Foscari Venezia, Italy
Clodoveu A. Davis, UFMG, Brazil
Dário Augusto Borges Oliveira, GE Global Research, Brazil
Daniel Zanotta, IFRS, Brazil
Dieter Pfoser, George Mason University, USA
Edzer Pebesma, University of Munster, Germany
Fabiano Morelli, INPE, Brazil
Fernando Bação, UNL, Portugal
Flávia Feitosa, UFABC, Brazil
Frederico Fonseca, The Pennsylvania State University, USA
Gilberto Câmara, INPE, Brazil
Gilberto Ribeiro de Queiroz, INPE, Brazil
Gilson Alexandre Ostwald Pedro da Costa, PUC-RJ, Brazil
Helen Couclelis, University of California, USA
Holger Schwarz, University of Stuttgart, Germany
Jorge Campos, UNIFACS, Brazil
José Antonio Macêdo, UFC, Brazil
João Paulo Papa, UNESP, Brazil
Jugurta Lisboa Filho, UFV, Brazil
Julio D'Alge, INPE, Brazil
Jussara O. Ortiz, INPE, Brazil
Karine R. Ferreira, INPE, Brazil
Lúbia Vinhas, INPE, Brazil
Luciana Alvim Santos Romani, Embrapa, Brazil
Luis Otavio Alvares, UFSC, Brazil
Marcelino P. S. Silva, UERN, Brazil
Marcus Vinicius A. Andrade, UFV, Brazil
Maria Isabel S. Escada, INPE, Brazil
Mário J. Gaspar da Silva, Universidade de Lisboa, Portugal
Pedro R. Andrade, INPE, Brazil
Raul Q. Feitosa, PUC-RJ, Brazil
Ricardo R. Ciferri, UFSCAR, Brazil
Rodrigo da Silva Ferreira, IBM Research Brazil - Natural Resources Analytics, Brazil
Rogério Galante Negri, UNESP, Brazil
Sergio Costa, UFMA, Brazil
Sergio D. Faria, UFMG, Brazil
Sergio Rosim, INPE, Brazil
Silvana Amaral, INPE, Brazil
Stephan Winter, University of Melbourne, Australia
Thales Sehn Körting, INPE, Brazil
Thomas Kemper, JRC, Italy
Valéria C. Times, UFPE, Brazil
Vania Bogorny, UFSC, Brazil
W. Randolph Franklin, Rensselaer Polytechnic Institute, USA

Contents

DBCells – an open and global multi-scale linked cells

Sérgio Souza Costa[1], **Evaldinolia Gilbertoni Moreira**[2],
Micael Lopes da Silva[1], **Thamyla Maria de Sousa Lima**[1]

[1]Curso de Engenharia da Computação – Universidade Federal do Maranhão (UFMA)
São Luís – MA – Brazil

[2]Departamento Acadêmico de Informática – Instituto Federal do Maranhão (IFMA)
São Luís – MA – Brazil

sergio.costa@ufma.br, evaldinolia@ifma.edu.br

micaelopes32@gmail.com, thamyla.sl@gmail.com

Abstract. *The land change models require large amounts of data, are difficult to be reproduced, as well as to be reused. Some initiatives to open and link data increase the reproducibility of scientific experiments and data reuse. One pillar of the linked data concept is the use of Uniform Resource Identifier (URI). In this paper, we propose DBCells – an architecture for publication of a global cellular space where each cell has a URI. This new approach will allow comparison, reproduction and the reuse of models and data. However, in order to succeed, this proposal requires participation, partnerships and investments. Our main purpose in this paper is to present the architecture, benefits and challenges for debating with the scientific community.*

1. Introduction

The reproducibility is a crucial characteristic for experimental science and requires access to data and tools [Molloy 2011]. Furthermore, the comparison and reuse of data and results play critical roles. The achievement of these requirements is a great challenge in experiments that demand large volumes of data, like land change models. These models demand data from environmental, social, technological, and political drivers [Moran et al. 2005, Turner et al. 2007]. In general, each driver is represented as a value into spatial unit, pixel or cell. A pixel is the smallest addressable element in the raster layer that represents a spatial variable, like slope or distance to roads. The cell space is an alternative representation, where each cell handles one or more types of attribute [Câmara et al. 2008]. In both cases, cells and pixels are not treated as unique and distinct entities, but as partitions of a continuous space. Then, even the smallest differences in the bounding box of the study area can generate different cell spaces. These differences make the comparison and reuse of data a great challenge. In this paper, we propose that each cell from each resolution is a unique and distinct entity that has a universal identifier, what we call DBCells architecture.

The Uniform Resource Identifier (URI) is one of the pillars of the web data architecture, which links data instead of pages. The architecture proposed by Tim Berners-Lee is referred to as linked data [Berners-Lee 2006] and provides support for large datasets, such as DBpedia [Auer et al. 2007] and GeoNames [Wick and Vatant 2012]. The DBpedia describes all the concepts from wikipedia through URI, for example, the National Institute for Space Research is identifiable by `http://dbpedia.org/data/`

`National_Institute_for_Space_Research.rdf`. This institute is located in São José Dos Campos, and is identified in the GeoNames by the following url: `http://sws.geonames.org/6322578/about.rdf`.

Several authors have argued that linked data allows experiments to become more reproducible, which depends on large volumes of data [Kauppinen and De Espindola 2011, Molloy 2011]. In addition, some authors argue that linked data can be explored to share large volumes of data among the scientific community [Quoca et al. 2014, Baučić and Medak 2014]. In [Quoca et al. 2014] the authors describe how NOAA dataset can be transformed and published as linked data. The data from 20.000 weather sensor stations over the world were converted to 177 billion triples. Other example of linked open data is the Linked Brazilian Amazon Rainforest Data [Kauppinen et al. 2014]. This dataset is openly available for anyone as non-commercial research use. However, in this dataset each variable (land use, demography, environmental, accessibility to markets technology) is strongly coupled to the cells. In our architecture proposal, described in Section 3, the cells are distinct entities that have an universal identifier, which can be linked from other data. In other words, each cell is a spatial unit that can link results and data from land change models. This paper is organized as follows: Section 2 presents the two major concepts – the open linked data and cellular-space; Section 3 describes DBCells – the architecture proposed; Section 4 summarizes the main benefits and challenges to achieve the link between the models in global scale.

2. Theoretical foundation

2.1. Open linked data

First of all, it is necessary to distinguish data, linked data and open data, shown in Figure 1. Data are the base of the pyramid, and are defined as symbols that represent properties of objects, events and their environment [Ackoff 1989]. Open data are all those that can be freely used, modified, and shared by anyone for any purpose [The Open Definition 2013]. The linked data refers to a set of best practices for publishing and interlinking structured data on the Web [Heath and Bizer 2011].

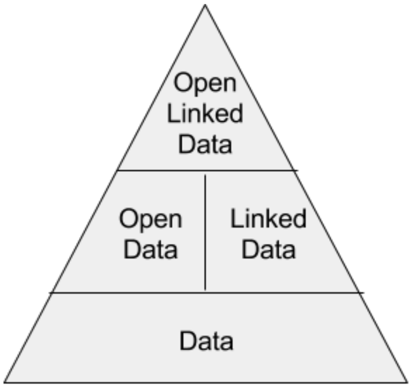


Figure 1. From data to open linked data

The movement of open data is inspired by the open source and consists of three major concepts: openness, participation and collaboration [Chignard 2013]. These concepts are present in the following three key features: (a) Availability and access – the data must be available as a whole and in a way that does not create complicated processes for the interested party in copying it; (b) Reuse and Redistribution – the data must be provided under terms that permit reuse and redistribution, including combining this data with other datasets; (c) Universal Participation – everyone must be able to use, reuse and redistribute; there should be no discrimination against fields of endeavour or against persons or groups [Dietrich et al. 2009]. The open data movement can bring democratic gain, like better transparency of public action, citizen participation and response to the crisis of confidence towards politicians and institutions [Chignard 2013, Janssen et al. 2012]. However, authors point out some prerequisites: the availability on the web and the machine readable. In other words, they must follow the three laws proposed in [Eaves 2009], which are:

1. If the data cannot be spidered or indexed, it does not exist;
2. If the data is not available in open and machine readable format, it cannot engage;
3. If a legal framework does not allow it to be repurposed, it does not empower.

Being readable for the machines is also one of the characteristics required for the linked data. However, in the linked data concept, it is necessary to link and allow it to be linked by other datasets, which is summarized in the following principles [Berners-Lee 2006]:

1. Use URIs as names for things;
2. Use HTTP URIs, so that people can look up those names;
3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL);
4. Include links to other URIs, so that they can discover more things.

In [Berners-Lee 2006], the author describes the datasets in terms of five-stars. Each rating represents a progressive transition from data to Linked Data. Every data is available on the web (at any format), but the ones with an open license have 1 star. In addition to that, if the data is available as machine-readable structured data (e.g., Microsoft Excel instead of a scanned image of a table) then it has 2 stars. To have 3 stars, the data needs to be available at a non-proprietary format (e.g., CSV instead of Excel). The next star requires the data to be available according to the previous constraints, plus the use of open standards from the W3C (RDF and SPARQL), in order to identify things, so that people can link to it. Finally, to have 5 stars, data needs to be available according to all the above criteria, plus to provide context via outgoing links to other people's data. It is important to emphasize that the opening is not a prerequisite for linked data. For example, a private company can link their data, but does not necessarily make them open. Figure 2 shows the linking open data cloud in 2014. The DBpedia [Auer et al. 2007] and GeoNames [Wick and Vatant 2012] datasets are located in the center.

The open and linked data is an important element to open science [Murray-Rust 2013, Kauppinen and De Espindola 2011]. In [Kauppinen and De Espindola 2011], the authors propose the Linked Open Science aiming to be a standardized and generic recipe for executable papers. This concept was built on these four key elements: (a) Linked Data, (b) OpenSource and Web-based Environments, (c) Cloud Computing and (d) Creative Commons. An example of linked

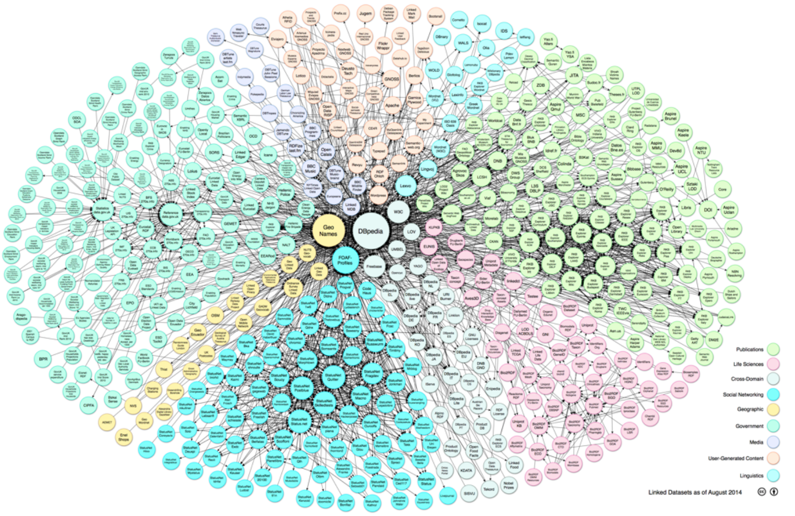


Figure 2. Linking open data cloud. Source: [Cyganiak and Jentzsch 2014]

open data is the Linked Brazilian Amazon Rainforest Data [Kauppinen et al. 2014]. This dataset is openly available for anyone that will make non-commercial research use of it. The data was produced by the Institute for Geoinformatics, University of Muenster, Germany and the National Institute for Space Research (INPE) in Brazil. However, in this dataset, each variable (land use, demography, environmental, accessibility to markets technology) is strongly coupled to the cells. In our proposal, described in the Section 3, the cells are distinct entities that have a universal identifier, which can be linked from other datasets.

2.2. From geo-fields to cellular space

Our focus is on data from land change models. In general, these models describe phenomena that vary continuously in space and time, as deforestation in the Brazilian Amazon region. Their input and output are represented as geo-fields. Together with geo-objects, the geo-fields are the two fundamental spatial representations [Kuhn 2012, Câmara 2005]. Geo-objects describe entities that have an identity as well as spatial, temporal, and thematic properties [Kuhn 2012]. However, geo-fields have been shown to be more fundamental than geo-objects and are capable of integrating both representations [Liu et al. 2008, Camara et al. 2014, Costa et al. 2007]. As data structure, the geo-fields are discretized and used two ways [Kuhn 2012]:

1. through a finite number of cells, within each one the attribute is assumed to remain constant;
2. through a finite set of sample points with interpolation rules for positions among them.

In this paper, we are interested in the first way, where the study area is partitioned forming a regular grid of square, triangular, hexagonal, or cubic cells as in raster based layers or a cellular space. The raster model can be compared with a bitmap image, which consists of a number of pixels organized in rows and columns. Basically, in most cases, raster data is indeed derived from satellite images, which serve as a basis for observing weather, vegetation or electromagnetic radiation. A cellular space is an alternative model to represent geo-fields. It is a spatial data type where each cell handles one or more types of attribute [Câmara et al. 2008]. In [Câmara et al. 2008],the authors argue that cellular spaces were part of the early GIS implementations, but now it is time to reconsider this decision and reintroduce it as a basic data type; they also argue that the usage of one-attribute raster data in the storage of results for dynamical models requires the storage of information in different files. By the other hand, a cellular space stores all attributes of a cell together, with significant benefits for modeling, in contrast to the more cumbersome single value raster approach. Together with the concept of Generalized Proximity Matrix (GPM), it is possible to represent hierarchical and network relations [de Aguiar et al. 2003] and [Moreira et al. 2008]. In [Moreira et al. 2008], the authors use these concepts to represent hierarchical and network spatial relations in multi-scale land change models.

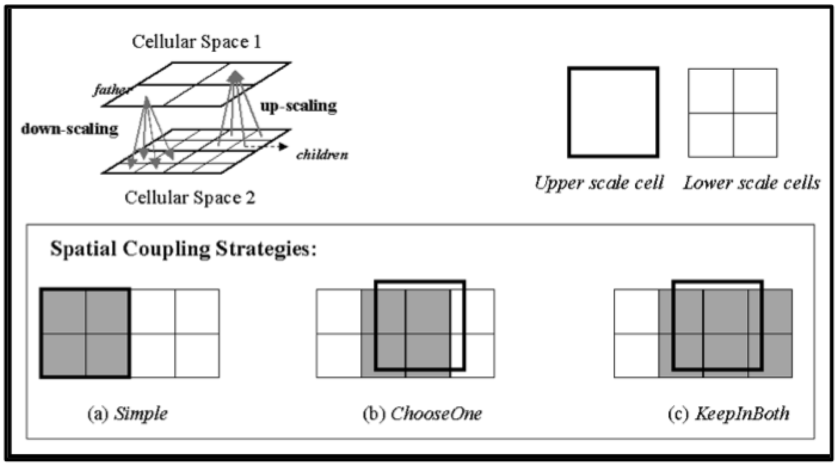


Figure 3. Representation of strategies for spatial coupling in the case of regular cells. Source: [Moreira et al. 2008]

Cellular spaces have been used for simulation of urban and environmental models as part of cellular automata models (Batty 2000). In TerraLib ([Câmara et al. 2008]) and TerraME ([de Senna Carneiro et al. 2013]) the cellular space is a native building block. These concepts and tools have supported the development of models published in the literature [Aguiar et al. 2007, Moreira et al. 2009, Aguiar et al. 2012, Espindola et al. 2012, Andrade et al. 2009].

3. The architecture

We propose a layered architecture (in five layers), adapted from [Heath and Bizer 2011], as showed in Figure 4. The publication layer includes a dataset of cellular space and multiscale relationships. The web of data layer links the cellular space to existing datasets, like Geonames, DBPedia and SWEET Ontology. The data access and storage layer integrate local and web data, providing a transparent access and storage for modeling tools. The model layer uses and shares data provided by the lower layers. For example, coarser scales run models of climate, and finer scales run environmental and social models. The user layer runs and reproduces the experiment of a particular model. A user can publish the results and the data of an experiment in the web of data, allowing replication of experiments, an essential characteristic of science.

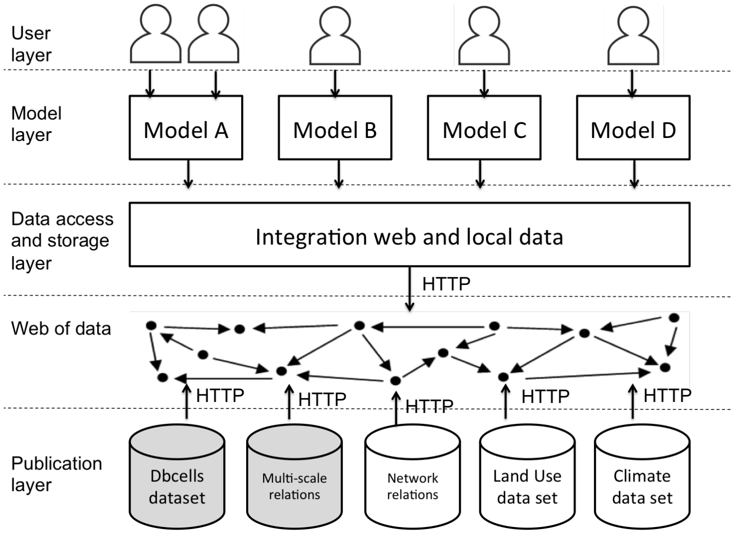


Figure 4. The DBCells Architecture Proposal

In the dataset of cellular space, each cell is identified by a URI and described as a RDF graph, see Figure 5. The RDF (Resource Description Framework) is the data model, standardized by W3C for representing Semantic Web resources. It expresses information as graphs consisting of triples with subject, property and object [Klyne and Carroll 2006]. These three graph elements are identifiable through URI. In the dataset of cellular space, each graph consists of minimal set of properties to describe a cell, such as its position and bounding box. These graphs can be stored in a graph database, like the Neo4J[1], and serialized as RDF/XML, see Code 1.

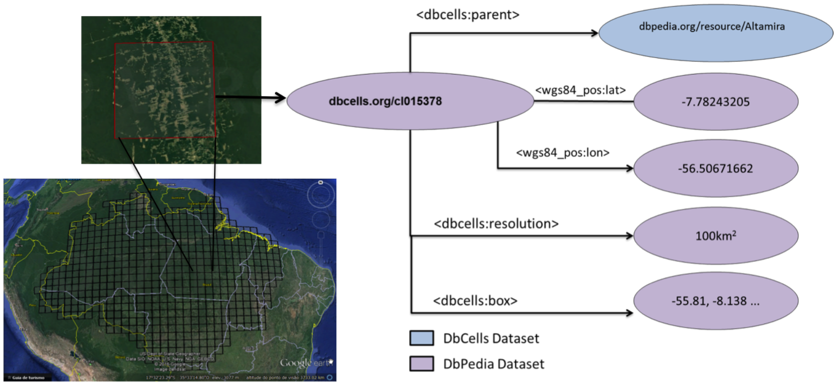


Figure 5. A specific cell as a RDF graph

[1] https://neo4j.com/

```

...
<rdf:Description rdf:about=”http://dbcells.org/cl015378”>
        <wgs84_pos:lat rdf:datatype=”http://www.w3.org/2001/
            XMLSchema#double”>−7.782432055255575</wgs84_pos:lat>
        <wgs84_pos:long rdf:datatype=”http://www.w3.org/2001/
            XMLSchema#double”>−56.50671662245421</wgs84_pos:long
            >
        <parent xmlns=”http://dbcells.org/”>
                http://dbpedia.org/resource/Altamira</parent>
        <resolution xmlns=”http://dbcells.org/”>100000</
            resolution>
        <dbcells:box>
                −55.81526040160506,−8.138132254243271,0
                    −54.90595645484754,−8.138133321360099,0
                    −54.90595545105769,−7.2313407607591,0
                    −55.81525938511868,−7.23133981183386,0
                    −55.81526040160506,−8.138132254243271,0
        </dbcells:box>
</rdf:Description>
...
```

Code 1. A specific cell graph serialized as a RDF/XML

Our proposal is to describe both, the cells and their relationships, through RDF graphs. Graphs express different relations, including: (a) topological relations; (b) network connectivity, both physical (e.g., transportation infrastructure) and logical (e.g., trade fluxes); (c) vicinity in cell spaces and grids; (d) coupling between spatial scales [Moreira et al. 2008]. We propose to describe the relationships in different datasets, allowing a model to select one or more relationships. The Figure 6 shows an example where the relationships describes a spatial coupling between cellular spaces in different resolutions. In this case, the nodes are cells and the edges represent the hierarchical relations. Similarly, these relationships can be stored in the graph database and serialized as RDF / XML.

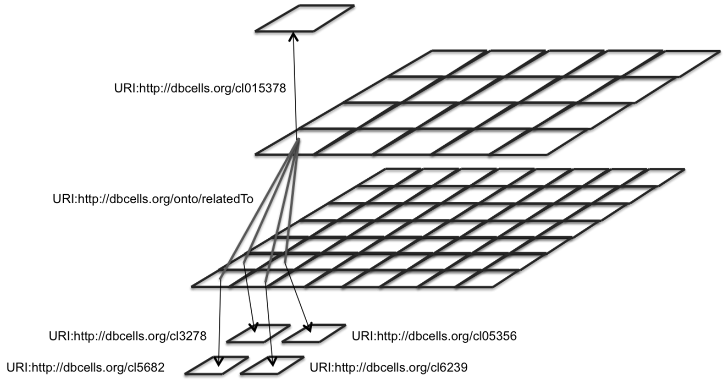

URI:http://dbcells.org/cl015378
URI:http://dbcells.org/onto/relatedTo
URI:http://dbcells.org/cl3278
URI:http://dbcells.org/cl05356
URI:http://dbcells.org/cl5682
URI:http://dbcells.org/cl6239

Figure 6. Relationships between multi-scale cellular spaces as a graph

The DBCells architecture is under development, and will require partnerships and investments. This article is intended to present and validate this proposal together with the scientific community, as its success will depend on the interest of this community. In the next section we present some benefits and challenges to complete this project.

4. Benefits and challenges

The implementation of this proposal brings several benefits and challenges. Similar to DBPedia and GeoNames, the DBCells may be a dataset that will link datasets from different spatial models. Since each cell has a universal identifier, the models can link their data and results to it. This will allow sharing data and results, and the reuse of datasets already published, illustrated in Figure 7.

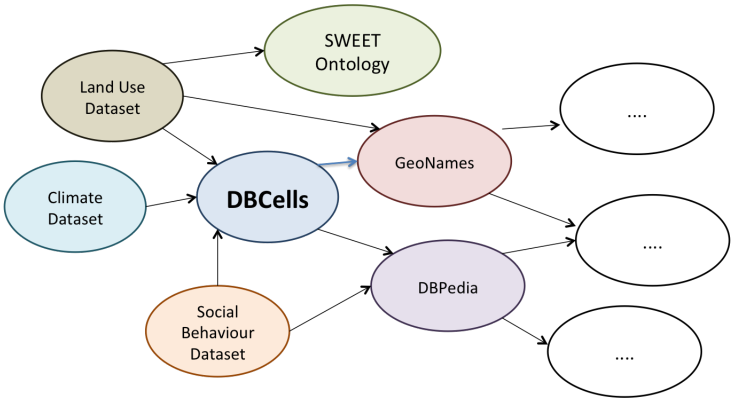


Figure 7. Integration between datasets

The open data is crucial for reproducibility of data demanding experiments [Murray-Rust 2013, Kauppinen and De Espindola 2011, Molloy 2011].

According to [Molloy 2011]:

> *"The more data is made openly available in a useful manner, the greater the level of transparency and reproducibility and hence the more efficient the scientific process becomes, to the benefit of society"*.

The relationships between cellular spaces allow the reuse of models at different scales and resolutions. For example, a land use model at a finer scale can use results of a climate model in a coarser scale, which is represented in Figure 8.

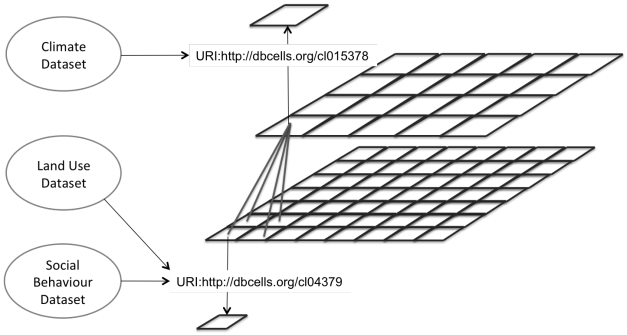

Climate Dataset
URI:http://dbcells.org/cl015378
Land Use Dataset
Social Behaviour Dataset
URI:http://dbcells.org/cl04379

Figure 8. Reuse of model data and results between cellular spaces

The proposed architecture will also contribute to a better reproducibility and comparison of the data demanding experiments. This benefit is enhanced through open source environment tools [Kauppinen and De Espindola 2011]. This architecture presents also several challenges, for example, the participation of the scientific community. The benefits previously mentioned will depend on the community interest in making their data open and linked. Furthermore, each modeling tool will need to implement the data access and storage layer to retrieve and store data on the web. Another challenge is the conflict between vocabularies from different models. Therefore, it will be necessary to use the already established vocabularies, whenever possible. At last, an efficient and distributed computing will be necessary for storage and retrieval of data from a global cellular space at different scales. For that reason, we will conduct the initial experiments in areas of greatest interest by the scientific community like Amazon rainforest.

5. Final remarks

This paper introduced an innovative architecture – DBCells – that integrates two concepts: cellular spaces and linked data. The pillar of integration is to treat each cell as a unique and distinct entity that has a universal identifier. To achieve this integration, we propose four steps: 1) divide the space in regular cells, 2) associate each cell to an identifier, 3) represent each cell as an RDF graph available on the web and 4) connect data and results models to these identifiers. The main benefits of the new approach are the reuse, sharing, comparison and reproduction of land change models. The main challenges are the participation and interest of the scientific community, and an efficient architecture to store and retrieve large volume of data. Thus, the success of this proposal requires partnerships and investments. Based on that, by presenting our vision, we expect to raise an engaging debate with the scientific community.

References

Ackoff, R. L. (1989). From data to wisdom. *Journal of applied systems analysis*, 16(1).

Aguiar, A. P. D., Câmara, G., and Escada, M. I. S. (2007). Spatial statistical analysis of land-use determinants in the brazilian amazonia: Exploring intra-regional heterogeneity. *Ecological modelling*, 209(2):169–188.

Aguiar, A. P. D., Ometto, J. P., Nobre, C., Lapola, D. M., Almeida, C., Vieira, I. C., Soares, J. V., Alvala, R., Saatchi, S., Valeriano, D., et al. (2012). Modeling the spatial and temporal heterogeneity of deforestation-driven carbon emissions: the inpe-em framework applied to the brazilian amazon. *Global Change Biology*, 18(11):3346–3366.

Andrade, P. R., Monteiro, A. M. V., Câmara, G., and Sandri, S. (2009). Games on cellular spaces: How mobility affects equilibrium. *Journal of Artificial Societies and Social Simulation*, 12(1):5.

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Baučić, M. and Medak, D. (2014). Building the semantic web for earth observations. In *DailyMeteo. org/2014 Conference*.

Berners-Lee, T. (2006). Linked data.

Camara, G., Egenhofer, M. J., Ferreira, K., Andrade, P., Queiroz, G., Sanchez, A., Jones, J., and Vinhas, L. (2014). Fields as a Generic Data Type for Big Spatial Data. *Geographic Information Science*, page in press.

Câmara, G., Vinhas, L., Ferreira, K. R., De Queiroz, G. R., De Souza, R. C. M., Monteiro, A. M. V., De Carvalho, M. T., Casanova, M. A., and De Freitas, U. M. (2008). Terralib: An open source gis library for large-scale environmental and socio-economic applications. In *Open source approaches in spatial data handling*, pages 247–270. Springer.

Chignard, S. (2013). A brief history of open data. *Paris Tech Review*, 29.

Costa, S. S., Câmara, G., and Palomo, D. (2007). Terrahs: integration of functional programming and spatial databases for gis application development. In *Advances in Geoinformatics*, pages 127–149. Springer.

Cyganiak, R. and Jentzsch, A. (2014). Linking open data cloud diagram. *LOD Community (http://lod-cloud. net/)*, 12.

Câmara, G. (2005). Representação computacional de dados geográficos. *CASANOVA, MA et al. Banco de dados geográficos. Curitiba: Mundogeo*, pages 11–52.

de Aguiar, A. P. D., Câmara, G., Monteiro, A. M. V., and de Souza, R. C. M. (2003). Modelling spatial relations by generalized proximity matrices. In *GeoInfo*.

de Senna Carneiro, T. G., de Andrade, P. R., Câmara, G., Monteiro, A. M. V., and Pereira, R. R. (2013). An extensible toolbox for modeling nature–society interactions. *Environmental Modelling & Software*, 46:104–117.

Dietrich, D., Gray, J., McNamara, T., Poikola, A., Pollock, P., Tait, J., and Zijlstra, T. (2009). Open data handbook.

Eaves, D. (2009). The three laws of open government data.

Espindola, G. M., De Aguiar, A. P. D., Pebesma, E., Câmara, G., and Fonseca, L. (2012). Agricultural land use dynamics in the brazilian amazon based on remote sensing and census data. *Applied Geography*, 32(2):240–252.

Heath, T. and Bizer, C. (2011). Linked data: Evolving the web into a global data space. *Synthesis lectures on the semantic web: theory and technology*, 1(1):1–136.

Janssen, M., Charalabidis, Y., and Zuiderwijk, A. (2012). Benefits, adoption barriers and myths of open data and open government. *Information Systems Management*, 29(4):258–268.

Kauppinen, T. and De Espindola, G. M. (2011). Linked open science-communicating, sharing and evaluating data, methods and results for executable papers. *Procedia Computer Science*, 4:726–731.

Kauppinen, T., De Espindola, G. M., Jones, J., S??nchez, A., Gr??ler, B., and Bartoschek, T. (2014). Linked Brazilian Amazon Rainforest Data. *Semantic Web*, 5(2):151–155.

Klyne, G. and Carroll, J. J. (2006). Resource description framework (rdf): Concepts and abstract syntax.

Kuhn, W. (2012). Core concepts of spatial information for transdisciplinary research. *International Journal of Geographical Information Science*, 26(12):2267–2276.

Liu, Y., Goodchild, M. F., Guo, Q., Tian, Y., and Wu, L. (2008). Towards a general field model and its order in gis. *International Journal of Geographical Information Science*, 22(6):623–643.

Molloy, J. C. (2011). The open knowledge foundation: open data means better science. *PLoS Biol*, 9(12):e1001195.

Moran, E., Ojima, D., Buchmann, N., et al. (2005). Global land project: Science plan and implementation strategy. *IGBP Report*, 33.

Moreira, E., Costa, S., Aguiar, A. P., Câmara, G., and Carneiro, T. (2009). Dynamical coupling of multiscale land change models. *Landscape Ecology*, 24(9):1183–1194.

Moreira, E., de Aguiar, A. P. D., Costa, S. S., and Câmara, G. (2008). Spatial relations across scales in land change models. In *GeoInfo*, pages 95–108.

Murray-Rust, P. (2013). Open Data in Science. *Serials Review*, 34(1):52–64.

Quoca, H. N. M., Quoca, H. N., Hauswirtha, M., and Le Phuoca, D. (2014). Global weather sensor dataset.

The Open Definition (2013). Open definition. Acessed in `http://opendefinition.org/od/2.0/pt-br/`.

Turner, B. L., Lambin, E. F., and Reenberg, A. (2007). The emergence of land change science for global environmental change and sustainability. *Proceedings of the National Academy of Sciences*, 104(52):20666–20671.

Wick, M. and Vatant, B. (2012). The geonames geographical database. *Available from World Wide Web: http://geonames. org*.

Comparison of supervised classification methods of Maximum Likelihood image, Minimum Distance, Parallelepiped and Neural network in images of Unmanned Air Vehicle (UAV) in Viçosa-MG

Daniel Camilo de Oliveira Duarte[1], Juliette Zanetti[1], Joel Gripp Junior[1], Nilcilene das Graças Medeiros[1]

[1]Civil Engineering Department– Federal University of Viçosa (UFV) - Viçosa, MG-Brazil

daniel.duarte@ufv.br,juliette.zanetti@ufv.br,jgripp@ufv.br, nilcilene.medeiros@ufv.br

Abstract: *The aim of this work was testing the classification techniques in digital air images of spatial high resolution obtained by Unmanned Air Vehicle (UAV). The images recover an area of the Federal University of Viçosa, campus Viçosa-MG. From the orthophoto generated, the classification test was made, by using four classifiers: Parallelepiped, Average Minimum Distance, Maximum Likelihood and Artificial Neural Networks. The classification that best delimited the different features present in the image was the classification by Artificial Neural Networks. In order to prove statistically the classification efficiency, the validation was carried out through Kappa index and visual analysis.*

1. Introduction

Nowadays, there are many methods for digital images treatment of remote sensing which allow to carry out tasks of manipulation, analysis and images comprehension. In the images processing of remote sensing, the target nature is determined based on the fact that different materials are characterized by interacting in different ways in each band of electromagnetic spectrum (JENSEN, 2009).

The use of Air Vehicles Unmanned (AVUs) is the study field that has grew fast in the technologies of remote sensing, by offering an option of low cost that allows to measure and monitor aspects of the environment with the possibility of the images acquisition (HONKAVAARA, et al., 2013). Air images with spatial and time high resolution contribute for obtainment of field information, characterization of the problem and even thematic maps generation of elevated detail. Lian e Chen (2011) worked with guided classification to object in satellite images of spatial high resolution and concluded that the precision of classification is directly related to spatial resolution.

According to Queiroz et al. (2004), the information contained in the images can be extracted through the classification process. There are various classification methods which search through sundry approaches identify with accuracy the information that each image pixel, by classifying it in category. The methods of image classification can present different levels of accuracy, depending on the approach used by the method and the specification of its parameters.

Thereby, the aim of this study was evaluating and comparing the performance of four classifiers: Parallelepiped (PPD), Minimum Distance Average (MDA), Maximum Likelihood (ML) and Artificial Neural Network (ANN) to determine the use and cover map with the classes: Forest, Water, Urbanization, Agriculture, Exposed Soil, in UAV images, in Viçosa, in order to verify which method offers best results through the validation by the Kappa Index.

2. Images classification

The methods of classification can be divided in classifiers per pixel or per regions and can take into account one or more bands of images. The classifiers per pixels use the spectral of each pixel isolated to find homogenous regions, defined as classes. Classifiers per regions are based on information of a group of neighboring pixels (INPE, 2014). In accordance with Santos (2003), the method of ADM assigns each unknown pixel to class whose average is next to it. Each pixel inside and out of the areas of training is evaluated and marked to class which it is more likely to belong to (Figure 1–A). The method PPD defines square areas, by using units of standard deviation or minimum and maximum reflectance values into each training area, according to Figure 1-B.

However, the MAXVER method, according is the most used in remote sensing within the statistical approach (JENSEN, 2005). This method suits ellipses, so that the location, shape and ellipse size reflect the average variance and covariance of two variables. The distribution of reflectance values is described by a probability function that evaluates the possibility of a given pixel belongs to a category and classifies the pixel to a category which it is more likely to associate, (SANTOS, 2013) as shown in the Figure 1c.

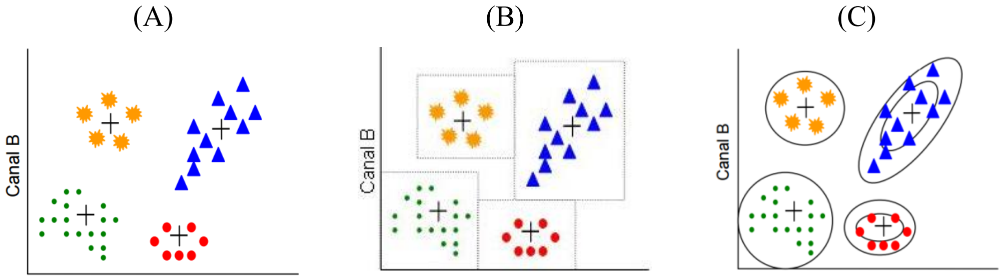


Figure 1: A - Rating method scheme supervised of Minimum Distance to the Average, B - Scheme of classification method supervised of parallelepiped and C - Scheme of classification method supervised of Maximum Likelihood.
Source: Santos (2013).

The RNA networks are also used in images processing. In Queiroz et al.'s (2004) opinion, as RNA are algorithms whose operation is based on structure of the human brain because it acquires and keeps knowledge through the learning process, which happens through neurons connections, what is also known as synapses. According to Abdalla and Sá Volotão (2013), there are neural networks of simple layer which consist of a group of neurons arranged in a single layer only and multilayer networks, formed by numerous hidden layers or the combination of several networks of simple layers.

To Belinda et al (2013), in a neural network the input layer X_i is one in which patterns are presented to the network. The intermediate layers (also called occult or hidden) which can be more than one, are responsible for most of the processing. At this stage the input data is multiplied by the weight W_{ji} and it is also added polarization $\theta_j$to adjust the residual error. An activation or transference function f calculates output Y_j of the neuron, by using a predefined logic. The output of the transfer function goes to other neurons or environment through the output layer. The operation of a typical neuron in a network is shown in Figure 2, can be written mathematically by equation 1 and 2:

$$I_j = W_{ji} X_i + \theta_j \tag{1}$$

$$Y_i = f\left(I_j\right) = f\left(W_{ji} X_i + \theta_j\right) \tag{2}$$

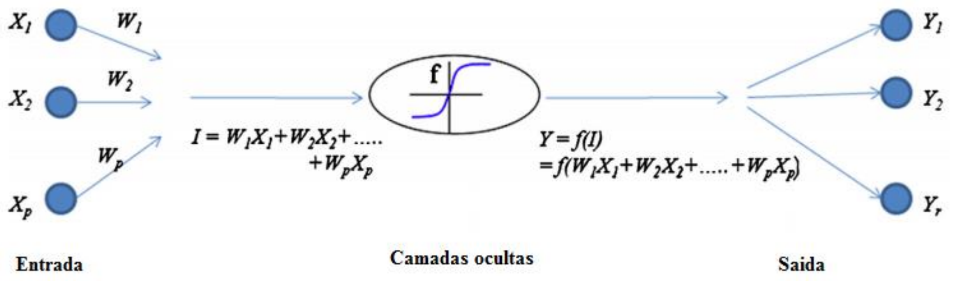


Figure 2: Architecture of a RNA with two hidden layers.
Source: Mazhar et al (2013).

The neurons number of the first layer corresponds to the dimensionality of input attributes vector. The output layer will have as many neurons as there are classes to be separated. The biggest problem is in the definition of hidden layers number, and the neurons number that composes them. In practical this problem has been generally solved by attempt and error, or by previous experience in domain of a given situation (GALO, 2000).

3. Classification rating

The classification rating can be determined by the *Kappa* index method, calculated based on an error matrix and by using as measuring of agreement between the map and the reference adopted for the estimative of the accuracy, in this case, the orthophoto. The equation 3 calculates the *Kappa* coefficient (COHEN, 1960):

$$K = \frac{N\Sigma X_{ii} - \Sigma(X_{i+} \times X_{+i})}{N^2 - \Sigma(X_{i+} \times X_{+i})} \quad (3)$$

Considering that:

K = *Kappa* coefficient of agreement;

N= Number of observations (sample points);

X_{ii} = Observation in the line i and column i;

X_{i+}= Total marginal of the line i;

X_{+i}= Total marginal of the column i;

The results of the *Kappa* index calculated for each test of classification can be understood according to Mangabeira et al. (2003) (Table 1).

Kappa index (%)	Estimative quality
80 a 100	Excellent
60 a 80	Very good
40 a 60	Good
20 a 40	Reasonable
0 a 20	Bad
<0	Very bad

Table 1: Table for *Kappa index interpretation* .
Source : Mangabeira et al. (2003).

4. Experiments and Results

The study area is part of the Federal University of Viçosa (UFV), Viçosa campus, Minas Gerais. The UFV landscape has classes variety of use and soil occupation such as: forest remnants; experimental fields of agriculture and bare soil; buildings and patio area with different characteristics; water bodies such as rivers and lakes, among others. Due to these characteristics classes of forest, agriculture, bare soil, urbanization and water were chosen for the experiment.

In this study the equipment UAV Echar 20A manufactured by XMobots (2015) was used, coupled with Sony ILCE camera - 7R, 36.4 MP full-frame CMOS sensor Exmor®. The photos processing was carried out in PhotoScan Professional Edition 1.0.2 of Agisoft software. Points Control and validation were collected by using the Global Navigation Satellite System (GNSS), Javad TRIUMPH 1 receptor with application of the method Real Time Kinematic (RTK) for georeferencing to the Brazilian Geodetic System.

It also used the Geographic Information System (GIS) IDRISI version 17 - *Jungle*, developed by Clark University. The software was chosen because of various processing tools and analysis of digital images.

5. Methodology

In order to facilitate the methodology comprehension applied, a flow diagram of the activities performed is presented in the Figure 3.

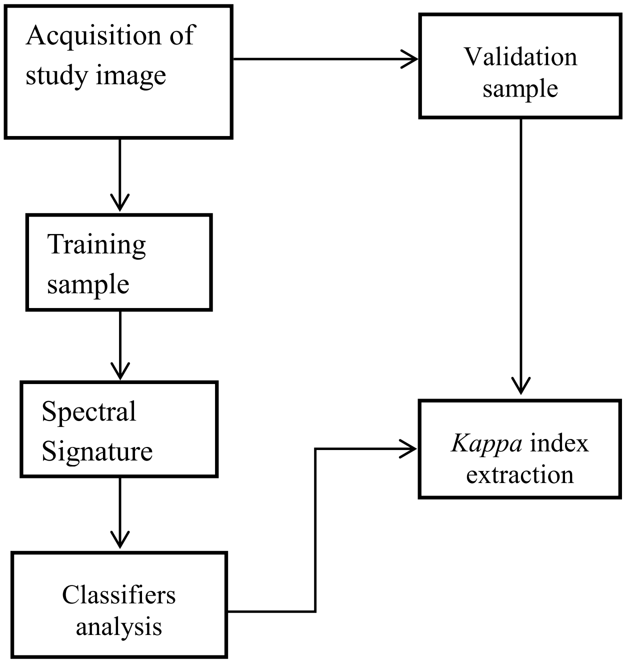


Figure 3: Flow diagram of the methodology proposed.

The UAV images were obtained on 10/08/2015 at 11:41. Possessing the UAV and control points, it was generated orthophoto with spatial resolution of 0.5 cm. In order to minimize the computational effort of the classifiers, the orthophoto was cut within the limits of the study area.

Then the class samples of Forest, Water, Urbanization, Agriculture and exposed soil were collected, divided into two groups: training and validation. On average, there were 61,995 training pixels and 12,810 validation pixels for each class. It is important to highlight that the sample size of each class was a control factor in the experiment enabling that the classifiers analysis were independent of the sample size of each class.

The spectral signatures of classes were extracted with the training samples and then the classifiers analysis PPD MDM MAXVER and RNA were performed. Possessing the four maps generated by using classifiers along with the validation sample it was extracted Kappa index of each classifier tested thus to carry out the analysis. Qualitative analysis was also conducted, based on the visual analysis, where the results of the classified images with the original image were compared, aiming to verify if the identification of classes was consistent with reality.

6. Results

It was observed that the four classifiers rated, RNA and MAXVER demonstrated the best performance, with Kappa index of 93% and 87%, respectively, according to the Figure 4. The Figure 5 illustrates the classifiers maps used.

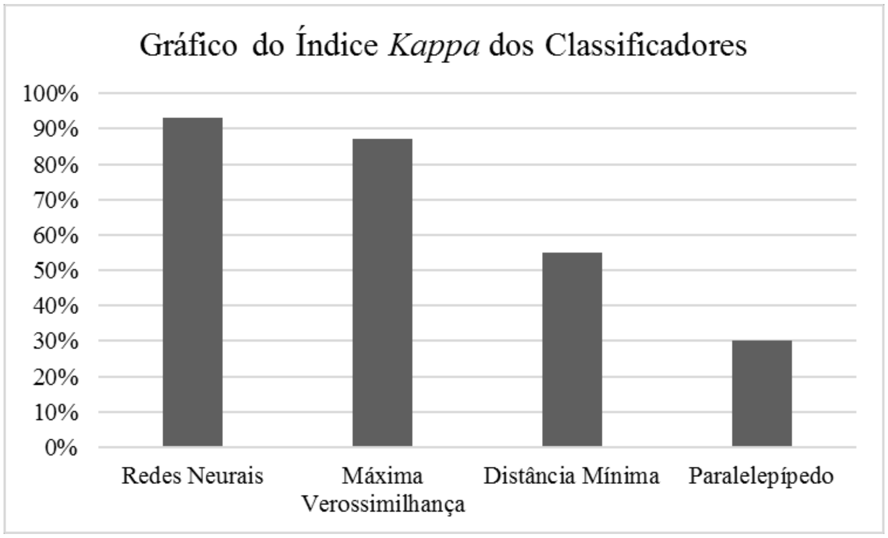


Figure 4: Graphic of Kappa index of the dos classifiers tested.

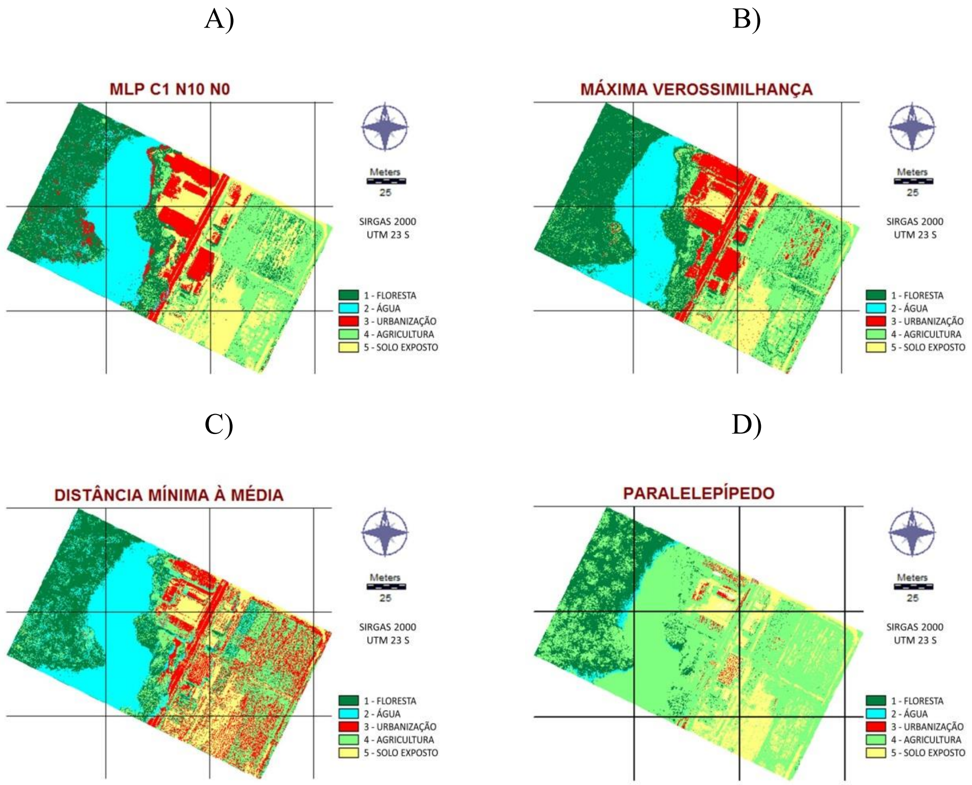


Figure 5: Map with best classification.

It can be observed that the parallelepiped method had the worst result (Figure 5d), since it is more appropriate for images that have classes with well-defined shapes, different from the image used, obtained by UAV, which has a low definition of class boundaries. In accordance with Crósta (1992) one of the problems is that an image that contains thousands of pixels most likely fell out of the decision limits of classes, no matter classes to define. Nevertheless, the method result of the minimum distance is due to the fact that interest classes were worked in the image that was better suited to the method algorithm, with good spectral similarity, which facilitated the classification.

Evaluating visually the images generated by the classifiers, it is observed that the method of neural networks and maximum likelihood were more compatible with reality, by distinguishing better the targets. To obtain the image classified as RNA, the parameters were changed, and by trial and error, an acceptable result was obtained.

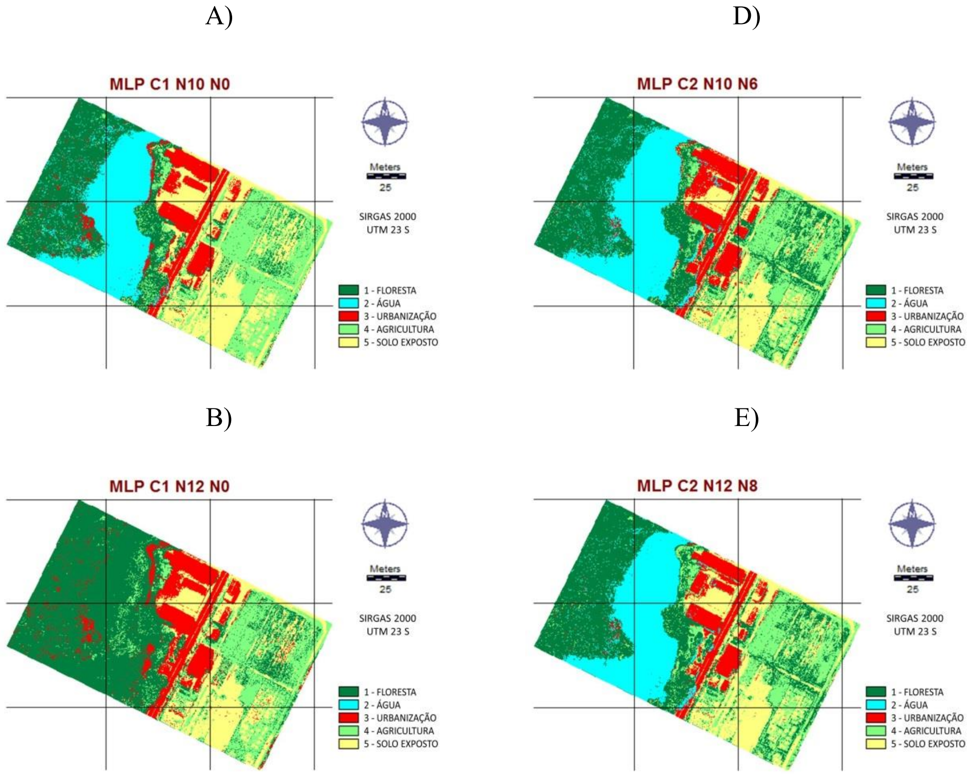
The settings used for test in an attempt to obtain the network that best classifies the image are shown in Table 2.

Table 2. Setting used for the tests performance.

HIDDEN LAYERS	NODES 1	NODES 2	RMS TRAINING	RMS TESTING	*KAPPA*
1	10	0	0,2	0,2	89%
1	12	0	0,3	0,3	66%
1	14	0	0,2	0,2	91%
2	10	6	0,1	0,1	93%
2	12	8	0,1	0,1	93%
2	14	10	0,1	0,1	93%

In accord with Table 2, it is stated that with a layer and increasing the knots number, the *Kappa* has random behavior. The neural network had best performance with two layers and independent when the second layer is inserted.

In the tested carried out, it was observed that, increasing the number of second layer, the network did not product significant results in the image and its *Kappa* index stayed constant. Figure 6 illustrates the maps obtained by Artificial Neural Network method.



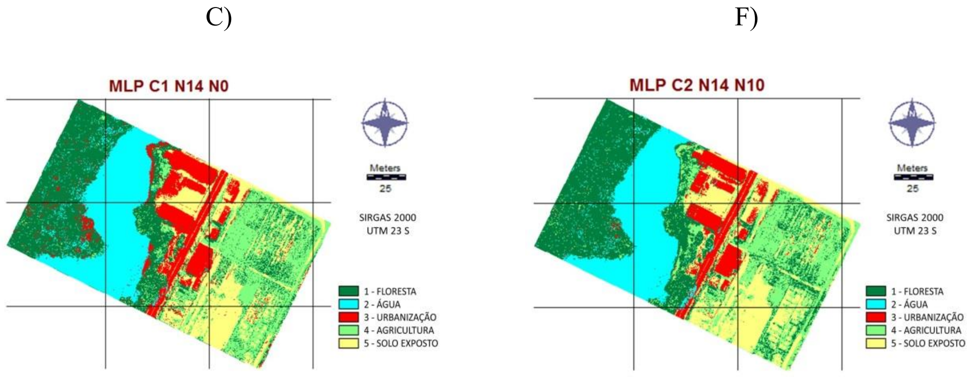


Figura 6: Mapas das classificações por Redes Neurais Artificiais.

7. Conclusion

Higher spatial resolutionimages can improve the classification, due to better identification of objects on the soil. Thus, after the work accomplishment, it was found that the use of UAV images was efficient to define targets of interest, avoiding the scan, based on photo-interpretation. It is also to note that these analyses can be made extremely quickly and dynamically, by enabling the monitoring of physical, environmental and urban social evolution, for example.

The choice of the best results in this work was based on the results of the Kappa index and visual analysis of the results generated thereby it was concluded that the use of the classification method by neural networks was more efficient than other tested methods, however the definition of the parameters and their training were long, requiring tests with modified parameters, in order to reach an acceptable result. In this context, the data generated by this research, can bring an effective contribution, once they can be considered as an alternative to systematization in the detection of classes in the image, not limited to traditional techniques.

References

ABDALLA, L.S.; SÁ VOLOTÃO, C.F. Estudo da configuração de diferentes arquiteturas de redes neurais artificiais MLP para classificação de imagens ópticas, Anais XVI Simpósio Brasileiro de Sensoriamento Remoto - SBSR, Foz do Iguaçu, PR, Brasil, p. 8200-8207, 2013.

COHEN, J.A. Coefficient of agreement for nominal scales. Educational and Psychological Measurement, 20:37-46, 1960.

GALO, M. L. B. T., Aplicação de redes Neurais Artificiais e Sensoriamento Remoto na Caracterização Ambiental do Parque Estadual Morro do Diabo. Tese de Doutorado, Escola de Engenharia de São Carlos da Universidade de São Paulo, São Carlos, SP, 2000.

HONKAVAARA, E.; SAARI H.; KAIVOSOJA, J.; POLONEN, I; HAKALA T.; LITKEY, P; MAKYNEN J.; PESONEN L.. Processing and assessment of

Spectrometric, Stereoscopic Imagery Collected Using a Lightweight UAV Spectral Camera for Precision Agriculture. Remote Sensing, 5:5006-5039,2013.

INPE. Instituto Nacional de Pesquisas Espaciais. Disponível em http://www.dpi.inpe.br/spring/portugues/tutorial/classific.html. Acesso em 23 de outubro de 2016.

JENSEN, J. R. Introductory Digital Image Processing: a remoting sensing perspective. 3ºedição. Universidade da Califórnia: Prentice Hall. 2005. 526 p.

JENSEN, J. R. Sensoriamento remoto do ambiente: uma perspectiva em recursos terrestres. São José dos Campos: Parêntese, 2009. 4-29 p.

LIPPMANN, R. P., An introduction to computing with neural nets. IEEE ASSP Magazine, v.4., 1987.

Lian, L. & Chen, J. 2011. Research on segmentation scale of multi-resources remote sensing data based on object-oriented. Procedia Earth and Planetary Science, 2(1):

352-357.

MANGABEIRA, J.A.C.; AZEVEDO, E.C. & LAMPARELLI, R.A.C. 2003. Avaliação do levantamento de uso das terras por imagens de satélite de alta e média resolução espacial. Campinas: Embrapa. Comunicado técnico, 11: 1-15.

MOREIRA, M. A. Fundamentos do sensoriamento remoto e metodologias de aplicação. Ed. UFV. 2ª ed. Viçosa, MG, 2003.

SANTOS, R. D. B. Dinâmica espaço-temporal (1990 - 2010) do uso da terra no município de Seropédica, RJ, determinado por classificação automatizada. 2013. Trabalho de conclusão de curso de Engenharia florestal - Universidade Federal Rural do Rio de Janeiro (UFRRJ).

RIBEIRO R. J. C., et al. Comparação dos métodos de classificação supervisionada de imagem Máxima Verossimilhança e Redes Neurais em ambiente urbano. 2007

QUEIROZ, R. B.; RODRIGUES, A. G.; GÓMEZ, A. T. Estudo comparativo entre as técnicas máxima verossimilhança gaussiana e redes neurais na classificação de imagens IR-MSS CBERS 1", WORKCOMP 2004, 2004.

Fast Spatially Coupled Bayesian Linearized Acoustic Seismic Inversion in Time Domain

Fernando Bordignon[1], **Leandro Figueiredo**[2], **Mauro Roisenberg**[1], **Bruno B. Rodrigues** [3]

[1]Instituto de Informática e Estatística
Universidade Federal de Santa Catarina (UFSC)

[2]Departamento de Física
Universidade Federal de Santa Catarina (UFSC)

[3]CENPES/PDGP/CMR
Petróleo Brasileiro SA - PETROBRAS

{bordi,mauro}@inf.ufsc.br

Abstract. *Bayesian methods for seismic inversion assume multi-Gaussian distributions for the variables involved and can apply a linearization to the forward model to offer a mathematically tractable solution and uncertainty analysis. Results of the maximum a posteriori are robust, produced fast and have a high synthetic to real seismic correlation. The drawbacks of this inversion method are related to the lack of spatial continuity patterns, when the solution is computed trace-by-trace. This work proposes a methodology to tackle the scalability problem of the Bayesian linearized inversion when adding spatial continuity in the time domain. A sliding window and a manipulation of the equations are used to solve the problem locally and avoid boundary effects.*

1. Introduction

Acoustic seismic inversion aims to infer the acoustic impedance property of the Earth's subsurface via recorded seismic reflection data. It plays a key role in hydrocarbons reservoir modeling and characterization, and it is an ill-posed, nonlinear inverse problem [Tarantola 2005]. Despite its nonlinearity and ambiguousness, methodologies that linearizes the problem were proposed in the last decades [Buland and Omre 2003, Figueiredo et al. 2014]. The methodology results that are comparable to commercial software and is broadly used in the industry nowadays [Figueiredo et al. 2014, Grana and Della Rossa 2010].

The linearized Bayesian acoustic inversion uses a framework based on the Bayes rule and assumes log-normal distributions for the acoustic impedance and normal distribution for the error term of the seismic data. The solution is computed fast in the time domain if done trace-by-trace, because one can reuse calculations when the area of interest have stationary covariances. Performing the inversion trace-by-trace limits the spatial coupling of the results because no horizontal correlation is imposed, inheriting the continuity solely from the seismic data [Buland et al. 2003].

To impose lateral continuity into the solution, it is necessary to define a model covariance matrix that is $n \times n$ being n the number of cells in the area of interest. This leads to an exponential complexity on n, i.e. $O(n^3)$, which makes the process impractical due to the amount of memory needed to process big volumes of seismic data. To overcome this issue, [Buland et al. 2003] proposes an inversion in the frequency domain, lowering the complexity to $O(n \log n)$.

This paper proposes a moving window technique to account only for the neighboring seismic traces while imposing lateral continuity, overcoming the border effects by using a sliding window that inverts only the central trace at a time. The algorithm execution time is highly dependent on the window size but it is of linear complexity on the number of inversion cells, i.e. $O(n)$. Two applications examples are shown comparing the trace-by-trace inversion with the proposed method. The run time of the proposal is acceptable, exhibiting up to a tenfold increase in run time compared to the trace-by-trace method for the studied cases, in addition to the algorithm being easy to implement.

2. Bayesian Linearized Inversion

In the discrete domain, the seismic data $\boldsymbol{d}$ is given from its relation to the reflectivity $\boldsymbol{r}$ by:

$$\boldsymbol{d} = \boldsymbol{S}\boldsymbol{r} + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{S}$ is the convolutional matrix constructed with a known wavelet. The relation between the reflectivity and the impedance z is given by:

$$r(t) = \frac{z(t+\delta t) - z(t)}{z(t+\delta t) + z(t)} \tag{2}$$

With the reflectivity smaller than 0.3, the relation above can be approximated by [Stolt and Weglein 1985]:

$$r(t) = \frac{1}{2}\Delta \ln(z(t)) \tag{3}$$

If we consider the model vector to be log-normal, e.g. $\boldsymbol{m} = \ln(\boldsymbol{z})$, and the differential matrix $\boldsymbol{D}$, the relationship between the seismic and the acoustic impedance is given by the linear operator $\boldsymbol{G} = \boldsymbol{\frac{1}{2}}\boldsymbol{S}\boldsymbol{D}$ plus a white noise error term $\boldsymbol{e}$ as follows:

$$\boldsymbol{d} = \boldsymbol{G}\boldsymbol{m} + \boldsymbol{e} \tag{4}$$

The Gaussian likelihoods between the seismic data and the model parameters are written as in the following equations:

$$p(\boldsymbol{d}|\boldsymbol{\mu_d}, \boldsymbol{\Sigma_d}) = N(\boldsymbol{\mu_d}, \boldsymbol{\Sigma_d}), \tag{5}$$

$$p(\boldsymbol{m}|\boldsymbol{\mu_m}, \boldsymbol{\Sigma_m}) = N(\boldsymbol{\mu_m}, \boldsymbol{\Sigma_m}), \tag{6}$$

where $\boldsymbol{\mu_d} = \boldsymbol{G}\boldsymbol{m}$, $\boldsymbol{\Sigma_d}$ the covariance matrix of the seismic, $\boldsymbol{\mu_m}$ the low frequency model (LFM) and $\boldsymbol{\Sigma_m}$ the covariance matrix for the model parameters, e.g. defined by an normal decaying correlation neighborhood as in [Figueiredo et al. 2014]:

$$\boldsymbol{\nu}_{t,t'} = \sigma_m^2 exp\left(-\frac{(t-t')^2)}{L^2}\right), \tag{7}$$

where L is the desired correlation distance and $\boldsymbol{\nu}_{t,t'}$ is then multiplied by the variance to obtain the covariance.

Making use of this Bayesian linearized framework, the posterior distribution is written according to:

$$p(m|d, s, \mu_m, \sigma_d^2, \sigma_m^2) \propto p(d|s, m, \sigma_d^2)p(m|\mu_m, \sigma_m^2) \tag{8}$$

where s is the wavelet and σ_d^2 and σ_m^2 are seismic and model variances respectively.

The posterior mean and covariance matrix are given by [Figueiredo et al. 2014]:

$$\boldsymbol{\mu}_{m|} = \boldsymbol{\mu}_m + \boldsymbol{\Sigma}_m \boldsymbol{G}^T(\boldsymbol{G}\boldsymbol{\Sigma}_m\boldsymbol{G}^T + \Sigma_d)^{-1}(\boldsymbol{d}_o - \boldsymbol{G}\boldsymbol{\mu}_m), \tag{9}$$

$$\boldsymbol{\Sigma}_{m|} = \boldsymbol{\Sigma}_m - \boldsymbol{\Sigma}_m\boldsymbol{G}^T(\boldsymbol{G}\boldsymbol{\Sigma}_m\boldsymbol{G}^T + \Sigma_d)^{-1}\boldsymbol{G}\boldsymbol{\Sigma}_m. \tag{10}$$

where the mean is also referred to as maximum a posteriori (MAP), as the mean of a multivariate normal distribution is the most likely values of the vector, or the vector values with the highest likelihood.

The methodology presented is to invert a 3D seismic cube in a trace-by-trace manner. In this case, nothing but vertical correlations are imposed to the results, allowing horizontal continuity to be derived from seismic data alone. When performing the trace-by-trace inversion, one can reuse the computation of the matrix inversion in Equation (9) for an area with similar covariances, reducing the computational cost to a matrix inversion, sized according to the number of vertical samples, and matrix-vector products for each trace.

It is possible to invert all the 3D region of interest at once, defining a model covariance matrix $\boldsymbol{\Sigma}_m$ that imposes correlations in every desired direction. A drawback of this approach is the quadratic growth of the covariance matrices, as the number of cells to be inverted grows, which then propagates its size to the matrix inversion in Equations (9) and (10) yielding an exponential computational complexity, i.e. $O(n^3)$.

Results of the trace-by-trace method are reported to be robust and can be considered a smooth representation of the subsurface properties of interest. The procedure is easy to implement and faster compared to commercial available software whereas producing similar impedance models [Figueiredo et al. 2014]. Although the methodology does not have the practical ability to model complex spatial continuities, it is suitable for an expeditious inversion used for interpretation or simpler purposes.

3. Fast Bayesian Linearized Inversion with Spatial Coupling

The Bayesian inversion methodology presented is the solution to a linear problem in the form $\boldsymbol{x} = \boldsymbol{A}\boldsymbol{y} + \boldsymbol{b}$, where $\boldsymbol{x}$ is the acoustic impedance, $\boldsymbol{A} = \boldsymbol{\Sigma}_m\boldsymbol{G}^T(\boldsymbol{G}\boldsymbol{\Sigma}_m\boldsymbol{G}^T + \Sigma_d)^{-1}$, $\boldsymbol{y} = (\boldsymbol{d}_o - \boldsymbol{G}\boldsymbol{\mu}_m)$ and $\boldsymbol{b} = \boldsymbol{\mu}_m$.

Adding spatial coupling, as the problem is linearized, does not change the form of the solution. The spatial coupling is added via a linear function of the $\boldsymbol{y}$ vector. In other words, $\boldsymbol{A}$ gets bigger and more complicated but it is still a linear operator over $\boldsymbol{y}$. As the area of interest grows, $\boldsymbol{A}$ becomes even more sparse. This effect is due to the correlation length being limited to the nearest neighbours, i.e. the solution of the problem only depends on the samples that are in a close horizontal vicinity, having very little relation to distant samples.

Using the Equation 7 to define horizontal correlation, the sparsity of the matrix A is notable for $L = 2$. At the fourth horizontal index away from the diagonal, i.e. $t - t' = 4$, the covariance is already 50 times smaller than the diagonal. Therefore, it is reasonable to truncate the horizontal covariance at two times the L distance in this case.

To implement a linear complexity inversion methodology we explore the linearity of the solution and use a truncation of the lateral continuity model. Assume the goal is to invert a 3D grid with coordinates $G \in \mathbb{Z}^3$ where $|G| = n = ijk$, being i the number of cells in x direction, j the number of cells in y direction and k the number of cells in time t. Assume a contiguous window $W \subset G$ where $|W| = w^2k$ being w the size of the window both in x and y directions with $w < i$ and $w < j$.

Following, the problem is fully defined as we would invert only the traces inside the 3D window W, i.e. we define the covariance matrices and stack the seismic traces and LFM traces in its respective vectors $\boldsymbol{d}_o$ and $\boldsymbol{\mu}_m$. Accordingly, the covariance matrices will have size $w^2k \times w^2k$. Next, a new covariance matrix $\boldsymbol{\Sigma}'_m$ is defined using only the lines of $\boldsymbol{\Sigma}_m$ correspondent to the central trace of W, i.e. only the lines relative to the coordinates $(\lceil \frac{w}{2} \rceil, \lceil \frac{w}{2} \rceil, t) \forall t \in k$ are selected from $\boldsymbol{\Sigma}_m$ to compose $\boldsymbol{\Sigma}'_m$.

Finally a operator is computed with a modified Equation 9 as follows:

$$\boldsymbol{O} = \boldsymbol{\Sigma}'_m \boldsymbol{G}^T (\boldsymbol{G} \boldsymbol{\Sigma}_m \boldsymbol{G}^T + \Sigma_d)^{-1} \tag{11}$$

The matrix $\boldsymbol{O}$ has k lines and w^2k columns, which when multiplied by $(\boldsymbol{d}_o - \boldsymbol{G}\boldsymbol{\mu}_m)$, defined with all samples inside the window W, gives the impedance solution for the central trace of the window. The window is then slided to the next x or y coordinate of the 3D grid by 1 index, overlapping most of the samples with the previous window. The $\boldsymbol{O}$ operator is applied again to compute the next central impedance trace of the window, wich is 1 index far from the previous central trace. The steps are repeated for every possible window in coordinates of the 3D grid G. The central traces are saved to compose the final solution to the problem. Figure 1 shows the graphic representation of the sliding window inside the 3D grid with its central trace at a (x, y) plane.

The truncation of $\boldsymbol{O}$ can also be applied at the vertical direction so that theoretically the algorithm will have $O(n)$ complexity. But, for most cases, the size k of the 3D grid is limited to a maximum of 1000 samples. Therefore, the procedure described above is of complexity $O(n)$ if k is considered a constant, i.e. the growth of n is mainly due to the growth of i and j.

4. Application Examples

In this section two datasets were used to compare the results of the inversion with omnidirectional correlation, also called spatially coupled inversion or coupled inversion for short, against the trace-by-trace inversion.

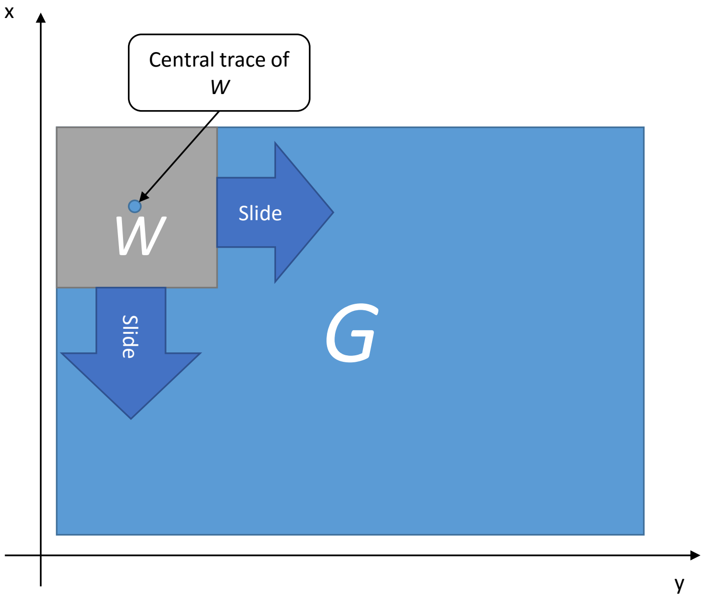


Figure 1. Graphic representation of the sliding window at the 3D grid

4.1. Synthetic Dataset

The first experiment was conducted on a synthetic dataset with $101 \times 101 \times 90$ cells. The seismic data has no noise, hence a trace-by-trace inversion of the original data is used as a gold standard for the inversion. A white noise with amplitude of 10% of the seismic data RMS value was added to the seismic data, therefore it is possible to compare the trace-by-trace inversion with the coupled inversion, evidentiating the lateral continuity imposed. Figure 2 depicts a vertical section of the trace-by-trace inversion with the original noise free seismic data.

For this case, the correlation distance L was set to 1.3 in all directions. A constant low frequency model was used with $13000 g/cm^3 m/s$ for all cell nodes. A previously extracted wavelet was provided by geophysicists. The model variance σ_m^2 from Equation 7 was set to 0.0077 which was extracted from the logarithm of the high-pass filtered impedance well logs (cutoff frequency of $8Hz$). The seismic variance σ_d^2 was set to 10% of the seismic data RMS value. The window W was defined as being of size $5 \times 5 \times 90$ for the inversion with horizontal correlation.

The trace-by-trace inversion took 2 seconds to run and yielded a synthetic to real seismic correlation coefficient of 0.99. Figure 3 shows a vertical section of the trace-by-trace inversion of the noisy seismic data. Notice the vertical stripes of noise that appear due to the lack of lateral coupling of the results.

The coupled inversion took 18 seconds to run for the entire grid, yielding a synthetic to real seismic correlation coefficient of 0.99. The impedance results are shown at Figure 4. As a result of spatial coupling, some noise is filtered and it is possible to see the improvement at the definition of the deepest layer at around the time index 80, which is now smoother and has more contrast compared to the trace-by-trace inversion.

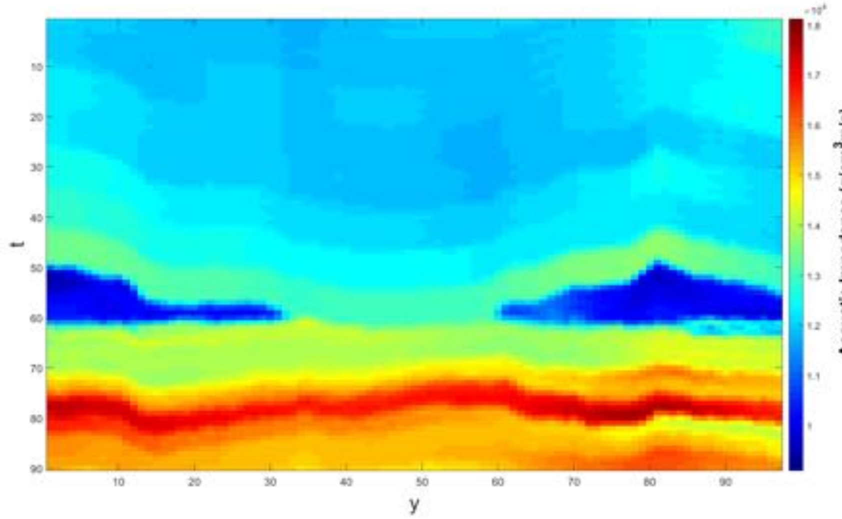


Figure 2. Impedance result from trace-by-trace inversion without noise for the synthetic dataset

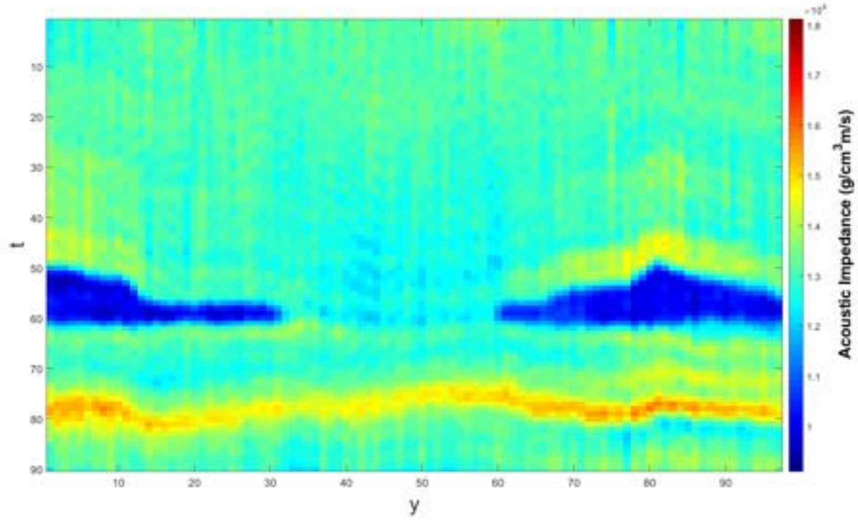


Figure 3. Impedance result from trace-by-trace inversion with noise for the synthetic dataset

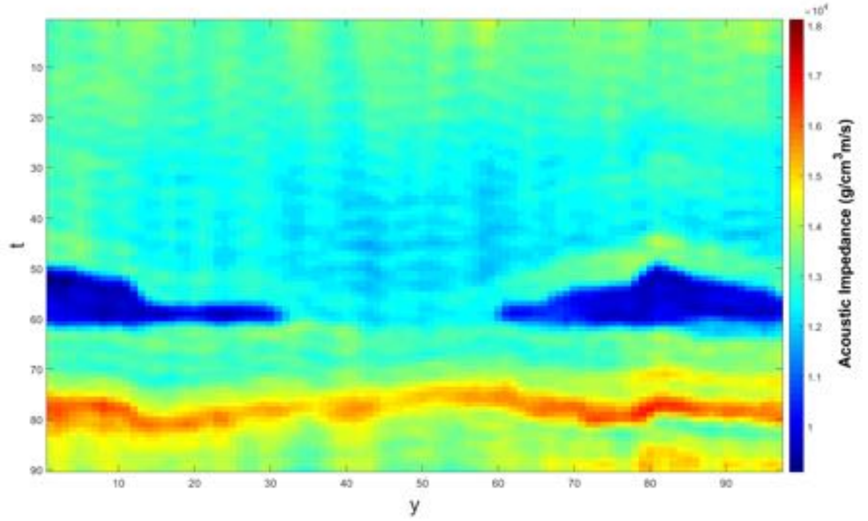


Figure 4. Impedance result from the coupled inversion for the synthetic dataset

4.2. Real case application

For this case, a real dataset was used that has a relatively high signal to noise ratio. A 2D arbitrary line that passes through the 4 wells was selected for the inversion. The section has 707 traces with 250 time samples each. The window W has size 5×250 because the inversion is 2D. The trace-by-trace inversion took 0.6 seconds to execute, while the coupled inversion 2.2 seconds. The acoustic impedance result of the trace-by-trace inversion is shown at Figure 5.

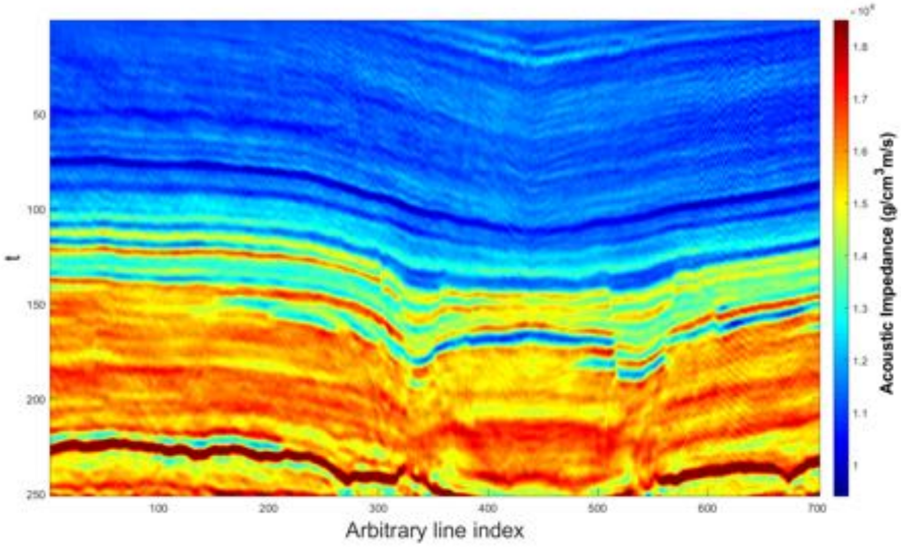


Figure 5. Impedance result from trace-by-trace inversion with noise for the real dataset

In this case it is possible to notice some noise throughout the section, specially at the upper right part. Note that the vertical continuity is present, while at the noisy areas there are abrupt discontinuities in the horizontal direction, indicating noise still left on the data.

The coupled inversion result is depicted at Figure 6. In this case the horizontal continuity is present, as it is possible to see mainly at the upper right area of the section.

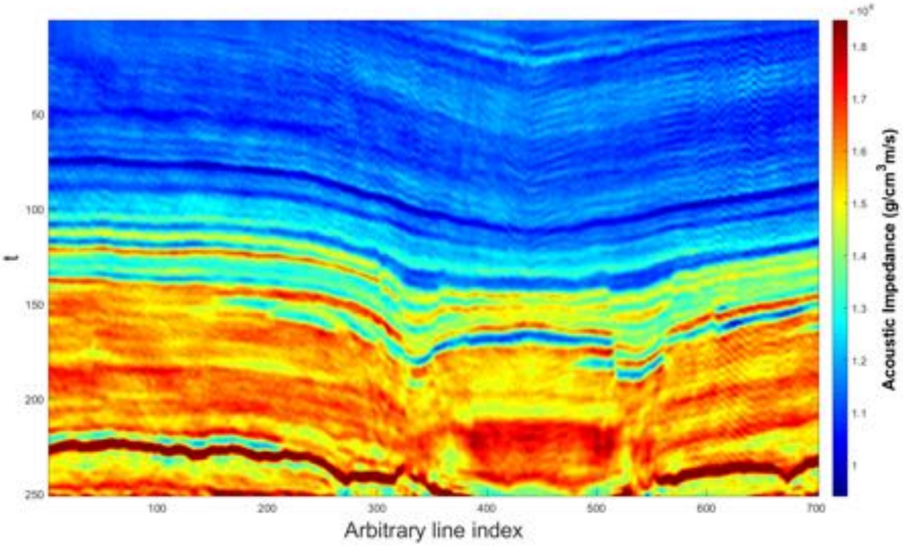


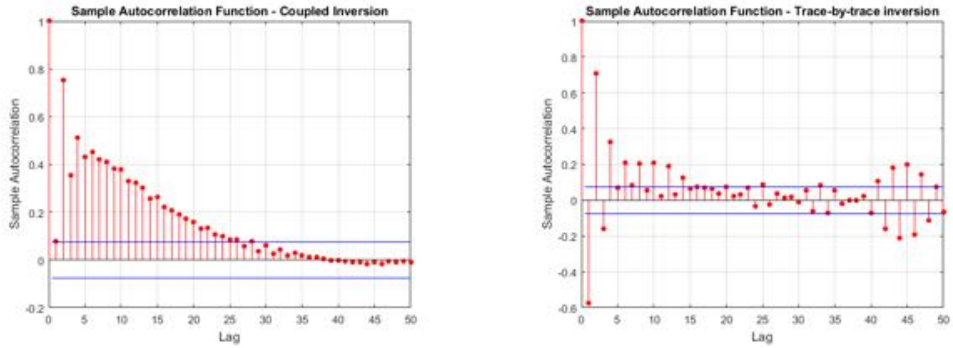
Figure 6. Impedance result from the coupled inversion for the real dataset

To better demonstrate the horizontal correlation imposed on the results, a high pass filter with cutoff frequency of $8Hz$ was applied vertically to the results and the autocorrelation function was calculated for a horizontal line in both cases. The filtering was applied to remove the low frequency provided by the low frequency model, leaving only the higher frequencies provided by the seismic data, containing noise and signal, to be examined by the autocorrelation function. Figure 7 shows both sample autocorrelation functions.

As expected, the coupled inversion has a greater correlation distance due to higher correlation at the near lags, reducing slowly until lag 25 while the trace-by-trace inversion has lower correlations and goes to near zero at lag number 15.

5. Discussion

The proposed methodology adds spatial coupling to the impedance results of the seismic inversion via a simple manipulation of the inversion operator. This manipulation was possible because of the linear nature in which the problem was casted and its solution. The assumptions are the same as in [Buland et al. 2003], which is a stationary prior model for all traces in the 3D model. The boundary effects cited by the same authors are now treated, as the sliding operator proposed in this paper explores the linearity and sparsity of the solution.



(a) Autocorrelation function for the coupled inversion

(b) Autocorrelation function for the trace-by-trace inversion

Figure 7. Autocorrelation of a horizontal 1D line at $\mathbf{t = 150}$

The main reason why the proposal works is that, even when coupling all cell nodes at once, the correlations imposed comes from a linear operator on the seismic data. Hence, the result is mathematically independent from the neighbor impedance traces, in other words, the resulting spatial coupling is the result of a linear filter applied to the seismic data.

Another positive effect observed in this proposal is the coupling of samples at greater distances than the window size w, which is evidentiated by having correlations up until lag 25 with a window size of 5, shown at Figure 7.

6. Conclusion

In this paper we presented a simple technique for spatially coupled acoustic seismic inversion in the time domain, which has linear computation time, i.e. $O(n)$ on the number of cells to be inverted. The two test cases showed the addition of spatial coupling without any boundary effects. The proposed technique has a time overhead compared to the trace-by-trace inversion that is a constant which depends on the window size. The window size is determined based on the correlation distance L, in a similar way in which is defined the search radius of kriging algorithms [Caers 2011].

The proposal can be further optimized if the vertical sparsity of the solution is explored. In the cases studied, the vertical number of cells was not enough to justify the truncation of the operator in t. Considering that the vertical dependency of samples are larger vertically due to the effect of the wavelet.

7. Acknowledgement

The authors would like to thank CAPES - Brazilian Federal Agency for Support and Evaluation of Graduate Education - scholarship process no. 4041/15-1, PETROBRAS and CERENA/Instituto Superior Técnico for supporting this work. We also thank CGG for providing an academic license for the Jason software package, used in this work for data preparation and other manipulations.

References

Buland, A., Kolbjørnsen, O., and Omre, H. (2003). Rapid spatially coupled avo inversion in the fourier domain. *Geophysics*, 68(3):824–836.

Buland, A. and Omre, H. (2003). Bayesian linearized avo inversion. *Geophysics*, 68(1):185–198.

Caers, J. (2011). *Modeling Uncertainty in the Earth Sciences*. Wiley.

Figueiredo, L. P., Santos, M., Roisenberg, M., Neto, G., and Figueiredo, W. (2014). Bayesian framework to wavelet estimation and linearized acoustic inversion. *Geoscience and Remote Sensing Letters, IEEE*, 11(12):2130–2134.

Grana, D. and Della Rossa, E. (2010). Probabilistic petrophysical-properties estimation integrating statistical rock physics with seismic inversion. *Geophysics*, 75(3):O21–O37.

Stolt, R. H. and Weglein, A. B. (1985). Migration and inversion of seismic data. *Geophysics*, 50(12):2458–2472.

Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics.

Analyzing mobility patterns from social networks and social, economic and demographic open data

Caio Libânio Melo Jerônimo, Claudio E. C. Campelo, Cláudio de Souza Baptista

Systems and Computing Department, Federal University of Campina Grande (UFCG) - Campina Grande, Pb - Brazil

caiolibanio@copin.ufcg.edu. br {campelo, baptista} @ computacao.ufcg.edu.br

***Abstract**. With the increased facility in acquiring georeferenced data from social networks, the interest in studying human mobility based on these data has grown, bringing new challenges and opportunities for knowledge discovery in GIScience. Even with this favorable scenario, few studies have attempted to analyze how the information produced from these networks may correlate with other aspects of human life. This paper presents an approach to extracting mobility patterns from Twitter messages and to analyzing their correlation with social, economic and demographic open data. The proposed model was evaluated using a dataset of georeferenced Twitter messages and a set of social indicators, both related to Greater London. The results revealed that social indicators related to employment conditions present higher correlation with the mobility patterns than any other social indicators investigated, suggesting that these social variables may be more relevant for studying mobility patterns.*

1. Introduction

Urban Mobility Patterns represent models of human behavior in an urban environment (Luo et. al., 2016) and are especially relevant as the analysis of these patterns may influence public transportation systems, public safety, traffic engineering, health systems, and many other fields related to the planning of urban centers (Noulas et al, 2012; Wilson and Bell, 2004). Mobility patterns are also present in studies related to recommendation systems (Hao et al., 2010; Ye et al., 2011; Zheng et al., 2010) as well as in researches on trajectories (Bagrow and Elin, 2012; Hsieh et al., 2012).

There are many studies addressing this subject, however, most of them are concerned with data from cell phone networks (Gonzalez et al., 2008; Jiang et al., 2013; Palchykov et al., 2014), Wifi networks (Chaintreau et al., 2007; Zhang et al., 2012) or GPS signals (Rhee et al., 2011; Zhao et al., 2015). Although these studies help understand mobility patterns, they have restrictions on privacy, as well as on precision, specially when using cell phone networks data.

The massive usage of social networks by different classes of people have lead to an increase in the amount of data that are generated in the internet, enabling the rise of new studies related to knowledge discovery. This phenomenon, allied to the usage of smartphones in daily life, and the capability of these devices to generate georeferenced data, have also favored studies related to mobility patterns, specially in urban centers.

It is known that large cities might have considerable economic, social and demographic discrepancies among their regions, which can influence the way locals move within such urban centers. Analyzing how these factors influence urban mobility is a major challenge to be considered, for example, in Points of Interest (POI) recommendation, in route prediction, or in urban planning systems.

In this paper we propose a model to extract mobility patterns from georeferenced data collected from social networks and to find statistical correlations between these patterns and social, economic and demographic open data from an urban center. Georeferenced information obtained from social media is typically imprecise and fragmented. For example, many people only post messages from certain locations (e.g from home, from work), even though they visit many other places; many users remain inactive for long periods. Thus, this research aims at verifying the feasibility of extracting mobility information from social media data that is relevant enough to be correlated with other open social data.

To evaluate our model, we collected Twitter[1] messages from Greater London for one year, totalizing 19,456,798 messages. For the social data, we used the public platform London Datastore (http://data.london.gov.uk/). These data were supplied to our model, allowing the discovery of some correlations between these informations, which indicates the practicability of using a model of this kind. Through the experiments we conducted, we found that the social variables related to employment conditions tend to correlate with the mobility patterns analyzed in this study, specially the following variables: economically inactive people, employment rate, unemployment rate and persons with no qualifications.

The remainder of this paper is organized as follows. The next section presents related work. Section 3 describes the mobility pattern properties used and some relevant related concepts. Then the experiments conducted to validate our model are addressed in Section 4. Section 5 discusses obtained results. Section 6 concludes the paper and points to future directions of this research work.

2. Related work

Most studies related to mobility patterns use data derived from cell phone networks, RFID devices, GPS based data, or Wifi networks. Recently, studies have addressed the task of extracting and identifying mobility patterns from social media data. This tendency is a consequence of the way that these networks offer their data, since this information is mostly available for public access, reducing financial costs applied to research projects.

Yuan et al. (2013) proposed a probabilistic model called W4 (Who + Where + When + What) to extract from Twitter messages aspects of mobility related to the users of this social network. The authors considered the spatial and temporal dimensions, and also the activities performed by the users.

Considering social networks as new data sources for current and future research in many different fields, some researchers have analyzed the suitability of this kind of data source. Jurdak et al. (2015) analyze mobility patterns considering spatial and temporal aspects related to the major cities of Australia, in order to demonstrate that

[1] Twitter - https://twitter.com

Twitter can be quite efficient when working with mobility patterns using georeferenced messages, where similar features were found by both Twitter and mobile phone networks data. Similarly, the research presented by Hawelka et al. (2014) only considers spatial and temporal aspects of Twitter data, however, they deal with global mobility scales (between countries). The study aims at revealing global mobility patterns related to these messages, demonstrating that these data have similar properties to other kinds of data sources used in different studies.

Hasan et al. (2013) categorize mobility patterns through user's activities around three major cities: New York, Chicago and Los Angeles. The differential in their research is that they consider, in addition to spatial and temporal aspects, the semantics of displacements. To do this, they analyze georeferenced messages from Twitter, using links to the Foursquare platform, which allows them to identify and categorize the check-ins as: (1) at home; (2) work; (3) meal; (4) entertainment activity; (5) recreation and (6) shopping.

The vast majority of researches relating mobility patterns and social network data only focus on the spatial and temporal aspects of these patterns. However, other aspects might be considered when studying mobility patterns, specially economic, social and demographic factors. In this context, the works that consider these dimensions are restricted and limited to a few variables in a social context.

Shelton et al. (2015) provide a conceptual and methodological framework to analyze inequalities in different regions of Louisville, Kentucky. The authors explore the spatial imaginaries of the citizens to divide the city into areas that they think it is more or less segregated in comparison to the rest of the city, mainly because of economic and racial/ethnic factors. They use Twitter messages to analyze how citizens travel around those divided areas and conclude that the popular imaginary could not be confirmed.

Cheng et al. (2011) investigate georeferenced Twitter messages, considering, in addition to spatial and temporal aspects, variables related to income, popularity on the social network, and the content of the messages. The authors try to find out some relations between these variables and the mobility patterns encountered in the Twitter messages. They conclude that people who live in cities with a higher average income, tend to get around for longer distances. Luo et al. (2016) investigate, in addition to spatial and temporal aspects, the following variables: ethnicity, age and gender. These social variables have been inferred from the users' profiles on Twitter and from public information provided by the government. The authors analyze how these three variables influence the mobility patterns extracted from the messages of Twitter related to the city of Chicago. They conclude that ethnicity was the most determining factor with regard to mobility patterns, possibly because this variable may express some socioeconomic characteristics of these users, demonstrating some level of segregation imposed to foreign people.

The social networks have made informations about mobility publicly available by the use of API's designed to explore the data generated everyday by these networks. Additionally, the governments are being encouraged to keep their populational data open for public access, generating a scenario where the information is easily accessible for any researcher who wants to discover new and hidden knowledge. Even with this favorable context, information related to population income, value of the properties of a region, crime rates and others are not analyzed nor correlated with mobility patterns, bringing the necessity of models that are able to perform such analysis, which could help in understanding the cities dynamics, guiding further studies related to urban centers and how people behave in these centers.

3. Mobility patterns and social analysis

Figure 1 shows an overview of our approach to detecting correlations between mobility patterns and social, economic and demographic data.

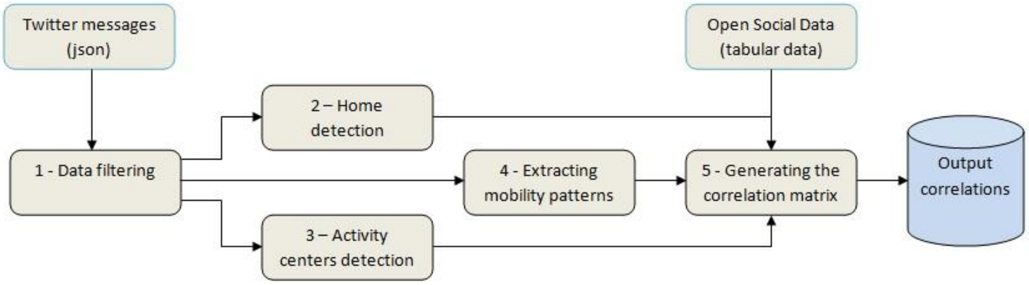

Twitter messages (json)
Open Social Data (tabular data)
2 – Home detection
1 - Data filtering
4 - Extracting mobility patterns
5 - Generating the correlation matrix
Output correlations
3 – Activity centers detection

Figure 1 - Basic flow process of the proposed model

Our model accepts as input a set of Twitter messages in *json* format. The first step consists in filtering the Twitter messages. Initially, the model filters out messages without geographic coordinates information and those whose latitude/longitude coordinates do not point to a location inside the area of interest. Then, still in this filtering process, messages posted by stationary users are also filtered out. We consider as stationary those users whose messages are located within the same area, based on a predefined radius. Filtering this kind of users is important due to the fact that these users generally represent companies that report community information such as weather or traffic conditions, which are not relevant for the proposed model. Finally, similarly to Birkin et al. (2014), we remove all users with less than 20 messages, avoiding noise in the data provided by users with few posts.

In the second and third steps, the model detects the home and the activity centers (most visited regions) for each user. These concepts are used by the model to identify social characteristics of the users. The fourth step extracts statistical properties to express mobility patterns, that are: radius of gyration and user displacement distance (Luo et al., 2016; Cheng et al., 2011; Gonzalez et al., 2008; Hasan et al., 2013). These concepts will be explained in further sessions.

The fifth step consists in the correlation analysis, where the model receives the mobility patterns extracted by the previous process, and accepts tabular data containing the social variables and the polygon related to each region of the city/region to be analyzed, allowing the model to calculate the correlations between mobility and social data.

3.1. Radius of Gyration

The radius of gyration represents the standard deviation of distances between points of a trajectory and the center of mass of these points. This metric can measure how far and how frequently a user moves. A low radius of gyration indicates that a specific user tends to travel mostly locally, with few long-distance checkins, while a high value of this metric generally indicates that the user moves predominantly for long distances (Cheng et. al., 2011). This metric can be formalized in Equation 1 as:

$$r = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(\mathrm{p_i} - \mathrm{p_c})^2}$$

Equation 1. Calculus of radius of gyration

Where:

- r represents the value for the radius of gyration for a user;
- m is the number of messages for a user;
- $\mathrm{p_i}$ represents a particular point where the message was posted;
- $\mathrm{p_c}$ represents the user center of mass (centroid);
- $(\mathrm{p_i} - \mathrm{p_c})$ is the distance between a particular point from the user's centroid;

3.2. User displacement distance

The user displacement distance represents the sum of the distances between all consecutive messages or checkins, reflecting the total distance traveled by the user around the analyzed geographic region.

Cheng et al. (2011) suggest that the behavior of user's displacement for Twitter messages follows a Lévy Flight distribution, which is characterized as a mixture of short and random displacements, with occasional long jumps. Shin et al. (2008) find similar results for the displacement distribution by analysing GPS data in different scenarios, such as metropolitan area and college campuses. In opposition to prevailing Lévy flight random walk models, Gonzalez et al. (2008), by analysing data from cell phone calls, highlight that human displacements have a significant level of temporal and spatial regularity, mainly because they tend to return to a few highly frequented locations.

3.3. Activity centers

An activity center can be defined as a location that a user frequently visits. The locations for an activity center could be a restaurant, home, place of work or any location where the user post his/her messages with some frequency. This concept appears to be an important parameter to express life patterns of a user, indicating the user's preferences for certain places or regions in a city.

To identify such activity centers, we used a popular clustering algorithm called DBSCAN (Density Based Spatial Clustering of Applications with Noise) (Ester et al. 1996). This is a density based clustering algorithm that clusters points that are closely located, and it can deal with points that are located in a low density regions, treating these points as noise. The algorithm has two parameters: the maximum radius of the neighborhood to be considered to form a cluster (ε); and the minimum number of points that a cluster must have (minPts). This algorithm presents advantages over other clustering algorithms, such as: it does not require the user to specify the number of clusters for the execution; it can deal with noise data efficiently; it can find a cluster surrounded by another cluster; and it relies on two variables only.

3.4. Home detection

The region of a user's home represents an important characteristic that needs to be considered in a study that analyze social, economic and demographic data. This is mainly because the user's home location may express social conditions, and these conditions might, in part, influence how the citizens move across an urban center.

Detecting home location based on user's center of mass of all checkins can lead to problems of splitting-the-difference, where a user that travels to distant regions over the city will have her home located at the middle of these regions (Cheng et al., 2011).

In this study, we consider home locations as the most intense activity centers during the night time, following other existing works (Luo et. al., 2016; Huang et al., 2014). For this, we select the messages posted between 8pm and 6am (on weekdays only); then we apply the DBSCAN algorithm to cluster the points of these messages; and finally we select the cluster with the greatest number of points as the user home.

4. Experimental Evaluation

This section presents the experimental evaluation conducted to validate our proposed model for identifying possible correlations between mobility patterns extracted from Twitter messages and social open data provided by governmental organizations.

4.1. Social data and maps

To perform the experiments, the city of London was chosen as a case study, specially due to the large volume of social, economic and demographic data publicly available for this city. In this study, data were collected from the governmental platform "London Datastore" (http://data.london.gov.uk/).

The social data collected for these experiments is related to the year of 2011, since this was the most recent data for the majority of the collected variables. Observed variables are related to the following categories: population age, family structure, ethnic groups, country of birth, house prices, economic activity, qualifications, health, car or van availability and religion. Additionally, we used the data level of LSOA (Lower Super Output Area), which is the smallest area used to divide the city of London for the available data. Each LSOA has an average population of 1,722 inhabitants. Figure 2 shows the map of the city of London used in this research. The map is divided by LSOA regions, and is originally made by the ONS (Office for National Statistics). This is under the terms of the Open Government Licence (OGL)[2] and UK Government Licensing Framework and is therefore free for this kind of use. This map was also acquired from the London Datastore platform.

[2] Contains National Statistics data © Crown copyright and database right [2012] and Contains Ordnance Survey data © Crown copyright and database right [2012].

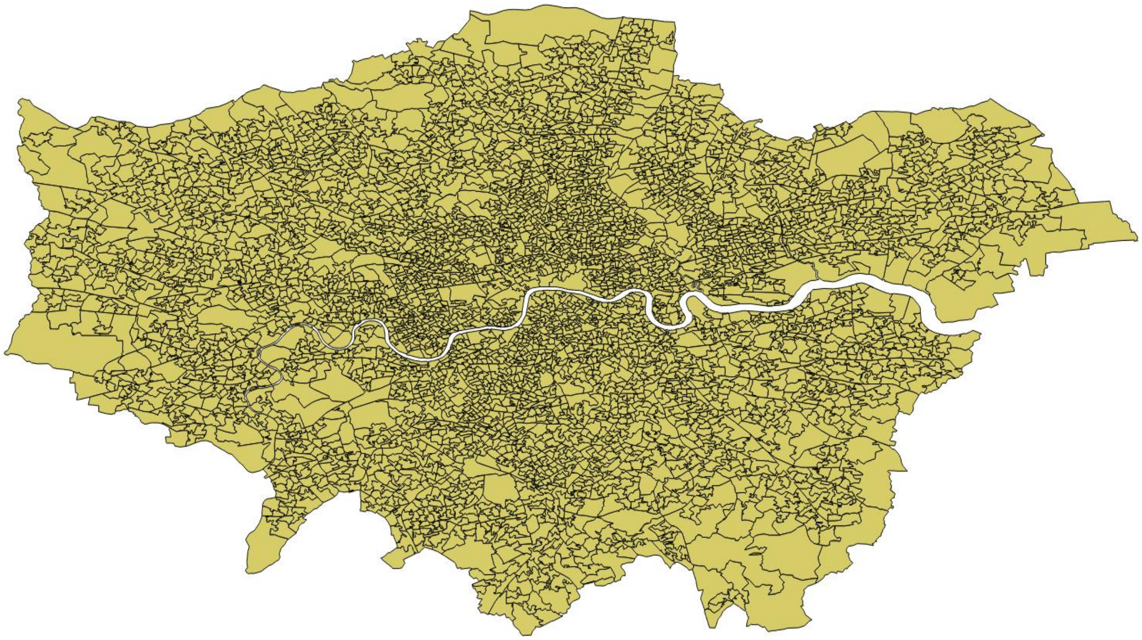


Figure 2 – City of London divided by LSOA regions

4.2. Twitter dataset

For this set of experiments, we collected georeferenced Twitter messages from the city of London via the Twitter API. The messages were collected from November 26th 2014 to November 22nd 2015, totalizing 19,456,798 messages from the region of London. From the initial set of messages, our model filtered out the messages without geographic coordinates information, resulting into 7,680,200 georeferenced messages and a total of 351,656 who have posted these messages. Then, after removing messages whose locations are outside the region of interest, messages posted from stationary users, and messages posted by light users, the dataset was further reduced to 6,203,474 georeferenced messages and 52,974 users, with an average of 117.1 messages per user.

4.3. Design of Experiments

The experiments executed in this research had the prior objective of answering the following questions:

- *Research question (Q1):* Do the characteristics of a user's home region influentiate his/her mobility patterns?
- *Research question (Q2):* Do the characteristics of the regions of a user’s activity centers influentiate his/her mobility patterns?

To answer these questions, the proposed model performs the Kendall’s correlation test to calculate possible correlations between user’s mobility patterns and open social data. We chose this test due to the fact that it neither requires a specific distribution of data nor a linear relation among the variables within the dataset.

We divided the experimental evaluation into two experiments. Experiment 1 consisted in analysing the first two metrics (radius of gyration and displacement distance). For this, we performed the correlation test between these metrics and all social data variables related to users' home locations (previously detected), selecting the correlations with $\tau >= 0.25$ and with statistical significance (*p-value* < 0.05). This approach enables us to identify situations where the social, economic and demographic status of the region where a user's home is located may influentiate his/her mobility patterns, allowing us to answer Question *Q1*.

Experiment 2 has been conducted aiming at answering Question *Q2*. For this, we used the concept of activity centers described in Section 3.3. First, we clustered the dataset of points for each user with the DBSCAN algorithm using ε = 45 meters and *minPts* = 4 in order to find the user's activity centers. After this clustering process, we calculate the medians for all social data variables related to the regions in which these clusters were formed, for each user. Then we performed the correlation tests over these medians and the users' mobility patterns properties, allowing us to answer Question *Q2*. For both experiments, we adopted the value of 45 meters for filtering stationary users.

To perform these experiments, we divided the users into three categories: Category 1, 2 and 3. Respectively, they group the users who have posted at least 1,000 messages (679 users); 2,500 messages (168 users); and 5,500 messages (33 users). This division was made with the objective of identifying correlations that can only be found for heavy users, possibly due to the imprecision and fragmentation of messages posted in the Twitter network.

5. Results and Discussion

The histograms of the two mobility variables extracted by the model and used in this study are shown in Figure 3. In these histograms, it is possible to visualize the frequency at which the values of both variables occur in the extracted mobility dataset. The displacement distance histogram is shown in a log10 scale while the radius of gyration histogram is represented by its original values. Both variables were calculated in meters. In the first histogram, it can be seen that the majority of users have their radius of gyration between 3,000 meters and 4,000 meters, totalizing 7,554 users. The second histogram shows that most users have their displacement distance between 100,000 meters and 316,228 meters, representing 19,514 users.

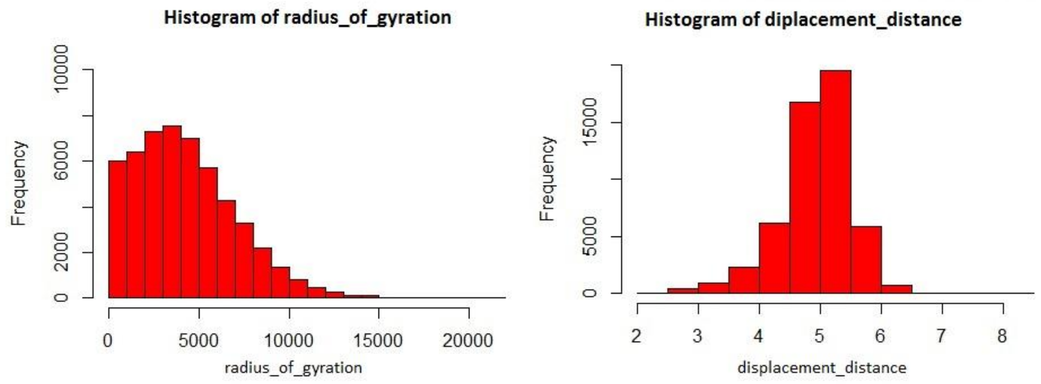


Figure 3 - Radius of Gyration and displacement distance (log10 scale) histograms

Aiming at answering Question *Q1*, after the generation of the correlation matrix, we selected the most relevant correlations found by the model. No significant correlation has been found for users from Categories 1 and 2 (users with at least 1,000 and 2,500 messages); however, some significant correlations were found for users from Category 3. The most significant correlations found for this group are shown in Table 1. In this table, we used the notation of a tuple (*τ, p-value*), where the first element represents the Kendall's *tau* (the correlation coefficient for this test), and the second represents the significance of the executed test, where a *p-value* less than 0.05 allows the rejection of the null hypothesis for the correlation test, denoting that there may be a real correlation between the variables of mobility patterns and social data.

Table 1 - Results for Experiment 1 (correlating radius of gyration and displacement distance with social variables). Values for users from Category 3.

Mobility / Social Data	Age 0-15	Couple with children	Economically inactive people	Employment rate	Unemployment rate	Persons with no qualifications
Radius of Gyration	-	-	(-0.35, 0.003)	(0.26, 0.03)	-	(-0.27, 0.02)
Displacement Distance	(-0.27, 0.02)	(-0.31, 0.01)	(-0.28, 0.02)	(0.29, 0.01)	(-0.38, 0.001)	(-0.29, 0.01)

From the results shown in Table 1 (which are related to Experiment 1), it can be observed that the highest correlation values were found for the analysis of social variables related to employment conditions. For instance, for the correlation between "Displacement Distance" and "Unemployment rate", we obtained $\tau = -0.38$. This negative correlation expresses that as the "Displacement Distance" of a user increases, the value for the variable "Unemployment rate" (related to the user's home region) tends to decrease, indicating that users with longer traveled distances tend to live in regions with low unemployment rate. A similar finding is shown for the variables "Radius of Gyration" and "Economic Inactive People", with $\tau = -0.35$. For the social variable "Employment Rate", when tested with the "Displacement Distance", we found a positive correlation, with $\tau = 0{,}29$, denoting that the greater is the displacement distance of users, the greater is the employment rate of the region they live.

The results obtained for Experiment 1 allowed us to answer Question *Q1*, since we could find correlations with some statistical significance, denoting that there are correlations between users' mobility patterns and social aspects of their home location, specially for variables related to employment conditions.

For Experiment 2, where we analyze possible correlations between the two mobility metrics (radius of gyration and displacement distance) and the social variables related to users' activity centers, we found no significant correlations for users from Categories 1 and 2. Again, the most significant correlations were found for user from Category 3.

Table 2 - Results for Experiment 2 (correlating mobility metrics and social variables related to users' activity centers). Values for users from Category 3.

Mobility / Social Data	Employment rate	Unemployment rate	Persons with no qualifications
Radius of Gyration	(0.25, 0.03)	-	(-0.31, 0.01)
Displacement Distance	(0.36, 0.002)	(-0.32, 0.007)	(-0.28, 0.02)

The results obtained for Experiment 2 are shown in Table 2. These results show some conformance with the first experiment. Here, again, the social variables related to employment conditions presented higher correlation coefficients. For the variable "Displacement Distance" the highest correlation was found with the variable "Employment Rate", with a τ = 0.36. For the "Radius of Gyration", the highest correlation found was with the variable "Persons with no Qualifications" (also related to employment conditions), with τ = -0.31. Given these results, we can answer Question *Q2* by stating that users' mobility patterns may be correlated with social attributes of regions that they visit in an urban center. For example, we found that users that have higher "Displacement Distance" tend to visit regions with higher "Employment rates" (τ = 0.36). Moreover, for these same users, the incidence of activity centers in regions with highest "Unemployment rates" tend to decrease when the "Displacement Distance" increases (τ = -0.32).

It is important to note that even finding possible correlations between the mobility patterns and social data, these correlations were not classified as strong correlations, as the highest value was of τ = -0.38. We believe that the fragmented nature of Twitter messages can add some inaccuracies to the results. For example, poor users might post significantly more than users from rich locations, bringing imprecisions to the results. This kind of problem was partially mitigated by the segmentation of users based on the number of messages they have posted (Categories 1, 2 and 3). For users in Category 3, which provided the best results, we could visually observe that they were homogeneously distributed over the city of London. Furthermore, it is not possible to extrapolate the results obtained from the correlations to the whole population of London, as these results were based on certain Twitter profiles.

6. Conclusions

This research presented a model to allow the identification of correlations between mobility patterns and social, economic and demographic variables. This model identifies mobility patterns from georeferenced Twitter messages, detect users' home locations and activity centers, then looks for correlations with the social data supplied to the model. An experimental evaluation was conducted using data from the city of London. This city was chosen due to the high availability of Twitter messages in the time interval where these messages were collected and also for the availability of many social indicators for this city.

This study confirms that it is possible to identify some correlations between mobility patterns extracted from social media and social indicators. In the results obtained from our experiments, relevant correlations were found for variables associated with employment conditions (economically inactive people, employment rate, unemployment rate and persons with no qualifications).

Additionally, the fragmented nature of Twitter messages makes the task of finding correlations even challenging, forcing us to reduce the number of users to be considered in the experiments (only those with a large number of posts). This indicates the need of performing additional experiments involving more heavy users, to make the correlations more significant. Further work also includes applying this model to the analysis of other regions in the world and comparing the results between them. Moreover, we intend to formulate additional mobility metrics, enhancing the analysis of mobility patterns and the discovery of relevant correlations.

References

Luo, F., Cao, G., Mulligan, K., & Li, X. (2016) "Explore spatiotemporal and demographic characteristics of human mobility via Twitter: A case study of Chicago", *Applied Geography*, *70*, 11-25.

Noulas A, Scellato S, Lambiotte R, Pontil M, Mascolo C (2012) "A tale of many cities: universal patterns in human urban mobility", PLoS ONE;7(5).

Wilson T, Bell M. (2004) "Comparative empirical evaluations of internal migration modelsin subnational population projections", Journal of Population Research;21(2):127-160.

Zheng, V. W., Zheng, Y., Xie, X., & Yang, Q. (2010) "Collaborative location and activity recommendations with gps history data", In Proceedings of the 19th international conference on World wide web (pp. 1029-1038). ACM.

Jurdak, R., Zhao, K., Liu, J., AbouJaoude, M., Cameron, M., & Newth, D. (2015) "Understanding human mobility from Twitter", *PloS one*, *10*(7), e0131469.

Birkin, M., Harland, K., Malleson, N., Cross, P., & Clarke, M. (2014) "An Examination of Personal Mobility Patterns in Space and Time Using Twitter", International Journal of Agricultural and Environmental Information Systems (IJAEIS), 5(3), 55-72.

Hawelka, B., Sitko, I., Beinat, E., Sobolevsky, S., Kazakopoulos, P., &Ratti, C. (2014) "Geo-located Twitter as proxy for global mobility patterns", Cartography and Geographic Information Science, 41(3), 260-271.

Yuan, Q., Cong, G., Ma, Z., Sun, A., &Thalmann, N. M. (2013) "Who, where, when and what: discover spatio-temporal topics for twitter users", In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 605-613). ACM.

Hasan, S., Zhan, X., &Ukkusuri, S. V. (2013) "Understanding urban human activity and mobility patterns using large-scale location-based data from online social media", In Proceedings of the 2nd ACM SIGKDD international workshop on urban computing (p. 6). ACM.

Cheng, Z., Caverlee, J., Lee, K., & Sui, D. Z. (2011) "Exploring Millions of Footprints in Location Sharing Services", ICWSM, 2011, 81-88.

Hao Q, Cai R, Wang C, Xiao R, Yang J, Pang Y, Zhang L (2010) "Equip tourists with knowledge mined from travelogues", In: Proceedings of the 19th international conference on World Wide Web, pp 401–410

Bagrow, J. P., & Lin, Y. R. (2012) "Mesoscopic structure and social aspects of human mobility", PloS one, 7(5), e37676.

Hsieh H-P, Li C-T, Lin S-D (2012) "Exploiting large-scale check-in data to recommend time-sensitive routes", In: Proceedings of the ACM SIGKDD international workshop on urban computing. ACM, New York, pp 55–62

Gonzalez, M. C., Hidalgo, C. A., &Barabasi, A. L. (2008) "Understanding individual human mobility patterns", Nature, 453(7196), 779-782.

Jiang, S., Fiore, G. A., Yang, Y., Ferreira Jr, J., Frazzoli, E., & González, M. C. (2013) "A review of urban computing for mobile phone traces: current methods, challenges and opportunities", In Proceedings of the 2nd ACM SIGKDD International Workshop on Urban Computing (p. 2). ACM.

Palchykov, V., Mitrović, M., Jo, H. H., Saramäki, J., & Pan, R. K. (2014) "Inferring human mobility using communication patterns", Scientific reports, 4.

Rhee, I., Shin, M., Hong, S., Lee, K., Kim, S. J., & Chong, S. (2011) "On the levy-walk nature of human mobility", IEEE/ACM transactions on networking (TON), 19(3), 630-643.

Zhao, K., Musolesi, M., Hui, P., Rao, W., & Tarkoma, S. (2015) "Explaining the power-law distribution of human mobility through transportation modality decomposition". Scientific reports, 5.

Chaintreau, A., Hui, P., Crowcroft, J., Diot, C., Gass, R., & Scott, J. (2007) "Impact of human mobility on opportunistic forwarding algorithms", Mobile Computing, IEEE Transactions on, 6(6), 606-620.

Zhang, Y., Wang, L., Zhang, Y. Q., Li, X. (2012) "Towards a temporal network analysis of interactive WiFi users", EPL (Europhysics Letters), 98(6), 68002.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996) "A density-based algorithm for discovering clusters in large spatial databases with noise", In Proceedings of 2nd international conference on knowledge discovery and data mining (vol. 96).

Huang, Q., Cao, G., & Wang, C. (2014) "From where do tweets originate?: a GIS approach for user location inference", In Proceedings of the 7th ACM SIGSPATIAL International Workshop on Location-Based Social Networks (pp. 1-8). ACM.

Shin, R. M., Hong, S., Lee, K., & Chong, S. (2008) "On the Levy-walk nature of human mobility: Do humans walk like monkeys?", In Proc. IEEE INFOCOM (pp. 924-932).

Shelton, T., Poorthuis, A., & Zook, M. (2015). Social media and the city: Rethinking urban socio-spatial inequality using user-generated geographic information. Landscape and Urban Planning, 142, 198-211.

Exact intersection of 3D geometric models

Salles V. G. de Magalhães [1,2], **Marcus V. A. Andrade** [1], **W. Randolph Franklin**[2], **Wenli Li**[2], **Maurício G. Gruppi** [1]

[1]Departamento de Informática – Universidade Federal de Viçosa (UFV)
Viçosa – MG – Brazil

[2]Rensselaer Polytechnic Institute (RPI), Troy – NY – USA

{salles,marcus,mauricio.gruppi}@ufv.br, mail@wrfranklin.org, liw9@rpi.edu

Abstract. *We present* 3D-EPUG-OVERLAY*, an exact algorithm for computing the intersection of two 3D triangulated meshes. This is useful in GIS and CAD.* 3D-EPUG-OVERLAY *has several innovations, including the use of exact rational arithmetic to avoid floating roundoff errors and the ensuing topological impossibilities. It also uses a uniform grid to efficiently index the geometric data.* 3D-EPUG-OVERLAY *was designed to be easily parallelizable. We are now incorporating Simulation of Simplicity, to correctly handle geometric degeneracies (coincidences). Our current implementation can easily process examples with millions of triangles.*

1. Introduction

Computing intersections or overlays is important to CAD, GIS, computer games and computational geometry. E.g., in 2D, consider two maps A and B, each composed of faces or polygons representing a partition of the E^2 plane. The overlay of A with B is a map C where each polygon of C is the intersection of a polygon of A with a polygon of B. For example, if A represents the coterminous states of the United States, and B represents drainage basins, then C is a new map where each polygon represents the part of each basin that is in each state.

While GIS usually deal with 2D geometric data, there are several applications for 3D GIS. For example, while a 2D map could model the street network of a city, the hydrological network of a state, or the different kinds of soil in a region, a 3D model could model more complex features such as layers of soil in a mine, the subway tunnels in a city, the buildings in region, etc Yanbing et al. (2007). Computing intersections is an important operation often required by these systems. An example of application is to intersect polyhedra representing layers of soil with a polyhedron representing a section of the soil to be digged in a mine. The resulting intersection represents the different kinds of soil that will be extracted during the excavation.

According to Feito et al. (2013), although 3D models have been widely used, processing is still a challenge. Due to the algorithm complexity caused by the need to handle special cases, the necessity of processing big volumes of data, and the loss if precision problems caused by floating point arithmetic, they note that software packages occasionally “fail to give a correct result, or they refuse to give a result at all”. The likelihood of failure increases as datasets get bigger.

An algorithm that occasionally fails might be acceptable. Nevertheless, an efficient, robust, and even exact, algorithm is especially important when it is a subroutine of another algorithm.

Hachenberger et al. (2007) presented, and CGAL (2016) implemented, an algorithm for computing the exact intersection of Nef polyhedra. A Nef polyhedron is a finite sequence of complement and intersection operations on half-spaces. However, according to Leconte et al. (2010), these algorithms have some limitations such as poor performance. Another limitation is their use of Nef Polyhedra, which are uncommon. Bernstein and Fussell (2009) also presented an intersection algorithm that tries to achieve robustness. Their basic idea is to represent the polyhedra using binary space partitioning (*BSP*) trees with fixed-precision coordinates. They mention that the main limitation is that the process to convert BSPs to widely used representations (such as meshes) is slow and inexact.

In previous works we have developed exact and efficient algorithms for processing 2D (polygonal maps) and 3D models (triangulated meshes). More specifically, we have developed algorithms for intersecting polygonal maps (Magalhães et al., 2015) and performing point location queries (Magalhães et al., 2016) in both polygonal maps and 3D meshes. These algorithms employ a combination of five separate techniques to achieve both robustness and efficiency. Exact arithmetic is employed to completely avoid errors caused by floating point numbers. Special cases (geometric degeneracies) are treated using *Simulation of Simplicity* (SoS) (Edelsbrunner and Mücke, 1990). The computation is performed using simple local information to make the algorithm easily parallelizable and to easily ensure robustness. Efficient indexing techniques with a uniform grid, and High Performance Computing (HPC) are used to mitigate the overhead of exact arithmetic.

In all these algorithms our spatial data is represented using simple topological formats. The 2D maps are represented using sets of oriented edges where each edge contains the labels of the polygons on its positive and negative sides. In 3D, the meshes are represented using a set of oriented triangles and each triangle has the labels of the polyhedra on its positive and negative sides.

In this paper we will present a brief description of these previous works and present our current research: 3D-EPUG-OVERLAY (3D-Exact Parallel Uniform Grid-Overlay), a parallel algorithm for exactly intersecting 3D triangulated meshes.

2. Roundoff errors

Non-integer numbers are usually approximated with floating-point values. The difference between a non-integer and its approximation is often referred as roundoff error. Even though these differences are usually small, arithmetic operations frequently create more errors, which accumulate, becoming larger.

In geometry, roundoff errors can generate topological inconsistencies causing globally impossible results for predicates like point-inside-polygon. For example, Kettner et al. (2008) presented a study of the failures caused by roundoff errors in geometric problems such as the planar orientation computation.

Several techniques have been proposed to overcome this problem. The simplest one consists of using an ϵ tolerance, and then consider two values x and y as equal if $|x - y| \leq \epsilon$. However this is not a good strategy because equality is no longer transitive, nor invariant under scaling. In practice, epsilon-tweaking fails in several situations, (Kettner et al., 2008).

Snap rounding is another method to approximate arbitrary precision segments into fixed-precision numbers (Hobby, 1999). However, snap rounding can generate inconsistencies and deforms the original topology if applied consecutively on a data set. Some variations of this technique attempt to get around these issues (Hershberger, 2013; Belussi et al., 2016).

Shewchuk (1996) presents the Adaptive Precision Floating-Point technique, that focus on exactly evaluating predicates. The idea is to perform this evaluation using the minimum amount of precision necessary to achieve correctness. As mentioned by the author, this technique focus on geometric predicates and it is not suitable to solve all geometric problems. For example, "a program that computes line intersections requires rational arithmetic".

The formally proper way to eliminate roundoff errors and guarantee robustness is to use exact computation based on rational numbers with arbitrary precision (Li et al., 2005; Hoffman, 1989; Kettner et al., 2008). In this work, our algorithms perform computation using arbitrary precision rationals provided by the GMP library. Computing in the algebraic field of the rational numbers over the integers, with the integers allowed to grow as long as necessary, allows the traditional arithmetic operations to be computed exactly, with no roundoff error. The cost is that the number of digits in the result of an operation is about equal to the sum of the numbers of digits in the two inputs. This behavior is acceptable if the depth of the computation tree is small, which is true for the algorithms we will present.

Besides ensuring exact results, the use of arbitrary precision rationals has other advantages. First, Simulation of Simplicity, a technique for treating degeneracies, requires exact arithmetic. Second, our algorithms will be able to support input data where the coordinates are represented using rationals and, thus, we will be able to process meshes that cannot be exactly represented using floating point numbers.

3. Previous works

In this section, EPUG-OVERLAY (Magalhães et al., 2015) and PINMESH (Magalhães et al., 2016), two previous algorithms developed for, respectively, intersecting maps and performing point location queries in 3D meshes will be presented.

First, two important techniques applied in these works will be briefly described: the use of a uniform grid for indexing the data and the application of the Simulation of Simplicity technique for handling special cases. Both techniques will be also applied in the intersection algorithm described in this paper.

Understanding EPUG-OVERLAY, PINMESH and Sections 3.1 and 3.2 is important because techniques similar to the ones described in these sections are applied to 3D-EPUG-OVERLAY.

3.1. Indexing data with a uniform grid

Franklin et al. (1989) proposed a uniform grid to accelerate his algorithm for computing the area of overlaid polygons. When a polygonal map (or triangular mesh) is indexed with a uniform grid, a 2D grid (or 3D grid for meshes) is created, superimposed over the input datasets, and then the edges (or triangles) intersecting each cell c are inserted into c. The efficiency of this idea depends on a careful choice of the concrete data structure. After the grid is created, it can be employed to accelerate the geometric algorithms. For example, given two maps indexed by the grid, the intersection of pairs of edges from the two maps can be found by processing each cell and comparing the edges in that cell pair-by-pair (one edge from each map) to compute the intersection points.

The uniform grid works well even for unevenly distributed data for various reasons (Akman et al., 1989; Franklin et al., 1988). First, the total time is the sum of one component (constructing the grid) that runs slower with a finer grid, plus other components (e.g., intersecting edges) that run faster. The total running-time varies slowly with changing grid resolutions. Second, an empty grid cell is very inexpensive, so that sizing the grid for the densest part of the data works.

Nevertheless, to process very uneven data, in EPUG-OVERLAY and PINMESH we have incorporated a second level grid into those few cells that are densely populated. The exact criteria for determine what cell to refine depends on the algorithm that will use the grid. For example, since in the intersection computation pairs of edges in the cells are tested for intersection, one could refine the grid cells where the number of intersection tests (i.e, the number of pairs of edges from the two maps) is greater than a threshold.

This nesting could be recursively repeated until all grid cells have fewer elements than a given threshold, creating a structure similar to quadtree (or octree), although with more branching. However, the general solution uses more space for pointers (or is expensive to modify) and is irregular enough that parallelization is difficult. Also, experiments have shown that the best performance is achieved using just a second level (Magalhães et al., 2015). This can be explained because the first level grid, in general, has many cells with more elements than the threshold justifying the second level refinement. But, in the second level, only a few number of cells exceed the threshold and the overhead (processing time and memory use) to refine those cells is never recaptured.

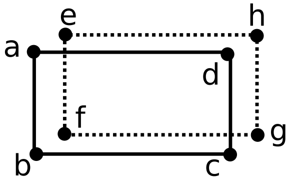
3.2. Simulation of Simplicity

To correctly handle special cases such as coincident edges when intersecting maps, we apply Simulation of Simplicity (SoS) Edelsbrunner and Mücke (1990). This is a general purpose symbolic perturbation technique designed to treat special (degenerate) cases. The inspiration for SoS is that if the coordinates of the points are perturbed, the degeneracies disappear. However, too big a perturbation may create new problems, while a too small one may be ineffective because of the limited precision of floating point numbers.

SoS is a solution that uses a symbolic perturbation by an indeterminate infinitesimal value ϵ^i, for some natural number i. Its mathematical formalization extends some exactly computable field, such as rationals, by adding orders of infinitesimals, ϵ^i. Floating point numbers with roundoff error cannot be the base. The infinitesimal ϵ is an *indeterminate*. It has no meaning apart from the rules for how it combines. All positive first-order infinitesimals are smaller than the smallest positive number. All positive second-order infinitesimals are smaller than the smallest positive first-order infinitesimal, and so on. All this is logically consistent and satisfies the axioms of an abstract algrebra field.

The result of SoS is that degeneracies are resolved in a way that is globally consistent. For example, consider Figure 1. Two identical rectangles ($abcd$ represented using solid edges and *efgh* represented using dashed edges) are overlaid, but all the vertices of *efgh* are slightly translated using the vector (ϵ, ϵ^2). This translation is globally consistent, i.e., even if the rectangle is stored as separate edges an intersection test with edge *ef* will return true only when this test is performed against the edge *ad* while an intersection test performed with *gf* will return true only when the test is performed against *cd*.

Figure 1. Effect caused by SoS during the intersection computation.



The infinitesimals do not need to be explicitly used in the program since they will be used only to determine signs of expressions. The only time that the infinitesimals change the result is when there is a tie in a predicate. Then, the infinitesimals break the tie. The effect is to make the code harder to write and longer. However, unless a degeneracy occurs, the execution speed is the same. When a degeneracy does occur, the code is slightly slower.

3.3. Point location

PINMESH (Magalhães et al., 2016) is an exact and efficient algorithm for performing point location queries in 3D meshes. It is based on the idea of ray-casting: given a query point q, a semi-infinite vertical ray r is traced from q, and then the triangle t whose intersection with r is the lowest is used to determine q's location. Since t is the lowest triangle to intersect r, because of the Jordan Curve Theorem, q will necessarily be on the polyhedron below t (this polyhedron can be quickly determined since all triangles contain the labels of the two polyhedra it bounds).

A uniform grid is used to reduce the number of ray-triangle intersection computation tests. Also, empty grid cells, which are each necessarily completely inside one polyhedron, are labeled with that containing polyhedron. That accelerates many queries. As a result of a careful implementation and use of parallelization, PINMESH is very efficient, being able to index a dataset and perform 1 million queries on a 16-core processor up to 27 times faster than RCT (Liu et al., 2010), a sequential and inexact algorithm, which was previously the fastest.

To summarize, PINMESH, represents coordinates with rational numbers to completely prevent roundoff errors, and handles special cases with simulation of simplicity.

3.4. Exact 2D map overlay

EPUG-OVERLAY (Magalhães et al., 2015) is an exact and efficient algorithm for overlaying two polygonal maps. Given two maps $\mathcal{A}$ and $\mathcal{B}$ composed of faces represented implicitly as sets of edges, the goal is to create a map where each face represents the intersection of a face of $\mathcal{A}$ with a face of $\mathcal{B}$. Parallel programming plus efficient indexing make EPUG-OVERLAY very efficient. It can process maps with more than 50 million edges faster than GRASS GIS, which is sequential and subject to roundoff error, since it does not use exact arithmetic.

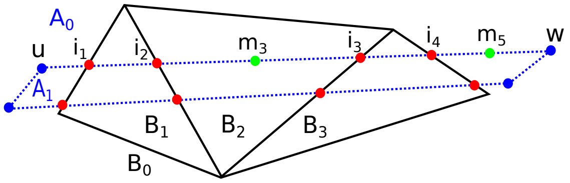
As described in (Magalhães et al., 2015), EPUG-OVERLAY has the following basic steps:

1. **Create a 2-level uniform grid** to index the edges from the two input maps A and B.
2. **Compute all intersection points between an edge of** A **with an edge of** B using the uniform grid is applied to accelerate the process, by iterating through the grid cells and testing all pairs of edges in each grid cell for intersection. The intersecting edges are split at the intersection point. After that, edges intersections will happen only at vertices.
3. **Label the resulting split edges** with their adjacent polygons.

Figure 2 illustrates this process: map $\mathcal{A}$ (in dotted blue) contains 4 edges and two polygons (polygon A_1 and polygon A_0, representing the exterior of the map) while map $\mathcal{B}$ (solid black lines) contains 7 edges and 4 polygons. After the intersections are detected and the edges are split at the intersection points (in red) the resulting edges are classified. For example, edge (u, w) bounds polygons A_0 (positive side) and A_1 (negative side). Edge (i_2, i_3) (generated after (u, w) was split) is inside polygon B_2 of the other map and, thus, in the output map (i_2, i_3) will bound polygon $A_0 \cap B_2$ (this polygon is equivalent to the exterior of the resulting map) on its positive side and $A_1 \cap B_2$ on the negative side.

Since the edges are split at the intersection points, after this process all edges will be completely inside a polygon of the other map. Thus, one strategy to determine in what polygon an edge e is consists in using a fast 2D point location algorithm to locate a point from e in the other map (for example, the location of m_3 from Figure 2 can determine in what polygon (i_2, i_3) is).

Figure 2. Intersecting two polygonal maps.



This strategy uses only local information to compute intersections. That is, instead of intersecting pairs of faces, the individual edges are intersected and classified; the resulting faces will be represented implicitly by the edges. This has several advantages. First, it is easier to test a pair of edges for possible intersection than to test a pair of faces, which would devolve to testing pairs of edges anyway. Second, knowing an intersection of a pair of edges contributes information about four output faces. Third, as an edge is fixed size but a face is not, parallel operations on edges are more efficient.

Degenerate cases are handled with *Simulation of Simplicity (SoS)*. The idea is to pretend that map $\mathcal{A}$ is slightly below and to the left of map $\mathcal{B}$. Thus no edge from $\mathcal{A}$ will coincide with an edge from $\mathcal{B}$ during the intersection computation. Oversimplified slightly, the process proceeds by translating map $\mathcal{B}$ by (ϵ, ϵ^2), where ϵ is an infinitesimal. As mentioned before, we do not actually compute with infinitesimals, but instead determine the effect that they would have on the predicates in the code, and modify the predicates to have the same effect when evaluated as if the variables could have infinitesimal values. For instance, the test for $(a_0 \leq b_0)\&(b_0 \leq a_1)$ becomes $(a_0 \leq b_0)\&(b_0 < a_1)$. With SoS, no point in $\mathcal{A}$ is identical to any point in $\mathcal{B}$, and neither do two any edges coincide.

4. Exact 3D mesh intersection

Similarly to our 2D intersection algorithm, in 3D the computation is performed using only local information stored in the individual triangles. That is, the triangles from one mesh are intersected with the triangles from the other one. Then a new mesh containing the triangles from the two original meshes is created and the original triangles are split at the intersection points. That is, if a pair of triangles in this new mesh intersect, then this intersection will happen necessarily in a common edge or vertex. Finally, the adjacency information stored in each triangle is updated to ensure that the new mesh will consistently represent the intersection of the original ones.

4.1. Intersecting triangles and remeshing

For performance, a strategy similar to the one used in EPUG-OVERLAY was adopted: for each uniform grid cell, the intersections between pairs of triangles from the two triangulations are computed. The pairs of triangles are intersected using the algorithm presented by Möller (1997), that uses several techniques to avoid unnecessary computation by detecting as soon as possible if the pair of triangles does not intersect.

More specifically, a two-level 3D uniform grid is employed to accelerate the computation using an strategy similar to the one we used in the 2D map intersection algorithm. That is, the grid will be created by inserting in its cells triangles from both meshes M_1 and M_2. Then, for each grid cell c, the pairs of triangles from both meshes in c are intersected. If the resolution of the uniform grid is chosen such that the expected number of triangles per grid cell is a constant K, then it is expected that each triangle will be tested for intersection with the other K triangles in its grid cell. Thus, the expected total number of intersections tests performed will be linear in the size of the input maps.

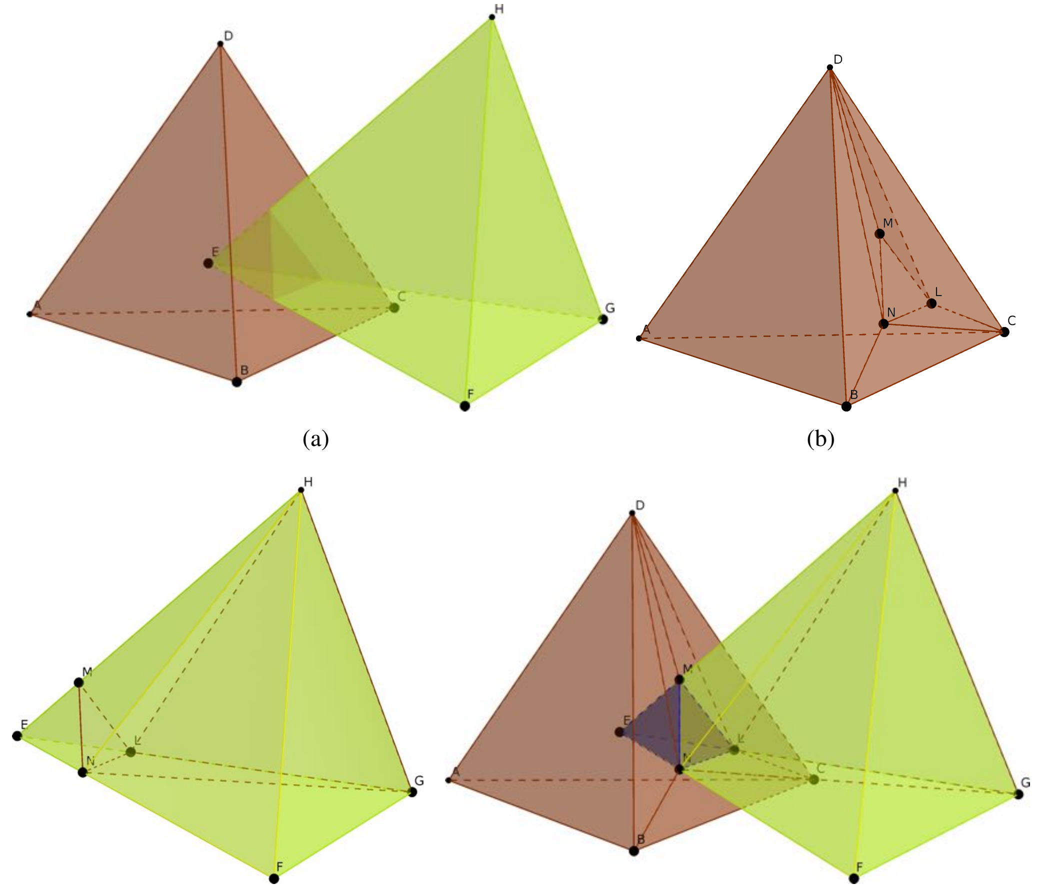
Since the cells do not influence each other, the process of intersecting the triangles can be trivially parallelized: the grid cells can be processed in parallel by different threads using a parallel programming API such as OpenMP.

After computing the intersections between each pair of triangles, the next step is to split the triangles where they intersect to create new ones, so that now all the intersections will happen only on common vertices or edges. When a triangle is split, the labels of its two bounding objects will be copied to the new triangles. This process is similar to the 2D map overlay step where the edges are split at the intersection points to ensure that all intersections happen in vertices.

Figure 3 presents an example of intersection computation. In Figure 3(a), we have two meshes representing two tetrahedra with one region in each one: the brown mesh (mesh M_1) bounds the exterior region and region 1 while the yellow mesh (mesh M_2) bounds the exterior region and region 2.

After the intersections between the triangles are computed, the triangles from one mesh that intersect triangles from the other one are split into several triangles, creating meshes M_1' and M_2' (for clarity, these two meshes are displayed separately in Figures 3(b) and (c), respectively). The only triangle from mesh M_1 that intersects mesh M_2 is the triangle BCD. Since BCD intersects three triangles from M_2, it was split in 7 triangles when M_1' was created (triangles LMN, CLN, CBN, BDN, DMN, DLM and CDL). Similarly, each of the three triangles from M_2 intersecting M_1 was split into 3 smaller triangles.

Figure 3. Computing the intersection of two tetrahedra.



(a)

(b)

(c)

(d)

4.2. Classifying triangles

After the intersections are detected and all the triangles that intersect other triangles are split at the intersection points, two new meshes M_1' and M_2' are created such that each new mesh M_i' will have the following two kinds of triangles:

- Triangles from the original mesh: if a triangle t from M_i did not intersect any triangle from the other mesh (or if this intersection was located on a vertex or edge), then t will be in M_i'.
- New triangles: if a triangle t from M_i intersects one or more triangles from the other mesh (and this intersection is not located on a common vertex or edge), then t will be split into several smaller triangles and these smaller triangles will be inserted into M_i'.

It is clear that each mesh M_1' will exactly represent the same regions that M_1 represents. In fact, if no triangle from M_1 intersects the mesh M_2, then M_1' will be equal to M_1. Otherwise, each triangle t from M_i that intersects M_2 will be split in n triangles $t_1, t_2, ..., t_n$ and these new triangles will be inserted into M_i' instead of t. Since the union of the triangles $t_1, t_2, ..., t_n$ is t and these split triangles contain the same attributes as t, then M_1' represents the same regions M_1 represents. This observation is also valid for M_2'.

Thus, computing the intersection between M_1' and M_2' is equivalent to computing the intersection of M_1 with M_2. However, M_1' and M_2' are easier to process: since the triangles from one mesh intersect with the triangles of the other one only in common vertices or edges, then each triangle t from M_1' will be completely inside a region from M_2'. Suppose a triangle t from M_1' bounds regions R_a and R_b and is completely inside region R_c from mesh M_2'. When M_1' is intersected with M_2', t will be in the resulting mesh and it will bound regions $R_a \cap R_c$ and $R_b \cap R_c$. The same process can be performed with the triangles from M_2.

Therefore, the process of classifying the triangles to create the output mesh consists in processing each triangle t from the mesh M_1', determining in what region of M_2' t is and, then, updating the information about the regions t bounds such that we will have a consistent mesh. The same process needs to be performed with triangles from M_2'.

To determine in what region from the other mesh a triangle is, the point location algorithm from Section 3.3 is applied. That is, since point location queries can be quickly performed, an efficient way to locate a triangle that is completely inside a region consists in locating one of its interior points (for example, its centroid).

This classification can also be performed in parallel since updating the regions that a triangle bounds does not influence other triangles.

If a triangle t is in the exterior of the other mesh, in the resulting mesh the two regions t bounds will be the exterior region. To maintain the mesh consistency, the triangles bounding only the exterior region can be ignored and not stored in the output mesh.

Figure 3(d) illustrates the classification step. All the intersections happen at common edges, and the only triangle from M_1' that is completely inside region 2 (of M_2') is triangle LMN. Since LMN bounds region 1 and the exterior region in M_1', in the resulting intersection LMN will bound region $1 \cap 2$ and the exterior region. All the other triangles from M_1' are in the exterior region of M_2' and, thus, they will only bound the exterior region in the resulting intersection (therefore, they will be ignored when the output mesh is computed). Similarly, in M_1' the only triangles that are inside region 1 of M_1' are triangles EMN, ELM and ELN. These three triangles will also bound the exterior region and region $1 \cap 2$ in the resulting mesh.

4.3. Handling the special cases

The current version of 3D-EPUG-OVERLAY does not handle special cases (degeneracies) yet. However, the ideas we intend to apply in order to handle these cases have already been successfully implemented for EPUG-OVERLAY and PINMESH, and therefore we believe they will suitable to 3D-EPUG-OVERLAY.

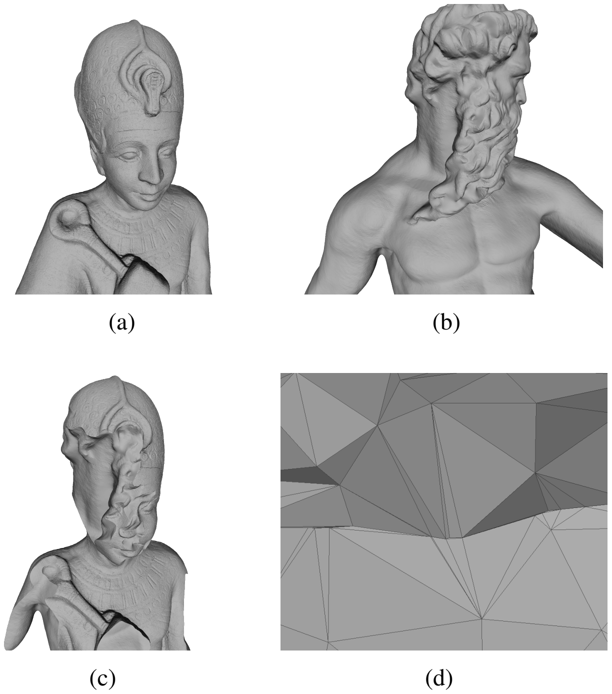
Without SoS, it would be too difficult to guarantee that all degeneracies are considered (this is particularly true in 3D). An adequate perturbation scheme associated with the use of exact arithmetic and a careful implementation will ensure our intersection algorithm is robust.

5. Preliminary results

3D-EPUG-OVERLAY was implemented in C++, and several experiments performed. Figure 4 presents an example of intersection computed using 3D-EPUG-OVERLAY: the model Ramesses (a) and Neptune (b) were intersected. These two models were downloaded from the repository in AIM@SHAPE (2016), and were produced by, respectively, Marco Attene and Laurent Saboret. The Ramesses model contains more than 1 million triangles while the Neptune model contains more than 4 million triangles. Figure 4(c) shows the result of the intersection.

Figure 4(d) presents a zoom that detaches the region of the resulting mesh where the triangles from the two models intersect. We see that the remeshing process generates several thin triangles (displayed in the vertical center of the figure), which are usually hard to process with methods using floating-point arithmetic.

Figure 4. Computing the intersection of two 3D models.



(a)

(b)

(c)

(d)

Since some features of 3D-EPUG-OVERLAY, such as SoS, are still under implementation, and the main feature of 3D-EPUG-OVERLAY is its exactness, we intend to optimize its performance only after those features are implemented. However, we intend to employ the same strategies successfully used in EPUG-OVERLAY and PINMESH. They include:

1. Trading memory for computation, pre-computing and storing results that will be needed several times.
2. Parallelization of the bottlenecks of the algorithm using OpenMP: similarly to our previous work, 3D-EPUG-OVERLAY was designed specifically for being easily parallelizable.
3. Reduction of memory allocations on the heap since they cannot be efficiently performed in parallel. Our previous experience has showed that this should be avoided especially inside parallelized blocks of code. However, as rationals grow, memory needs to be allocated. Therefore we pre-allocate enough temporary rationals that creating them inside parallelized functions is not necessary.

Since these techniques were so sucessful in our previous works, so that they even outperformed inexact algorithms, we believe they will also make 3D-EPUG-OVERLAY very efficient.

6. Conclusion and future work

We have presented 3D-EPUG-OVERLAY, an exact and parallel algorithm for computing the intersection of 3D models represented by triangulated meshes. 3D-EPUG-OVERLAY uses arbitrary precision rational numbers to store all the geometric coordinates and perform computation, and so is roundoff error free.

Even though the current implementation of 3D-EPUG-OVERLAY does not treat special cases, preliminary experiments have indicated that 3D-EPUG-OVERLAY can successfully intersect some big meshes available in public repositories.

Next, we intend to implement a symbolic-perturbation scheme on 3D-EPUG-OVERLAY to ensure that all the special cases are properly handled. Furthermore, the optimization techniques that have been so successful in our previous works will be also applied to 3D-EPUG-OVERLAY.

7. Acknowledgement

This research was partially supported by CNPq, CAPES (Ciência sem Fronteiras), FAPEMIG and NSF grant IIS-1117277.

References

AIM@SHAPE (2016). AIM@SHAPE-VISIONAIR Shape Repository. http://visionair.ge.imati.cnr.it// (accessed on Sep-2016).

Akman, V., Franklin, W. R., Kankanhalli, M., and Narayanaswami, C. (1989). Geometric computing and the uniform grid data technique. *Comput. Aided Design*, 21(7):410–420.

Belussi, A., Migliorini, S., Negri, M., and Pelagatti, G. (2016). Snap rounding with restore: An algorithm for producing robust geometric datasets. *ACM Trans. Spatial Algorithms Syst.*, 2(1):1:1–1:36.

Bernstein, G. and Fussell, D. (2009). Fast, exact, linear booleans. *Eurographics Symposium on Geometry Processing*, 28(5):1269–1278.

CGAL (2016). CGAL, Computational Geometry Algorithms Library. http://www.cgal.org (accessed on Sep-2016).

Edelsbrunner, H. and Mücke, E. P. (1990). Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics (TOG)*, 9(1):66–104.

Feito, F., Ogayar, C., Segura, R., and Rivero, M. (2013). Fast and accurate evaluation of regularized boolean operations on triangulated solids. *Computer-Aided Design*, 45(3):705 – 716.

Franklin, W. R., Chandrasekhar, N., Kankanhalli, M., Seshan, M., and Akman, V. (1988). Efficiency of uniform grids for intersection detection on serial and parallel machines. In Magnenat-Thalmann, N. and Thalmann, D., editors, *New Trends in Computer Graphics (Proc. Computer Graphics International'88)*, pages 288–297. Springer-Verlag.

Franklin, W. R., Sun, D., Zhou, M.-C., and Wu, P. Y. (1989). Uniform grids: A technique for intersection detection on serial and parallel machines. In *Proceedings of Auto Carto 9*, pages 100–109, Baltimore, Maryland.

Hachenberger, P., Kettner, L., and Mehlhorn, K. (2007). Boolean operations on 3d selective nef complexes: Data structure, algorithms, optimized implementation and experiments. *Computational Geometry*, 38(1):64–99.

Hershberger, J. (2013). Stable snap rounding. *Computational Geometry*, 46(4):403–416.

Hobby, J. D. (1999). Practical segment intersection with finite precision output. *Comput. Geom.*, 13(4):199–214.

Hoffman, C. M. (1989). The problems of accuracy and robustness in geometric computation. *Computer*, 22(3):31–40.

Kettner, L., Mehlhorn, K., Pion, S., Schirra, S., and Yap, C. (2008). Classroom examples of robustness problems in geometric computations. *Comput. Geom. Theory Appl.*, 40(1):61–78.

Leconte, C., Barki, H., and Dupont, F. (2010). Exact and efficient booleans for polyhedra. Citeseer.

Li, C., Pion, S., and Yap, C.-K. (2005). Recent progress in exact geometric computation. *The Journal of Logic and Algebraic Programming*, pages 85–111.

Liu, J., Chen, Y. Q., Maisog, J. M., and Luta, G. (2010). A new point containment test algorithm based on preprocessing and determining triangles. *Comput. Aided Des.*, 42(12):1143–1150.

Magalhães, S. V., Andrade, M. V., Franklin, W. R., and Li, W. (2016). Pinmesh - fast and exact 3d point location queries using a uniform grid. *Computers & Graphics*, 58:1 – 11. Shape Modeling International 2016.

Magalhães, S. V. G., Andrade, M. V. A. A., Franklin, W. R., and Li, W. (2015). Fast exact parallel map overlay using a two-level uniform grid. In *Proc. of the 4th ACM Bigspatial*, BigSpatial '15, New York, NY, USA. ACM.

Möller, T. (1997). A fast triangle-triangle intersection test. *Journal of graphics tools*, 2(2):25–30.

Shewchuk, J. R. (1996). Adaptive precision floating-point arithmetic and fast robust geometric predicates. *Discrete & Computational Geometry*, 18:305–363.

Yanbing, W., Lixin, W., Wenzhong, S., and Xiaomeng, L. (2007). On 3d gis spatial modeling. In *Proceedings of the ISPRS Workshop on Updating Geo-spatial Databases with Imagery and the 5th ISPRS Workshop on DMGISs, Urumchi, Xinjiang, China*, pages 237–240. Citeseer.

Multiple Aspect Trajectory Data Analysis: Research Challenges and Opportunities

Carlos Andres Ferrero[1,2], **Luis Otavio Alvares**[1], **Vania Bogorny**[1]

[1]Programa de Pós-Graduação em Ciência da Computação
Departamento de Informática e Estatística (INE)
Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brasil.

[2]Instituto Federal de Santa Catarina (IFSC), Lages, SC, Brasil.

andres.ferrero@ifsc.edu.br, vania.bogorny@ufsc.br

Abstract. *Trajectory data analysis and mining has been largely studied in the past years. Although trajectories are multidimensional data, that have space, time, and semantic information, only a few works on the literature have considered all three dimensions. Indeed, we claim in this paper that not only the three dimensions must be considered in trajectory data analysis, but that trajectories can be represented from different points of view, that we call* multiple aspect representation. *Existing works, in general, are limited to one single trajectory representation, what narrows the discovery of several types of interesting patterns. In this vision paper we show that there is a need for a change of paradigm in trajectory data analysis, and present new research challenges in movement analysis.*

1. Introduction and Motivation

We are living the era of movement tracking and mining, where huge volumes of data about our daily lives are being collected and stored in several sources and formats. Examples include our smartphones, from which Google and Apple collect all details about our daily routines, including the places we visit and the time we stay there. Facebook captures our location, stores our friendship relationships, as well as our thoughts and opinions about things and people. More recently, the Pokémon GO emerges to capture not only our movement, but photos of places we visit when capturing Pokémons, what certifies with a high accuracy where we are. In summary, when an individual is moving, the application collects his/her location over time, in the form of sequential spatio-temporal points, called *raw trajectories*, as shown in Figure 1 (left). A raw trajectory is a complex data type, which has *space* and *time* information associated with each trajectory point.

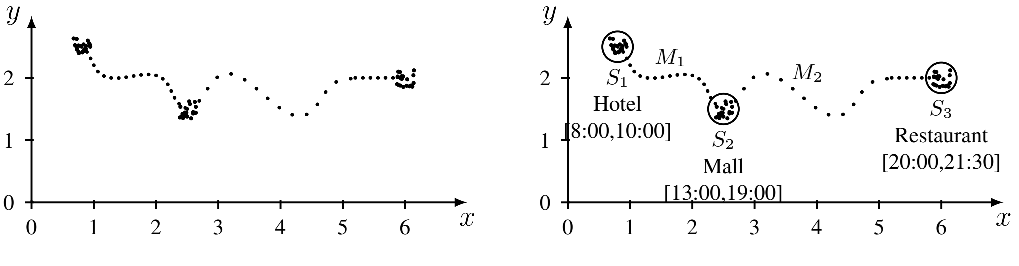


Figure 1. (left) Example of a raw trajectory T, and (right) the corresponding semantic trajectory.

Raw trajectory is the most simple trajectory representation. Since 2007, because of the explosion of social networks (e.g. Foursquare, Facebook) and navigation applications (e.g. Google Maps, Waze), raw trajectories are being enriched with more information, such as the name of the place visited by an object, called Point of Interest (POI), and the amount of time the individual stayed at each POI. With more information associated to raw trajectories, a new representation is defined, the *semantic trajectories* [Spaccapietra et al. 2008, Alvares et al. 2007, Parent et al. 2013]. In other words, the movement is a sequence of stops (visited places) and moves (spatio-temporal points between stops) [Spaccapietra et al. 2008]. An example is shown in Figure 1 (right). With the new representation of trajectories, movement becomes a more complex data type, having now more dimensions to be considered: *space, time,* and *semantics*.

More recently, in 2014, Bogorny [Bogorny et al. 2014] proposed a new trajectory representation model, where the same trajectory can be represented according to several aspects. In other words, the same trajectory can have *multiple aspect representations*. For instance, a raw trajectory can be represented as a sequence of stops and moves, a sequence of transportation means used during the movement, the sequence of weather conditions, the sequence of activities performed during the movement, and so on. In another recent work [Noël et al. 2015], the authors propose a semantic trajectory data model composed of multiple aspects, which are all different points of view from which a trajectory can be observed. They apply the model for life trajectories considering several high level aspects as residential, professional, and familial, where each aspect contains specific information. Although the works of [Bogorny et al. 2014] and [Noël et al. 2015] address the need for multiple aspects, they are limited to a model for multiple aspect representation, and do not present the challenges related to multiple aspect trajectory data analysis and mining.

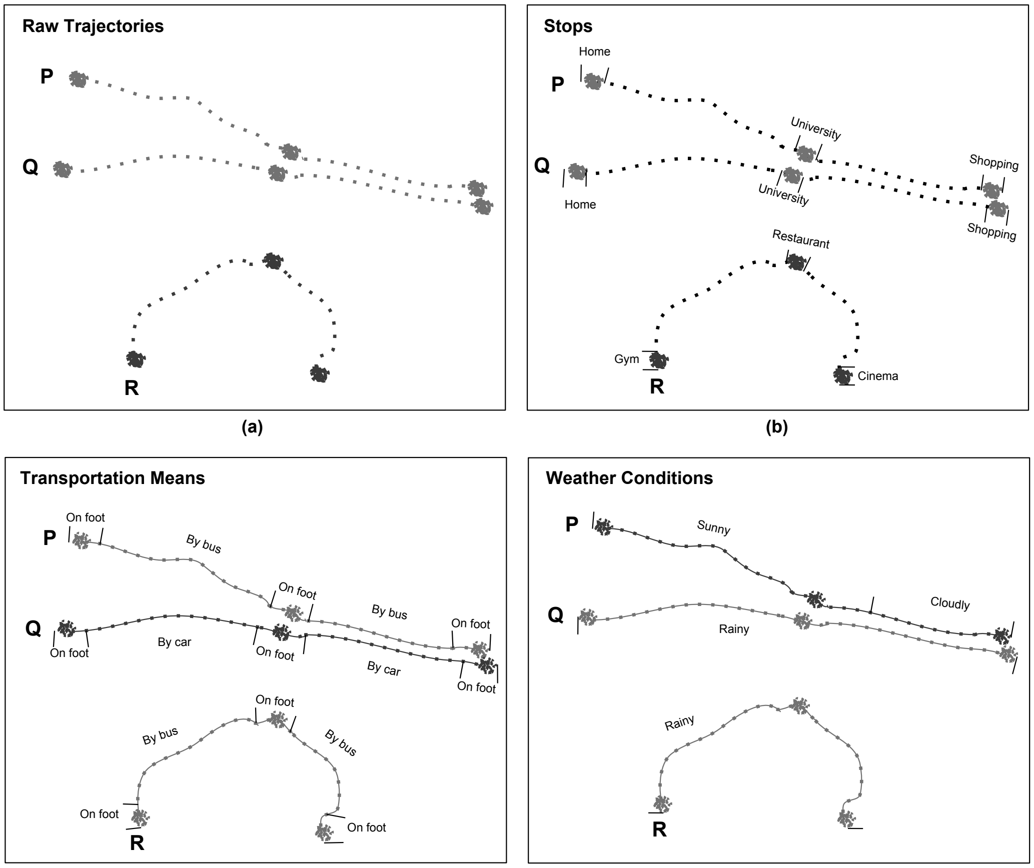
To the best of our knowledge, there is no work in the literature on trajectory data mining and similarity analysis that considers different representations for a single trajectory. So far, existing works consider either raw trajectories or semantic trajectories in the form of stops and moves. Most of the existing works do not even consider the three dimensions of space, time and semantics. In similarity analysis, for instance, only the work of [Furtado et al. 2016] considers all three dimensions, while works as [Vlachos et al. 2002, Chen et al. 2005, Ying et al. 2010, Pelekis et al. 2012, Liu and Schneider 2012] consider only two dimensions as space and time, or space and semantics. In trajectory clustering, for instance, only the single raw trajectory representation has been considered [Lee et al. 2007, Abul et al. 2010, Hung et al. 2015] and a few works are starting semantic trajectory mining as [Pelekis et al. 2011, Lv et al. 2013, Xiao et al. 2014, Ying et al. 2014, Wu et al. 2015, Cai et al. 2016].

By considering only one aspect in movement analysis and mining, several types of interesting patterns cannot be discovered. For instance, how do individuals move when it is raining, by car, bike, or public transportation? Do groups of friends visit specific places only by bus and with good weather? How do weather conditions affect traffic jams? Which is the transportation pattern at a beach town on a rainy weekend and a sunny weekend? In this paper we analyze some existing works and show their limitations related to multiple aspects trajectory representation, and that there is a need for a change of paradigm on trajectory analysis and mining, which is not limited to only a few trajectory attributes.

The rest of this paper is organized as follows: Section 2 presents a multiple aspect trajectory data analysis, that introduces the similarity analysis on multiple aspects. It also presents a comparative study of similarity measures proposed in the literature and their limitations, and the need of new proposals to multiple aspect trajectory data mining. In Section 3 we present a discussion of our vision about the future in trajectory data analysis methods.

2. Multiple Aspects Trajectory Data Analysis

Let us consider three trajectories, P, Q, and R, shown in Figure 2. These trajectories can be represented as, for instance, four different aspects: as raw trajectories, in Figure 2(a); as stops and moves (Figure 2(b)), where the labeled parts are the stops; as transportation means (Figure 2(c)); and according to weather conditions (Figure 2(d)).



(a)

(b)

(c)

(d)

Figure 2. Multiple trajectory representation.

By considering every aspect separately, and considering only the *space* and *semantic* dimensions, excluding time for simplification, trajectories would be analyzed as follows:

Raw Trajectories: from this aspect (Figure 2(a)), trajectories P and Q are spatially closer than P and R or Q and R. For any spatial point of P, the closest spatial point is always from Q, not from R, and vice versa. So P and Q are the most similar.

Stops: from this aspect (Figure 2(b)), trajectories P and Q visit the same POI types $Home$, $University$, and $Shopping$, and in the same order. On the other hand, trajectory R visits different POI types (Gym, $Restaurant$, and $Cinema$). So from the Stops and moves aspect trajectories P and Q are the most similar.

Transportation Means: from this aspect, P moves $On\ \ foot$ and $By\ \ bus$ while Q moves $On\ foot$ and $By\ car$. Trajectory R uses exactly the same transportation means of P $On\ foot$ and $By\ bus$, and in the same sequence. So in this aspect, trajectories P and R are the most similar.

Weather Conditions: from this aspect (Figure 2(d)), P and Q occur at different weather conditions: P under $Sunny$ and $Cloudy$, and Q under $Rainy$. On the other hand, trajectories Q and R occur under the same weather condition, $Rainy$. This is possible because P and Q occur at different days. Then, Q and R are the most similar.

Figure 3 summarizes the similarity analysis for every aspect for the trajectories P, Q, and R. For instance, P and Q have higher similarity for the aspects *Raw Trajectory* and *Stops* than for the other two aspects. Trajectories P and R show higher score for the aspect *Transportation Means*, while Q and R are more similar on *Weather Conditions*.

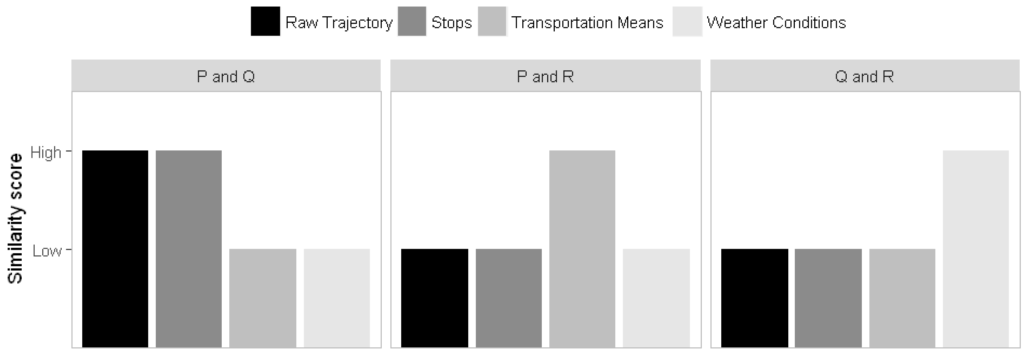


Figure 3. Similarity scores between pairs of trajectories, considering the aspects: raw data, stops, transportation means, and weather conditions.

In the following sections we analyze the multiple aspect trajectory data representation from two perspectives: similarity analysis and trajectory data mining.

2.1. Multiple Aspect Similarity Analysis

In this section we compare several trajectory similarity measures, applied to the trajectories of Figure 2, which mainly consider two aspects: raw trajectories and stops. Table 1 shows the results. This table shows the aspect considered by each approach and which dimensions are taken into account, where Ti represents time, Sp represents space and Se represents semantics.

The measures DTW [Vlachos et al. 2002], LCSS [Vlachos et al. 2002], and EDR [Chen et al. 2005] consider only raw trajectories, specifically the spatial dimension. DTW returns the distance between points, so the closest trajectories are P and Q. LCSS

Table 1. Similarity Values of existing approaches.

		Raw Data		**Stops**			**Transp. Means**			**Weather Cond.**			
#	**Measure**	Ti	Sp	Ti	Sp	Se	Ti	Sp	Se	Ti	Sp	Se	The Most Similar Trajectories
1	DTW Distance [Vlachos et al. 2002]		✓										P and Q
2	LCSS Ratio [Vlachos et al. 2002]		✓										P and Q
3	EDR Ratio [Chen et al. 2005]		✓										P and Q
4	MSTP [Ying et al. 2010]					✓							P and Q
5	[Liu and Schneider 2012]				✓	✓							P and Q
6	[Lv et al. 2013]			✓		✓							P and Q
7	MTM [Xiao et al. 2014]			✓		✓							P and Q
8	MSM [Furtado et al. 2016]			✓	✓	✓							P and Q
	Multiaspect Similarity Measure	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	Raw Data	✓	✓										P and Q
	Stops			✓	✓	✓							P and Q
	Transp. Means						✓	✓	✓				P and R
	Weather Cond.									✓	✓	✓	Q and R

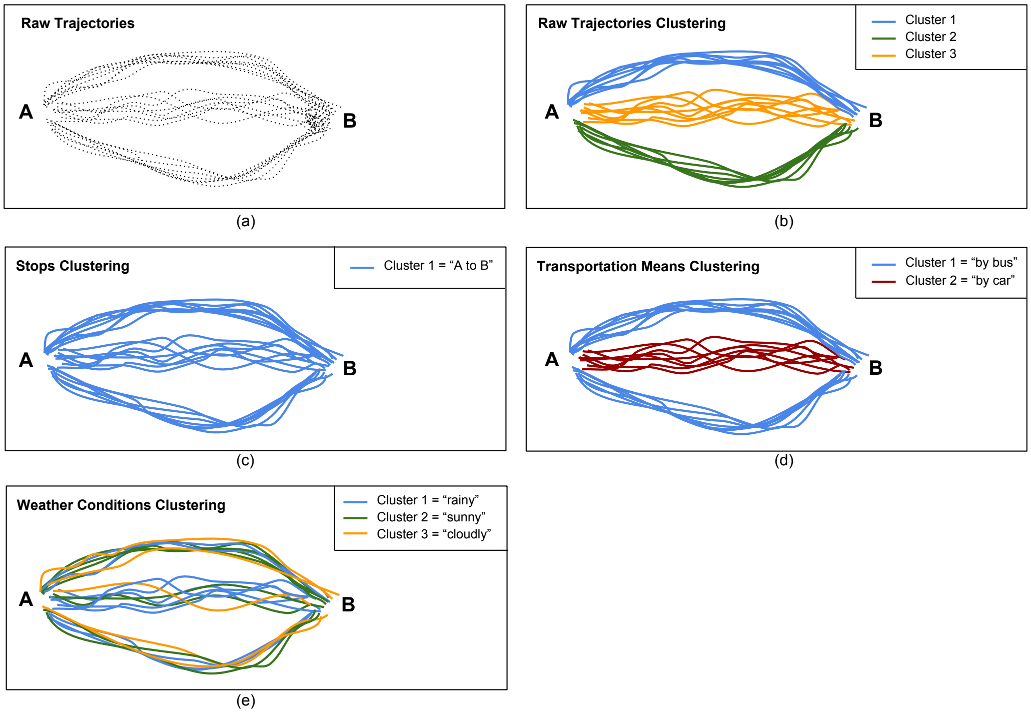
also considers trajectories P and Q as the most similar, because this measure uses the sequence of nearest spatial points between two trajectories to calculate their similarity. EDR also considers trajectories P and Q as the most similar, because the cost to transform P into Q is lower than the cost to transform P in R or Q in R. The similarity measures proposed by Ying [Ying et al. 2010], Liu [Liu and Schneider 2012], Lv [Lv et al. 2013], Xiao [Xiao et al. 2014], and the MSM [Furtado et al. 2016] were developed for the stops and moves representation, so they all consider the semantic dimension. Therefore, all these measures return a high similarity for P and Q, and low similarity for P and R, and Q and R.

As can be seen in the last rows of Table 1, the multiple aspect similarity measures should first consider all three dimensions: space, time, and semantics. By considering all dimensions and more aspects, not only trajectories P and Q would be similar, but also P and R and Q and R, depending on the aspect(s) considered. From a similarity analysis point of view, we have two problems: first, the trajectories have several *aspects* to be considered, such as raw data, stops, transportation means, activities, weather conditions, and others; second, existing measures consider only one representation, either raw trajectories or stops and moves. In the following section we show how the multiple aspect trajectory data analysis can lead to new types of trajectory patterns that cannot be detected so far by existing data mining methods.

2.2. Multiple Aspect Trajectory Data Mining

In this section we show two simple examples of interesting patterns that can only be discovered when considering trajectories over multiple aspects, not simply stops and moves or raw trajectories.

Figure 4(a) shows several raw trajectories, where the moving objects are traveling from region A to region B. By applying a clustering technique over these raw trajectories we can obtain three clusters, as shown in Figure 4(b), since most works as [Lee et al. 2007, Lee et al. 2008, de Vries and van Someren 2010, Liu et al. 2010, Liu et al. 2013, Chen et al. 2014] consider proximity in *space*. If we consider the *stops* representation, as all trajectories have the same two stops, first A and after B, probably they will be in the same cluster, as shown in Figure 4(c).



(a)

(b)

(c)

(d)

(e)

Figure 4. (a) raw trajectories between A and B, (b) clusters of raw trajectories, (c) clustering under stops representation, (d) clustering by transportation means, and (e) clustering by weather conditions.

Now let us consider the aspects that have not been considered so far, as transportation means. With this representation we can obtain two clusters, as shown in Figure 4(d), one cluster at the center that represents objects moving *by car*, and the second cluster (in blue) objects moving *by bus*, indicating that there are two different bus lines to move from region A to region B. Another possible clustering is the one shown in Figure 4(e), where the trajectories were grouped by weather conditions. Notice that there are three clusters, corresponding to *rainy*, *sunny*, and *cloudy*. Existing clustering methods would not be able to detect these clusters, since most of them consider the spatial distance as similarity measure.

Considering the examples in Figure 4, by mixing the different representations, different clusters could be generated. Besides, we could also include many other variables in the clustering analysis, as the duration time and the average speed of each transportation means, or the duration and the length of each weather condition, and so on.

For other data mining techniques, as classification, several aspects besides raw trajectories and stops are very important. And more interesting is the combination of different aspects in the same mining step. For instance, if we want to distinguish two classes of transportation means: *on foot* and *not on foot*, a classification algorithm could find patterns as:

- $if\ (temperature\ >\ 35\ degrees)\ then\ class = "not\ on\ foot"$
- $if\ (weather = "rainy"\ and\ length > 1000m)\ then\ class = "not\ on\ foot"$
- $if\ (weather = "sunny"\ and\ temperature < 30\ degrees\ and$ $length < 5000m)\ then\ class = "on\ foot"$

The weather conditions are also very important in the analysis of traffic jams since in most cities they are more frequent and stronger when it is raining. To the best of our knowledge, existing trajectory similarity and mining methods are limited to a few trajectory attributes, and one single aspect. None of the existing works have considered multiple aspects. In the following section we present a discussion about the challenges behind considering multiple aspects together for similarity analysis and trajectory data mining.

3. Research Challenges and Opportunities

In this section we present some major challenges on multiple trajectory representation analysis and how they lead to new research opportunities. Tables 2, 3 and 4 give an overview of the complexity related to multiple aspect trajectory data analysis. Every table shows just a few examples of different features that can be extracted from trajectories over one single aspect. For instance, in Table 2 (the stops representation), information as stop duration, traveling time between stops, stop name, route followed between stops, etc, can be extracted. The complexity relies on the amount of information that can be obtained over each dimension: space, time, semantics, and the combination of dimensions, as for instance, the name and location of the stops with duration above 1 hour. This example refers to one aspect and three dimensions.

Now consider the combination of multidimensional features of three aspects, such as (i) the *average speed* of the moving object when traveling *by car* when it is *raining*; (ii) average traveled distance by car under rain; (iii) average traveled time by bus under rain; (iv) total traveled distance on foot in a sunny day.

As mentioned previously, existing works in the literature do not support multiple aspect trajectory data analysis. But one may ask why not simply include all aspects information into a unique trajectory representation? For instance, considering the *stops representation* one could argue that it is simple to load (enrich) the trajectory with all aspect information such as weather conditions, transportation means, activities, etc. The problem is not so simple. Let us suppose that during one stop the object is moving on foot

Table 2. Examples of information to be extracted for the aspect of stops.

Information	Dimensions
Stop duration	Time
Traveled time between the stops	Time
Traveled distance between the stops	Space
Geographical position of the stops	Space
Visited POI types	Semantics
Amount of time at each POI type	Time and Semantics
Name and location of the stops with duration greater than 1 hour	Time, Space and Semantics

Table 3. Examples of information to be extracted for the aspect of transportation means.

Information	Dimensions
Duration of each transportation means	Time
Traveled distance of each transportation means	Space
Type of transportation means	Semantics
Total duration of each transportation means	Time and Semantics
Distance traveled on foot	Space and Semantics
Average speed by car	Time, Space and Semantics

Table 4. Examples of information to be extracted for the aspect of weather conditions.

Information	Dimensions
Travel duration under rain	Time and semantics
Traveled distance under rain	Space and Semantics
Average speed under sun	Time, Space, and Semantics

and the weather condition changes from $Sunny$ to $Rainy$. And in another aspect as *Activity* the object changed its activity from $drinking\ coffee$ to $teaching$ when the weather is $Rainy$. All these things happening at one single stop with label *University*. It would be very hard to correctly split and annotate the stop into two weather conditions, each one having a different start and end time, splitting it in different activities. The same stop would have several semantic labels for weather, transportation means, stop name, activity name, etc, and several time intervals associated to each semantic label, as for instance, the start time and end time of a stop, duration of a transportation means, distance traveled by one transportation means, etc., and probably different space information as well.

There are six main challenges in multiple aspect trajectory data analysis:

Multiple Aspect Representation. The first point is how to compute different aspect information since it involves heterogeneous data sources. For instance, climate conditions can be taken from the web, stops labeling from Open Street Maps, transportation means from raw trajectories or manual annotation, activity inference from social networks, etc. While some aspects may be easy to represent and to obtain as weather conditions, others are more complex, as transportation means and activity inference, which by their own are still challenging open research fields. Each aspect has space, time, and semantic information as well as their combinations. The second point is how to represent trajectories with all this information.

Feature Extraction. New efficient algorithms must be developed for trajectory segmentation and feature extraction, considering each aspect and their combinations. The algorithms should handle different aspects and different segmentation forms, avoiding trajectory feature recomputation, mainly when dealing with large trajectory datasets.

Data Storage. New efficient algorithms are needed to store, besides raw trajectories, the information of different aspects and the extracted features. New indexes and data structures have to be proposed for efficiently storing and querying these complex data.

Similarity Analysis and Data Mining. Similarity measures and data mining algorithms do not yet consider all three dimensions (space, time and semantics) or are still limited to raw trajectories or stops and moves representation. There is a need for similarity measures that are not limited to a predefined set of variables, and which do not only give a similarity score to express how similar two trajectories are. Note that a low global similarity can hide a strong similarity for a specific aspect. New data mining algorithms are needed to find more complex patterns than a group of objects moving together in space and time or that visit similar places. Algorithms should infer if the objects moving together have a relationship, how much moving objects are aware of each other in the group, and how the movement of a single individual influences the group.

Visualization. Tools to visualize trajectories from different aspects, and their information, is crucial. It is important to note that each aspect has, in general, time, space and semantics dimensions, and the relationship between these dimensions should be treated by visualization approaches. In addition, there is a need of visualization techniques to show the patterns found in data mining tasks, to make them easier to evaluate and validate.

Privacy Protection. Using more information about the moving objects is a crucial problem related to the privacy preserving of users and protecting their sensitive information. Multiple aspects reveal more details about users, so privacy preserving data mining methods become more challenging.

We believe that new methods and tools are needed to simultaneously process multiple aspect information. This will lead to a new era in movement data processing and to the discovery of more complex and interesting patterns which have not been addressed so far. We believe that new similarity measures will need to output not only a single number that represents the similarity degree of a pair of objects considering two or three features, but in which aspect the movement of individuals is more or less similar. These measures will allow answering questions as: (i) In which aspect two trajectories T1 and T2 are more similar? (ii) In which aspect two trajectories are less similar? (iii) In which aspect two trajectories T1 and T2 have a similarity degree higher than δ? (iv) Which trajectories are more similar in a given aspect α? (v) Which trajectories are more similar considering all aspects?

In summary, we strongly believe that multiple aspect representation is a big issue in future trajectory data analysis and a challenge for researchers to develop new concepts and methods in this promising area.

Acknowledgment

This work was supported by CNPq, UFSC and IFSC-Lages.

References

Abul, O., Bonchi, F., and Nanni, M. (2010). Anonymization of moving objects databases by clustering and perturbation. *Information Systems*, 35(8):884–910.

Alvares, L. O., Bogorny, V., Kuijpers, B., de Macedo, J. A. F., Moelans, B., and Vaisman, A. (2007). A model for enriching trajectories with semantic geographical information. In *Proceedings of the 15th ACM SIGSPATIAL*, pages 22:1–22:8, New York, NY, USA. ACM.

Bogorny, V., Renso, C., de Aquino, A. R., de Lucca Siqueira, F., and Alvares, L. O. (2014). CONSTAnT - A Conceptual Data Model for Semantic Trajectories of Moving Objects. *Transactions in GIS*, 18(1):66–88.

Cai, G., Lee, K., and Lee, I. (2016). Discovering common semantic trajectories from geo-tagged social media. In *Proceedings of the 29th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems: Trends in Applied Knowledge-Based Systems and Data Science*, pages 320–332, Cham. Springer International Publishing.

Chen, L., Özsu, M. T., and Oria, V. (2005). Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data (SIGMOD '05)*, pages 491–502, New York, NY, USA. ACM.

Chen, Y., Shen, H., and Tian, H. (2014). Clustering subtrajectories of moving objects based on a distance metric with multi-dimensional weights. In *Proceedings of the Sixth International Symposium on Parallel Architectures, Algorithms and Programming*, pages 203–208.

de Vries, G. and van Someren, M. (2010). *Clustering Vessel Trajectories with Alignment Kernels under Trajectory Compression*, pages 296–311. Springer Berlin Heidelberg, Berlin, Heidelberg.

Furtado, A. S., Kopanaki, D., Alvares, L. O., and Bogorny, V. (2016). Multidimensional similarity measuring for semantic trajectories. *Transactions in GIS*, 20:280–298.

Hung, C.-C., Peng, W.-C., and Lee, W.-C. (2015). Clustering and aggregating clues of trajectories for mining trajectory patterns and routes. *The VLDB Journal*, 24(2):169–192.

Lee, J.-G., Han, J., Li, X., and Gonzalez, H. (2008). Traclass: Trajectory classification using hierarchical region-based and trajectory-based clustering. *VLDB*, 1(1):1081–1094.

Lee, J.-G., Han, J., and Whang, K.-Y. (2007). Trajectory clustering: A partition-and-group framework. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, SIGMOD '07, pages 593–604, New York, NY, USA. ACM.

Liu, H. and Schneider, M. (2012). Similarity measurement of moving object trajectories. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on GeoStreaming (IWGS)*, pages 19–22, New York, NY, USA. ACM.

Liu, S., Liu, Y., Ni, L. M., Fan, J., and Li, M. (2010). Towards mobility-based clustering. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 919–928, New York, NY, USA. ACM.

Liu, S., Wang, S., Jayarajah, K., Misra, A., and Krishnan, R. (2013). Todmis: Mining communities from trajectories. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management (CIKM)*, pages 2109–2118, New York, NY, USA. ACM.

Lv, M., Chen, L., and Chen, G. (2013). Mining user similarity based on routine activities. *Information Sciences*, 236:17–32.

Noël, D., Villanova-Oliver, M., Gensel, J., and Le Quéau, P. (2015). Modeling semantic trajectories including multiple viewpoints and explanatory factors: Application to life trajectories. In *Proceedings of the 1st International ACM SIGSPATIAL Workshop on Smart Cities and Urban Analytics*, UrbanGIS'15, pages 107–113, New York, NY, USA. ACM.

Parent, C., Spaccapietra, S., Renso, C., Andrienko, G. L., Andrienko, N. V., Bogorny, V., Damiani, M. L., Gkoulalas-Divanis, A., de Macêdo, J. A. F., Pelekis, N., and Yan, Z. (2013). Semantic trajectories modeling and analysis. *ACM Computing Surveys (CSUR)*, 45(4):42.

Pelekis, N., Andrienko, G., Andrienko, N., Kopanakis, I., Marketos, G., and Theodoridis, Y. (2012). Visually exploring movement data via similarity-based analysis. *Journal of Intelligent Information Systems*, 38(2):343–391.

Pelekis, N., Kopanakis, I., Kotsifakos, E. E., Frentzos, E., and Theodoridis, Y. (2011). Clustering uncertain trajectories. *Knowledge Information Systems*, 28(1):117–147.

Spaccapietra, S., Parent, C., Damiani, M. L., de Macedo, J. A., Porto, F., and Vangenot, C. (2008). A conceptual view on trajectories. *Data and Knowledge Engineering*, 65(1):126–146.

Vlachos, M., Kollios, G., and Gunopulos, D. (2002). Discovering similar multidimensional trajectories. In *Proceedings of the 18th International Conference on Data Engineering*, pages 673–684, San Jose, CA, USA. IEEE.

Wu, X., Zhu, Y., Xiong, S., Peng, Y., and Peng, Z. (2015). A new similarity measure between semantic trajectories based on road networks. In *Proceedings of the 17th Asia-Pacific Web Conference*, pages 522–535, Guangzhou, China. Springer International Publishing.

Xiao, X., Zheng, Y., Luo, Q., and Xie, X. (2014). Inferring social ties between users with human location history. *Journal of Ambient Intelligence and Humanized Computing*, 5(1):3–19.

Ying, J. J.-C., Lee, W.-C., and Tseng, V. S. (2014). Mining geographic-temporal-semantic patterns in trajectories for location prediction. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(1):2:1–2:33.

Ying, J. J.-C., Lu, E. H.-C., Lee, W.-C., Weng, T.-C., and Tseng, V. S. (2010). Mining user similarity from semantic trajectories. In *Proceedings of the 2nd ACM SIGSPATIAL*, pages 19–26, New York, NY, USA.

TOWARDS ACTIVITY RECOGNITION IN MOVING OBJECT TRAJECTORIES FROM TWITTER DATA

Marco Aurelio Beber [1], **Carlos Andres Ferrero** [2], **Renato Fileto** [1], **Vania Bogorny** [1]

[1]Programa de Pós-Graduação em Ciência da Computação (PPGCC)
Universidade Federal de Santa Catarina (UFSC), Florianópolis, SC, Brasil

[2]Coordenação de Informática e Ciência da Computação
Instituto Federal de Santa Catarina (IFSC), Lages, SC, Brasil

marco.beber@posgrad.ufsc.br, andres.ferrero@ifsc.edu.br, {r.fileto,vania.bogorny}@ufsc.br

Abstract. *The knowledge about people daily activities is of great value for several application domains. On the one hand, the activity recognition in trajectories has not been deeply investigated. On the other hand, social media data such as tweets can be rich in information about where people go and what they do. We strongly believe that the integration of trajectory data and social media can reveal the activities performed by individuals in daily life. In this paper we propose a new method to infer moving object activities from their trajectories, using knowledge extracted from Twitter data. We evaluate the proposed approach with two datasets and show that it outperforms current works.*

1. Introduction

The knowledge about which activities people do at certain Points Of Interest (POIs) can be of great value for several applications. For instance, recommendation systems could infer user activities from visited locations obtained from their Google account [1] and suggest new places based on the inferred activities. Architects and city planners could project better public spaces, such as parks, based on how people perform activities. Augmented reality games, such as Pokemon Go [2], could improve the security of the players by warning them about suspicious activities at POIs to prevent cases of robbery.

We are living the era of big data, where individuals are constantly leaving traces of their movements and their activities. Even though we are not fully aware of it, we are tracked everyday. Our spatiotemporal traces can be delineated as *moving object trajectories*. A raw trajectory is a temporally ordered sequence of geographical coordinates associated with a timestamp, which does not present explicit semantics. A semantic trajectory [Spaccapietra et al. 2008] is represented as a sequence of stops and moves, where stops are the places visited by the object. Bogorny in [Bogorny et al. 2014] extends the concept of semantic trajectory by considering other important aspects such as activities and goals of trajectories. Several recent works address semantic trajectory data analysis [de Aquino et al. 2013, Ying et al. 2014, de Alencar et al. 2015, Furtado et al. 2016], but only a few have focused on activity recognition. While it might be easy to discover visited places in many situations, determining the activities performed at these places is

[1] https://accounts.google.com/

[2] http://www.pokemongo.com

not a trivial task. There is no unique association between each POI (or POI type) and the activities that can be performed at that POI. Several activities may be performed in the same POI (or POI type). There is a wide range of possibilities that vary in number and nature according with the POI type. For instance, at a shopping mall, one could be eating, purchasing, working, socializing, watching a movie, or at a commercial building one could be drinking a coffee, working, visiting, purchasing and at a company one could be working or visiting.

We strongly believe that the main limitation of existing works for activity recognition from GPS trajectory data, such as the works [Weerkamp et al. 2012, Furletti et al. 2013, Njoo et al. 2015] is the assignment of only one activity at a POI, and relying on specialists to manually label POIs (or POI types) with activities that can happen at these POIs (or POI types). Another limitation is the dependence of an annotated trajectory dataset to generate a classification model. In this paper we overcome these problems by proposing a novel solution to recognize activities in moving object trajectories based on tweets sent from the visited POIs. First, we assume that more than one activity can be performed at each POI (e.g. one can study, socialize and eat at a University). Second, we enrich Foursquare POI types with statistics about the activities observed in tweets sent from the respective POI types. Third, we propose a matching process to infer activities based on the similarity between the trajectory and the POI type profiles extracted from Twitter. To the best of our knowledge, our proposal is the first to extract knowledge from Twitter data to infer activities in moving object trajectories. In summary, we make the following contributions: (i) we build a knowledge base with POI type profiles based on activities observed in tweets sent from each POI type; (ii) we propose the algorithm T-Activity to infer activities in trajectory data, by matching the POIs visited by trajectories with POI type profiles in the knowledge base; (iii) we evaluate the proposed approach with real trajectories and census data to evaluate our method in real case scenarios.

The rest of this paper is organized as follows. Section 2 discusses related work. Section 3 presents the main definitions. Section 4 describes our proposal for activity recognition. Section 5 reports the experimental evaluation, and finally, Section 6 presents our conclusions.

2. Related Work

There are different works in the literature related to human activity recognition using different types of data, such as social media and GPS trajectories. The works based on social media focus on text classification, and extract features from text and POIs to build classifiers for activity recognition. For instance, Liu in [Liu et al. 2012] builds a classifier over tweets in order to predict the POI type of tweets linked to Foursquare. Although this work does not recognize activities, it recognizes POI types that can be related to different activities. Weerkamp in [Weerkamp et al. 2012] proposed an approach to predict the popular activities that will happen in a future time window, such as tonight, tomorrow, and next week, by using a future time-window and keywords related to activities. Zhu in [Zhu et al. 2016] builds a multi-label classifier using tweets manually annotated with activities in order to predict up to three activities. To build the classifier, it considers the tweet text, the tweet posting time, the POI type from Foursquare and POI name from Foursquare.

On the other hand, only a few works try to recognize activities on GPS trajectories. Moreno in [Moreno et al. 2010] proposed an algorithm that given a set of stops of a trajectory, uses a predefined set of rules that considers the minimum time and maximum speed to infer the goal of movement. However, the set of rules has to be defined by a specialist, and the matching process is based on movement aspects of the goal, such as its minimum time and its maximum speed. Therefore, it ignores important aspects such as the place and the time of the goal. Our work, instead, focuses on recognizing activities, and we do not depend on a specialist, since we extract the knowledge from Twitter in order to build a knowledge base that describes the activities that can be performed at a POI. Furletti in [Furletti et al. 2013] proposed a method for activity recognition where a set of activities is manually defined for POI types, and given a trajectory, it finds the stops and matches the POI type of the stop with the manually defined activities. Our work on the other hand considers that multiple activities can happen at each POI and computes the similarity of the trajectory and the activities in the knowledge base in order to infer activities. Reumers in [Reumers et al. 2013] uses a dataset of semantic trajectories annotated with activities and proposes to infer activities using a decision-tree based model. To build the tree, it uses the start time, duration and activity of each stop, but does not consider the place where the activity happened and depends on the annotated trajectory data to build the model. Kim in [Kim et al. 2014] builds a classification model to recognize groups of activities, as for instance, home, work and transportation. It uses spatial regions annotated with the frequency of time, duration and frequency of the activities. However, it does not infer activities, just groups of activities, and it also depends on the annotated trajectory data to build the model. Njoo in [Njoo et al. 2015] also manually defines an activity for each POI type, and using a dataset of semantic trajectories annotated with activities it builds classifiers with trajectories from the same moving object, to represent the routine of a moving object. However, if the moving object goes to a place that was not previously seen, it matches the POI type with the manually defined activity.

Overall, our work is different from the previous approaches since we do not depend on a specialist to build the knowledge base, instead, we extract the knowledge from Twitter data. We also consider that multiple activities can take place at each POI and we propose an algorithm to match trajectories with our knowledge base to infer activities.

3. Main Definitions

There are several definitions of semantic trajectories in the literature, such as [Bogorny et al. 2014] and [Spaccapietra et al. 2008]. In this work we adapt the definition of semantic trajectory defined by Spaccapietra, considering a semantic trajectory as a set of stops and the POI type of the stops. Definition 1 shows our formal definition of semantic trajectory.

Definition 1 (Semantic Trajectory). A semantic trajectory $S = \{s_0, s_1, ..., s_n\}$ is a set of stops, where the ith stop is a tuple $s_i = (x_i, y_i, st_i, et_i, poi_i)$, with x_i and y_i being the spatial coordinates of the stop from the start time st_i to the end time et_i at the POI poi_i.

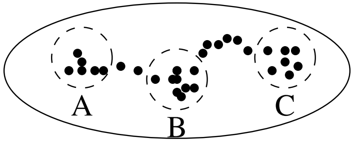
A stop of a trajectory occurs at a place, called POI, given in Definition 2.

Definition 2 (Point of Interest). A point of interest is a tuple $poi = (type, x, y, ot, ct)$ where $type$ is the type of the POI (e.g. Restaurant, Gym, University, Shopping Mall), x

and y are its spatial coordinates, and ot and ct are, respectively, the opening and closing times of the point of interest.

The moving object can perform multiple activities at each stop, then how could we distinguish, for instance, at a shopping mall, if people are purchasing items at a store, eating at a restaurant or watching a movie? In order to identify multiple activities at the same stop, we split the stop into sub-stops, which are smaller stops happening inside a bigger stop, as proposed by [Moreno et al. 2010]. For example, let us consider Figure 1 as a trajectory of a student that has a stop at a university. Inside this stop, he has a sub-stop at a classroom (A), a sub-stop at the laboratory (B) and a sub-stop at the university cafeteria (C). Therefore, by using the concept of sub-stops we can identify more than one activity at each stop. The formal definition of sub-stop is given in Definition 3.

Figure 1. Example of Stop with Sub-Stops



Definition 3 (Sub-Stop). A sub-stop is a tuple $sub = (s, st, et, x, y)$ where sub is the sub-stop from the start time st until the end time et inside the stop s with x and y being the centroid of the sub-stop.

We extend the definition of semantic trajectory to cope with activities. Definition 4 shows our formal definition of activity trajectory.

Definition 4 (Activity Trajectory). An activity trajectory $T = \{t_0, t_1, ..., t_n\}$ is a set of stops, where the ith stop is a tuple t_i = $(x_i, y_i, st_i, et_i, poi_i, A_i)$, with x_i and y_i being the spatial coordinates of the stop from the start time st_i to the end time et_i with the set of activities $A_i = \{a_0, a_1, ..., a_n\}$ performed at the POI poi_i, where the jth activity a_j is an activity label.

In this work we extract activities from georeferenced tweets associated with Foursquare, and collect the POI information through the Foursquare API[3]. The definition of georeferenced tweet is given in Definition 5.

Definition 5 (Georeferenced Tweet). A georeferenced tweet is a tuple $(text, time, day\_week, POI, act)$, where act is the activity extracted from the tweet text $text$ shared at the time $time$ at the day of the week day_week and at the point of interest POI (as in Definition 2).

We consider that every POI type has a set of activities that can be performed at a given time and with a certain duration. We define this set of activities as the ***POI Type Profile***, given in Definition 6.

Definition 6 (POI Type Profile). A POI type profile is a tuple pro = $(POItype, act, meanTime, sdTime, meanDuration, sdDuration, frequency)$, where $meanTime$ is the mean time of the observed ocurrences of the activity act at the POI type $POItype$; $sdTime$ is the standard deviation of this time, $meanDuration$ is

[3] https://developer.foursquare.com

the mean duration time of act at $POItype$, $sdDuration$ is the standard deviation of this duration, and $frequency$ is the frequency of act at $POItype$ relative to the total number of activity occurrences observed at $POItype$.

In the following section we present the proposed approach.

4. Proposed Approach

Our approach to infer activities in moving object trajectories has two steps. The first one is to build a knowledge base in the form of POI type profiles, which are extracted from Twitter, Foursquare and census data. The second step is to infer activities in trajectories by matching the object movement with the POI type profiles in the knowledge base. For that end, we propose an algorithm called *T-Activity*. These steps are described in the following sections.

4.1. Building the Knowledge Base

The knowledge base is a representation of the distribution of activity time and duration that happen at each POI type. It contains the following information: POI type, activity name, average duration, duration standard deviation, average time, time standard deviation and relative frequency. Table 1 shows an example of the knowledge base for the POI Type *Shopping Mall*.

Table 1. Knowledge Base for POI Type *Shopping Mall*

POI Type	Activity Name	Avg Time (hrs)	Time Std Dev	Avg Duration (hrs)	Duration Std Dev	Relative Frequency (%)
Shopping Mall	Consumer Purchases	13.98	3.57	0.74	0.86	0.17
Shopping Mall	Socializing, Relaxing, and Leisure	14.57	4.00	0.78	0.99	0.45
Shopping Mall	Eating & Drinking	14.17	3.52	0.63	0.48	0.28
Shopping Mall	Movies	16.35	3.12	2.27	0.77	0.10

Each attribute is described as follows: (i) POI Type is extracted from Foursquare, and it is present in the tweet to show where the activity happened; (ii) Activity Name is the activity we want to infer. We can extract this information from any taxonomy of activities; (iii) Average Duration is the average time people spend doing an activity at a POI type. We can extract this information from any set or taxonomy of activities; (iv) Duration Std Dev is the standard deviation of the average duration. This information can be extracted from any set or taxonomy of activities; (v) Average Time is the post time of the activity at the POI type. This information is extracted from the tweets; (vi) Time Std Dev is the standard deviation of the average time. This information is extracted from the tweets; and finally (vii) Relative Frequency is the proportion of tweets of an activity that happened at the POI type.

Having described the attributes that compose the knowledge base, Algorithm 1 describes how we build it from a dataset of georeferenced tweets annotated with activities (see Section 5.1 for details) and using any set or taxonomy of activities to obtain the average time spent with each activity at a POI Type. The input of the algorithm is a corpus of georeferenced tweets and any dataset containing the average time people spend with daily activities. It iterates the corpus (lines 4 to 8), extracting the tweet post time to an auxiliary structure of POI types and activities (line 5). Then it adds one to the frequency for the POI type and activity of the same auxiliary structure (line 6), and adds one to the frequency regardless of the activity (line 7) in order to obtain the relative frequency of the activities at each POI type. After that, it iterates the auxiliary structure A (lines 9 to 18) and obtains the POI type and activity of each instance (lines 10 and 11). After that, it calls the method $C.getDuration$, which has a list of durations for each POI type/activity, and gets the activity duration list filtered by POI type and activity (line 12) and calculates the mean and standard deviation (lines 13 and 14). Then it also calculates the mean and standard deviation of the tweet post time (lines 15 and 16) and the relative frequency of the activity for the POI type (line 17). Finally, it returns the POI Type profiles as the knowledge base K (line 19). As this algorithm iterates the corpus of tweets and the auxiliary dictionary A only once, the complexity is $O(n_d + n_p)$, where n_d is the corpus size and $n_p = n_t * n_a$, with n_t being the number of unique POI types in the corpus of tweets D and n_a being the number of unique activities in the corpus of tweets D.

Algorithm 1 Knowledge-Base Builder

```
Require:
     $D$ // corpus of tweets
     $C$ // census dataset / activity taxonomy
 1: $K$ = empty dictionary;
 2: $A$ = empty dictionary;
 3: $T$ = empty dictionary;
 4: for each $tweet$ in $D$ do
 5:     $A[tweet.poi.type, tweet.act].time.append(tweet.time)$;
 6:     $A[tweet.poi.type, tweet.act].frequency$ += 1;
 7:     $T[tweet.poi.type].frequency$ += 1;
 8: end for
 9: for $i$ = 0; $i < A.size()$; $i$ = $i$ + 1 do
10:     $ptype$ = $A.getPOIType(i)$;
11:     $act$ = $A.getActivity(i)$;
12:     $duration\_list$ = $C.getDuration(ptype, act)$;
13:     $K[ptype, act].meanDuration$ = $mean(duration\_list)$;
14:     $K[ptype, act].sdDuration$ = $sd(duration\_list)$;
15:     $K[ptype, act].meanTime$ = $mean(A[ptype, act].time)$;
16:     $K[ptype, act].sdTime$ = $sd(A[ptype, act].time)$;
17:     $K[ptype, act].frequency$ = $A[ptype, act].frequency$ / $T[ptype].frequency$;
18: end for
19: return $K$;
```

In the next section we describe the algorithm T-Activity and show how to infer activities using the knowledge base.

4.2. T-Activity

Before we describe the algorithm that performs activity inference, we introduce the activity inference model, which is based on time, duration and the relative frequency. The time similarity is computed in Equation 1, where K is the POI type profile, st is the sub-stop start time, $meanTime$ is the average time the activity starts in the POI type profile and $\frac{K_{sdTime}}{K_{meanTime}}$ is the variation coefficient of the time in the POI type profile.

$$T(K, st) = 1 - \left| \frac{K_{avgTime} - st}{K_{avgTime}} \right| * \frac{K_{sdTime}}{K_{meanTime}} \qquad (1)$$

The duration similarity between the sub-stop and each activity in the POI type profile is computed using Equation 2, where K is the POI type profile, d is the sub-stop duration, $meanDur$ is the average duration of the activity in the POI type profile and $\frac{K_{sdDur}}{K_{meanDur}}$ is the variation coefficient of the duration in the POI type profile.

$$D(K,d) = 1 - \left|\frac{K_{meanDur} - d}{K_{meanDur}}\right| * \frac{K_{sdDur}}{K_{meanDur}} \tag{2}$$

However, if the score between the activities is too much similar, the time and duration cannot describe which activity happened. Therefore, our model considers the frequency of the activities, according to Equation 3, where K is a POI type profile and $frequency$ is the relative frequency of the activity at the POI type.

$$M(K,d,st) = K_{frequency} * D(K,d) * T(K,st) \tag{3}$$

Considering the matching metrics, Algorithm 2 describes the activity recognition. It receives as input a semantic trajectory S, a knowledge base in the form of POI Type profiles K, a radius that intersects the sub-stop locations $radius$, and the parameters of the algorithm CB-SMoT [Palma et al. 2008]. In order to identify sub-stops, we followed the steps described in [Moreno et al. 2010], which runs the algorithm CB-SMoT over stops to find the sub-stops.

Algorithm 2 T-Activity

```
Require:
    $S$ // semantic trajectory
    $K$ // set of POI Type Profiles
    $radius$ // radius to intersect sub-stop centroids
    $MaxAvgSpeed, MinTime, MaxSpeed$ // CB-SMoT parameters
 1: $T$ = $computeSubStops(S, MaxAvgSpeed, MinTime, MaxSpeed)$;
 2: for each $stop$ in $T$ do
 3:     for each $sub$ in $stop$ do
 4:         $area$ = $buffer(sub.x, sub.y, radius)$;
 5:         $near\_substops$ = $intersect(area, stop)$;
 6:         $duration$ = $sum(near\_substops.getDuration())$;
 7:         $freq$ = $getFrequency(K, sub.s.poi.type)$;
 8:         $score\_time$ = $getSimTime(K, sub.s.poi.type, sub.st)$;
 9:         $score\_duration$ = $getSimDuration(K, sub.s.poi.type, duration)$;
10:         $ranked\_activities$ = $getRankedActivities(freq, score\_time, score\_duration)$;
11:         $sub\_act$ = $max(ranked\_activities)$;
12:         $stop.A.append(sub\_act)$;
13:     end for
14: end for
15: return $T$
```

The algorithm starts by initializing the activity trajectory T, computing the sub-stops by calling the method *computeSubStops*, which considers the points of the stop as a trajectory and calls algorithm CB-SMoT, and if no sub-stop is found, it considers the whole stop as the sub-stop (line 1). After that, it iterates the stops and sub-stops of the trajectory (lines 2 to 14). Then, for each sub-stop, it creates an area with a radius of size $radius$ from the sub-stop centroid (line 4) and calls the method $intersect$ to find all sub-stops that intersect the area (line 5) in order to group them as they happened at the same location, and sum the sub-stop duration as the whole time spent at the same location (line 6). After that, it computes the similarity between the sub-stop sub and the activities in the knowledge base K. First, it gets the relative frequency of the activities at the POI type of the stop from the knowledge base K (line 7). Then it computes the time similarity score of each activity at the POI type of the stop using Equation 1 (line 8). After that it computes the duration similarity score of each activity at the POI type of the stop using Equation 2 (line 9). Then, it computes the score of each activity by multiplying the activity scores of each set using Equation 3 (line 10), and selects the activity with the highest score (line 11). Finally, it appends the activity with the highest score to the stop (line 12), and returns the activity trajectory (line 15).

The complexity of this algorithm is $O(n_s + n_{sub}^2)$, where n_s is the number of stops and n_{sub} is the number of sub-stops. Also, as the algorithm CB-SMoT is executed outside the loop, it does not increase our complexity. In the next section we describe the performed experiments and compare our results with other work.

5. Experiments

In this section we describe the experiments performed in two trajectory datasets and how we extracted activities from tweets in order to build the knowledge base.

5.1. Building the Knowledge Base from Twitter

We use the Twitter Public Streaming API [4] to gather tweets for the knowledge base. We selected 137,509 instances of georeferenced tweets generated from Foursquare from 14/09/2010 to 11/05/2015. The collection was filtered by Portuguese written tweets inside Brazil's bounding box and with at least 3 words.

To build and evaluate the knowledge base, the first step is to extract the POI information from Foursquare, using the Venue Search API [5]. We do that by looking the Venue ID present in the tweet text. As a result we have the tweet text, the tweet time, the POI type and the POI name. Then, for this experiment we filtered the tweets by the following POI types: Restaurant, Gym, Supermarket, University and Shopping Mall, which are types we assumed to have more than one activity. The result was a corpus with 45,209 tweets.

To identify the activities from tweets, we follow the method proposed by Zhu in [Zhu et al. 2016], which consists of using the activities defined in the American Time-Use Survey (ATUS) [Shelley 2005] to build a classification model to assign each tweet to an activity. We randomly selected a sample of tweets stratified by POI type, with the size determined by a confidence level of 95% and a confidence interval of 5%, and manually classify each one to an activity present in the ATUS taxonomy accordingly to the text of the tweet. Having the annotated tweets, we build a classification model considering the following features: (i) POI Type: as each tweet is georeferenced to a Foursquare POI, we extract the POI type and construct a matrix containing 887 binary features, each one representing a POI type; (ii) Tweet Text: by extracting the most relevant unigrams and bigrams weighted by TF-IDF, we obtain 9394 features from the text; (iii) POI Name: the same way we extract features from the tweet text we extract the POI name, as some POI names can be indicative of activities (e.g. Japanese Restaurant, Fourth Street Market); Posting Time: we chunk the tweet posting time by hour of the day to construct a matrix of 24 features. We combine the previous features in a matrix and apply

[4] https://stream.twitter.com/1.1/statuses/sample.json

[5] https://developer.foursquare.com/docs/venues/search

the Linear SVM package from Scikit-Learn library [Pedregosa et al. 2011] to build the classification model. We select L1-regularization with squared hinge loss and keep the default parameters. We evaluate the tweet classification model using a 10-fold cross-validation, obtaining an average accuracy of 76%.

After building and evaluating the classification model, we classify the remaining tweets. In addition, we run Algorithm 1 to extract the mean and standard deviation of time and duration and also the relative frequency of the activities to store in the knowledge base. Table 2 shows the knowledge base generated for this experiment.

Table 2. Entire Knowledge Base

POI Type	Activity Name	Avg Time (hrs)	Time Std Dev	Avg Duration (hrs)	Duration Std Dev	Relative Frequency (%)
Shopping Mall	Consumer Purchases	13.98	3.57	0.74	0.86	0.17
Shopping Mall	Socializing, Relaxing, and Leisure	14.57	4.00	0.78	0.99	0.45
Shopping Mall	Eating & Drinking	14.17	3.52	0.63	0.48	0.28
Shopping Mall	Movies	16.35	3.12	2.27	0.77	0.10
Supermarket	Consumer Purchases	15.12	4.52	0.69	0.52	0.73
Supermarket	Eating & Drinking	14.50	5.17	0.54	0.47	0.27
Restaurant	Eating & Drinking	12.24	6.83	1.00	0.62	0.78
Restaurant	Socializing, Relaxing, and Leisure	14.60	6.59	1.47	1.34	0.18
Restaurant	Consumer Purchases	14.51	5.28	0.16	0.19	0.04
University	Socializing, Relaxing, and Leisure	14.52	5.69	0.73	0.88	0.02
University	Education	13.38	5.37	3.16	1.90	0.96
University	Eating & Drinking	13.96	4.51	0.51	0.30	0.02
Gym	Sports, Exercise, and Recreation	13.25	5.98	0.99	0.67	1.00

As we can see in Table 2, the majority of the POI types have multiple activities. For instance, *Shopping Mall* has four different activities, where the most common activity is *Socializing, Relaxing, and Leisure*. The relative frequency is a reflex of what people do and tweet about. It is important to notice that the average time is an approximation of multiple distributions, which explains the high standard deviations of time.

Having the knowledge base, we run the algorithm *T-Activity* using two different datasets, a semantic trajectory dataset built from census data (Section 5.2), and a semantic trajectory dataset collected in Florianópolis, Brazil (Section 5.3).

5.2. Census Trajectory Dataset

Considering the difficulty for obtaining a semantic trajectory dataset with ground truth, we evaluate our algorithm with a dataset generated from the ATUS dataset. This dataset consists of activity diaries, where each diary corresponds to the semantic trajectory of an individual, resident of the United States of America. The diary contains the activity, the place where the activity was performed (POI), and the start and end times of the activity. From this dataset we selected all households that have activity entries with more than 5 days, resulting in 41 households with 5246 stops. Every POI where an activity was performed is considered as a stop, and as we have multiple entries at the same POI, we consider them as sub-stops. However, and as we do not have the POI coordinates in this dataset, we consider the parameter $radius$ as 0 meters, as we cannot match the location of sub-stops.

We compare the algorithm *T-Activity* with the works [Furletti et al. 2013], [Reumers et al. 2013] and [Njoo et al. 2015]. In order to compare the works, we consider only the activity with the highest score at each sub-stop to calculate f1-score and accuracy. On the other hand, as the POIs can have several activities, the metrics of the related work are calculated based on the activity that consumed more time at the POI. Figure 2 shows the f1-score for the methods of Furletti [Furletti et al. 2013] and Reumers [Reumers et al. 2013]. From the f1-score we can see that our method outperforms the existing works. The work of Furletti has a lower or equal score for all classes. This happens because each POI type is matched to an exclusive activity, and our method considers the similarity of the activities along the relative frequency of the activities. However, considering the relative frequency is problematic for activities that are too similar and have a low frequency, such as *Socializing, Relaxing and Leisure*. On the other hand, as Reumers does not consider the POI type, the f1-score has the lowest result.

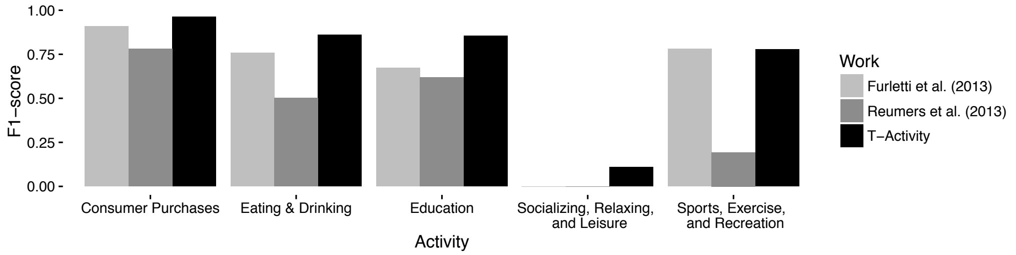


Figure 2. Comparison of F1-Score

In order to demonstrate the multi-activities, we analyze the accuracy at the POI type *University* in Figure 3. It shows that our method outperforms the works of Njoo and Reumers for the activities *Eating & Drinking* and *Socializing, Relaxing, and Leisure*, and that our work is the only one that recognizes the activity *Socializing, Relaxing, and Leisure*. On the other hand, as Furletti and Njoo consider one activity, they have an accuracy of 1.00 for *Education*.

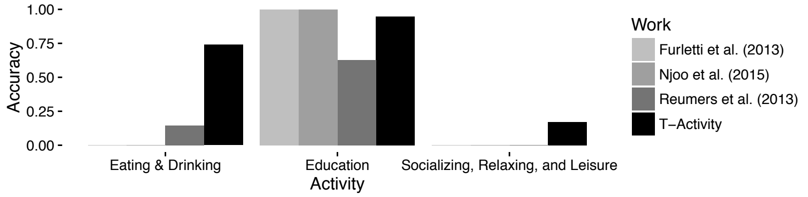


Figure 3. Accuracy Comparison at POI Type University

5.3. GPS Trajectory Dataset

The GPS trajectory dataset is a ground-truth dataset of semantic trajectories annotated with activities collected from 14/04/2016 to 08/06/2016 by 8 participants in the city of Florianopolis, Brazil. The dataset has 59 trajectories, 100 stops and 128 sub-stops. For this experiment we use a $radius$ of 10 meters to group sub-stops at the same location, and consider the activity with the highest score at the sub-stop to calculate f1-score and accuracy. However, as the stops can have several activities, the metrics of the related work are calculated based on the activity that consumed more time at the stop. Figure 4 shows the f1-score in comparison to [Furletti et al. 2013, Reumers et al. 2013]. Analyzing the f1-score we can see that our work has the best result. In addition, Furletti has a score of 1.00 where the main activity is the only activity in the GPS trajectory dataset, such as *Consumer Purchases* for *Supermarket* and *Sports, Exercise and Recreation* for *Gym*, otherwise it has a lower score. Reumers on the other hand, considers the duration and start time of the activities, and as some activities have a similar start time and duration the f1-score is affected.

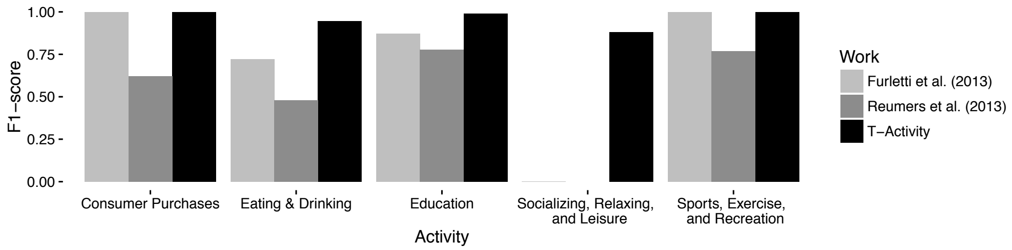


Figure 4. Comparison of F1-Score

We also analyze the accuracy of the works at the POI type *Restaurant* for the activities *Eating & Drinking* and *Socializing, Relaxing and Leisure* in Figure 5. It shows that our work is the only one to recognize the activity *Socializing, Relaxing and Leisure*, as we consider sub-stops to identify multiple activities and the other works can only identify one activity at each stop. Nevertheless, our work also has the highest accuracy.

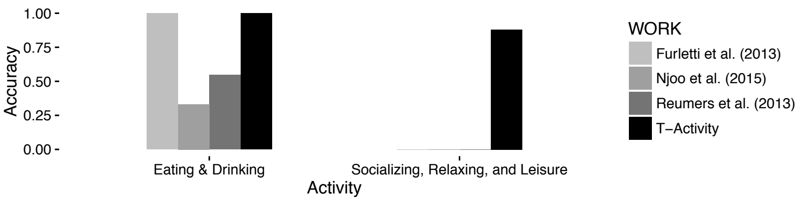


Figure 5. Accuracy Comparison at POI Type Restaurant

6. Conclusion

In this paper we proposed a new method for activity recognition in moving object trajectories, extracting the possible activities from tweets posted at POI types visited by the trajectories. Even though activity recognition is broadly performed with different types of data, infer activities in moving object trajectories is not a trivial task. In this paper we proposed a POI Type profile, in the form of a knowledge base, extracted from georreferenced tweets, to represent the activities that can happen at a POI Type. We proposed the algorithm T-Activity that matches the trajectory and the POI Type profiles for detecting the trajectory activity. As future work, we will go deeper in the activity analysis, defining and recognizing unusual activities, and using Gaussian Mixture Models to calculate the statistics for time and duration in the knowledge base.

7. Acknowledgment

This work was supported by the Brazilian agencies CAPES and CNPQ.

References

Bogorny, V., Renso, C., Aquino, A. R., Lucca Siqueira, F., and Alvares, L. O. (2014). Constant–a conceptual data model for semantic trajectories of moving objects. *Transactions in GIS*, 18(1):66–88.

de Alencar, L. A., Alvares, L. O., Renso, C., Raffaetà, A., and Bogorny, V. (2015). A rule-based method for discovering trajectory profiles. In *SEKE 2015*, pages 244–249, Pittsburgh, PA, USA.

de Aquino, A. R., Alvares, L. O., Renso, C., and Bogorny, V. (2013). Towards semantic trajectory outlier detection. In *GeoInfo*, pages 115–126.

Furletti, B., Cintia, P., Renso, C., and Spinsanti, L. (2013). Inferring human activities from gps tracks. In *2nd ACM SIGKDD*, pages 5:1–5:8, New York, NY, USA. ACM.

Furtado, A. S., Kopanaki, D., Alvares, L. O., and Bogorny, V. (2016). Multidimensional similarity measuring for semantic trajectories. *Transactions in GIS*, 20(2):280–298.

Kim, Y., Pereira, F. C., Zhao, F., Ghorpade, A., Zegras, P. C., and Ben-Akiva, M. (2014). Activity recognition for a smartphone based travel survey based on cross-user history data. In *ICPR 2014*, pages 432–437. IEEE.

Liu, H., Luo, B., and Lee, D. (2012). Location type classification using tweet content. In *ICMLA, 2012*, volume 1, pages 232–237.

Moreno, B., Times, V. C., Renso, C., and Bogorny, V. (2010). Looking inside the stops of trajectories of moving objects. In *Geoinfo*, pages 9–20.

Njoo, G. S., Ruan, X. W., Hsu, K. W., and Peng, W. C. (2015). A fusion-based approach for user activities recognition on smart phones. In *DSAA*, pages 1–10.

Palma, A. T., Bogorny, V., Kuijpers, B., and Alvares, L. O. (2008). A clustering-based approach for discovering interesting places in trajectories. In *Proceedings of the 2008 ACM SAC*, pages 863–868, New York, NY, USA.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830.

Reumers, S., Liu, F., Janssens, D., Cools, M., and Wets, G. (2013). Semantic annotation of global positioning system traces: Activity type inference. *Transportation Research Record: Journal of the Transportation Research Board*, (2383):35–43.

Shelley, K. J. (2005). Developing the american time use survey activity classification system. *Monthly Lab. Rev.*, 128:3.

Spaccapietra, S., Parent, C., Damiani, M. L., de Macedo, J. A., Porto, F., and Vangenot, C. (2008). A conceptual view on trajectories. *DKE*, 65(1):126 – 146.

Weerkamp, W., Rijke, M. d., et al. (2012). Activity prediction: A twitter-based exploration. In *Proceedings of the TAIA'12 Workshop Associated to SIGIR'12*.

Ying, J. J.-C., Lee, W.-C., and Tseng, V. S. (2014). Mining geographic-temporal-semantic patterns in trajectories for location prediction. *ACM TIST*, 5(1):2:1–2:33.

Zhu, Z., Blanke, U., and Tröster, G. (2016). Recognizing composite daily activities from crowd-labelled social media data. *PerCom*, 26:103 – 120.

An algebra for modelling the simultaneity in agents' behavior in spatially explicit social-environmental models

Washington Sena de França e Silva[1], **Tiago Garcia de Senna Carneiro**[1]

[1]Departamento de Computação – Universidade Federal de Ouro Preto (UFOP)
Ouro Preto – MG – Brazil

wsenafranca@gmail.com, tiagogsc@gmail.com

Abstract. *Humanity is the major driver of spatial changes resulting from interactions between social and environmental systems. Environmental models usually apply the agent-based modeling paradigm to describe the social aspects of spatial changes. For this reason, these models have incorporated challenges inherent to this paradigm. One of these challenges is how to provide a semantically correct way to describe and simulate the simultaneity in execution of agents. In this context, this work describes an algebra to the development of spatially explicit agent-based models in a way that the algebra operators implicitly treat the simultaneity in agent's execution.*

1. Introduction

The most indicated modelling paradigm to describe social aspects of spatial change processes is the Agent-Based Model [Parker et al. 2003]. In this paradigm, different types of individuals (agents) are able to communicate and change the space through function that represents the agent's behavioral rules [Macal and North 2005]. However, this paradigm does not have a syntactic structure for directly representing heterogeneous spaces. This type of spatial structure needs different sets of attributes, resolutions and neighborhood relations at different locations. For this reason, it is useful to combine Cellular Automaton (CA) and Agent-Based Model (ABM) paradigms to describe biophysical and social aspects of spatially explicit environmental models [d'Aquino et al. 2002].

The use of the ABM presents issues that are inherent to this paradigm. Among these issues, there is a need to provide a semantically correct way to describe and simulate the simultaneity in execution of agents. This simultaneity means that agents may perform changes in the current state of simulation (present) considering the last synchronized state of simulation (past), and perceive the changes into environment only after they synchronize their own information [Michel et al. 2001]. Analogously, agents may perceive also the changes into environment instantaneously when they perform their changes from the present state into the same present state [Brown et al. 2005].

Ideally, the manner that agents perceive the environment or execute their behavioral rules should be independent of simulation platforms and their architectures (parallel or sequential). A same set of rules, i.e., a model should have the same semantics in any simulator. Despite this, platforms that use the sequential architecture tend to ignore the simultaneity in execution of agents. Consequently, these approaches deal with a strictly simultaneous behavioral rule as a sequential one. For this reason, simulations may present incorrect results caused by computational artifacts [Coakley et al. 2012].

On the other hand, platforms for simulation of agents that execute under some parallel architecture like REPAST-HPC [Collier and North 2011], FLAME [Coakley et al. 2012] and D-MASON [Cordasco et al. 2013], are able to deal with the simultaneity in the execution of agents. They deal with it using parallel programming and defining strategies to allow that tasks like scheduling, communication and synchronization [Shook et al. 2013, Fujimoto 2015, Rousset et al. 2016] perform in automatized and transparent way from the modeler's perspective. This way, modelers can explore high performance simulations even when they are inexperienced to deal with parallel programming issues. However, these approaches force modelers to describe behavioral rules in a manner that such rules have to guarantee the coherence and consistency of simulations. This can be highlighted in cases in which exist concurrent access to shared resource like in collision avoidance [Torrens and McDaniel 2013], matching and reproduce [Lysenko et al. 2008].

An alternative way to guarantee coherence and consistency in simulations is to provide such simultaneity control in the language level instead of doing it in level of the simulation engine. In this way, modelers can clearly express the expected semantics from his/her code diminishing ambiguity in rules semantics. Sequential architectures will find in the model code the information required to simulate simultaneity. Parallel architectures will find the necessary information to simulate sequential behaviors. It is possible to guarantees that a correct model will perform a correct simulation as well. In this context, this paper defines and evaluates one approach for the specification of simultaneity in the execution of agents through an algebra for spatially explicit agent-based model development.

This paper is organized as follows. Section 2 highlights some related works. Next, section 3 describes the algebra, its types, operators, syntax and semantics. Section 4 shows the experiments developed to demonstrate how the algebra has solved some problems faced in agent's modelling and simulations and then we present the algebra usage through a classical model. Finally, section 5 presents the conclusions of this work.

2. Related Works

Providing the simultaneity in execution of agents in modelling level is a manner to guarantee consistence and coherence for semantically correct models in simulation time. In these approaches, the modelling language is able to deal with these issues. The early works to present solutions for this are DESIRE [Dunin-Keplicz and Treur 1994], Concurrent METATEM [Fisher 1994], ConLog [De Giacomo et al. 2000] e AALAADIN [Michel et al. 2001].

Recently, the works that much relate to our approach are the languages ALOO and SARL. The agent-oriented language ALOO [Ricci and Santi 2013] uses the concept of agents (mobility entities) and objects (stationary entities) to define one semantic for mutual exclusion on language level for scenarios where several agents are trying to access or change a same object. ALOO is therefore able to guarantee concurrency control in accesses to shared resource.

The general-purpose agent-oriented programming language SARL [Rodriguez et al. 2014] provides a manner to encapsulate the model's partition, and the communication and synchronization of agents through concepts of multi-contexts and spatial hierarchy. Briefly, agents can communicate only with other agents that are located at same space and are able to access the same context. In this way, the language make explicit the groups of agents that are able to communicate and therefore, need to keep their states synchronized.

Comparing the previous approaches with our own. Both ALOO and our approach have mechanisms to guarantee coherence in a scenario where exist concurrent access to shared resource in a way that modelers do not need to deal with the concurrency control directly. Comparing our algebra and SARL, they both use concepts as groups of agents in language level for providing coherency in communication and synchronization of agents.

The main aspects that differ our algebra and these languages are: (1) Modelers can clearly express the expected semantics for agent's rules; (2) The algebra provides two ways (simultaneous or sequential) to execute a same agent's rule.

3. An Algebra for describing social-environmental spatially explicit model

An algebra specifies his components in a manner that makes possible to abstract the implementation of these components [Frank 1999]. Hence, algebras for ABM are independent of programming languages and of simulator's architectures as well. In this paper, we define an algebra by a set of types and operators applicable to these types.

3.1. Types

Types in an algebra are the kind of entities that the algebra's operators are able to manipulate. Types define in which kind of entities the modeled phenomenon can be decomposed and represented. In this work, modelers are able to describe their models in terms of agents, collections of agent, cellular spaces, and social and spatial relations. These types are grouped into three main categories: (1) basic, (2) collections and (3) relations.

An agent is a basic type that performs changes in the environment. An agent has an attribute list. Each attribute in this list is a pair key-value (Figure 1). A key represents the name of an agent property and the value represents the current state of the correspondent property.

$Attribute : (Key, Value)$

- $Key : Indentifier$
- $Value : Boolean|Number|String|Agent|Cell|[Attribute]|Null$

Figure 1. Formal definition of an attribute.

All agents have a non-null attribute that locate them in the space. This attribute is a reference to a certain cell. The main function of this attribute is to enable agent movement. To perform this movement, an agent just need to replace the current value of his location attribute by another cell. A cell describes properties of a spatial location. Besides the attribute list, a cell also has an agent list to store all agents placed inside it and a list of its neighbor cells (Figure 2a).

A collection represents a set of same-type entities. The figure (Figure 2b) shows the definition of all collections in this algebra. A society is a collection for same-type agents. Two agents have the same type when they present the same internal states and the

(a)

$Agent : (Attributes, Location)$
- $Attributes : [Attribute]$
- $Location : Cell$

$Cell : (Attributes, Agents, Neighbor)$
- $Attributes : [Attribute]$
- $Agents : [Agent]$
- $Neighbor : [Cell]$

(b)

$Society : Agents$
- $Agents : [Agent]$

$Group : Agents$
- $Agents : Sorted([Agent])$

$CellularSpace : Cells$
- $Cells : [Cell]$

Figure 2. (a) Definition of basic types. (b) Definition of agent and cell collections.

same behavioral rules. Group is a set of agents in which every agent must satisfy a given selection function. For example, a group of agents of same gender or a group of agents where every agent is older than a given age. A group can sort its agents to define a kind of precedence between them. Thus, groups are filters defined over societies, selecting the agents that will activated in some action.

A Cellular Space is a grid of cells described by the same attributes. Each cell has a set of cells that defines its neighborhood. This neighborhood is essential to simulate spatial process using the CA paradigm.

A relational type is responsible for connecting agents enabling communication between them. The relational type social network can represents any relation between agents. A modeler defined function generate a social network. This function must determine the weight of a connection between two agents in a society. A weight with 0% means that there is no connection and 100% means a connection of maximum intensity. In a social network, agents are nodes in a graph, their connection are edges and the edges' weight are the strength agent's connection [Andrade et al. 2010]. In this algebra, social networks are like maps in which agents work as indexes which maps to lists containing agents and weights representing their connections (Figure 3).

$SocialNetwork : Agent : Connections$
- $Connections : [(Agent, Weight)]$
- $Weight : Number$

$SpatialNeighborhood : (CellularSpace, N, M)$
- $N : Number$
- $M : Number$

Figure 3. Definition of algebra's relational types.

Besides the social network, neighborhoods represent spatial relations between agents. In this relation, an agent connects to another agent through the cell's neighborhood structure. Using the cell's list of agents, an agent can access all agents from a neighbor cell. In this manner, agents are able to connect by proximity relations. A spatial neighborhood is defined by a reference to a cellular space and by its dimension. The size of a neighborhood is a pair N and M (MxN), where N is the number of cells in vertical and M is the number of cells in horizontal.

3.2. Operators

Operators are a set of functions applicable to the types previously presented. Next, we present the syntax and semantic of each operator defined in this algebra.

The ask operator uses a message passing schema for providing interaction between agents and other types (Figure 4). Agents send messages to receivers requesting them to perform some actions (tasks).

$ask : Function(Receiver, Action, Args)$
- $Receiver : Agent|Cell|Society|Group|CellularSpace|[Receiver]$
- $Action : Function(Receiver, Args)$
- $Args : [*]$

Figure 4. Definition of ask operator.

Since actions like move, die and reproduce are very common in ABM, these operators were pre-defined in this algebra using the ask operator (Algorithm 1).

Algorithm 1 Defining operators move, die and reproduce using operator ask.

```
function MOVE($ag$, $cell$)
    $loc = ag.location$
    $ask(loc.agents, pop, ag)$
    $ask(ag, setLocation, cell)$
    $ask(cell.agents, push, ag)$

function DIE($ag$)
    $loc = ag.location$
    $ags = loc.agents$
    $ask(ags, pop, ag)$
    $ask(soc, pop, ag)$

function REPRODUCE($ag$)
    $loc = ag.location$
    $child = copy(ag)$
    $ask(society, push, child)$
    $move(child, ag.location)$
```

In this algebra, there is not a way for directly create basic types. Modelers must use collection construction operators to instantiating entities of these types. In this manner, every basic entity will be enclosed in at least one collection. Figure (Figure 5) briefly defines the construction operators for collections and relations.

$createSociety : Function(Instace, Quantity) \rightarrow Society$
- $Instace : Agent$
- $Quantity : Number$

$createGroup : Function(Society, Filter, Compare) \rightarrow Group$
- $Filter : Function(Agent) \rightarrow Boolean$
- $Compare : Function(Agent, Agent) \rightarrow Boolean$

$createCellularSpace : Function(Instace, Dimension) \rightarrow CellularSpace$
- $Instace : Cell$
- $Dimension : (Width : Number, Heigh : Number)$

$createSocialNetwork : Function(Society, Connection) \rightarrow SocialNetwork$
- $Connection : Function(Agent, Agent) \rightarrow Number$

$createSpatialNeighborhood : Function(Space, N, M) \rightarrow SpatialNeighborhood$
- $Space : CellularSpace$
- $N : Number$
- $M : Number$

Figure 5. Definition of agent and cell collections.

The construction operator of society creates a society using an agent definition and a given quantity. This definition works as a template that enables the operator to instantiate any quantity of agents in a society. The operator creates each agent as a copy of the archetype agent. The parameter quantity determines the number of agents that the operator will create. The construction operator for group uses a society, a selection function and a compare function to create a group. In the same way, the construction operator for cellular spaces instantiates cells by copying the archetype cell received as parameter. The cellular space dimension determines the quantity of cells that the operator will create. The construction operator for social network uses a society and a function that determines the intensity of connections between each pair of agents to create a social network. The construction operator for spatial neighborhood uses a cellular space and the required neighborhood dimension to create a spatial neighborhood.

Modelers should use execution operator to simulate collection of agents provoking changes described by the behavioral rule received as parameter. The modeler defines these rules as functions that govern the behavior of some types of agents. This approach allows the reuse of rule definitions, allowing the modeler to apply them to any collections able to execute it. Three factors determine the semantics of the operator execute: The type of collection received as parameter, the use of any relational as parameter, and the type of behavioral rule received as parameter (Table 1).

Table 1. Syntactic and Semantic definition of execute operator.

Operator	Syntax	Semantic
Simultaneous local execution	$execute(Society, Rule)$ • $Rule : Function(Agent)$	Each agent in a given society simultaneously applies a given rule independently of the other agents.
Sequential local execution	$execute(Group, Rule)$ • $Rule : Function(Agent)$	Each agent in a given group sequentially applies a given rule independently of the other agents.
Simultaneous shared execution	$execute(Society, Relation, Rule)$ • $Relation :$ – $SocialNetwork$ – $SpatialNeighborhood$ • $Rule : Function(Agent, Agent)$	Each agent in a given society simultaneously applies a given rule that enables communication between agents.
Sequential shared execution	$execute(Group, Relation, Rule)$ • $Relation :$ – $SocialNetwork$ – $SpatialNeighborhood$ • $Rule : Function(Agent, Agent)$	Each agent in a given group sequentially applies a given rule that enables communication between agents.

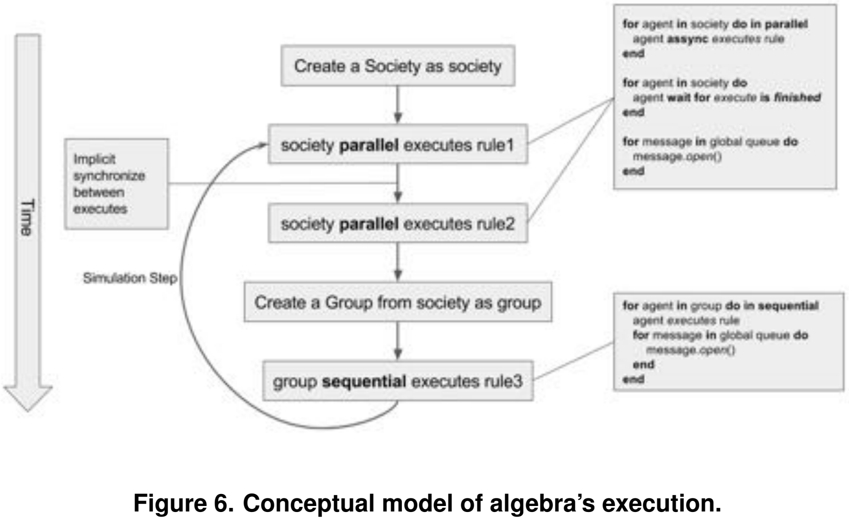
When a society executes a rule, all agent simultaneously performs changes. This means that agents will perceive the provoked changes only after all of them have accomplished their execution. A group enables that agents instantaneously perceives any provoked change. Group executes agent by agent in a sorted and sequential manner. This mode of execution guarantees mutual exclusion for agents performing the same rule. In this context, the rule code works as a critical section. Thus, agents perceive changes as soon as each agent finishes its execution. The group's order function determines in which order agents will execute.

Relational types determines how the communications between agents of a given collection will occur. When execute operator does not receive a relation, changes are local. This mean that, an agent will apply a change independently of the others. When an execution has a relation, the rules will receive two agents as parameter. The first agent will apply the rule while the second one will only take in a communication process. Usually, these rules describe behaviors that collect information from other agents to support the decision-making process. The second agent is a read-only object. The only way to change the value of an attribute from the second agent is through requests using the *ask* operator. This is a convention in order to guarantees coherent computations for all agents. Requests sent through the *ask* operator will be served only after the simulation synchronization stage, causing the changes requested.

The execute operator is also responsible for performing communication and synchronization of agents. Synchronization of agents is transparent to the modelers. Modelers do not need to deal with concurrency control to guarantee coherent computation. For this, the ask operator sends asynchronous messages and senders do not need to wait for receivers' responses to resume their execution. The execute operator will synchronize agents and process messages according to the semantics desired by the modeler, depending on the type of collection received as parameter: Society or group.

When a society invokes the execute operator, all agents perform their simulations in parallel and a synchronization barrier forces agents to wait until all other agents to finish. Only then, all agents will process the received asynchronous messages. This guarantees that all agents' rules will execute taking in consideration the same model state, which immediately precedes the invocation of the execute operator. In addition, no communication happens while agents' internal states are changing.

On the other hand, when a group invokes the execute operator, each agent will execute the behavioral rule sequentially. Immediately after the execution of each agent, all agents will perform the received messages and all agents will perceive the changes caused by the last behavioral rule executes (Figure 6).


Time
Create a Society as society
society parallel executes rule1
Implicit synchronize between executes
society parallel executes rule2
Simulation Step
Create a Group from society as group
group sequential executes rule3

```
for agent in society do in parallel
  agent assync executes rule
end

for agent in society do
  agent wait for execute is finished
end

for message in global queue do
  message.open()
end
```

```
for agent in group do in sequential
  agent executes rule
  for message in global queue do
    message.open()
  end
end
```

Figure 6. Conceptual model of algebra's execution.

In this semantic, if a rule demands an intermediate synchronization, the modeler should split the rule's code into two rules: (1) One containing the code that precedes the synchronization point, and (2) another one containing the code that comes after the synchronization point.

To execute more than one society at the same time, the modeler should use a list of tuples (Figure 7). When an execution will perform a local rule, a tuple has a society and a local rule. In case of a rule demands communication between agents, the tuple will contain a relational type as well. Semantically, a tuple execution is equals to the execute operator (Table 1). In a practical manner, all tuples simultaneously execute before that agents perceive any change (communication and synchronization). Execution by tuples allows for example that two different societies, using different rules, change the space at the same time.

$ExecutionTuple : (Society, Rule)|(Society, Relation, Rule)$

- $Relation : SocialNetwork|SpatialNeighborhood$
- $Rule : Function(Agent)|Function(Agent, Agent)$

Figure 7. Definition of an execution tuple.

4. Experiments

We have done two kind of experiments. (1) The first one demonstrated how the Algebra solves some problems that are related with the simultaneity and how they may affect the simulation results. In addition, these experiments also demonstrated how the simultaneity and the semantic of execution are related. (2) The second one demonstrates the algebra usage in implementing of Predator-Prey model. This classical model has features that are relevant and frequently used in spatial-explicit social-environmental models.

4.1. Effects of simultaneity in simulation results

The experiments demonstrated the effects of simultaneity in the simulation results using three simple models (Figure 8). In the first model, agents are trying to change simultaneously the energy in one shared cell of space (Figure 8a). When simulated through a society, changes performed by most of these agents have no effects. Agents update the cell computing their rules from the simulation past state, overwriting changes in the current simulation state and ignoring any other changes previously simulated. This way, only changes performed by one unique agent will be persisted and perceived in the future computations. On the other hand, when simulate via groups, changes are sequenced and agents will perceive the changes instantaneously.

In another model, eight agents are simultaneously trying to move to one a same empty cell (Figure 8b). Disregarding the collision, simultaneity semantics have shown to have a huge impact in model results. The execution by society resulted in a scenario where all agents moved to the same cell. In this case, they sensed the environment's state where this cell was empty. In the other hand, execution by group resulted in another mobility pattern, in this case only one agent has moved to the target cell. This because after his move the other agents perceived the environment's state where this cell was not empty anymore.

The third model describes a rule where each agent have to collect information from neighbor cells to decide whether he should put fire in his location (Figure 8c). When simulating this model through groups, the order of execution of the agents affects the simulation results, possibly introducing computational artifacts in it. However, this artifact did not appear when agents' rules are simulated through a society, because agents sense and change the space simultaneously, there is order in agents' execution.

	(a) Model	(b) Model	(c) Model

	`function change(ag, cell)`
	`    ask(cell, addEnergy, 2)`
	`end`
		`function moving(ag, cell)`
		`    if cell is empty then`
		`        move(ag, cell)`
		`    end`
		`end`
			`function burn(ag, cell)`
			`    c=count(cell.neighs,'burning')`
			`    if c >= 2 then`
			`        ask(cell, burn, true)`
			`    end`
			`end`

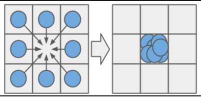
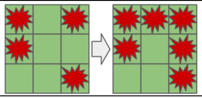
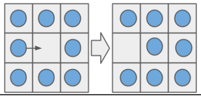
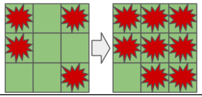
Society	cell.energy: 2		
Group	cell.energy: 4		

Figure 8. Comparison between the simulation using society and group

These experiments shows that a unique model code (syntax) can lead to different results depending on the simultaneity semantics adopted by a given modeling language or simulation engine. In the proposed algebra, modelers can clearly state the expected semantic diminishing ambiguity in model specification and allowing for simulations that produce exactly the same result no matters whether models are been executed by sequential simulators or not. Modelers may choose which semantics fits better the phenomenon being modeled. In addition, the algebra promotes code reuse allowing modelers to apply a same rule with different semantics.

4.2. Description of Predator-Prey model in the proposed algebra

The Predator-Prey model used in these experiments is an adaptation of Wilensky's version of Wolf-Sheep [Wilensky 1997]. In this simplified version, there are three types of entities: Wolves, Sheep and Space. Wolves and Sheep are agents that, in each simulation step, randomly move in space. Agents spend energy to move and dies once they spent all energy. Agents eat to recover their energy. Sheep eat grass from their cells. Wolves prey sheep that are in their cell. The space is modeled as a cellular automaton that simulates grass periodical regrowth. Agents also reproduce by losing half of their energy for the newborn agents.

This model demonstrates the usage of execute and ask operators in describing interactions between two different societies, which individuals compete for access to shared resources (preys and grass). The algorithm (Algorithm 2) shows the *hunting* rule of predators. If a predator meets a prey, predator will target this prey. After marking a prey as target, a predator will attack it. The algorithm (Algorithm 2) describes the behavior of an attack. The *attacking* rule recovers predator's energy by the amount of target's energy and informs that the target is going to lose its energy and then die after the next synchronization.

One can interpret the prey's behavior analogously to predator's behavior (Algorithm 2). Preys will target their own location cells. Then, preys will eat grass from these cells. This case shows that different types of agent can perform a same rule since theys

Algorithm 2 Behavioral rules of hunting.

```
function HUNTING($predator, prey$)
    $predator.target = prey$
function TRYEAT($prey$)
    $prey.target = prey.location$
function ATTACKING($predator$)
    $predator.energy{+}= predator.target.energy/2$
    $ask(predator.target, setEnergy, 0)$
```

have a same set of internal states. In this experiment, preys and predators have energy and target as attributes, and cells have the unique attribute energy (Algorithm 3). In this manner, any agent can perform the *attacking* rule.

Algorithm 3 Defining predator, prey and cell.

```
$predator = Agent\{energy = 40, target = null\}$
$prey = Agent\{energy = 40, target = null\}$
$cell = Cell\{energy = 40\}$
```

The algorithm (Algorithm 4) shows the execution of predator and prey rules. Predators and preys must sense the same environment state. Therefore, they must execute simultaneously. For this, agents execute their rules (moving, hunting and tryEat) using two tuples (*movingExecuteTuples* and *huntingExecuteTuples*). These tuples allow both societies to execute at same time.

In contrast, some agent behavior demands that only one agent executes per time in order to guarantee coherence to model results. For instance, only one predator can kill a given prey. Therefore, a group must execute the *attacking* rule sequentially, meaning that only one agent will attack a target. This way, the *attacking* rule is a critical section of code in which mutual exclusion to resources (target) is guaranteed and all agents will sense attacks at the same instant as they occurs.

Algorithm 4 Scheduling for executions of predators and preys.

```
function MAIN($POPULATION, DIMENSION$)
    $predators = createSociety(predator, POPULATION)$
    $preys = createSociety(prey, POPULATION)$
    $space = createCellularSpace(cell, DIMENSION)$
    $neighs = createNeighborhood(space, 1, 1)$
    $movingTuples = [(predators, moving), (preys, moving)]$
    $huntingTuples = [(predators, neighs, hunting), (preys, tryEat)]$
    for $t = 1...1000$ do
        $execute(movingExecuteTuples)$
        $execute(huntingExecuteTuples)$
        $predatorsGroup = createGroup(predators, hasTarget)$
        $preysGroup = createGroup(preys, hasTarget)$
        $execute(predatorsGroup, attacking)$
        $execute(preysGroup, attacking)$
```

Group also filters the society allowing only few agents to execute the attacking rule, the ones who have targets. The *predatorsGroup*) and preys (*preysGroup* groups do not have an order function defined by the modeler. By default, groups will randomly organize their agents. Hence, all agents have the same chance to attempt an attack to a prey.

In order to evaluate the performance that can be attained by a C++ and OpenMP [Dagum and Menon 1998] implementation of this algebra (Figure 9), the predator-prey model was simulated for spaces of different sizes and, therefore, different population sizes.

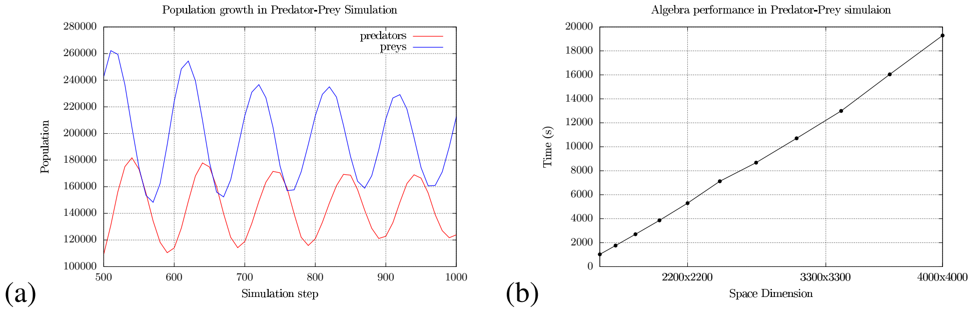


Figure 9. (a) Population growth of predators and preys in a cellular space of dimension 1000 x 1000. (b) Performance results of Predator-Prey simulation.

Initial experiments have shown that simulations using the proposed algebra may achieve a performance curve near to linear in relation of the number of cells in space. This demonstrates that this algebra is also a viable solution from the performance point of view.

5. Conclusions

This paper has presented an algebra for modeling the social aspects of spatial changes in accordance to the Agent-Based Model paradigm. Experiments have demonstrated how this algebra can handle some problems that relates to the simultaneity in execution of agents. Beside this, experiments have demonstrated also the usage of the algebra in the development of a model that has features that are common to many spatially explicit socio-environmental models. The contributions of this algebra are as follow:

1. To allow the definition of behavioral rules independently of the agents that will execute them.
2. To show one way of decoupling model description from the issues that rises from the parallel simulation of multiple agents.
3. To allow for modeler decides the execution semantics of agent's rules.

For these reasons, we believe that this algebra can facilitate the development of models that use the agent-based modeling paradigm. It is still necessary to evaluate the algebra in the development and simulation of other models. Thus, determining if there are models that this algebra is not sufficient to describe them. Furthermore, we wish to evaluate this algebra in large-scale simulations in order to understand the pros and cons of this approach from the high performance point of view.

References

Andrade, P. R., Monteiro, A. M. V., and Camara, G. (2010). Entities and relations for agent-based modelling of complex spatial systems. In *Social Simulation (BWSS), 2010 Second Brazilian Workshop on*, pages 111–118. IEEE.

Brown, D. G., Riolo, R., Robinson, D. T., North, M., and Rand, W. (2005). Spatial process and data models: Toward integration of agent-based models and gis. *Journal of Geographical Systems*, 7(1):25–47.

Coakley, S., Gheorghe, M., Holcombe, M., Chin, S., Worth, D., and Greenough, C. (2012). Exploitation of high performance computing in the flame agent-based simulation framework. In *High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems (HPCC-ICESS), 2012 IEEE 14th International Conference on*, pages 538–545. IEEE.

Collier, N. and North, M. (2011). *Repast HPC: A platform for large-scale agentbased modeling*. Wiley.

Cordasco, G., De Chiara, R., Mancuso, A., Mazzeo, D., Scarano, V., and Spagnuolo, C. (2013). Bringing together efficiency and effectiveness in distributed simulations: the experience with d-mason. *Simulation*, 89(10):1236–1253.

Dagum, L. and Menon, R. (1998). Openmp: an industry standard api for shared-memory programming. *IEEE computational science and engineering*, 5(1):46–55.

De Giacomo, G., Lespérance, Y., and Levesque, H. J. (2000). Congolog, a concurrent programming language based on the situation calculus. *Artificial Intelligence*, 121(1):109–169.

Dunin-Keplicz, B. and Treur, J. (1994). Compositional formal specification of multi-agent systems. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 102–117. Springer.

d'Aquino, P., August, P., Balmann, A., Berger, T., Bousquet, F., Brondízio, E., Brown, D. G., Couclelis, H., Deadman, P., Goodchild, M. F., et al. (2002). Agent-based models of land-use and land-cover change. In *Proc. of an International Workshop*, pages 4–7.

Fisher, M. (1994). Representing and executing agent-based systems. In *International Workshop on Agent Theories, Architectures, and Languages*, pages 307–323. Springer.

Frank, A. U. (1999). One step up the abstraction ladder: Combining algebras-from functional pieces to a whole. In *International Conference on Spatial Information Theory*, pages 95–107. Springer.

Fujimoto, R. (2015). Parallel and distributed simulation. In *Proceedings of the 2015 Winter Simulation Conference*, pages 45–59. IEEE Press.

Lysenko, M., D'Souza, R. M., et al. (2008). A framework for megascale agent based model simulations on graphics processing units. *Journal of Artificial Societies and Social Simulation*, 11(4):10.

Macal, C. M. and North, M. J. (2005). Tutorial on agent-based modeling and simulation. In *Proceedings of the 37th conference on Winter simulation*, pages 2–15. Winter Simulation Conference.

Michel, F., Ferber, J., and Gutknecht, O. (2001). Generic simulation tools based on mas organization. In *10th European Workshop on Modelling Autonomous Agents in a Multi Agent World MAMAAW*, volume 1.

Parker, D. C., Manson, S. M., Janssen, M. A., Hoffmann, M. J., and Deadman, P. (2003). Multi-agent systems for the simulation of land-use and land-cover change: a review. *Annals of the association of American Geographers*, 93(2):314–337.

Ricci, A. and Santi, A. (2013). Concurrent object-oriented programming with agent-oriented abstractions: the aloo approach. In *Proceedings of the 2013 workshop on Programming based on actors, agents, and decentralized control*, pages 127–138. ACM.

Rodriguez, S., Gaud, N., and Galland, S. (2014). Sarl: a general-purpose agent-oriented programming language. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences on*, volume 3, pages 103–110. IEEE.

Rousset, A., Herrmann, B., Lang, C., and Philippe, L. (2016). A survey on parallel and distributed multi-agent systems for high performance computing simulations. *Computer Science Review*.

Shook, E., Wang, S., and Tang, W. (2013). A communication-aware framework for parallel spatially explicit agent-based models. *International Journal of Geographical Information Science*, 27(11):2160–2181.

Torrens, P. M. and McDaniel, A. W. (2013). Modeling geographic behavior in riotous crowds. *Annals of the Association of American Geographers*, 103(1):20–46.

Wilensky, U. (1997). Netlogo wolf sheep predation model. *URL http://ccl. northwestern. edu/netlogo/models/WolfSheepPredation. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.*

Accessibility and flood risk spatial indicators as measures of vulnerability

Juliana Siqueira Gay[1], Mariana Abrantes Giannotti[1], Diego Bogado Tomasiello[1]

[1]LabGEO – Dept. of Transportation Eng. – Polytechnic school at University of São Paulo

siq.juliana@gmail.com, mariana.giannotti@usp.br, diegobt86@gmail.com,

Abstract. *Based on the recent literature relating transport service level and social exclusion, as well as perceptions about the capacity to cope with the occurrence of flood events, this paper identified the spatial pattern related to flood risk and accessibility to urban facilities. For that, composite spatial indicators are developed and compared with socioeconomic data. The analysis shows the outskirts of the city with the most vulnerable places, with high levels of flood risk and low levels of accessibility. Besides that, the high-risk areas are characterized by low income level as well as the low percentage of residents with sewage system is a typical condition of regions with low level of accessibility and close to flood prone areas.*

1. Introduction

The capacity to adapt of population and systems are a relevant focus of the literature (Smit & Wandel, 2006). Vulnerability could be understood as the sensitivity or susceptibility to harm and lack of capacity to cope and adapt facing the occurrence of an extreme event (IPCC, 2014). For the discussion about adaptation measures, vulnerability assessment represents a considerable tool. In this context, the spatial analysis have been notably used to explore spatial data and maps, to inform and communicate different stakeholders about the relation between community and the environment risks at a given scale (Preston, Yuen, & Westaway, 2011).

Different frameworks are formulated to understand the relation between systems, environment, population and risks (Alves, 2013; Anazawa, Feitosa, & Monteiro, 2013; Cutter, 1996; Cutter, Boruff, & Shirley, 2003; Hogan, 1993; Turner et al., 2003). For Cutter (1996) vulnerability is defined as a coupled concept between the social vulnerability and the biophysical risk, located in a specific area. This place-based vulnerability concept involves components of risk, as the proximity to hazards, furthermore social impacts, as the infrastructure availability to support basic needs (Cutter, 1996). According to Hogan & Marandola (2005):

> "Vulnerability is associated with the social disadvantages which simultaneously produce and are reflections and products of poverty. […] Disadvantages are understood as social conditions which negatively affect people, communities or places."

Beyond that (Hogan & Marandola, 2005; Vignoli, 2000) emphasize that these disadvantages correspond to the lack of access and capacity to deal with the availability of resources and opportunities.

One dimension that could affect the social exclusion/inclusion is the level of accessibility to different places and opportunities (Lucas, van Wee, & Maat, 2015). Therefore, accessibility measures are addressed to understand the social exclusion (Lucas, 2012) and equity (Neutens, Schwanen, Witlox, & de Maeyer, 2010). For Wee & Geurs (2011), indicators that include distribution effects should be explored, for instance, accessibility to achieve schools and medical services. In vulnerability index formulation, indicators of the level of access to opportunities, sometimes are considered as components of sensitivity and adaptive capacity, (Moss, Brenkert, & Malone, 2001; Weis et al., 2016).

Based on the motivation of the vulnerability mapping importance as an integrative approach, this work aims at mapping flood risk areas and accessibility measures to urban facilities in the city of São Paulo (Brazil). The main hypothesis is: are there spatial pattern regarding accessibility conditions and flood risk areas? To answer the question, a flood risk indicator was calculated and compared with measures of accessibility to leisure, education and health.

2. Materials and Methods

Firstly, the flood risk areas in São Paulo city were identified. Then, a field work, to better understand the relations between accessibility and flood risk areas was done, as a preparation for further analysis regarding measures, indexes development and mapping as described in this section.

2.1. Field visit

An area characterized by consolidated flood risk was investigated by a field visit. The select area is Jardim Pantanal region, close to Tietê river (Figure 1 and Figure 2). The region is an Environmental Protection Area and presents a land use conflict between irregular occupation and the environment legislation. The local was visited on June, 30th, 2016.

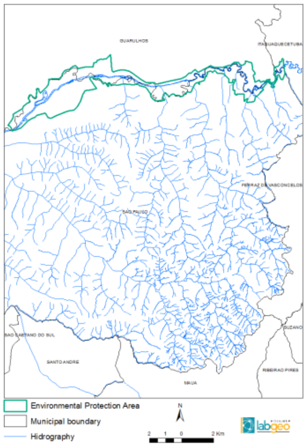


Figure 1 Environmental Protection Area "Várzea do Rio Tietê" in the east region of São Paulo

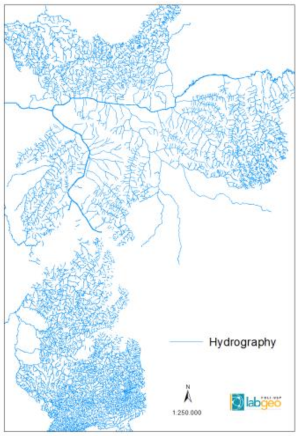


Figure 2 - Hydrography

<u>Outward trip</u>: Brás Metro Station– Line 12 Safira: Itaim Paulista Station. On foot until Rua Tietê (25min)

<u>Return trip</u>: Bus stop (25/30 min on foot) – Bus 273G-10 Metrô Arthur Alvim (50 minutes)

This local analysis serves as experiencing routine of the local population regarding the use of public transport infrastructure. The local impressions helped to confirm the hypothesis that the level of attendance of transportation infrastructure and opportunities is also an indicative of the degree of social inclusion/exclusion of the population living at risk areas.

2.2. Spatial data analysis

The procedure follows the data acquisition, indicators calculation, normalization, and composition of indicators. The analysis and maps were done using ArcGIS 10.4.1. Data used are summarized in Table 1.

Table 1 - The spatial data used for indicators construction

Category	Indicators	Data	Metadata		
			Responsible	Source	Year
Accessibility to public schools	Number of public schools to be accessed in 45 minutes by public transportation	Public schools	Municipal Secretary of Education/Municipal Secretariat of Urban Development	*Geosampa* (Prefeitura de São Paulo, 2016)	2014
		Public Transportation Network	Diego Bogado Tomasiello	(Tomasiello, 2016)	2015
		Metro Origin Destination Survey of São Paulo	São Paulo Metropolitan Company – Metro	(Companhia do Metropolitano de São Paulo, 2007)	2007
Accessibility to health facilities	Number of health facilities (hospitals and health centers) to be accessed in 60 minutes by public transportation	Health facilities (Hospitals and basic health centers)	Municipal Secretary of Health	*Geosampa* (Prefeitura de São Paulo, 2016)	2010
		Public Transportation Network	Diego Bogado Tomasiello	(Tomasiello, 2016)	2015
		Metro Origin Destination Survey of São Paulo	São Paulo Metropolitan Company – Metro	(Companhia do Metropolitano de São Paulo, 2007)	2007
Accessibility to culture facilities	Number of culture facilities to be accessed in 50 minutes by public transportation	Cultural facilities (Libraries, Museums, cultural centers, arts gallery)	Municipal Secretary of Urban Development	*Geosampa* (Prefeitura de São Paulo, 2016)	2015
		Public Transportation Network	Diego Bogado Tomasiello	(Tomasiello, 2016)	2015
		Metro Origin Destination Survey of São Paulo	São Paulo Metropolitan Company – Metro	(Companhia do Metropolitano de São Paulo, 2007)	2007
Flood risk	Population close to flood areas	Flood prone areas (Geotechnical chart)	Department of Planning, Budget and Management/Technology Research Institute (IPT)/Municipal Secretary of Public Safety/Municipal Secretary of Housing	*Geosampa* (Prefeitura de São Paulo, 2016)	1993

Flood risk indicators

The flood risk indicator was based in the general and basic definition of risk as:

$$\text{Risk} = \text{Hazard} \times \text{Exposure}$$

Hazard in the context of this work is represented by the flood risk and the exposure, by the population living in the flood prone area. The steps for the indicator construction is shown in Figure 3.

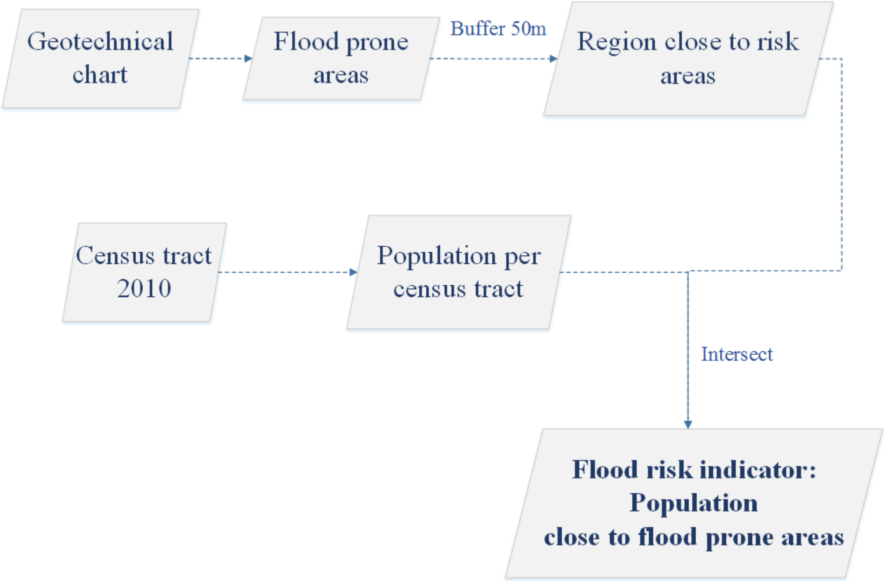

Geotechnical chart
Flood prone areas
Buffer 50m
Region close to risk areas
Census tract 2010
Population per census tract
Intersect
Flood risk indicator: Population close to flood prone areas

Figure 3 - Methodology of the flood risk indicator calculation

Accessibility indicators

The accessibility indicator evaluated was based on the cumulative opportunities (Páez, Scott, & Morency, 2012):

$$A_{ik}^{p} = \sum_{j} W_{jk} I(c_{ij} \leq \gamma_i^p)$$

Where:

W_{jk} = facility of type k at location j

c_{ij}= cost of travel, here is considered the travel time measured in the public transportation network

γ_i^p= threshold value

The value is calculated based on the centroids of census tracts, as a proxy from the origin and destination location, thus not considering an "exact" point coordinate. This approximation was necessary to make the process effort reasonable for this large volume of data. The threshold value is calculated based on the guideline of the Department for Transport Business Plan (2012) from UK and represents the median of the all travel with public transportation with specific reason: education for accessibility to public schools, health to accessibility to hospitals and health centers and leisure for cultural facilities. The steps for the indicator construction are summarized in Figure 4.

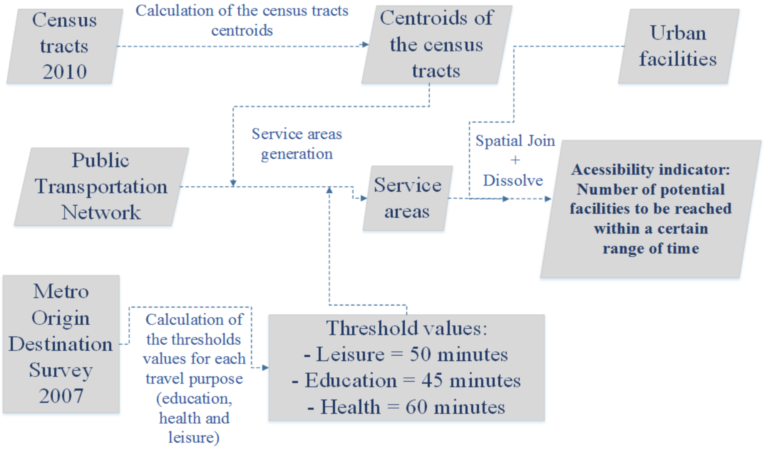

Census tracts 2010
Calculation of the census tracts centroids
Centroids of the census tracts
Urban facilities
Service areas generation
Spatial Join + Dissolve
Public Transportation Network
Service areas
Acessibility indicator: Number of potential facilities to be reached within a certain range of time
Metro Origin Destination Survey 2007
Calculation of the thresholds values for each travel purpose (education, health and leisure)
Threshold values:
- Leisure = 50 minutes
- Education = 45 minutes
- Health = 60 minutes

Figure 4 - Methodology of the accessibility indicators calculation

Indicator composition

All the components were calculated and aggregated in the census tract area. They were normalized to a scale from 0 to 1, according to the formula:

$$\text{Indicator} = \frac{\text{Value(x)} - \text{ValueMin}}{\text{ValueMax} - \text{ValueMin}}$$

Where:

Value (x) = value of the indicator in the referenced census tract

ValueMin = minimum value of the distribution of all the category

ValueMax = maximum value of the distribution of all category

The composition follows the methodology of works already developed in the field of environmental risk (Alves, 2013; Hogan, 1993). The sample of each indicator was reclassified and divided by the median value of the distribution.

High = distribution above the median

Low = distribution below the median

Without risk = outside the flood area

Table 2 - Indicators components and groups

Accessibility to facilities	Flood risk	Group		
Low	High	Aa	A	Low accessibility with the flood risk (high and low)
	Low	Ab		
High	High	Ba	B	High accessibility with the flood risk (high and low)
	Low	Bb		
High	Without risk	C		Without flood risk
Low				

3. Results and Discussions

The outcomes of the field visit are the perceptions about the transit system and the population at risk condition. The residents at Jardim Pantanal area live at border of the transit system. Besides that, they suffer with low level of attendance of sewage treatment system coverage (Figure 5) and accumulated garbage on the streets (Figure 6), causing a considerable harm to the public health. The high travel time to reach the place (more than one hour from metro station) shows that to achieve facilities and even go to work is a costly task for the population living there.

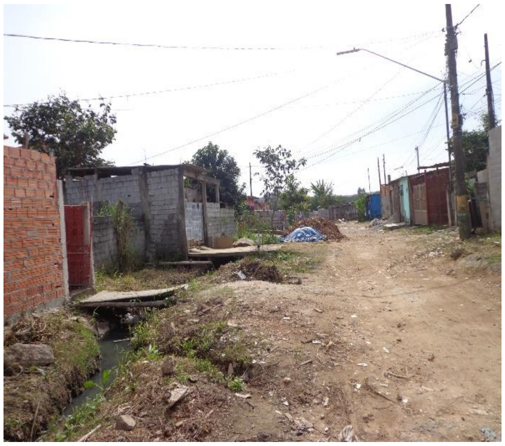


Figure 5 - Open sewage and unpaved street

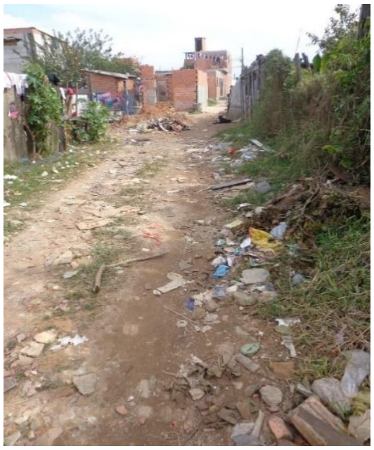


Figure 6 - Accumulated garbage

In the spatial analysis, each component has been calculated and grouped by the census tract. The map of accessibility to health facilities (Figure 7) shows a clear pattern related to the transit system. The facilities are concentrated close to the metro lines. The maps of accessibility to public schools reveal the plenty distribution of school at the east zone of São Paulo (Figure 8). Although, these measures did not consider, in their formulation, the vacancies and quality of the schools. The map of accessibility to cultural facilities displays the lack of cultural opportunities as libraries, museums, cultural centers and art galleries in the peripheral region.

In the flood risk map, the high concentration of population and the proximity of the Billings and Guarapiranga reservoirs, present a critical region in the watershed area. Other areas, as the Pinheiros river, which represents an economic development hub, however displays a population density level lower than the south of the city.

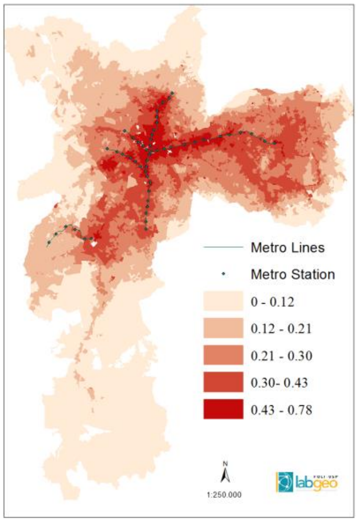


Figure 7 - Number of health facilities to be accessed in 60 minutes with public transportation (normalized)

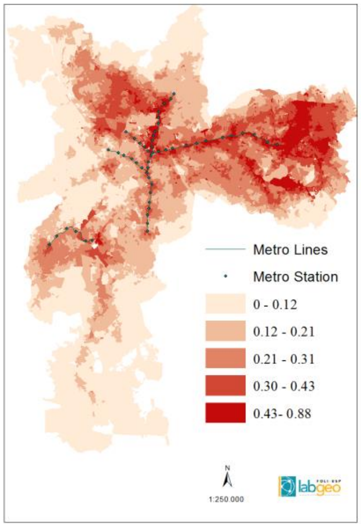


Figure 8 - Number of public school to be accessed in 45 minutes with public transportation (normalized)

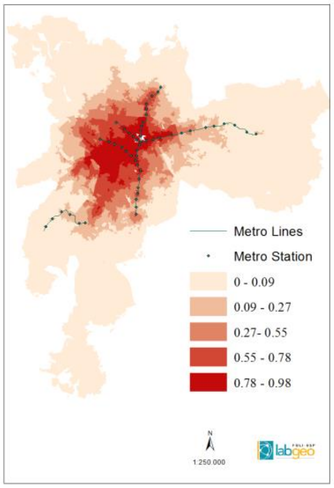


Figure 9 - Number of cultural facilities to be accessed in 50 minutes with public transportation (normalized)

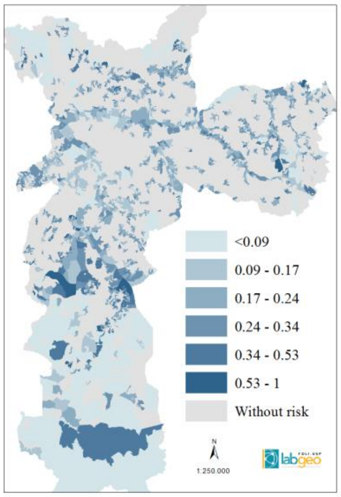


Figure 10 - Population close to flood risk areas

The indicators have been combined and 5 groups have been mapped. It is possible to note the difference between critical areas of accessibility to schools and health facilities compared to the cultural facilities.

In Figure 11 and Figure 13, which present the map of accessibility indicators for public schools and hospitals, the south region presents both, the high and low level of flood risk, and the low level of accessibility. In contrast, in Figure 12, the census tracts very close to the watershed, present high level of accessibility to cultural facilities.

Considering the center and the east zone of the city, the composite indicator for accessibility to cultural facilities (Figure 12) presents some tracts with high level of flood risk, especially along the Aricanduva river, unlike the pattern presented in the other maps.

In general, the risk areas follow the pattern of the hydrographic network of the city, although the outskirts concentrate the tracts classified in the group 1, namely the regions with low accessibility and risk areas.

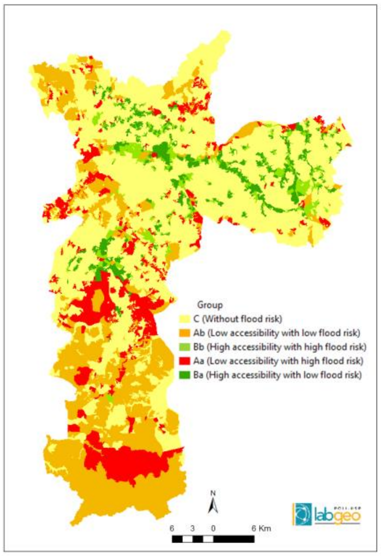

Group
C (Without flood risk)
Ab (Low accessibility with low flood risk)
Bb (High accessibility with high flood risk)
Aa (Low accessibility with high flood risk)
Ba (High accessibility with low flood risk)
N
6 3 0 6 Km
labgeo

Figure 11 - Accessibility to public schools and flood risk area

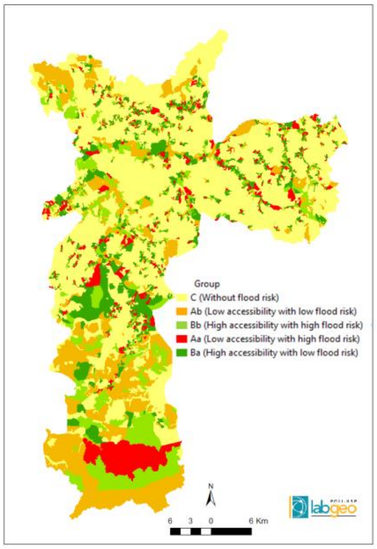

Group
C (Without flood risk)
Ab (Low accessibility with low flood risk)
Bb (High accessibility with high flood risk)
Aa (Low accessibility with high flood risk)
Ba (High accessibility with low flood risk)
N
6 3 0 6 Km
labgeo

Figure 12 - Accessibility to cultural facilities and flood risk areas

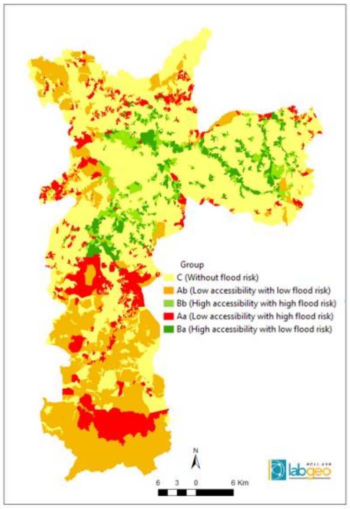

Group
C (Without flood risk)
Ab (Low accessibility with low flood risk)
Bb (High accessibility with high flood risk)
Aa (Low accessibility with high flood risk)
Ba (High accessibility with low flood risk)
N
6 3 0 6 Km
labgeo

Figure 13 - Accessibility to health facilities and flood risk areas

The analysis with the results Census 2010 (Table 3) aims to show the differences between the already defined groups (Table 2) concerning the variables of socioeconomic, households, race and vulnerable groups.

The proportion of the population in each group shows that around 10% are living in areas with low accessibility to health facilities and public schools and are living close to flood prone area. Of these, 7% are located in regions with high level of flood risk. The socioeconomic variable of average income of the group with high flood risk (Aa and Ba) is lower than values of the groups with low and without risk (Aa, Bb, C). The difference between groups with high (Aa and Ab) and low accessibility (Ba and Bb) is considerable only in the accessibility to health facilities analysis. These differences demonstrate that the poverty dimension is more coincident to risk pattern than to accessibility levels of public schools and cultural facilities.

In the households' attributes analysis, the variables of water supply, garbage and energy system did not present a clear pattern of correlation. Although, in the analysis of the private bathroom and sewage system, the group with low accessibility (A) shows a median lower than the group located at areas with low accessibility (B) and without risk (C). Is not possible to confirm some correlation between risk or accessibility, however it is clear the characterization of the most vulnerable regions (A) as precarious with respect to the sewage infrastructure. Other remarkable works in the vulnerability assessments in Brazil (Alves, 2013; Hogan, 1993) present similar analysis focused on the income variable, relating environmental risk and poverty. According to Alves (2013), in Cubatão city it is possible to say that the level of attendance of sewage treatment systems is very different between groups and strongly related to environmental risk.

About the race variables, it is possible to conclude that, for all accessibilities measures, the percentage of white people is lower than pardo people percentage in the most vulnerable group (A). From this analysis, there is no evidence of correlation between high risk level in these variables, meanwhile, for the black race, the percentage is higher or equal between the groups with high flood risk (Aa and Ba).

The consolidated literature about vulnerability, shows as critical groups the families headed by a female, children and elderly. Among these groups, children are the most related with risk and low accessibility and also this is the most susceptible group to waterborne diseases (Alves, 2013).

Table 3 - The summary of the Census 2010 indicators according accessibility measure and flood risk group

Group		Socioeconomic		Households				Race				Vulnerable groups		
		1	2	3	4	5	6	7	8	9	10	11	12	13
Accessibility to health facilities	Aa	**7%**	**1134**	0,97	**0,79**	1,00	1,00	**0,51**	**0,08**	0,01	**0,40**	**0,20**	0,06	0,43
	Ab	**3%**	1554	0,81	**0,65**	0,92	0,94	**0,53**	0,06	0,02	**0,33**	**0,18**	0,07	0,41
	Ba	5%	**2007**	0,99	0,94	1,00	1,00	0,66	**0,06**	0,03	0,26	0,16	0,10	0,44
	Bb	3%	2551	0,98	0,89	0,99	0,99	0,67	0,05	0,03	0,24	0,15	0,11	0,44
	C	82%	2249	0,95	**0,88**	0,95	0,96	**0,60**	0,06	0,02	**0,27**	**0,16**	0,08	0,43
Accessibility to public schools	Aa	**7%**	**1552**	0,97	**0,81**	1,00	1,00	**0,54**	**0,07**	0,01	**0,37**	**0,20**	0,07	0,42
	Ab	**3%**	2148	0,82	**0,69**	0,93	0,94	**0,57**	0,06	0,02	**0,29**	**0,17**	0,08	0,41
	Ba	6%	**1509**	0,99	0,91	1,00	1,00	**0,61**	**0,07**	0,02	0,30	0,17	0,09	0,44
	Bb	2%	1703	0,98	0,86	0,98	0,99	0,61	0,06	0,02	0,29	0,17	0,09	0,44
	C	82%	2249	0,95	**0,88**	0,95	0,96	**0,60**	0,06	0,02	**0,27**	**0,16**	0,08	0,43
Accessibility to cultural facilities	Aa	**4%**	**1472**	0,99	**0,86**	1,00	1,00	**0,57**	**0,07**	0,02	**0,34**	**0,18**	0,08	0,42
	Ab	**2%**	1871	0,82	**0,67**	0,91	0,93	**0,56**	0,06	0,02	**0,29**	**0,17**	0,08	0,41
	Ba	8%	**1558**	0,98	0,86	1,00	1,00	**0,58**	**0,07**	0,02	0,33	0,18	0,08	0,43
	Bb	3%	2076	0,94	0,83	0,98	0,99	0,61	0,06	0,02	0,29	0,17	0,09	0,43
	C	82%	2249	0,95	**0,88**	0,95	0,96	**0,61**	0,06	0,02	**0,27**	**0,16**	0,08	0,43

1. Proportion of population
2. Average income
3. Proportion of residences with water supply system
4. Proportion of residences with private bathroom and sewage system
5. Proportion of residences with garbage system
6. Proportion of residences with energy system
7. Proportion of residents from white race
8. Proportion of residents from black race
9. Proportion of residents from yellow race
10. Proportion of residents from pardo race
11. Proportion of population less than 11 years
12. Proportion of population 65 years or older
13. Proportion of residences headed by a female

4. Conclusion

This work is an exploratory analysis of the relation between risk and accessibility as a measure of social exclusion in the context of the vulnerability assessment. Some initial perception about the relation between transportation service level and the disadvantage by the lack of access and flood risks, helped to stablish the hypothesis and basic motivation for the spatial analysis.

The indicators of accessibility to urban facilities and flood risk are combined to compose groups with high and low level of attendance at flooded and non-flooded areas. It is possible to conclude that areas with low accessibility and risk are located in the outskirts of São Paulo city and present a different pattern according to the type of the facilities. The south region presents low and high levels of flood risk combined with low level of accessibility to public schools and health centers. In the east and central regions, mainly along Aricanduva river, there are some areas with high level of flood risk and also high level of accessibility to cultural facilities.

Some results of Census 2010, as socioeconomic, households, race variables and vulnerable groups are brought for discussion of the differences between groups. The accessibility to public schools and health facilities presents, in general, more discrepancy between the groups, while the accessibility to cultural facilities, presents more homogenous values. Besides that, the high-risk areas are characterized by low income level. The low percentage of residents with private bathroom and sewage system is typical of areas with low level accessibility and close to flood prone areas. Such areas are also characterized for higher percentages of children and people from pardo race and lower percentages of white people.

It is important to remark that these relations are only valid for accessibility considering the public transportation and flooding risk, therefore, it is not enough for evaluating all the vulnerability relations. For further developments, it could be tested others risks and equity values. Also improvements in the flood risk indicators could be made, for instance, considering the return period and the respective variation of the flooded areas, as well as interpolation of water surfaces and intersection with digital elevation model (Apel, Aronica, Kreibich, & Thieken, 2009). Regarding the accessibility and its relation with equity (Neutens et al., 2010), it would be interesting to consider the competition and more sophisticated measures. The formulation of vulnerability index with different technics as Principal Component Analysis and other weighting methods (Beccari, 2016) are further methods to be explored.

5. Acknowledgements

The authors acknowledge Julia de Mattos Cavalcanti for the discussion about Jardim Pantanal area and to Beatriz Moura for the help with the network. The authors also acknowledge the Coordination for the Improvement of Higher Education Personnel (CAPES) for graduate scholarships financial support.

6. References

Alves, H. P. D. F. (2013). Análise da vulnerabilidade socioambiental em Cubatão-SP por meio da integração de dados sociodemográficos e ambientais em escala intraurbana. *Revista Brasileira de Estudos de População*, *30*(2), 349–366. http://doi.org/10.1590/S0102-30982013000200002

Anazawa, T. M., Feitosa, F. da F., & Monteiro, Â. M. V. (2013). Vulnerabilidade socioecológica no litoral norte de São Paulo: medidas, superfícies e perfis de ativos. *Geografia*, *38*(1), 189–208.

Apel, H., Aronica, G. T., Kreibich, H., & Thieken, A. H. (2009). Flood risk analyses - How detailed do we need to be? *Natural Hazards*, *49*(1), 79–98. http://doi.org/10.1007/s11069-008-9277-8

Beccari, B. (2016). A Comparative Analysis of Disaster Risk, Vulnerability and Resilience Composite Indicators. *PLoS Currents Disasters*, *14*(1). http://doi.org/10.1371/currents.dis.453df025e34b682e9737f95070f9b970

Companhia do Metropolitano de São Paulo. (2007). Pesquisa Origem Destino. Retrieved from http://www.metro.sp.gov.br/

Cutter, S. L. (1996). Vulnerability to environmental hazards. *Progress in Human Geography*. http://doi.org/10.1177/030913259602000407

Cutter, S. L., Boruff, B. J., & Shirley, W. L. (2003). Social Vulnerability to Environmental Hazards. *Social Science Quarterly*, *84*(2), 242–261. Retrieved from http://gcal.summon.serialssolutions.com/search?s.q=Social+Vulnerability+to+environmental+hazards

Department for Transport Business Plan. (2012). *Accessibility Statistics Guidance* (Vol. 2012).

Hogan, D. J. (1993). População, pobreza e poluição em Cubatão, São Paulo. In G. (Org) MARTINE (Ed.), *População, meio ambiente e desenvolvimento* (pp. 101–131). Campinas: Editora Unicamp.

Hogan, D. J., & Marandola, E. (2005). Towards an interdisciplinary conceptualisation of vulnerability. *Population, Space and Place*, *11*(6), 455–471. http://doi.org/10.1002/psp.401

IPCC. (2014). *Climate Change 2014: Impacts, Adaptation, and Vulnerability. Part B: Regional Aspects. Contribution of Working Group II to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. (and L. L. W. Barros, V.R., C.B. Field, D.J. Dokken, M.D. Mastrandrea, K.J. Mach, T.E. Bilir, M. Chatterjee, K.L. Ebi, Y.O. Estrada, R.C. Genova, B. Girma, E.S. Kissel, A.N. Levy, S. MacCracken, P.R. Mastrandrea, Ed.)*IPCC*. United Kingdom and New York, NY, USA: Cambridge University Press. http://doi.org/10.1017/CBO9781107415324.004

Kowarick, L. (2002). Viver em risco: sobre a vulnerabilidade no Brasil urbano. *Novos Estudos*, (63), 9–30.

Lucas, K. (2012). Transport and social exclusion: Where are we now? *Transport Policy*, *20*, 105–113. http://doi.org/10.1016/j.tranpol.2012.01.013

Lucas, K., van Wee, B., & Maat, K. (2015). A method to evaluate equitable accessibility: combining ethical theories and accessibility-based approaches. *Transportation*. http://doi.org/10.1007/s11116-015-9585-2

Moss, R. H., Brenkert, a L., & Malone, E. L. (2001). Vulnerability to climate change: a quantitative approach. *U.S. Department of Energy, Oak Ridge, TN*, (September), 1–88.

Neutens, T., Schwanen, T., Witlox, F., & de Maeyer, P. (2010). Equity of urban service delivery: A comparison of different accessibility measures. *Environment and Planning A*, *42*(7), 1613–1635. http://doi.org/10.1068/a4230

Páez, A., Scott, D. M., & Morency, C. (2012). Measuring accessibility: Positive and normative implementations of various accessibility indicators. *Journal of Transport Geography*, *25*, 141–153. http://doi.org/10.1016/j.jtrangeo.2012.03.016

Prefeitura de São Paulo. (2016). Geosampa. Retrieved from http://geosampa.prefeitura.sp.gov.br/

Preston, B. L., Yuen, E. J., & Westaway, R. M. (2011). Putting vulnerability to climate change on the map: A review of approaches, benefits, and risks. *Sustainability Science*, *6*(2), 177–202. http://doi.org/10.1007/s11625-011-0129-1

Smit, B., & Wandel, J. (2006). Adaptation, adaptive capacity and vulnerability. *Global Environmental Change*, *16*(3), 282–292. http://doi.org/10.1016/j.gloenvcha.2006.03.008

Tomasiello, D. B. (2016). *Modelos de rede de transporte público e individual para estudos de acessibilidade em são paulo*. Dissertação de mestrado. Departamento de Engenharia de Transportes. Universidade de São Paulo.

Turner, B. L., Kasperson, R. E., Matson, P. A., McCarthy, J. J., Corell, R. W., Christensen, L., … Schiller, A. (2003). A framework for vulnerability analysis in sustainability science. *Proceedings of the National Academy of Sciences of the United States of America*, *100*(14), 8074–8079. http://doi.org/10.1073/pnas.1231335100

Vignoli, J. R. (2000). *Vulnerabilidad demografica: una faceta de las desventajas sociales*. Santiago de Chile.

Wee, B., & Geurs, K. (2011). Discussing equity and social exclusion in acessibility evaluations. *European Journal of Transport and Infrastructure Research*, *11*(4), 350–367.

Weis, S. W. M., Agostini, V. N., Roth, L. M., Gilmer, B., Schill, S. R., Knowles, J. E., & Blyther, R. (2016). Assessing vulnerability: an integrated approach for mapping adaptive capacity, sensitivity, and exposure. *Climatic Change*, *136*(3–4), 615–629. http://doi.org/10.1007/s10584-016-1642-0

Travel History: Reconstructing Travelers´ Trajectories Using Digital Footprints

Amon Veiga Santana[1] and Jorge Campos[1,2]

[1]Graduate Program in Systems and Computation – UNIFACS

[2]Bahia State University – UNEB

Salvador, Bahia, Brazil

amoncaldas@yahoo.com.br, jorge@unifacs.br

Abstract: *This work proposes a solution for reconstructing travel histories using heterogeneous social track posts in social networks, GPS positioning data, location history data generated by cloud services or any digital footprint with an associated geographic position. The solution encompasses a conceptual model; a methodology to reconstruct travel histories based on heterogeneous social tracks sources; and an application to present the reconstructed travel itinerary in a graphical and interactive fashion. An experiment conducted with real travelers showed that the proposed solution is a reasonable way to reconstruct semantic-rich travel histories in an automatic fashion.*

1. Introduction

The popularization of Online Social Network (OSN) and User Generated Content (UGC) have modified the way people search, find, read, access, and share information on the Internet (Ye et al. 2011). OSNs have an important role in the production and search for information. OSN users' are frequently involved in activities to find relevant contents, advices, opinions, or to simply interact with their mates to have fun (Lange-faria and Elliot 2012). UGCs (e.g., posts in social networks and comments in websites and forums) have become an important and recognized source of information in the tourism domain (Akehurst 2009). Travel specialized websites, for instance, have increased its sociability and usage by adopting mechanisms that facilitates content sharing in real time between users. A 2011 PhocusWright report[1] shows that nine of ten cyber travelers read and trust online reviews in touristic related sites. Unfortunately, there are far more people willing to consume this kind of content, than people disposed to generate them (Pan and Crotts 2012). It is because most people see UGC as a time consuming and boring task, but they will not mind to contribute if there exists some kind of application or service that captures their contribution in an automatic fashion.

A special kind of information incorporated by most OSNs that has attracted the attention of the travel and tourism community is the users' position while they are moving. The increasing number of location-enabled devices opens the possibility of making the position of the user a mandatory piece of information to virtually any kind of social interaction or user generated content. Moreover, the capability of keeping track of the position of a user at high detailed levels opens the possibility to combine traveler's

[1] http://www.researchandmarkets.com/reports/1866967/phocuswrights_social_media_in_travel_2011

trajectory data and georeferenced social interactions to produce, in an automatic fashion, a structured and semantic rich dataset of traveler's preferences and behaviors.

This paper introduces Travel History, a conceptual model and a methodology to reconstruct the trajectory of travelers based on records of their position and their interactions posted on social networks. Position information may vary from the usual detailed GPS logs to any evidence of places visited by the traveler and recovered from the traveler social network repository. Thus, Travel History model supports the representation of the trajectory with different levels of granularities mixed and interleaved with travelers' social interactions.

The remainder of this paper is structured as follows: Section 2 discusses related work. Section 3 presents the Travel History Model. Section 4 discusses the techniques used to instantiate entities of the model. Section 5 introduces a prototype tool and presents some results of an experimental evaluation with real travelers' volunteers. Section 6 presents conclusions and indicates future work.

2. Related work

The analysis of trajectories of moving objects has been intensively discussed by the GIS community over the last decade. Fed initially by the profusion of data captured from sensors and location devices, studies in this field have evolved from the generation of trajectories using GPS raw trajectory data to the use of novel means to enrich trajectories semantically. One salient new source of information comes from the growing habit among users to interact in social networks, posting, commenting, or sharing contents that contain geographic references. This source of information has proven its value for many different fields and purposes. It is of special interest of this work the combination of trajectory semantic enrichment techniques and georeferenced post in social networks to produce semantic rich set of information about travelers and their visits.

Semantic enrichment and annotation in trajectory data are very active research topics. (Spaccapietra et al. 2008) proposed the first model that treats trajectories of moving objects as a spatiotemporal concept. The *Stops* and *Moves* model is one of the most accepted model to represent trajectory of moving objects. (Andrienko, Andrienko, and Wrobel 2007) used the time for adding semantic annotation to the stationary part of a trajectory and argued that the more time is spent in a place more important it is the place to a person. In the (Zheng et al. 2009)'s work, it was exposed a technique that considered, beyond the spending time, the geographic coincidence with Points of Interest (POI) defined in the application. (Zheng et al. 2010) proposed a technique based on speed, acceleration and the orientation of the user to detect the transportation mode used to move from one place to another. A comprehensive set of solutions for semantic annotation of heterogeneous trajectories can be found in (Yan et al. 2011) and for semantic trajectories modeling and analysis in (Parent et al. 2013).

Researches in the trajectory domain provide a solid base for the development of effective solutions to extract information from raw trajectory data. In another branch, several initiatives focus in pattern and knowledge discovery from User Generated Georeferenced Content (UGGC). In our context, UGGC is defined as a UGC that carries some kind of information that allows the identification of the geographic location of the related content, not necessarily the location of the user. A georeferenced picture of

Copacabana beach posted in Instagram and a Web review made by someone in New York about the Copacabana Palace Hotel, for instance, are both examples of UGGC of the same geographic region. UGCC do not have the same spatial granularity of positioning devices, such as GPS, but allows a more refined semantic extraction.

Related with initiatives that deal with UGCC for semantic enrichment, (Ji et al. 2009)(Hao et al. 2010) proposed a solution for mining city attractions from touristic blogs posts, (Rattenbury, Good, and Naaman 2007) proposed an approach to extract semantic from georeferenced picture posted in the Flickr social network, and (Gao et al. 2010) proposed a method to identify touristic attractions from Flickr's georeferenced pictures and to enrich the description of such attractions with information extracted from collaborative websites like Yahoo Travel Guide[2] and WikiTravel[3]. (Lu et al. 2010) proposed a picture-based customized trip planning. This system allows trip planners to specify personal preferences and generates travel routes from geo-tagged photos. The proposed solution is limited to surrounding attractions in a given city or region and does not support travel plans lasting more than one day and involving multiple destinations. (Yoon et al. 2012) proposed a framework for itinerary social recommendation using trajectories generated by local residents and expert travelers.

Despite the enormous potential of aforementioned initiatives, few works have combined the use of trajectory data and UGCC in the process of trajectories reconstruction and semantic enrichment. (Gil et al. 2014) proposed a method for trajectory annotation based on the spatiotemporal compatibility of Twitter posts. Although the spatial component of the post is mentioned in the work of (Gil et al. 2014), the proposed methodology takes into account only the temporal compatibility between trajectory data and Twitter posts, that is, they do not use the posts' content or location to enrich the trajectory semantically.

Analyzing early related work, it is noticeable that most solutions used detailed logs of position devices to analyze people's movements. This tendency is switching to incorporate location information embedded in social interactions and stored in the cloud. In a social post, for instance, the position comes with some kind of information or even a personal opinion about the place visited. Thus, trajectory reconstruction using georeferenced social interactions can be the strategy to recover context elements and semantics of trajectory. As the process of trip planning involves information search and retrieval, it is natural that travelers also look for this kind of information among their friends and people from their social circle. At the best of our knowledge, however, there is no service that, considering previous users' experience registered as social tracks, offers efficient means for travelers to access information about structured travel itineraries, including attractions and transportation means. Next section introduces an attempt to represent this kind of information.

3. Travel History

Travel History conceptual model encompasses all information needed to represent relevant actions and movements of a traveler. The central entity of the model is *Travel History*, which is

[2] https://www.yahoo.com/travel/guides

[3] http://wikitravel.org

an entity that aggregates *Stays* and *Trails* traveled by an individual during a given time interval. Figure 1 shows the relationship among models entities using UML notation.

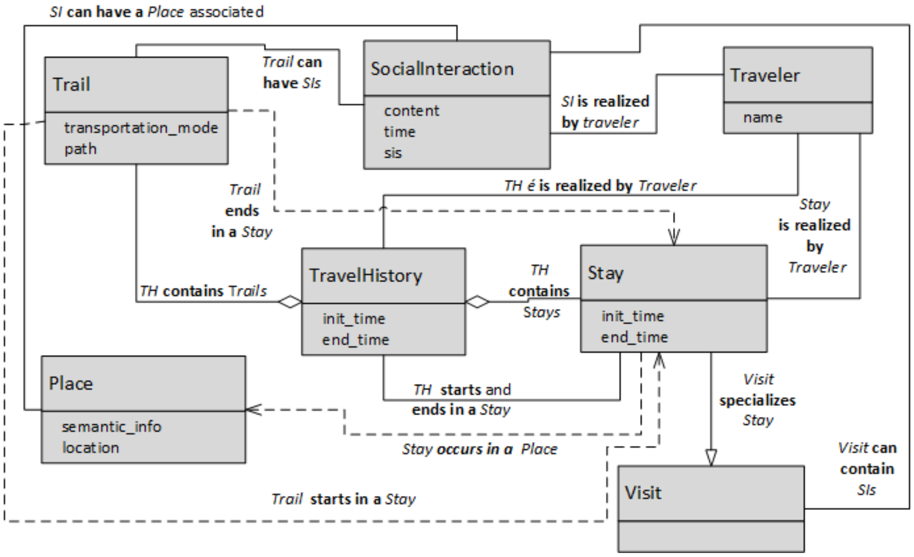


Figure 1: Travel History Conceptual Model.

A *Trail* is an entity that captures the traveler movement. Each *Trail* has an associated path and a transportation mode. The path is a collection of geographic points that represents the geometry of the movement. The path may vary from a pair of points indicating only the endpoints of the movement up to a collection of points representing the detailed path fulfilled by the traveler. The transportation mode indicates how the *Traveler* goes from one place to another (walking, by train, etc.).

Stays represent the places where the *Traveler* remained for a while or change the transportation mode. Each *Stay* occurs at a *Place*. A *Place* is a geographic location plus some semantic incorporated, like a description, political categorization or a combination of them. Consider, for instance, the place where the traveler stops walking and takes a cab. This place is represented as a *Stay* only because the change in the transportation mode. If the traveler remains around a place for a while, these place also becomes a *Stay* in our model.

A *Stay* may or may not have a special meaning for the trip. When the *Traveler*'s permanency at some place is relevant for the trip, the *Stay* is specialized and becomes a *Visit*. The relevance of the Stay takes into account the amount of time spent at the place or the amount of social interactions related to the place. Thus, a *Visit* represents a place where the *Traveler* has made and registered some social interaction or stayed for certain amount of time above a given threshold.

Visits and *Trails* may have one or more *Social Interactions*. These interactions are contents that help to understand Traveler's intentions or activities. The association of *Visits* and *Trails* with *Social Interactions* considers both temporal and geographical matches. Temporal matches consider the time of the realization of the *Social Interaction*, (i.e. the interaction is associated with a *Stay* or a *Trail* that is going on at the time of the occurrence of the post). Geographical matches occur when a *Social Interaction* has some geographic information associated to its content (i.e., it is an UGGC). In this case, the location of the interaction is determinant to establish the association between the *Social*

Interaction and the respective *Stay* or *Trail*. The amount of *Social Interactions* related to a *Stay* or *Trail* is also an indicator of the place's relevance for the trip.

Travel History model was conceived aiming at handling multiple types of UGGCs retrieved from different OSNs and combined with any sort of positioning data about the user movement. Next section discusses the mechanism of converting these kinds of heterogeneous data into entities of the model.

4. Rebuilding Travel Histories

The *Travel History* reconstruction process is based on heterogeneous sources of data. Sometimes it is available a very fine set of registers of a traveler's movement captured by some kind of position device. Other times there is a less fine position records, but the data comes with some kind of semantics attached, and not rarely, there is a social interaction that can be used as a source of information about the travel. Even coming from different sources, these datasets share common concepts and structures. Thus, they were grouped in three main categories: 1) Raw Trajectory Data (RTD), 2) Semantic Trajectory Data (STD), and 3) Social Interactions. When a Social Interaction has an associated geographic location, it is called Georeferenced Social Interaction (GSI).

RTD and STD are sequences of spatiotemporal records. Although they share the same basic structure, they differ significantly considering both the spatial-temporal granularity and content. RTD are generated by positioning devices and contains only registers with the position and the timestamp of each reading. RTD is usually obtained from a single device and store the data as they are produced. STD, on the other hand, is a result of preprocessed data acquired from many different sources. This information comes from some kind of cloud service such as "Google Takeout", which allows users to recover information about their movement. Users have to authorize Google to keep track of their whereabouts and Google uses this information in lots of different services. Although RTD records are denser than STD records, the semantic of the later is much more relevant to the trajectory reconstruction than the former. The last category of source of information is GSI. GSI records are even sparser than RTD and STD records. Thus, GSI alone contributes little to reconstruct the detailed trajectory geometry, but they are very important to enrich the trajectory semantically.

The process of rebuilding *Travel Histories* goes from gathering all pieces of information to the semantic enrichment of models entities. This process can be split in three phases: data acquisition; data processing; and entities generation (Figure 2). In the first phase, data are acquired from different sources, like social networks, location history web services, and location's tracks recorded in the user device. In the second phase, the data are processed to identify *Stay* candidates and transportation modes. In the last phase, *Stays*, *Visits* and *Trails* are generated and semantically enriched.

The data acquisition phase starts with the definition of the time window when the travel happened. After defining the temporal window of the trip, the reconstruction process continues by gathering all relevant information about user movement and his/her *Social Interactions*. The data acquisition strategy depends on the category of the sources available. RTD and STD are collected as a single file. RTD records come from mobile applications that continuously record the position of the travelers over time. These records are stored in the device internal memory and can be imported at any time. STD records come from cloud services (like Google Takeout). These records are requested by the owner of the data, the only person able to retrieve them. RTD and STD files are traversed and relevant information is extracted and stored in a local database. The process of gathering social interactions, on the other hand, can be fully automated. OSN's users authorize a computer application to search and retrieve all social interactions of a given period. These data are also stored in a local database.

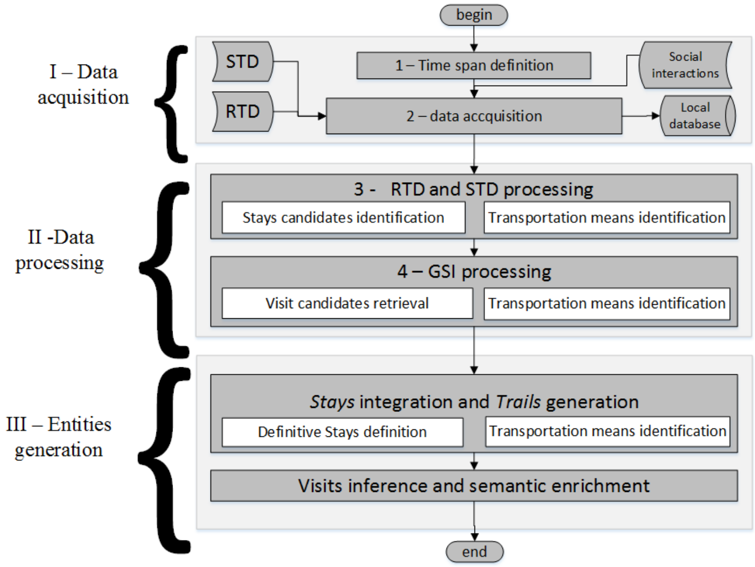

begin
I – Data acquisition
STD
RTD
1 – Time span definition
Social interactions
2 – data accquisition
Local database
II -Data processing
3 - RTD and STD processing
Stays candidates identification
Transportation means identification
4 – GSI processing
Visit candidates retrieval
Transportation means identification
III – Entities generation
Stays integration and Trails generation
Definitive Stays definition
Transportation means identification
Visits inference and semantic enrichment
end

Figure 2: The three phases of the Travel History generation process.

In order to illustrate the reconstruction process, consider a hypothetical travel with samples of information gathered from the traveler's digital footprints. Figure 3 shows a travel timeline with samples of information where, in the first segment there is only intermittent RTD records. In the second segment, RTD and GSI (content posted in social networks like Facebook or Instagram) are available. In the last segment, a combination of RTD, STD and GSI is available and in some parts they overlap.

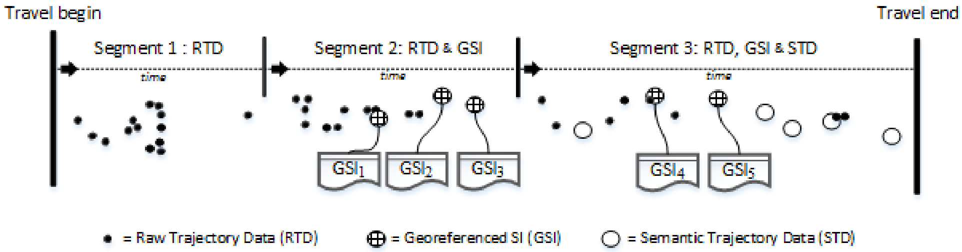

Travel begin
Travel end
Segment 1 : RTD
Segment 2: RTD & GSI
Segment 3: RTD, GSI & STD
time
time
time
GSI1
GSI2
GSI3
GSI4
GSI5
● = Raw Trajectory Data (RTD)
⊕ = Georeferenced SI (GSI)
○ = Semantic Trajectory Data (STD)

Figure 3: Graphical presentation of RTD, STD an GSI records.

With all pieces of information in place, the second phase of the reconstruction process starts by identifying *Stays* candidates and transportation modes between these candidates. *Stays* candidates represent the locations where the traveler hangs around for a while or change the transportation mode.

Stays candidates are generated considering the category of the data source to be processed. To detect *Stays* from RTD and STD sources, it was developed an algorithm capable of recognizing these entities based on the geometric configurations of the track (clusters of points or isolated points) and based on some semantic information already present in the data. When a traveler stays in a place, a cluster of points (i.e., a dense formation of points) is formed. The algorithm identify this formation of points and group these points to form a *Stay* candidate (Figure 4 - case A). On the other extreme, isolated points also becomes a *Stay* candidate (Figure 4 - case B). This case occurs when there is a record distant from both the previous and the next point in the sequence and it is not considered an outlier. Outliers are treated during the pre-processing phase of the reconstruction process. Most outliers are disregarded based on the physical unviability for a traveler being at a certain place considering, for example, the maximum speed of known transportation means. Outliers are disregarded form are disregarded from the dataset and do not reach this phase of the reconstruction process.

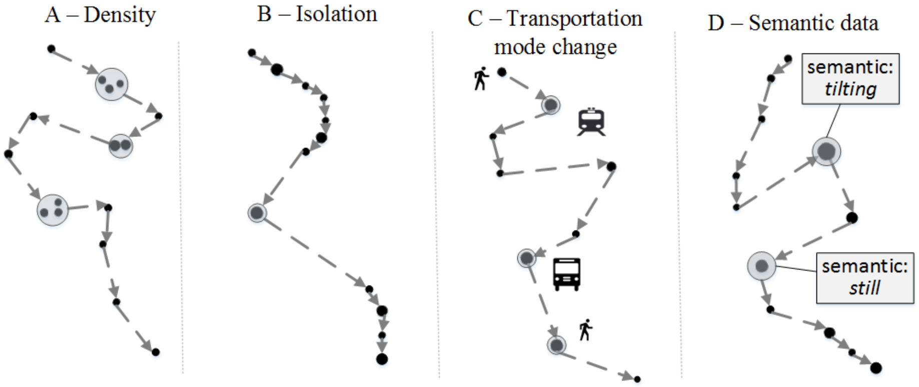


Figure 2: Stays identification techniques based in RTD and STD and the result when it is integrated with Social Interactions.

While processing RTD files, the transportation mode used between two *Stays* candidates is also computed. The definition of the transportation mode takes into account the following aspects: speed, speed variation, acceleration, orientation variation and continuity. Each transportation mode has a single combination of these factors. By taking them together, it is possible to infer how the *Traveler* moved between *Stays*. Due to the lack of space, it is abstracted away the details of determining transportation modes. *Stays* are also defined at every location where a transportation mode change occurs (Figure 4 - case C). Finally, a *Stay* can be inferred from the semantic already embedded in STD files. These data sometimes have semantic information like “still” or “tilting” associated with a place. (Figure 4 - case D). These places always become a *Stay* candidate.

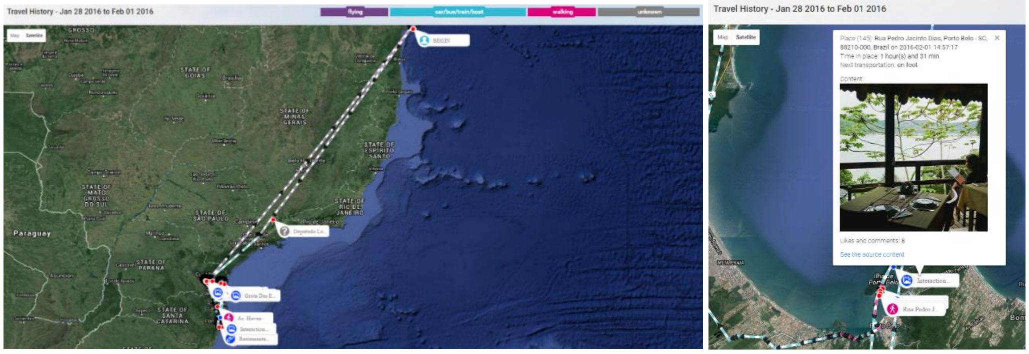
Stay candidates are also generated considering GSI information. In this case, the rule is simple, that is, every GSI generate a *Stay* candidate. Later, some of these *Stays* will be grouped, becoming a single *Visit,* others will not be confirmed as a *Stay* and will become a social interaction of a *Trail*. All *Stays* candidates, no matter the source of information, are stored in a common persistence entity.

During the last phase of the reconstruction process, all high-level entities of the model representing parts of the travel history are generated, integrated and semantically enriched. Throughout the integration step, issues related to the duplicity and overlaps are solved**.** At this point, each *Stay* candidate is processed, confirmed as a definitive *Stay*, promoted to a *Visit,* or merged with others *Stays*. Since *Stays* candidates are generated from different sources separately, it is possible that some *Stays* candidates refer the same event of the trip. The *Stays* merge process occurs when the distance between two *Stays* is less than a given threshold.

The next step in the reconstruction process is the *Trails* generation. *Trails* connect two *Stays* and describe the *Traveler* movement between them. During *Trails* generation, it is necessary to identify the transportation mode. If the *Stays* were generated from the same source, the transportation mode between them has been already defined in the second phase, but if the *Stays* were generated based on different sources, the same algorithm to detect transportation mode discussed earlier is used. The last step of the phase of the reconstructing process is the semantic enrichment of *Trails* and *Visits*. *Visits* are specialized versions of *Stays*. For a *Stay* to become a *Visit*, it is considered the amount of time spent on the site and the number of *Social Interactions* carried out by the traveler. Once all model entities are instantiated, an application using geovisualization techniques can easily depicts the graphical realization of the reconstruct travel history.

5. Experiment and results

To evaluate the *Travel History* model, it was developed a prototype tool that employs all techniques presented in this paper (available at http://th.fazendoasmalas.com). The prototype allows the acquisition, processing and generation of *Travel Histories*. At the end, the tool shows the user travel history in an interactive map. *Stays*, *Visits*, and *Trails* are presented in a graphical and user-friendly web application.



Place (145): Rua Pedro Jacinto Dias, Porto Belo - SC, 88210-000, Brazil on 2016-02-01 14:57:17
Time in place: 1 hour(s) and 31 min
Next transportation: on foot

Content:

Likes and comments: 8

See the source content

Figure 5: Graphical presentation of a Travel History generated using RTD, STD and GSI data sources: an overview of the entire trip and a detailed view of part of the trip.

Figure 5 shows the reconstruction of a Travel History generated using different and rich sets of information. This travel occurred between January 28 and February 1, 2016. It was a five days' trip in the south part of Brazil, including a visit to the capital of Paraná, a train trip between the capital and some place along the state coastline, a visit to Mel Island in the Paranaguá Bay, and a visit to the state of Santa Catarina, including the capital Florianópolis and other small towns nearby. To reconstruct this travel history, it was used as source of information GPS log files, location history files generated by Google and the online social networks Facebook, Instagram and Twitter. When RTD, STD, and GSI are all available to reconstruct the travel, it is possible to zoom in the map presentation to analyze the detailed path of the traveler and to visualize comments and social interactions posted along the path. Figure 5 shows on the left side an overview of the entire trip and on the right side a detailed view of part of the trip to Parana and Santa

Catarina and a box with the one of the GSI posted about the trip. We used this application to support the realization of an experimental evaluation of the proposed model and methodology. Next section discusses the result of such an experiment.

Volunteers from RBBV (acronym, in Portuguese, for Brazilian Travel Bloggers Network) have been invited to use the tool, submit their data, reconstruct their travels and evaluate the travel history generated by the application. A total of 58 volunteers started the experiment, but only 23 travelers completed the entire process successfully. Some volunteers did not submitted data, others submitted inconsistent data, and some did not perform the evaluation. The volunteers were oriented to answer a questionnaire after analyzing and exploring their reconstructed travel. The questions presented to the volunteers aimed at verifying the level of satisfaction with the accuracy and similarity of the Travel History created based on their digital footprints when compared with the events and destinations of the real trip they have made. An interactive map allowed volunteers to check visited *Places*, analyze the performed *Trails*, verified transportation means used, and examine associated semantics. The tool used to present the graphical realization of the trip was not evaluated.

In the process of evaluating the travel reconstruction process, the volunteers have answered five questions. For each question, volunteers were asked to give a grade ranging from 0 to 10. The results of the experiment are presented in the Table 1.

The first question aims to evaluate the accuracy of *Visits* identification. The result for this question indicates that most of the evaluators (~79%) considered that the identification of *Visits* was completely accurate or precise in most cases, while ~13% felt it was accurate in some places, and 8% found the process of *Visit* identification was slightly or completely inaccurate.

The second question measures the satisfaction with temporal order of visits and trails. This question is related to the integration process of *Stays* and *Visits*, which is responsible for identifying overlaps, to perform merges, and to sort these entities. The result indicates that most evaluators (82.6%) considered that the order of visits and trail order has been completely precise or precise in most cases, 13.05% found that the identification of the order was accurate in some places and only one evaluator (4.31%) considered that the order of *Visits* and *Trails* was imprecise.

Table 1: Travel History reconstruction evaluation results

Aspect analyzed	Avg. grade	Standard Deviation	Totally accurate or in most of the cases
Visits identification accuracy	7,71	2,7	78,25%
Visits and *Trails* order accuracy	7,82	2,03	82,6%
Transportation means identification accuracy	7,39	2,19	69,56%
Activities and semantic identification accuracy	7,06	1,79	74%
Travel History rebuilt represents the real travel made.	8,69	2,14	95,65%
Averages considering all aspects	***7,73***	***2,17***	***80,01%***

The third question evaluates the accuracy of identifying the transportation mode used in each *Trail*. Although the level of satisfaction with the identification of the transportation mean is close to 70%, this aspect has the worst evaluation on the survey. The identification of the transportation mode is directly linked to the existence and granularity of RTD and STD sources and the accuracy of the location of GSI. It is noticeable that the identification of transportation mean improves when interactions in OSNs are made in real time during the trip.

The fourth question is more subjective and it is related to the accuracy of the semantic enrichment process. In this regard, 73.9% of the evaluators answered that the identification of activities and interactions was completely accurate or accurate in most cases. The semantic enrichment process can be improved by incorporating the capability of including textual content of Social Interactions and with the ability to access structured information about users' activities. Facebook, for example, has such kind of information, but, at the time of writing, it is not possible to access such kind of information using third-party applications.

The fifth question evaluated the overall perception of the reconstruction process. It is by far the best-rated item of the survey. Almost 96% of the evaluators considered that the Travel History reconstructed represents, totally or in the major part, the travels they have made. Taking all aspects together, about 80% are satisfied with the proposal of reconstructing semantic trajectories based on heterogeneous social tracks sources.

6. Conclusion

Considering the digital socialization growth and the search for online social recognition, travelers begin to demand ways to share their travel experiences in a systematic and intuitive way. This paper proposes conceptual and data models and a methodology to reconstruct semantic-rich traveler's trajectories. The central entity of the model is *Travel History*, which is an entity that aggregates *Stays*, *Visits* and *Trails* traveled by an individual during a given time interval. A *Trail* is an entity that captures the traveler movement and the transportation mode used. A *Stay* represent places where the *Traveler* remained for a while or changes the transportation mode. A *Stay* becomes a *Visit* if it is a place of intense online social interaction or if it is a place where the traveler spent a considerable amount of time.

Model's entities are instantiated based on a myriad of sources of information, varying from detailed low-level GPS registries and going up to high-level georeferenced social interactions. Thus, the proposed methodology used to generate models entities and to identify transportation mode is based on techniques to process, analyze, and integrate data with different levels of semantic and spatial-temporal granularity. The ability to reconstruct the trip successfully is directly related to the quality and quantity of the sources of information available. Different, reliable, and abundant sources of information will produce rich and accurate travel histories. On one hand, RTD and STD are good sources of information for detailed analysis of the trip. On the other hand, GSI generates semantic richer entities. As expected, the combination of all sources produces the best result.

In order to evaluate the proposed model and methodology, an experiment with travelers from a social traveler's network was designed and run. The results of the experiment show an overall level of satisfaction of 80%, considering the identification of the model's entities (i.e., *Stays*, *Visits* and *Trails*), the temporal order of these entities, the identifications of the transportation mode used by the traveler, the identification of activities performed by the traveler during the trip, the semantic enrichment of travelers' activities, and the level of adherence of the modeled travel history with the real trip.

As future work, there are several aspects that can be investigated. Semantic enrichment, for instance, can be improved by incorporating text-mining algorithms. Moreover, a travel social media ontology can be developed to improve semantic identification. To improve the data accuracy and granularity, mobile applications can be used to collect other kinds of social interaction, like offline media capture or any other type of interactions on the device. Algorithms for transportation means identification can be improved to become more accurate and to support the identification of other kinds of transportation.

The use of the proposed model and methodology in Web application in the tourism domain will allow the reconstruction of large number of Travel Histories, which in turn, can be a way to generate a knowledge base for travel itineraries, preferences, attractions and other aspects and events inherent to travels. This knowledge can be used as the base of a travel recommendation system or other initiatives such as urban planning, demographic and behavioral studies, intelligent transportation systems, social recognizing research, among others. Despite the fact that the model is generic and that it can be, in principle, used in several domains to describe semantic trajectories, the usage for other domains requires further investigations.

7. Acknowledgements

This work was partially supported by CAPES.

8. References

Akehurst, Gary. 2009. "User Generated Content: The Use of Blogs for Tourism Organisations and Tourism Consumers." *Service Business* 3 (1): 51–61. doi:10.1007/s11628-008-0054-2.

Andrienko, Gennady, Natalia Andrienko, and Stefan Wrobel. 2007. "Visual Analytics Tools for Analysis of Movement Data." *ACM SIGKDD Explorations Newsletter*. doi:10.1145/1345448.1345455.

Gao, Yue, Jinhui Tang, Richang Hong, Qionghai Dai, Tat-Seng Chua, and Ramesh Jain. 2010. "W2Go: A Travel Guidance System by Automatic Landmark Ranking." *Proceedings of the International Conference on Multimedia - MM '10*: 123. doi:10.1145/1873951.1873970.

Gil, Ricardo, Belther Nabo, Renato Fileto, Mirco Nanni, and Chiara Renso. 2014. "Annotating Trajectories by Fusing Them with Social Media Users ' Posts." In *Geoinfo Synposiun 2014*. http://www.geoinfo.info/proceedings_geoinfo2014.split/Paper04-F-p21.pdf.

Hao, Qiang, Rui Cai, Changhu Wang, Rong Xiao, Jiang-Ming Yang, Yanwei Pang, and Lei Zhang. 2010. "Equip Tourists with Knowledge Mined from Travelogues." *Proc. of the 19th International World Wide Web Conference*: 1–10. doi:10.1145/1772690.1772732.

Ji, Rongrong, Xing Xie, Hongxun Yao, and Wei-Ying Ma. 2009. "Mining City Landmarks from Blogs by Graph Modeling." In *ACM International Conference on Multimedia*, 105–114. doi:10.1145/1631272.1631289.

Lange-faria, Wendy, and Statia Elliot. 2012. "Understanding the Role of Social Media in Destination Marketing." *Tourismos: An International Multidisciplinary Journal of Tourism* 7 (1): 193–211.

Lu, Xin, Changhu Wang, Jiang-Ming Yang, Yanwei Pang, and Lei Zhang. 2010. "Photo2Trip: Generating Travel Routes from Geo-Tagged Photos for Trip Planning." *Proceedings of the International Conference on Multimedia - MM '10*: 143–152. doi:10.1145/1873951.1873972.

Pan, Bing, and John C. Crotts. 2012. "Theoretical Models of Social Media, Marketing Implications, and Future." *Social Media in Travel, Tourism and Hospitality: Theory, Practice and Cases* (1965): 1–19.

Parent, Christine, Stefano Spaccapietra, Chiara Renso, Gennady Andrienko, Natalia Andrienko, Vania Bogorny, Maria Luisa Damiani, et al. 2013. "Semantic Trajectories Modeling and Analysis." *ACM Computing Surveys* 45 (4): 42:1–42:32. doi:10.1145/2501654.2501656.

Rattenbury, Tye, Nathaniel Good, and Mor Naaman. 2007. "Towards Automatic Extraction of Event and Place Semantics from Flickr Tags." *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval SIGIR 07* pages: 103. doi:10.1145/1277741.1277762.

Spaccapietra, Stefano, Christine Parent, Maria Luisa Damiani, Jose Antonio, José Antônio De Macedo, Fábio Porto, and Christelle Vangenot. 2008. "A Conceptual View on Trajectories." *Data & Knowledge Engineering* 65.1 (May 2007): 126–146.

Yan, Zhixian, Dipanjan Chakraborty, Christine Parent, Stefano Spaccapietra, and Karl Aberer. 2011. "SeMiTri: A Framework for Semantic Annotation of Heterogeneous Trajectories." In *Proceedings of the 14th International Conference on Extending Database Technology (EDBT/ICDT '11)*, 259–270. doi:10.1145/1951365.1951398.

Ye, Qiang, Rob Law, Bin Gu, and Wei Chen. 2011. "The Influence of User-Generated Content on Traveler Behavior: An Empirical Investigation on the Effects of E-Word-of-Mouth to Hotel Online Bookings." *Computers in Human Behavior* 27 (2): 634–639. doi:10.1016/j.chb.2010.04.014.

Yoon, Hyoseok, Yu Zheng, Xing Xie, and Woontack Woo. 2012. "Social Itinerary Recommendation from User-Generated Digital Trails." *Personal and Ubiquitous Computing* 16: 469–484. doi:10.1007/s00779-011-0419-8.

Zheng, Yu, Yukun Chen, Quannan Li, Xing Xie, and Wei-Ying Ma. 2010. "Understanding Transportation Modes Based on GPS Data for Web Applications." *ACM Transactions on the Web* 4 (1): 1–36. doi:10.1145/1658373.1658374.

Zheng, Yu, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. 2009. "Mining Interesting Locations and Travel Sequences from GPS Trajectories." *Proceedings of the 18th International Conference on World Wide Web - WWW '09*: 791. doi:10.1145/1526709.1526816.

NSDI-compliant reference map: experiences on implementing a user-centered cartographic symbology and standardized data modeling at large scale (1:2000)

Vitor Silva de Araújo[1], Claudia Robbi Sluter[1], Silvana Philippi Camboim [1]

[1]Programa de Pós-graduação em Ciências Geodésicas– Universidade Federal do Paraná (UFPR)
Caixa Postal 19001 – 81.531-980 – Curitiba – PR – Brazil

vitorsilvadearaujo@gmail.com, robbisluter@gmail.com, silvanacamboim@gmail.com

Abstract. *Standardized urban maps are essential cartographic products to address the challenges of cities management. Brazilian NSDI (National Spatial Data Infrastructure) was created to facilitate geospatial data sharing and use, but its standards are not completely adapted for large scale mapping for urban areas. This work aims to propose data models and symbology based on existent standards to support the generation of reference maps at large scale (1:2,000) that is logically consistent, standardized and aligned to the principles of the theory of Cartography. These characteristics will allow these maps to be maintained by several producers, shared among institutions and, most importantly, understandable and meaningful for their users. The results of this project are a geospatial data model that includes some new classes & object, and their correspondent relations, needed in a large-scale mapping, and the related symbology for visualizing any map generated from that data model.*

1. Introduction

Although the legal base of the National Topographic Mapping in Brazil, the decree 243/67, only defines standards for systematic mapping at 1:25.000 scale and smaller, large scale reference maps are an important source of geoinformation to urban areas planning and management. According to the 2010 Census [IBGE 2010], 84% of Brazilian population lives in cities. An estimation from 2016 Brazilian population is that more than 94 million inhabitants live in 26 metropolitan areas with 1 million people or more [IBGE 2016]. This large population faces the issues that are common for cities around the world. In the UN-habitat 2016 World Cities report, the most important challenges are the growing number of urban residents living in informal settlements, the problem of providing urban services, climate change; exclusion and rising inequality and insecurity [UN-Habitat 2016]. Many of these challenges can be adequately addressed only when spatial data is available.

Large-scale standardized mapping is an important information source for the spatial analyses needed to propose solutions for urban management. Brazilian National Spatial Data Infrastructure (NSDI) was created in 2008 to facilitate geospatial data generation, use, and dissemination. However, the initiative is only mandatory for the Federal government, and, consequently, there are few examples of SDI (Spatial Data Infrastructures) in Brazil based on urban spatial data at state and local level government. The mapping of urban areas is the responsibility of local governments, and applying standards could help municipalities to exchange open format solutions to maximize the use of resources at the local administration level. State and federal level agencies, like the ones dealing with urban planning, can also take advantages from the use of standardized database models and cartographic solutions. Also, data sharing is crucial when dealing with adjacent urban areas, such as Metropolitan Regions.

This paper describes a research work that aims to propose a conceptual model, based on NSDI standards, and related symbology, for large scale (1:2000) mapping in Brazil. It is a part of an extensive research developed in the Cartography and GIS research group at the Federal University of Paraná.

Brazilian NSDI recommends that the product of reference mapping is a geodatabase which implementation must be based on the conceptual model called EDGV (in Portuguese, Estruturação de Dados Geoespaciais Vetoriais) [CONCAR 2010]. However, the latest version (2.1.3) of EDGV adopted by the National Commission on Cartography (in Portuguese, Comissão Nacional de Cartografia - CONCAR) is only suitable for 1:25.000 to 1:250.000. The Brazilian Army recently released another version, the EDGV Força Terrestre [DSG 2015], that, though not officially yet approved by CONCAR, detailed the data model at large scale.

Additionally, standard cartographic symbology is an important characteristic for any reference mapping. Although the digital technology allows us to use a high number of possibilities to interact with the geodata and to develop geoinformation analyses, the limitations of our (humans) visual perception and cognition is still a challenge in making the geoinformation visible. This issue is especially challenging as the standards for symbology are not yet updated for the NSDI environment.

The Brazilian reference mapping at large scales presents challenges related to symbology and database modeling. In this paper we describe a proposed solution for a conceptual model for a large-scale reference mapping and, also, a proposed set of cartographic symbols which solution is based on the EDGV classes and the T-34 700 Guide of Cartographic Symbols for Topographic Mapping at 1:25000 scales and smaller [DSG 1998]. The T-34 700 is an almost 20 years old guide, and although a representation standard is planned by DSG [DSG 2016], there is no official symbology specification aligned to the EDGV for any scale, including large scales.

To integrate the cartographic symbols to the EDGV classes by the theory of Cartography, we defined a set of cartographic features, and their classifications, that should be included in a large-scale reference mapping. In the second step of this research, we compared the EDGV and that list of classes and their definitions. We, then, designed a geospatial data model based on the identified similarities and differences. In the next step, we implemented this geospatial database for four cities in the State of Paraná. In the final step, we applied a proposed set of symbols to these datasets. Additionally, we stored the symbology using the OpenGIS® Styled Layer Descriptor (SLD) of the Open Geospatial Consortium [OGC 2007] to allow the utilization of the proposed symbology when the geospatial database is shared.

2. Materials and Method

This work began as a proposal for standard symbols for urban maps of the State of Paraná, Brazil. At that time, the graphic solution for topographic maps at large scales, 1:2000, 1:5000 and 1:10000 was not efficient, as we can notice in Figure 1. This kind of cartographic solution had been adopted in the State of Paraná, Brazil, for three decades as it was enforced by Paranacidade, the state-level agency for urban planning and development of the State of Paraná, Brazil. The State of Parana, through this agency, has funded reference mapping of several municipalities since the 1990s.

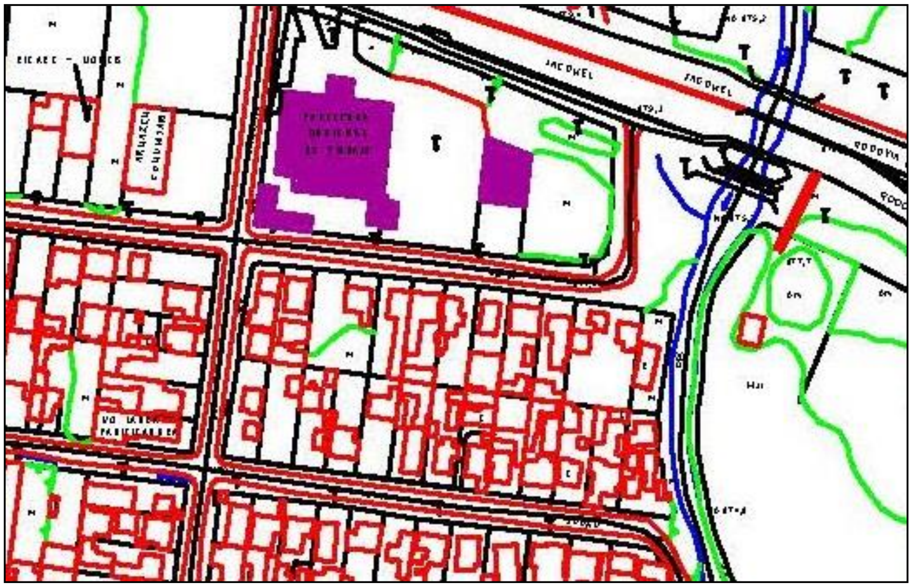


Figure 1. An example of the solution for topographic map symbols at 1:5000 scales in the State of Paraná, Brazil.

Hence, our research group proposed a new set of symbols for large scale base maps, as a consultant to a working group of the Technical Chamber of Cartography and Geoprocessing – (in Portuguese, Camara Técnica de Cartografia e Geoprocessamento – CTCG) of the State of Parana Government [CTCG 2009]. The first results of this project have been presented at ICC 2013 – International Cartographic Conference (Sluter et al. 2013). To achieve an efficient result in proposing the map symbols we developed a map design based on the theory of Cartography. Our first step was to set some premises as follow:

- The set of map features and their related symbols must be defined by the theory of topographic mapping.
- The large-scale maps must be totally integrated to the Brazilian standards for topographic mapping.
- Therefore, the decisions about symbols design must agree with the Brazilian standards for reference map symbols.

We defined the steps of the methodology by map design theory for generating topographic maps [Keates 1973], as follows:

- Defining the cartographic features that must be in a large scale (1:2000) reference mapping of an urban area.
- Establishing the meaning of every cartographic features based on the theory of topographic mapping.
- Grouping the features into classes by their meaning and by the EDGV conceptual model.
- Creating symbols for each kind of feature.
- Applying the symbols to urban areas of the State of Parana.

2.1. Study Area

We have tested all the results of this research work in four municipalities of the State of Parana (Figure 2). Those municipalities are Cascavel (24°57'21"S, 53°27'19"W), Guarapuava (25°23'43"S, 51°27'29"W), Ponta Grossa (24°05'42"S, 50°09'43"W) and São

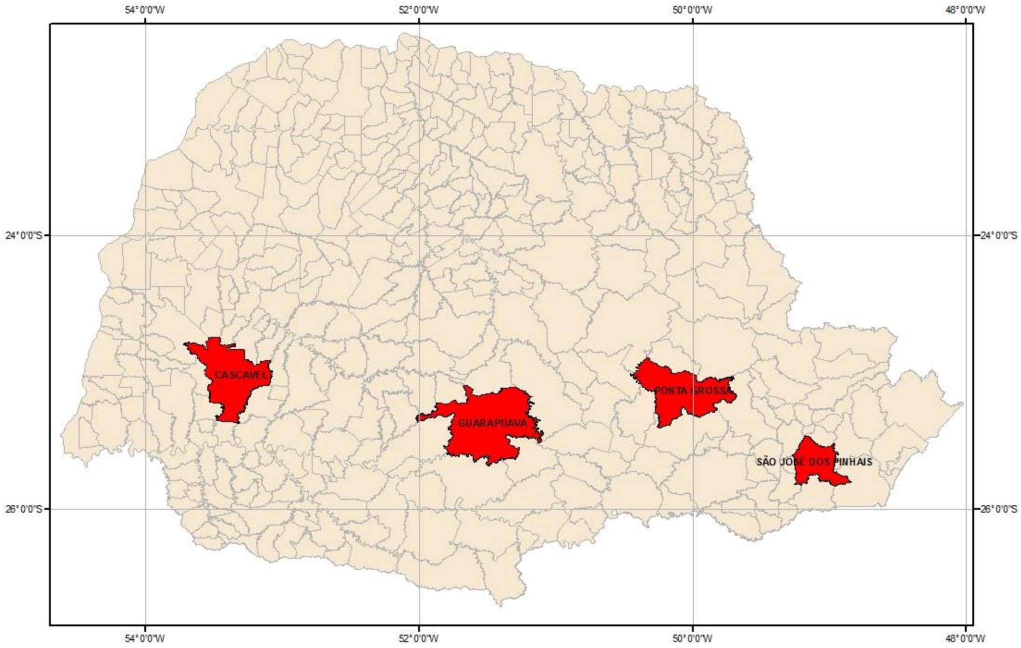


Figure 2. The State of Paraná, highlighting the studied areas.

2.2 Materials

We have been developing this research work using the following software:

- LibreOffice 2007 http://www.libreoffice.org/ - we used to generate the spreadsheets of definitions and symbology.

- Astah Community http://www.astah.net/ - we used this solution for the database modeling as UML class diagrams.
- PostgreSQL http://www.postgresql.org/ - with its extension PostGIS, we used this database management system to store the geographic database
- QGis http://www.qgis.org/ - we used this open source GIS software for data manipulation and definition of symbology. Additionally, we used the plugin DSG Tools created by the Brazilian Army to implement the EDGV model.

The large-scale topographic mapping (1:2000) was provided by Paranacidade in shapefile format.

3. Results

The results are presented in two sections: the first one describes the generation of the conceptual model for the reference maps and the database implementation, and the second one focuses on the symbology definition applied for the geographic database.

3.1. Data model

The definition of which cartographic features should be the represented at large scale topographic mapping is a result of several meetings with the members of the Technical Chamber of Cartography and Geoprocessing of the State of Parana, Brazil (in Portuguese, Camara Técnica de Cartografia e Geoprocessamento) [CTCG 2009]. After defining the cartographic features to be represented in topographic maps, we described the meaning of each one.

In the next step, we verified which classes are already part of EDGV data model, and which ones are not. We, then, grouped the classes were into EDGV categories: altimetry, recreation area, buildings, hydrography, infrastructure, boundaries, reference points, transportation, and vegetation. We organized a spreadsheet with symbology and definition of each class. This spreadsheet was the input to model the database with the relationships between objects in a logical structure as a class diagram. The conceptual modeling in UML focuses on the Class Diagrams for each category. Figure 3 shows an example of the diagram of the category *Boundaries*.

With the definition of all classes and categories proposed for a large-scale topographic mapping of municipalities under study, we defined the database scheme at the logical level. The results are the class diagrams and a general table that relate different classes and definitions from EDGV to the standards of CTCG.

We organized the general table as a set of spreadsheets. The spreadsheets refer to each category and, the first line of each specifies the category name and origin, which can be EDGV or CTCG. The first column of each spreadsheet shows the classes that compose it. The second column presents the classification criteria for each feature in the reference mapping. For example, in the *transportation* category, there is the "*Special Structures*" class in the CTCG standard, and several classes, e.g. Tunnel, Bridge, in the EDGV, therefore the classification criteria are different in both standards. The fourth column presents the definition of the feature. To be able to propose a solution for a geospatial data model based on both CTCG and EDGV, the differences between CTCG standards and EDGV for features definitions, classification, and classification criteria were highlighted.

Table 1 shows an example of a part of the relational table of the *Special Structure*s CTCG Class.

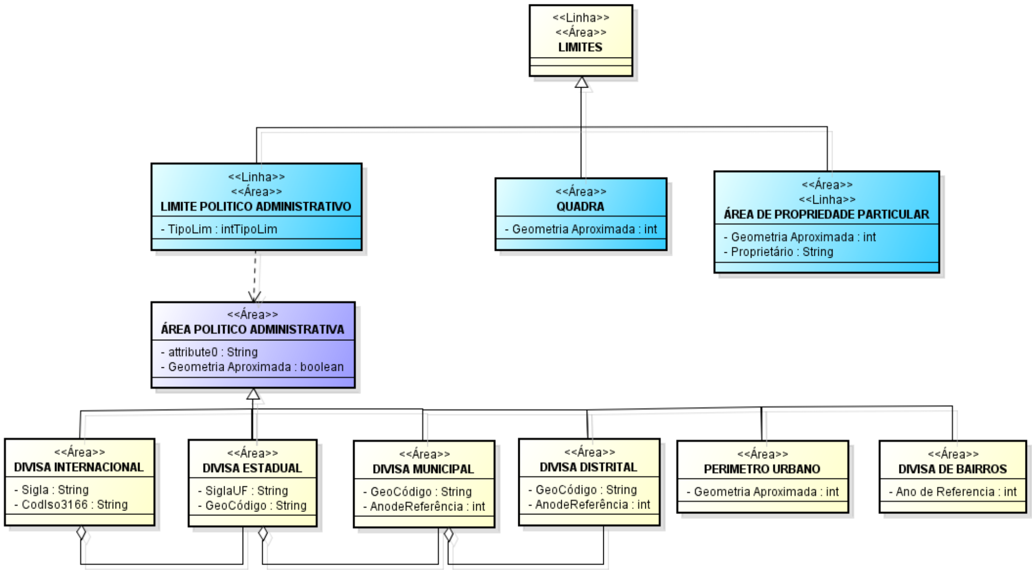


Figure 3. Class diagram of the category *Boundaries*.

CTCG			EDGV			
Classification criteria	**Feature**	**Definition**	**Class**	**Classification criteria**	**Feature**	**Definition**
Função como passagem de Nível no Sistema de Tráfego Terrestre	Túneis	Passagem ou caminho subterrâneo que serve de via de comunicação.	TUNEL	Altura	Túnel	Túnel é uma passagem subterrânea em uma via e no seu sentido longitudinal (Rodovia, Ferrovia, Dutos).
	Pontes	Construção que liga dois pontos separados por curso de água ou por uma depressão de terreno.	PONTE	Tipo de ponte/Vão livre horizontal/ Vão vertical/ Carga suportada Máxima	Ponte	Ponte é obra de arte especial destinada a permitir que uma via transponha um obstáculo líquido.

CTCG			EDGV			
Classification criteria	**Feature**	**Definition**	**Class**	**Classification criteria**	**Feature**	**Definition**
	Viadutos	Obra de construção civil destinada a transpor uma depressão de terreno, que não seja uma massa d'água, ou servir de passagem superior.	PASSAGEM ELEVADA/ VIADUTO	Tipo de Passagem/Viaduto /Vão livre horizontal/ Vão vertical/ Carga suportada Máxima/gabarito horizontal suportado/gabarito vertical suportado	Passagem Elevada/ Viaduto	Passagem elevada ou viaduto é uma obra destinada a permitir que uma via transponha vales, grotas, rodovias, ferrovias ou contorne encostas, bem como substitua aterros. Pode ser também uma via urbana para trafego rodoviário ou ferroviário em nível superior ao solo.
	Passagem de nível	Chama-se passagem de nível a um cruzamento não desnivelado entre uma ferrovia e um caminho ou estrada.	PASSAGEM DE NÍVEL	Nome	Passagem de Nível	Passagem de nível é um cruzamento de nível entre trechos rodoviários e um trecho ferroviário. Para efeito desta norma, também será considerado entre um trecho rodoviário e outro específico para o trânsito de Veículo Leve sobre Rodas.
	Pinguelas	Tronco ou prancha que serve de ponte sobre um rio.	TRAVESSIA PEDESTRE- Travessia de pedestre é uma estrutura, normalmente estreita, destinada a permitir a transposição, por pedestres, de um obstáculo natural ou artificial, geralmente construída sobre ou sob uma via.	Tipo de travessia de pedestres- Indica o tipo de travessia pedestre.	Pinguela	**A EDGV NÃO PREVÊ UMA DEFINIÇÃO DE PINGUELA**
	Passarelas	Ponte para pedestres,em geral estreita, construída sobre ruas ou estradas.			Passarela	**A EDGV NÃO PREVÊ UMA DEFINIÇÃO DE PASSARELA**
	Passagem a Vau	Local onde é possível atravessar o rio à pe, à cavalo ou de veicúlo traçado.		Tipo de travessia - Indica o tipo de travessia.	Vau construída / Vau Natural	Vau Construida: Travessia por região alagada ou massa d'água, após preparação especial. Também conhecida como "passagem molhada". Vau natural: Travessia por região alagada ou massa d'água, sem a necessidade de preparação especial.

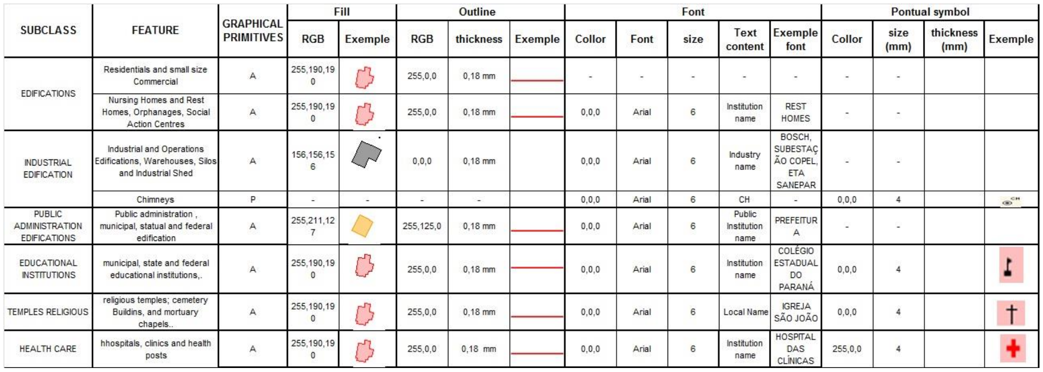
Table 1. Relation (in Portuguese) of both conceptual models (CTCG e EDGV). Category: Transportation **and Class: CTCG** Special Structures **(Obras de Arte in Portuguese)**

We created the geospatial database using this reference table and class diagrams. In this step, we used the QGIS Plugin DSG Tools, a free and open-source solution that enables the generation of geospatial data in full compliance with the NSDI, which makes the physical implementation of the EDGV standard in the geospatial databases possible [DSG 2015].

After the creation of the geospatial database, we could transfer the features geometries from shapefiles, provided from Paranacidade to the corresponding tables in the databases. We had to manually assign the features attributes from the original data source to the corresponding EDGV data model when it was a complete relation with CTCG data model. When it was necessary, we created additional compliant classes and their attributes accordingly with the conceptual data model.

3.2. Symbology

After establishing the relation between both conceptual models, we could apply the proposed symbology to the EDGV-based geographic database. In Figure 4 there is a sample of the detailed symbology definition that we adapted to the new geospatial data model. The symbols' definition includes area fill and outline colors, fonts and point symbols.



SUBCLASS	FEATURE	GRAPHICAL PRIMITIVES	Fill		Outline			Font					Pontual symbol			
			RGB	Exemple	RGB	thickness	Exemple	Collor	Font	size	Text content	Exemple font	Collor	size (mm)	thickness (mm)	Exemple
EDIFICATIONS	Residentials and small size Commercial	A	255,190,190	▨	255,0,0	0,18 mm	―	-	-	-	-	-	-	-		
	Nursing Homes and Rest Homes, Orphanages, Social Action Centres	A	255,190,190	▨	255,0,0	0,18 mm	―	0,0,0	Arial	6	Institution name	REST HOMES	-	-		
INDUSTRIAL EDIFICATION	Industrial and Operations Edifications, Warehouses, Silos and Industrial Shed	A	156,156,156	▨	0,0,0	0,18 mm		0,0,0	Arial	6	Industry name	BOSCH, SUBESTAÇÃO COPEL, ETA SANEPAR	-	-		
	Chimneys	P	-	-	-	-		0,0,0	Arial	6	CH	-	0,0,0	4		CH
PUBLIC ADMINISTRATION EDIFICATIONS	Public administration , municipal, statual and federal edification	A	255,211,127	▨	255,125,0	0,18 mm	―	0,0,0	Arial	6	Public Institution name	PREFEITURA	-	-		
EDUCATIONAL INSTITUTIONS	municipal, state and federal educational institutions,.	A	255,190,190	▨	255,0,0	0,18 mm	―	0,0,0	Arial	6	Institution name	COLÉGIO ESTADUAL DO PARANÁ	0,0,0	4		▨
TEMPLES RELIGIOUS	religious temples; cemetery Buildins, and mortuary chapels..	A	255,190,190	▨	255,0,0	0,18 mm	―	0,0,0	Arial	6	Local Name	IGREJA SÃO JOÃO	0,0,0	4		†
HEALTH CARE	hhospitals, clinics and health posts	A	255,190,190	▨	255,0,0	0,18 mm	―	0,0,0	Arial	6	Institution name	HOSPITAL DAS CLÍNICAS	255,0,0	4		✚

Figure 4. Symbology description adapted to the conceptual model (in Portuguese).

We applied the standard symbology to the datasets of the four municipalities we have chosen as study areas (Figure 5 e 6). We exported the resulting symbology using SLD style to be used in data sharing applications, for example, servers of WMS (Web Mapping Services).

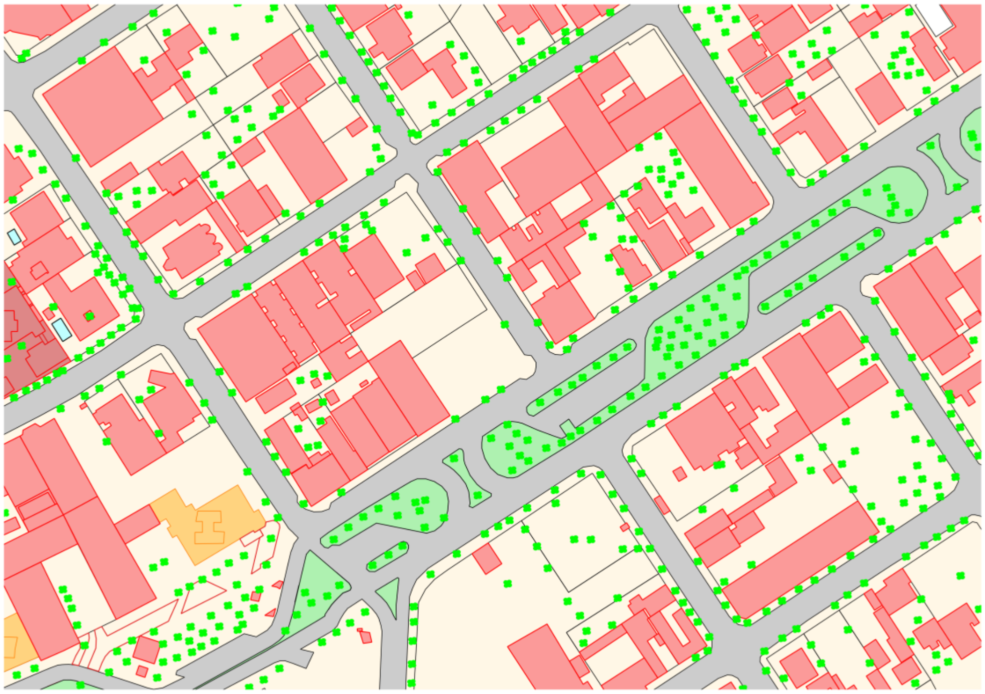


Figure 5. Extract of the results for the city of Cascavel

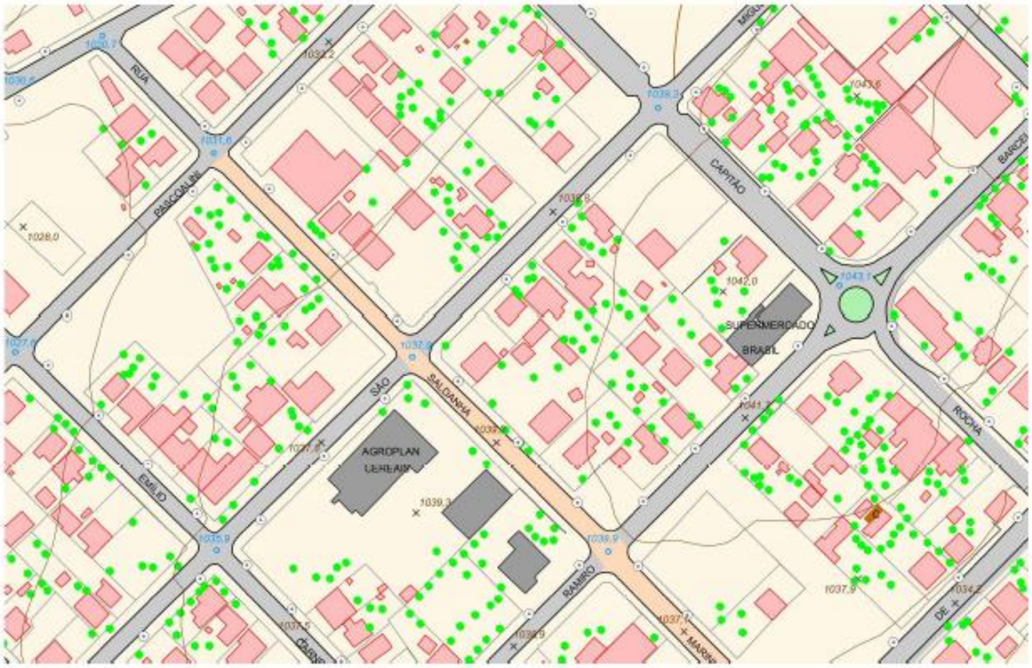


Figure 6. Extract of the results for the city of Guarapuava.

4. Conclusion

The standardized geodatabase and symbology of the data that compose the large-scale reference mapping guarantee the interoperability amid systems at different social and political levels (local, state or federal). The standardization enables the integration between different geospatial databases, which is an important result of this work in planning, engineering, and urban projects. The standards for geospatial database and symbology make spatial data review and update possible, minimizes the time for spatial analyses, and ensures information quality.

The main obstacles to developing this work were the data models discrepancies since EDGV and CTGC standards were proposed for different purposes in different time. The plugin DSG Tools facilitated the process to create EDGV compliant database and shows how the development of open source tools can be a benefit for several uses on a standardized environment.

The solution proposed in this research can be implemented in any geospatial database at 1: 2000 scale and using it at other scales could be adopted with some adjustments in the method. More users could be involved in the conceptual modeling and testing steps to expand the use out of the Parana context where we developed this the research work. The proposition and management of standards is a continuous task. Several studies on cartographic generalization, geospatial semantics, user's cognition and others can benefit from the results of this conceptual model.

5. References

CONCAR (2010), “Especificações Técnicas para Estruturação de dados Geoespaciais Digitais Vetoriais”. http://www.geoportal.eb.mil.br/images/PDF/ET_EDGV_Vs_2_1_3.pdf

CONCAR (2010), “Plano de Ação da INDE” www.concar.gov.br/pdf/PlanoDeAcaoINDE.pdf

CTCG – Câmara Técnica de Cartografia e Geoprocessamento do Estado do Paraná (2009) “Relatório Técnico Preliminar: Proposta de Convenções Cartográficas para o Mapeamento Topográfico em Grande Escala no Estado do Paraná.”

DSG – Diretoria do Serviço Geográfico do Exército (1998) “Manual Técnico T34-700 convenções cartográficas 1ª. parte: Normas para o emprego dos símbolos.”

DSG (2015) “Norma da Especificação Técnica Para Estruturação De Dados Geoespaciais Vetoriais de Defesa da Força Terrestre ET-EDGV-DefesaFT” . 1a Parte, 1ª Edição, http://www.geoportal.eb.mil.br/images/PDF/EDGV_Defesa-Forca_Terrestre_2015.pdf

DSG (2016) – “Geoportal do Exército Brasileiro” www.geoportal.eb.mil.br

IBGE (2010) “Censo Demográfico 2010”, http://www.ibge.gov.br/home/estatistica/populacao/censo2010/default.shtm

IBGE (2016) “Estimativas populacionais para os municípios e para as Unidades da Federação brasileiros em 01.07.2016” http://www.ibge.gov.br/home/estatistica/populacao/estimativa2016/estimativa_dou.shtm

KEATES, J.S. (1973) Map Design and Construction. New York, USA: John Wiley & Sons, Inc.

KEATES, J.S. (1982) Understanding maps. New York, USA: John Wiley & Sons, Inc.

OGC (2007) "OpenGIS Styled Layer Descriptor Profile of the Web Map Service Implementation Specification" http://portal.opengeospatial.org/files/?artifact_id=22364

PARANACIDADE (2010) Processo licitatório de Convite n°008/2010 - Plano Diretor Municipal - Município de Lunardelli - Termo de referência padrão elaborado pelo Paranacidade que atende a lei 15229/2006. Serviço Social Autônomo Paranacidade, Curitiba, Brasil.

SLUTER, C.R.; BRANDALIZE, M.C.B.; VAN ELZAKKER, C.J.P.; and IVÁNOVÁ, I. Defining Standard Symbols for Street Network Maps for Urban Planning Based on User Requirements. In Annals of the 26th Cartographic Conference – ICC 2013. Dresden, Alemanaha.

UN-HABITAT (2016) "World Cities Report 2016, Urbanization and Development: Emerging Futures" http://wcr.unhabitat.org/

A new Platform for Time-Series Analysis of Remote Sensing Images in a Distributed Computing Environment

Sávio S. Teles de Oliveira[1], Marcelo de C. Cardoso[1], Wisllay M. V. dos Santos[1], Paulo C. P. Costa[1], Vagner J. do Sacramento Rodrigues[1], Wellington S. Martins[2]

[1]GoGeo
Rua Leopoldo Bulhões, esquina com a Rua 1014
Quadra 31, Lote 07, Sala 9 Setor Pedro Ludovico
CEP 74820-270 - Goiânia - GO - Brazil

[2]Instituto de Informática - Universidade Federal de Goiás (UFG)
Alameda Palmeiras, Quadra D, Câmpus Samambaia
131 - CEP 74001-970 - Goiânia - GO - Brazil

{savio.teles, marcelo.castro, wisllay.vitrio, paulo.cezar, vagner}@gogeo.io,
wellington@inf.ufg.br

Abstract. *Time series analysis of remote sensing images is essential for the detection of patterns, trends and changes to allow the modeling and prediction of events in the earth's surface. Users of geographic information systems (GIS) are often involved in spatio-temporal remote sensing analysis. In case of applications with large volumes of data, this analysis should be carried out in an automated manner and allow spatiotemporal filtering in the image database. This work proposes a new platform, DistSensing, that can enable these analyses to be conducted with the aid of distributed indices. We show how this DistSensing platform outperforms the solutions found in the literature when there is the need to run queries on the database using temporal and spatial filters.*

1. Introduction

The Earth's surface is changing at an unprecedented rate, with the alarming disappearance of much of the forest ecosystem, while urban and farming areas are expanding around natural spaces. A time series analysis of remote sensing images is essential to detect these changes [Neves et al. 2015]. This entails, for example, providing information on shifts in the spatial distribution of bio-climatic zones, and indicating the variations in large-scale global circulation patterns or changes in land-use.

It is essential to make users aware of both the spatial and temporal dimensions in a Geographic Information System (GIS), because these may reveal implicit relationships which match the reality of the analyzed data [de Oliveira and de Souza Baptista 2012]. The GIS is a computer-based tool for mapping and analyzing feature events on earth, which must allow users to conduct a time-series analysis of remote sensing data filtering in specific geographical regions and at regular time intervals, for example, to trace the evolution of deforested areas in the Amazon forest from 1991 to 1997.

The incorporation of latest-generation sensors to airborne and satellite platforms has led to a nearly continuous stream of data [Smits and Bruzzone 2004], and this sharp rise in the amount of collected information has raised new processing challenges with regard to the time-series analysis of remote sensing data. In addressing these computational requirements, several research endeavors have recently been geared to incorporating high performance computing models in the remote sensing field [Plaza and Chang 2007].

Our work introduces a new platform called DistSensing, to perform the distributed processing involved in the time series analysis of remote sensing data, which allows users to have real time spatio-temporal filters. DistSensing obtained more impressive performance gains than other time series processing platforms found in the literature ([Van Den Bergh et al. 2012, Song et al. 2015]), when spatiotemporal filters are required. The main contributions to this field of studies made by this work are as follows: i) it sets out a strategy for partitioning the remote sensing images into cluster nodes; ii) it includes a search algorithm for the processing of the time series analysis of remote sensing images by means of distributed indexing.

The remainder of this work is structured as follows. Section 2 describes strategies found in the literature to process remote sensing images. Section 3 provides a review of time series processing based on remote sensing data. Section 4 outlines the DistSensing platform. Section 5 describes the methodology and the experimental results. Section 6 summarizes the conclusions and includes a brief description of further work that will be carried out in the future.

2. Related Work

[Ferreira et al. 2015] created a new RDF vocabulary to access spatiotemporal datasets from different kinds of data sources, including remote sensing images. Some works sought to automatically detect changes by using remote sensing images, like [Neagoe et al. 2014] who adopted non-supervised approaches with less manual intervention. In [Romani et al. 2009] a new algorithm was designed to reveal changes in weather patterns by plotting the pixels location from an image that was created to partition them. All these works relied on a single server to process the images, instead of using the computing resources in a *cluster*.

Several studies employed the MapReduce model. [da Silva Ferreira et al. 2015] introduced the architecture of a distributed platform named InterIMAGE Cloud Platform (ICP) to handle with very large volumes of data using clusters of low-cost computers with the Hadoop framework. In [Lv et al. 2010] a K-means clustering algorithm was implemented for remote sensing images and in [Wang et al. 2012] a classification algorithm was created for high resolution remote sensing images. The MapReduce model was also used by [Almeer 2012] to create a parallel processing platform for remote sensing images based on a cloud computing system. In [Lin et al. 2013], a platform was proposed and implemented using Hadoop to process remote sensing algorithms with MapReduce models. The work by [Rathore et al.] relies on an architecture to analyze remote sensing images in real time with *Big Data* using Hadoop. However, these works fail to provide any solution for conducting time series analysis by means of remote sensing images.

[Song et al. 2015] created the Spatiotemporal platform for the time series processing of spatial objects, through cloud computing. This includes HDFS, which is used as a distributed file system, and a MapReduce based computing service, which is used to analyze spatial data. This paper does not discuss spatial and temporal indexing techniques nor how the image distribution is carried out in the cluster. When using HDFS, it is important to define the data distribution algorithm, since images from the same geographical region and collected during the same time interval, should be stored in the same machine [Van Den Bergh et al. 2012]. This is to avoid generating a high volume of network traffic while the time series analysis is being processed.

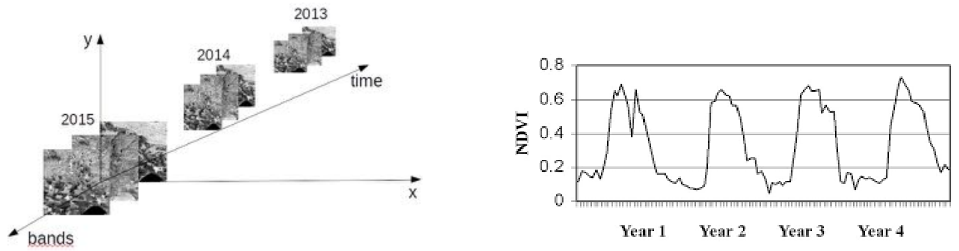
[Van Den Bergh et al. 2012] established the HiTempo platform to assist research in the area of time series analysis based on remote sensing images. The images were stored as 3D objects, with geographically closed images obtained in the same time interval, being stored in the same machine. HiTempo was designed to make possible the evaluation and comparison of remote sensing algorithms and, for that reason, the time-space filters are only applied before inserting the images into the platform, to prevent the filters from being applied at query time. This means that, it is not possible to perform spatial and temporal filtering after the data insertion and the query cannot apply filters to the image set stored in the HiTempo platform.

3. Time Series Analysis of Remote Sensing Images

There are a wide range of satellites and sensors with different resolutions. In our study, remote sensing images were chosen from LandSat-8, because they are public and easy to access. Each scene taken by a satellite in a given geographical area and time is called a scene. Each scene from LandSat-8 consists of nine different spectral bands, covering an area of around 170 km north to south and 183 km east to west.

A time series analysis of remote sensing images collected by satellite is needed to evaluate the features of terrestrial surfaces. A time series is a sequence of images, collected from successive, and usually uniformly spaced, time intervals. A time series analysis includes methods that are employed to identify patterns and trends, detect changes, and cluster and model data.

In Figure 1(a) it is possible to visualize the four dimensions of the time series of remote sensing images. In this graph, the x and y axes represent the spatial limits of each band from a scene, where x is the scene's geographical longitude, and y is the scene's geographical latitude. The “bands' 'axis represents the spectral bands of each scene, and the “time” axis represents the satellite images obtained from various points of time.



(a) Dimensionality of a remote sensing time series.

(b) Changes in the NDVI in a historical series.

Figure 1. Time series of remote sensing images.

To validate the DistSensing platform in this work, an algorithm for a time series analysis of remote sensing images was implemented to validate the DistSensing platform used in this study, by means the NDVI (Normalized Difference Vegetation Index) historical analysis. The NDVI is a numerical indicator correlated with certain physical properties of the vegetation canopy: leaf area index, vegetation condition, and biomass [Carlson and Ripley 1997]. Through this analysis, it is possible to visualize, for example, the increase in deforestation in certain regions throughout the period by looking at their NDVI time series [Morton et al. 2005]. The final result, for instance, is a graph, containing the NDVI mean for each date, as can be seen in the example from Figure 1(b).

The NDVI is calculated from the difference between the Infrared and Near Red bands, normalized by the sum of the same bands, using Equation 1. The NDVI is derived from spectral reflectance measurements acquired in the visible (RED) and near-infrared (NIR) regions, where NIR and RED are the reflectance in infrared and red spectral intervals, respectively. The result of the NDVI ranges between -1 and +1.

$$NDVI = \frac{NIR - RED}{NIR + RED} \tag{1}$$

4. DistSensing: A new Platform for Efficient Distributed Processing of Remote Sensing Time Series

This paper creates a new platform, DistSensing, for processing remote sensing time series in a distributed manner. DistSensing has an highly available, elastic and fault-tolerant architecture (Section 4.1) and provides efficient distributed algorithms for storing, indexing (Section 4.2) and querying images at its databases (Section 4.3).

4.1. DistSensing Architecture

DistSensing has a peer-to-peer architecture in which the cluster servers do not share CPUs, hard drives or memory and all communication is carried out via a message passing system. It is composed of a client layer and a server layer. Client applications interact with the platform through an API for updating and querying the databases. Each client application must use a client library for DistSensing API so that the platform's available services can be used. Read and write requests can be sent to any server of the cluster. The list of servers can be obtained from the platform's name service.

The server that receives the client request becomes the coordinator for that specific request. The coordinator acts as a proxy between the client application and the cluster servers. To ensure high availability, when the coordinator is unavailable another server is chosen as coordinator. The client library sends the request, receives the response and transfers the result of the operation to the client application.

Cluster nodes exchange information with each other every second using the gossip protocol [Subramaniyan et al. 2006]. The Gossip protocol is used for finding out and sharing location and status information about other cluster servers. Each server exchange messages not only about its status and other factors but also regarding other cluster data nodes with up to three other servers. Thus, all the cluster nodes become aware of the status of the other nodes of the cluster quickly. Failure detection occurs on the basis of the data exchanged via the gossip protocol. The platform uses this information to avoid requesting processing to unavailable servers.

DistSensing stores data replicas on multiple nodes to ensure fault tolerance and reliability. All the replicas are equally important, since there is no primary or master replica. Every object stored is given a single identifier key generated via hashing. Each server is responsible for an equal range of cluster keys, that are accessed through a Distributed Hash Table (DHT) [Karger et al. 1997]. When an object is required, the elected server is chosen by means of the DHT.

To ensure platform's elasticity, machines might be added or removed from the cluster at any moment. Whenever this happens, the DHT is rebuilt to reflect the new configuration of the cluster. When a new server joins the cluster it queries the DHT and sends a message to the server with the highest disk usage in the cluster to obtain half of the keys kept in this server. Upon removal of the server S its keys are distributed among the remaining servers and its objects are copied from replicas.

4.2. Distribution and Storage of Remote Sensing Images on Cluster Servers

The architecture of DistSensing allows several remote sensing images to be inserted in a distributed and parallel manner. Figure 2 shows step-by-step what happens when an image is inserted in the platform. Insertion starts with a client sending the image bands to the DistSensing platform using the API. Each image band is then sent to a randomly chosen server of the cluster.

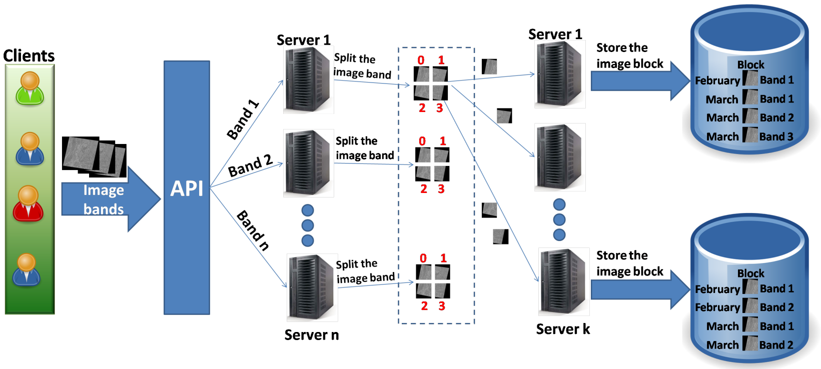


Figure 2. Distribution and Storage of Remote Sensing Images on DistSensing Platform.

In each server, each spectral band of the image is broken into smaller fixed-size blocks that allow a parallel processing of images during the queries. The block size directly affects the size of the job that has to be carried out in the query operations involving the remote sensing images. The greater the number of cores in the cluster servers, the smaller the remote sensing image block size must be to increase the parallelization and improve the efficiency of query processing. This block size must be set from tests conducted in the cluster environment where the platform is installed. To ensure reliability and fault tolerance, as well as providing load balancing, each block is stored on a server and replicated in two other servers.

As Figure 2 shows, block distribution is based on four dimensions of each block: the first two are x and y, and represent the spatial limits of the block, the third is the $time$

and the fourth one is related to the image $band$. Each server in the cluster has its own storage system regardless of the others servers, where the blocks are stored. Thus, blocks from the same geographical area collected at different times are stored at the same server, and this reduces network traffic during the execution of time series analysis. By means of this block distribution, the remote sensing algorithms that carry out the time-series analysis in each image pixel (e.g. NDVI) do not have to send data over the network, since the historical series of each pixel is stored in a server.

Each image block is linked to a geographical location and contains metadata from the original image from which it was generated, such as the date when the image was captured. The *Quad Tree* [Finkel and Bentley 1974] spatial index is built in each server of the cluster, allowing us to perform spatial filters. The index in each server is built from each spatial image block bounding stored in that server. The DistSensing platform is responsible for coordinating the distributed spatial filter amongst the several spatial indices.

The *Quad Tree* is a tree data structure in which each internal node has exactly four children and it was chosen because of its ability to filter remote sensing data [Fu et al. 2013]. The key idea is to partition the space into four quadrants. Each node in the tree either has exactly four children, or has no child. Each child represents a subquadrant of its parent. Quadrants that can no longer be subdivided are represented as leaf nodes that contain data corresponding to that specific subregion.

In addition, an index is built from the metadata (remote sensing image attributes, like the date) of each image in the database. This index is important since it provides with ways to, for instance, conduct an analysis at specific time intervals. The Apache Lucene [Lucene 2010] library was used to build this index. It is available at each server of the cluster and the DistSensing platform is responsible for coordinating the distributed filter amongst the Lucene search engines. The client sends a request that defines the restrictions using Apache Lucene query syntax, which allows it to filter images through phrase, wildcard, range and full-text-search queries.

4.3. Querying the Remote Sensing Database

The query execution process starts with the client sending the request to the DistSensing platform with the spatial filters and metadata attributes constraints. One of the platform's servers is then chosen as the Master that coordinates the execution of that request. Algorithm 1 describes the steps that must be followed inside the DistSensing platform.

The query request is sent to all the cluster servers, since each server stores remote sensing image blocks. After this, each server locally processes the spatial filters by means of the *Quad-Tree* and the image metadata, by using the Apache Lucene on the basis of client-specified restrictions. The result is stored in $block\_ids$ and sent to the Master server as a list of ids of image blocks, along with metadata from the blocks necessary for the execution.

The Master server is responsible for receiving the Ids and metadata and removing duplicates, since the data is replicated among the servers. The unique ids and metadata are stored in the variable $global\_block\_ids$. For each resulting id in $global\_block\_ids$, the DistSensing platform queries its DHT to discover which servers are storing the image block with that id. This DHT stores the id of each object as its key and the IP addresses of servers as value. A server S is randomly chosen from this list and combined with that id. The table $server\_ids$ is built using the server S as key and the block ids stored in S as value. For each server S in $server\_ids$, the list of ids and metadata is retrieved and then sent to server S.

```
Algorithm 1: $Query(spatial\_filter, metadata\_filter)$
    Data: $spatial\_filter$: spatial filter containing the spatial restriction,
          $metadata\_filter$: metadata filter using Lucene's query syntax
    Result: $final\_result$
    $Master$ server send $spatial\_filter$ and $metadata\_filter$ for all cluster servers
    for Server $S$ of the cluster do
        $metadata\_block\_ids \Leftarrow$ Ids from the block accepted by the metadata filter
        $spatial\_block\_ids \Leftarrow$ Ids from the block accepted by the spatial filter
        $block\_ids \Leftarrow metadata\_block\_ids \cap spatial\_block\_ids$
        Send $block\_ids$ for $Master$
    end
    $Master$ server receives $block\_ids$ and creates the set $global\_block\_ids$
    for $id \in global\_block\_ids$ do
        $S \Leftarrow$ IP from one of the servers storing block with id $id$
        Insert id $id$ on ids list with key $S$ of the table $server\_ids$
    end
    for $S \in server\_ids$ do
        Send the list $ids\_list$ associated with the key $S$ to the server $S$
    end
    for Server $S$ of the cluster that receives an $ids\_list$ do
        for $id \in ids\_list$ do
            $b \Leftarrow$ retrieve block with id $id$
            $b\_result \Leftarrow$ run the remote sensing time-series analysis algorithm using $b$ as input
            Add $b\_result$ on block list $local\_result$
        end
        $local\_result \Leftarrow$ result from local result aggregation algorithm on $local\_result$
        Send $local\_result$ to $Master$ server
    end
    $Master$ server receives $local\_result$ list from each server $S$ and aggregates the result in $final\_result$
    $Master$ server sends $final\_result$ to the client
```

In each server S, the blocks are retrieved from the storage system and the remote sensing time-series analysis is conducted with the algorithm requested by the client, such as the NDVI time-series analysis. The results obtained from processing this block list, are stored in $local\_result$ and sent to the master server. The master server receives replies from each server and aggregates them in order to create the final result, which is then sent to the client. This aggregation algorithm is defined by the clients of the DistSensing platform in accordance with their requirements.

On the basis of the analysis shown in Section 3, which employs the normalized difference vegetation index (NDVI), the client sends the spatial and metadata restrictions to the DistSensing platform. The date attribute is then used to select remote sensing image blocks in a given time slice. The coordinator sends the client request to each server of the cluster which calculates the average NDVI for each image block. Once the average NDVI is known for all the image blocks, the values are aggregated by date to create the average NDVI for each date. These aggregate values are sent to the Master server which processes the final aggregation on the basis of data obtained from all the servers and generates the average NDVI for each date. These values are sent to the client which is then capable of creating graphs of NDVI variations over a period of time.

5. Performance Evaluation

An algorithm was implemented to evaluate the performance of the DistSensing platform and thus allow an analysis of the temporal variations in the NDVI values, as shown in Section 3. The response times from the DistSensing platform are compared with those of the HiTempo platform recommended by [Van Den Bergh et al. 2012], which is similar to the proposal put forward by [Song et al. 2015], that are unable to perform temporal and spatial filters during the query execution. The image processing module from the solution found by [Van Den Bergh et al. 2012] was implemented to evaluate a HiTempo platform with spatial and temporal constraints and in the final stage.

5.1. Experiments and Database

A horizontal scalability test was conducted to measure the performance of the DistSensing platform when adding servers to the cluster. Ideally the system should have linear horizontal scalability, which is not always possible: a) since the tasks distributed in the cluster have to be synchronized and b) the devices are subject to physical constraints as in the case of those used for networking. The tests were run on *Intel Core i7 3.4 GHz* machines, with 16 GB of RAM and 1 TB hard drives. The machines were connected to a 1Gbit/s Ethernet network and a 6248P Dell PowerConnect switch. 53 remote sensing images from Brazil were collected by the LandSat-8 satellite from 2013 to 2015, each image with approximately 2 GB. Each of the image's band was broken into blocks of 1024 x 1024 pixels during the insertion procedure.

The algorithm for analyzing temporal variation of NDVI values was evaluated with clusters of 1, 2, 4 and 8 servers and took account of several spatial and temporal filter configurations: i) No filter, covering 100% of the database, and mirroring the HiTempo strategy (HiTempo), ii) spatial constraints covering 27,1% of the database (Large_Spatial), iii) spatial constraints covering 6,1% of the database (Small_Spatial), iv) temporal constraints covering 34% of the database (Date_Large), v) temporal constraints covering 5% of the database (Date_Medium), vi) temporal constraints covering 1% of the database (Date_Small), vii) having spatial and temporal constraints that result in an empty response. Names in brackets are used in the charts shown in Section 5.2.

The algorithm was executed 20 times for each test configuration and the response time on the charts is the average of the 20 response times observed. A client library was created so that data could be sent for insertion, and queries performed with spatial and relational constraints which involved measuring the response times for each specific test.

5.2. Evaluation

The first test was carried out to determine the scalability of the DistSensing and HiTempo platforms when processing remote sensing time-series analysis, as shown in Figure 3. The response time was obtained after the time-series analysis had been conducted on the entire database. Both platforms achieved a lower response time as more machines were added to the cluster. This scalable behavior was made possible by dividing the image into blocks, and allowing a distributed and parallel processing of these blocks on the cluster. The HiTempo platform had a similar behavior and response time, since there was no need for subsequent filtering.

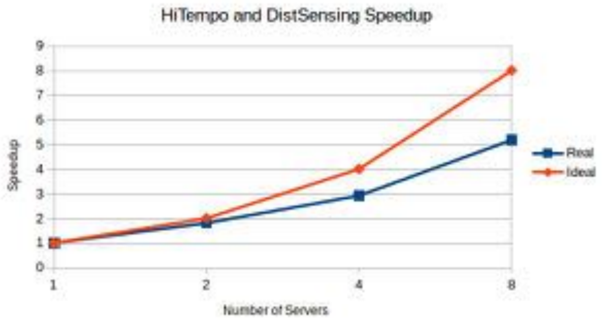
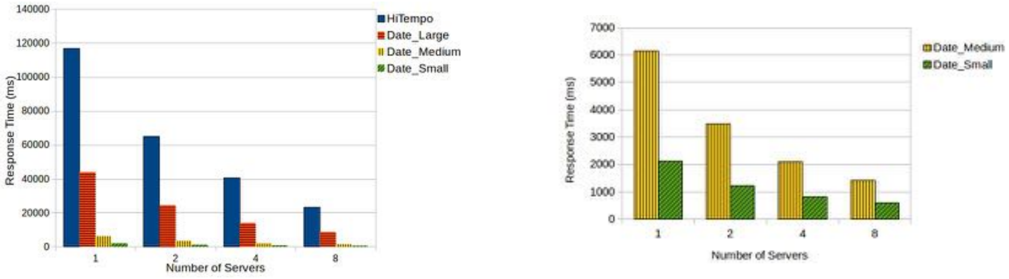


Figure 3. Horizontal Scalability.

Figure 4 shows a comparison between HiTempo and DistSensing platform when running the analysis with temporal filters on the image database. Compared with the HiTempo platform, Figure 4(a), DistSensing achieves response times up to 53 times lower if the "Date_Small" temporal filter is selected for the execution. Since the HiTempo platform does not allow temporal filters at the query time, it must process the complete database and apply the filter later. With "Date_Small" temporal filter, the DistSensing platform was able to filter 99% of the database, leaving only 1% for processing the analysis on remote sensing images.

The DistSensing platform shows scalability when processing the algorithm with temporal filters, especially with filters that are less restrictive like "Date_Large" where there is a greater need for computational resources during the analysis of the image time series. As shown in Figure 4(b), the DistSensing platform shows scalability when processing algorithms with more restrictive filters than "Date_Large", although not at the same levels of scalability. However, adding more servers for restrictive filters like "Date_Small" and "Date_Medium" do not result in proportionally lower response times, unless the image database increases and requires more computational resources.

Spatial filters are important for remote sensing specialists since they allow them to conduct an analysis that takes into account their geographic context and interest. Figure 5 shows the performance of the DistSensing platform when the algorithm is processed with spatial filters. The DistSensing platform was up to 9 times faster than the solution



(a) Comparison between DistSensing platform and the solution proposed by HiTempo platform.

(b) Scalability on temporal filters.

Figure 4. Database temporal filtering.

proposed by the HiTempo platform when the spatial filter “Small_Spatial” was used by the algorithm. With the “Date_Small” temporal filter DistSensing was up to 53 times faster because this filter covers 1% of the database while the “Small_Spatial” spatial filter corresponds to 6,1%.

Tests were carried out with spatial and temporal constraints which lead to empty responses. The DistSensing platform showed constant response times of approximately 10 ms in these cases, regardless of the cluster size. The HiTempo platform performed up to 11,600 times worse, with response times of 116 seconds when only one server was used. With 8 servers HiTempo took 23 seconds to process the data, which is a performance 2300 times worse than that of DistSensing. The reason for this huge difference is that DistSensing is capable of filtering out images that do not meet the constraints, before processing the algorithm on the remote sensing images.

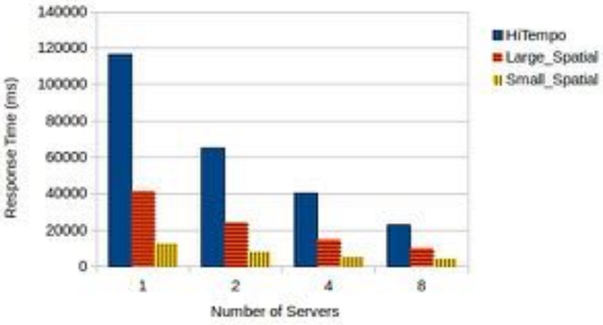


Figure 5. Database spatial filtering with HiTempo and DistSensing platforms.

The algorithm employed for analysing the temporal variation of the NDVI values, was found to be scalable, both with regard to the HiTempo platform and the DistSensing platform, as more servers were added to the cluster. The DistSensing platform had a better performance when temporal and spatial filters were used, with a speed up to 53 times faster when a temporal filter was used that corresponded to 1% of the database, and up to 9 times faster if a spatial filter was used that corresponded to 6,1% of the database. In cases where spatial and temporal filters lead to an empty dataset, the DistSensing platform was up to 11,600 times faster.

6. Conclusion

Time series analysis is crucial for detecting patterns, trends and changes, as well as providing us with ways to model and predict events on the earth's surface. The automatic execution of an analysis of remote sensing time series has become a challenge due to the increase in the amount of remote sensing data. To the best of our knowledge, there is no study on the methods employed for processing a time series analysis that allows the filtering of data based on a geographical region and time period.

This paper proposes a new platform called DistSensing, to conduct an analysis of remote sensing time series in a distributed manner. Spatial and relational indices were built to provide query functionality at the image database. DistSensing allows patterns, trends and changes on the earth's surface to be detected with lower response times. When temporal filters are used, DistSensing is up to 53 times faster than the HiTempo platform proposed by [Van Den Bergh et al. 2012], and if spatial filters are needed, the performance can be up to 9 times better than the HiTempo platform. Furthermore, if the combination of spatial and/or temporal filters generates an empty dataset, the DistSensing platform shows a response time that is 11,600 times lower the those of the HiTempo platform.

Tests on bigger clusters and with a larger volume of data will be conducted in future research. In addition, experiments will be carried out to find out what is the ideal size for the image blocks on the basis of to the cluster configuration.

References

Almeer, M. H. (2012). Cloud hadoop map reduce for remote sensing image analysis. *Journal of Emerging Trends in Computing and Information Sciences*, 3(4):637–644.

Carlson, T. N. and Ripley, D. A. (1997). On the relation between ndvi, fractional vegetation cover, and leaf area index. *Remote sensing of Environment*, 62(3):241–252.

da Silva Ferreira, R., Oliveira, D. A. B., Happ, P. N., da Costa, G. A. O. P., Feitosa, R. Q., and Bentes, C. (2015). Interimage cloud platform: Em direção à arquitetura de uma plataforma distribuída e de código aberto para a interpretação automática de imagens baseada em conhecimento. *XVII Simpósio Brasileiro de Sensoriamento Remoto - SBSR*, pages 5264–5271.

de Oliveira, M. G. and de Souza Baptista, C. (2012). Geostat-a system for visualization, analysis and clustering of distributed spatiotemporal data. In *GeoInfo*, pages 108–119.

Ferreira, K. R., de Oliveira, A. G., Monteiro, A. M. V., and de Almeida, D. B. (2015). Temporal gis and spatiotemporal data sources. In *GeoInfo*, pages 1–13.

Finkel, R. A. and Bentley, J. L. (1974). Quad trees a data structure for retrieval on composite keys. *Acta informatica*, 4(1):1–9.

Fu, G., Zhao, H., Li, C., and Shi, L. (2013). Segmentation for high-resolution optical remote sensing imagery using improved quadtree and region adjacency graph technique. *Remote sensing*, 5(7):3259–3279.

Karger, D., Lehman, E., Leighton, T., Panigrahy, R., Levine, M., and Lewin, D. (1997). Consistent hashing and random trees: Distributed caching protocols for relieving hot spots on the world wide web. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 654–663. ACM.

Lin, F.-C., Chung, L.-K., Wang, C.-J., Ku, W.-Y., and Chou, T.-Y. (2013). Storage and processing of massive remote sensing images using a novel cloud computing platform. *GIScience & Remote Sensing*, 50(3):322–336.

Lucene, A. (2010). Apache lucene.

Lv, Z., Hu, Y., Zhong, H., Wu, J., Li, B., and Zhao, H. (2010). Parallel k-means clustering of remote sensing images based on mapreduce. In *Proceedings of the 2010 International Conference on Web Information Systems and Mining*, WISM'10, pages 162–170, Berlin, Heidelberg. Springer-Verlag.

Morton, D. C., DeFries, R. S., Shimabukuro, Y. E., Anderson, L. O., Del Bon Espírito-Santo, F., Hansen, M., and Carroll, M. (2005). Rapid assessment of annual deforestation in the brazilian amazon using modis data. *Earth Interactions*, 9(8):1–22.

Neagoe, V., Ciurea, A., Bruzzone, L., and Bovolo, F. (2014). A novel neural approach for unsupervised change detection using som clustering for pseudo-training set selection followed by csom classifier. In *Geoscience and Remote Sensing Symposium (IGARSS), 2014 IEEE International*, pages 1437–1440. IEEE.

Neves, A. K., Bendini, H. N., Körting, T. S., and Fonseca, L. M. G. (2015). Combining time series features and data mining to detect land cover patterns: a case study in northern mato grosso state, brazil. In *GeoInfo*, pages 174–185.

Plaza, A. J. and Chang, C.-I. (2007). *High performance computing in remote sensing*. CRC Press.

Rathore, M. M. U., Paul, A., Ahmad, A., Chen, B.-W., Huang, B., and Ji, W. Real-time big data analytical architecture for remote sensing application.

Romani, L. A., de Ávila, A. M. H., Zullo Jr, J., Traina Jr, C., and Traina, A. J. (2009). Mining climate and remote sensing time series to discover the most relevant climate patterns. In *SBBD*, pages 181–195.

Smits, P. and Bruzzone, L. (2004). *Analysis of multi-temporal remote sensing images*, volume 3. World Scientific.

Song, W., Jin, B., Li, S., Wei, X., Li, D., and Hu, F. (2015). Building spatiotemporal cloud platform for supporting gis application. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1:55–62.

Subramaniyan, R., Raman, P., George, A. D., and Radlinski, M. (2006). Gems: Gossip-enabled monitoring service for scalable heterogeneous distributed systems. *Cluster Computing*, 9(1):101–120.

Van Den Bergh, F., Wessels, K. J., Miteff, S., Van Zyl, T. L., Gazendam, A. D., and Bachoo, A. K. (2012). Hitempo: a platform for time-series analysis of remote-sensing satellite data in a high-performance computing environment. *International journal of remote sensing*, 33(15):4720–4740.

Wang, G., He, G., and Liu, J. (2012). A new classification method for high spatial resolution remote sensing image based on mapping mechanism. In *Proceedings of the International Conference on Geographic Object-Based Image Analysis (GEOBIAâ12)*, pages 186–190.

GIULIA – A Spatial Decision Support System for Urban Logistics Interventions Analysis

Tatiana K. Ferrari[1], Cássio F. Rossetto[2], Maria Flávia Adorni[2], Gustavo B. Furuzawa[2], Paulo F. de Oliveira[3]

[1]National Institute for Space Research (INPE) – São José dos Campos - Brazil

[2]Geologística – R. Dr. Cândido Espinheira, 560 – Perdizes
São Paulo – SP – Brazil

[3]World Bank Group – Washington D.C - EUA

tatianak.ferrari@gmail.com,{cassio, flavia, gustavo}@geologistica.com.br, pfernandesdeoliv@worldbank.org

Abstract. *Decision in urban planning involves an understanding of complex interactions between different aspects of the city. The use spatial decision support systems (SDSS) became an important tool to support decision-making, given its capability to integrate spatial and descriptive data and allow simulation of alternative scenarios. In this work, we present a new tool that reproduces the dynamics of e-commerce and urban logistics and that supports decision making for urban planners. The tool allows for assessment of alternative logistics interventions, considering the efficiency aspects from the supply side and overall environmental and travel time savings benefits from the demand side.*

1. Introduction

Decision-making is an essential and vital part of human life and planning process. Real world problems can be very complex, and as argued by Simon (1955) humans have a bounded rationality. Contrary to what is assumed by mainstream economic theory, decision-making is not a perfectly rational decision, due to humans' limited cognitive system and the complexity of the environment.

In the urban space, the process of decision-making is even more complex due to its land heterogeneity and the conflicting interests from several different stakeholders. A variety of analytical techniques have been developed to help decision makers to solve problems. Spatial decision support systems (SDSS), are a particular case, designed to provide decision makers with a problem solving environment within, which they can explore structure and solve complex spatial problems (DENSHAM, 1991).

Aware of these systems' potential, the World Bank, through the Multi-donor Sustainable Logistics Trust Fund, supports a project with the objective to create a platform to analyze the impacts of benchmark urban logistics initiatives and support decision-making. The Project aims to look into the impacts of public policies on e-commerce urban logistics, and the potential of e-commerce to replace consumers individual trips and generate overall positive results in terms of vehicles-mil reduction.

In the last decades, due to technological improvements, on-line shopping has been growing and becoming an important player in the delivery of goods in urban areas, with a potential to grow even more, given the prospects of an increase in internet access[1]. In Brazil, e-commerce is in full expansion; the Webshoppers Report (2016) shows that between 2011 and 2015, e-commerce accumulated a growth of 98.3%, going from about 53 million requests in 2011 to 106 million in 2015.

In traditional retailing, a marketplace is a physical place in which the consumer needs to visit to complete the transaction, resulting in the movement of individuals. With the e-commerce development, this individual displacement of consumers is being replaced by freight deliveries. According to Visser, Nemoto and Browne (2014), the impact of the changes remains uncertain. On one hand, this retail channel could increase vehicle movements within the cities; on the other hand, e-commerce could change the consumers' travel behavior, which could lead to fewer car journeys.

From the point of view of sustainable cities, the growth of e-commerce implies a higher number of freight transportation circulating within the city, which is often associated with congestion problems, noise and air pollution. Further, urban logistics related to e-commerce has some particularities when compared to traditional logistics, with high impacts for cities: parking, more concentration in downtown areas, smaller size of vehicles (thus more), the fact that they are "disguised" as normal commuters. Thus, it is difficult to manage.

In order to reach urban sustainability, the understanding of transportation logistics in the city becomes a strategic issue. In addition, it is important to address in which way some policies and operational initiatives could promote an efficient last-mile delivery in urban areas.

This paper presents a new tool, called GIULIA - Geographic Information for Urban Logistics Interventions Analysis – designed to provide a decision-making environment that relies on the visualization and the modeling of the interrelationship between e-commerce and generated urban logistics, and enables the analysis of selected policies and interventions.

The rest of this paper is organized as follows. Section 2 discusses works that developed systems in the context of urban logistics. Section 3 presents the structure of the proposed system and describes the database. The model is defined in section 4 and section 5 presents the interventions that are analyzed. To complete the presentation of the platform, the section 6 shows an overview of the GIULIA interface. Section 7 concludes and present future steps associated to this work.

2. Related Works

In the last decades, the field of transportation and urban logistics proved to be quite fruitful with many studies and solutions developed to model and treat interventions. In this section, we review some projects related to the creation of the GIS platform and SDSS in the context of urban logistics, which were useful as a reference to this work.

[1] About the growth of internet use in last decades and prospects for next years, see: Global Internet Report, 2015. Internet Society; and specific to Brazil: ICT Households, 2014. Brazilian Internet Steering Committee, São Paulo, 2014.

The KM2[2] project was developed at the Massachusetts Institute of Technology (MIT) to function as an urban logistics atlas with information collected in megacities around the world. It consists of rich datasets of relevant and detailed logistics information and includes the factors that impact delivery performance in general. The project was a pioneer in providing a georeferenced logistics data in a web server, free of charge, for selected cities around the world.

The Luxembourg Institute of Science and Technology developed and hosted the Smart Cities Logistics[3] platform. It is a spatial decision support platform for urban logistics in European cities and depicts information on transportation networks, access restrictions, traffic measures, delivery and transport facilities, administrative units, population, land use and emissions. This information can be overlaid, allowing for cross-analysis. The system also offers a set of modules for hypothetical cost calculations, optimal meeting point, shortest path, sums layers, surface generations by Spline interpolation and traffic measurements to surface along road segments. The platform is for registered users-only, and enables scenario analysis on current nuisances associated to the urban delivery activity, highlighting the variations in CO^2 emission.

Eindhoven University of Technology developed a similar study with the scope of this work in collaboration with the World Bank Group and Agence Marocaine de Developpement de la Logistique (AMDL), entitled "Modelling Traditional City Logistics in Low and Middle Income Countries - Case Study Casablanca".

The goal of the project was to design a method and an associated decision support tool for city logistics policy and regulation making, considering the traditional channel of logistics, improving the livability and competitiveness of the city (BROFT, 2015).

The city was divided into zones, classified by assigned function (terminal, logistics zone, manufacturing, commercial or residential) and logistical characteristics (traditional, modern or mixed). The types of products were grouped into three categories: Fast Moving Consumer Goods, Consumer Goods and Materials. The model input consists of socioeconomic data and drivers observations, and the outputs are emission, congestion and logistics costs indicators.

The above projects created platforms to understand and analyze urban logistics, which enhance our understanding about the dynamics in the complex system of urban logistics. Nevertheless, to the extent of our knowledge, the above tools do not take into consideration demand-related aspects, in particular associated with e-commerce urban logistics. GIULIA platform incorporates these aspects and explores the roles and interrelationships between e-commerce, public policies and urban logistics.

3. Tool Architecture

Densham (1991) defines five elements that compose an operational spatial decision support system (SDSS): 1) the data subsystem; 2) the model subsystem; 3) display generator; 4) report generator; and 5) the user interface. The database provides storage of all the spatial and operational information in the system. The model subsystem specifies a number of mathematical models to manipulate the e-commerce demand, the logistics

[2] Available at: http://lastmile.mit.edu/km2.

[3] Available at: http://iguess-sl.tudor.lu/maps.

demand as well as parameters estimations and interventions results. The display and report generator refers to thematic maps, graphics and tables that report the results of the model. Finally, the user interface regards the visualization and how the user interacts with the system.

Before going deeper into those elements, we list some functional specifications and system language programming: The operational system used for servers is the *Windows Server* with *Internet Information Server* framework. The codes were created using C#.Net, Active Server Pages (ASP.Net) and JavaScript. To perform the georeferenced operations and support the presentation of maps, a free software component called *LeafLet* is adopted, complemented with *MapServer* tool, which work together with Geoservers like *OpenStreetMaps*, combined with specific geographic data layers in Shape File format.

3.1. Database

The database created is relational and georeferenced, including electronically stored data sets in a structured way, with update and recovery resources. The data needed as input in the system was organized in seven groups, according to the subject of the data and are presented in Table 1.

Table 1. Groups of Data Inputs

Group	Input Data
Zoning	Number of zones
	Zone area
Network	Transfer (inter-zone) average speed
	Local average speed
	Internode distance
	Real/straight distance factor
	Path distance between origin-destination zones
Socioeconomic	Population for base scenario
	Coincidental addresses factor
	Specific shopping trips factor
	Commercial address distribution factor
	Average travel time for buy
Commercial	Delivery average weight
	Delivery average volume
	Delivery deadline (days)
	Consumption factor
	Commercial address delivery factor
Vehicle	Fixed daily vehicle cost
	Variable vehicle cost per distance
	Vehicle weight capacity
	Vehicle volume capacity
	Vehicle delivery units capacity
	Vehicle daily journey time
	Interval between routes of the same vehicle
	Efficiency factor
Logistic	Zone origin probability
	Residential delivery sucess factor
	Commercial delivery success factor
	Number of shippers
	Vehicle usage factor
	Average stop time
Environment	Pollution factor

All spatial structure of the model is based on the adopted zoning areas. For this reason, it is important to adopt a consistent zoning with the granularity of the other parameters in the model, in order to allow the analysis of results at reasonable levels. It is also interesting to adopt divisions that may be compatible with other related studies.

Data concerning the commercial aspects are model parameters to characterize the products, consumers and respective consumption. The first thing that need to be considered, in the case of logistic modeling, is that the market needs to be segregated based on the categories of products. The reason it is that logistics deliveries depend heavily on the characteristics of the product and the deadline required. However, it is practically impossible to input each product. Therefore, based on its characteristics, the products can be grouped; the proposed classification is: Fashion and accessories; Cosmetics, perfumes and health; Home appliances; Home decor; Electronics, computing, telephony and mobile; Books, magazines and subscriptions; Sports and leisure; and Food and beverages.

Finally, in order to build a more consistent baseline scenario, it is interesting to divide the population based upon their social class. The objective here is to create homogeneous groups of consumption. Several studies show a direct correlation between social class and consumption. Warner (1960) states that each social class has unique motivations and buying behaviors. Surveys have shown a large disparity among social classes in the online shopping (ICT Households, 2014). Moreover, the configuration of urban spaces is, in general, heterogeneous and organized by clusters of similar groups in certain locations and it usually has a strong correlation with social classes. The consequence is that the delivery flow will not to be similar in all zones of the city.

4. Modelling Specification

4.1 E-commerce demand modeling

Modeling the demand for e-commerce products, is an essential part of the model. This computation is going to determine the amount of deliveries needed in the city and serve as a base for the logistics transportation modeling.

The first step it is to determine the quantity of orders (*DE*) for each zone (*z*) and by product segment (*p*). The total of orders will be the sum of the population of each social class (*c*) in the zone *z* ($P_{z,c}$), multiplied by a consumption factor ($fc_{c,p}$) by the social class c in the segment p.

$$DE_{z,p} = \sum_{C} [P_{z,c} \times fc_{c,p}] \tag{1}$$

The number of orders must be transformed into demand deliveries. Another issue considered in the model is the unsuccessful deliveries. Sometimes, more than one delivery attempt is needed, since it is mandatory the presence of someone responsible for receiving the goods. As a result, the total amount of supply operations exceeds the total of orders.

Another issue is that part of the deliveries is to commercial addresses, since some consumers prefer, by convenience, to receive their orders at their work place, instead at home. Thus, the demand is divided according to the address delivery type in residential (*DH*) or commercial (*DC*). The demand deliveries are determined by:

$$DH_{z,g} = \frac{(1 - fa_g)}{fsh} \times \sum_{p \in g} DE_{z,p} \tag{2}$$

$$DC_{z,g} = \frac{fa_g}{fsc} \times fcd_z \times \sum_{p \in g} \sum_{k}^{NZ} DE_{k,p} \tag{3}$$

Where, the index g indicates a group of joinable products, that is, they may be grouped for delivery in the same loading; fa_g indicates a factor of deliveries in commercial addresses, while fcd_z is a factor of the location of these commercial addresses in each zone; finally, fsh and fsc indicate the success factor in residential and commercial deliveries, respectively.

Finally, one more aspect was considered, related to matching addresses. The goal here is to determine the amount of stops performed in the delivery operation. Frequently, the delivery carries different orders, but it is targeted to one place, requiring only one stop to deliver more than one order. This becomes quite common in densely populated areas, with the presence of horizontal condominiums. Therefore, the number of delivery addresses indicator it is reached as:

$$DP_{z,g}^{m} = DG_{z,g}^{m} \times fac_{z} \tag{4}$$

fac_z represents the coincidental addresses factor in the zone *z,* and DG refers to consolidation of demand deliveries (*DH e DC*), that can be expressed by zone and product group for individual delivery (*I*), pick-up points (*P*) or last mile deliveries (*L*), for (m = *{I,L}*).

4.2. Logistics Distribution Modeling

The interrelationship between demand and logistics is represented by a set of stops (S). The number of delivery addresses($DP_{z,g}^{m}$) will determine the number of deliveries needed and the stops, conditioned by different vehicle types (*v*), which are parameterized by the division factor (*fv*) which may vary by time of day (*t*) and, product group (*g*). Hence, for (m = *{I,L}*), a set of stops is:

$$S_{z,t,v}^{m} = \sum_{g} [DP_{z,g}^{m} \times fv_{t,v,g}^{m}] \tag{5}$$

In the case of deliveries in pick-up or concentration points, the stops are estimated by the number of concentration (*NC*) and pick-up points (*NP*) in each zone *z*, divided by interval of supply measured in terms of day:

$$S_{z,t,v}^{P} = \frac{(NC_z + NP_z)}{si} \times \sum_{g} fv_{t,v,g}^{P} \tag{6}$$

The vehicles usage factor, expressed in the division factor, must total 100% for operations $m = \{I, P, L\}$, i.e., $\sum_{t,v} fv_{t,v,g}^{m} = 1$. The total load in terms of weight (*W*) and volume (*V*) can be calculated by adding the average weight value (*wm*) and volume (*vm*) in a multiplicative form in equation 5.

Finally, in order to structure the results it is required to measure the amount of vehicles needed, the cost of transport and the emissions during delivery processes. Thus, more than the amount of supplies/stops, one needs to know the traveled route, both in terms of distance and in terms of time spent.

To measure the number of routes needed for transport (*R*), it must be considered the maximum constraint between the weight capacity (*CW*) of the vehicle type (*v*), volume capacity (*CV*) and delivery unit's capacity (*CU*) or a function of travel time (*Rt*). Using also an efficiency factor (*fe*) since, in practice, vehicles do not have their full capacity used:

$$R_{z,t,v}^{m} = fe_v \times \max\left\langle \frac{W_{z,t,v}^{m}}{CW_v} \middle| \frac{V_{z,t,v}^{m}}{CV_v} \middle| \frac{U_{z,t,v}^{m}}{CU_v} \middle| Rt_{z,t,v}^{m} \right\rangle \tag{7}$$

The travel time restriction should consider the maximum trip journey (*J*) and stop times (*ts*), travel time for path between stops (*tt*), travel time from origin (*ta*) and travel time to return (*tr*), for final deliveries:

$$Rt_{z,t,v}^{m} = \frac{S_{z,t,v}^{m} \times \left(ts_{z,t,v}^{m} + tt_{z,t,v}^{m}\right)}{\left(J_v - ta_{z,t,v} - tr_{z,t,v} + tt_{z,t,v}^{m}\right)} \tag{8}$$

In the case of routes serving concentration/pick-up points, it should only consider the capacity of the vehicle and not the time, since the carriers may use more appropriated vehicles.

In order to evaluate the travel route times, it is considered an estimation obtained by Daganzo (1984) method, based on the number of stops in each area and the zone dimension. In addition, it was considered a correlation with the network density in each zone. With this method, it´s possible to calculate the distance and time between consecutives route stops.

Another portion of the route time is the distance and time to access the zone. For that, it is taken into account the probability of the zone for supplying the destination zones. It is used a previous calculated origin-destination matrix of routes between zones that give the estimation of route distance and time for each flow between zones.

4.3 Indicators Calculation

The model seeks to provide four indicators as a way to give an overview of transport logistics impacts related to urban e-commerce in the simulated scenarios.

The first indicator refers to the transportation cost (C). The cost needs to be viewed only as an estimation for operational vehicles costs in the logistics for e-commerce goods; this model was not structure to be a distribution cost analysis tool. For each vehicle type ($CV_{z,v}$), the cost is based on fixed the cost (cf_v) times the number of routes needed to transport$\left(R_{z,t,v}\right)$, plus the variable cost (cv_v) that varies in terms of the travelled distance ($D_{z,t,v}$).

$$CV_{z,v} = \sum_{t}\left(R_{z,t,v}\right) \times cf_v + \sum_{t}\left(D_{z,t,v}\right) \times cv_v \tag{9}$$

It is considered important to measure, even if in a simplified way, the direct benefits for e-consumers, when compared to the conventional alternative shopping in physical stores. The proposed indicator calculates the hours saved by the buyers (B), considering the total number of e-commerce purchases ($DE_{z,p}$) multiplied by the average time of travels with shopping purpose.

$$B_z = \left(\sum_{p} DE_{z,p}\right) \times tb_z \tag{10}$$

Where, tb_z is the average travel time for buying.

The other two indicators are related to the environmental impact. The model calculates the energy efficiency (EE) and total emission (E) generated by transport. Both the energy efficiency and emission considered the total travelled distance ($D_{z,t,v}$) by motorized vehicles (M), and are multiplied by a factor of fuel consumption in the case of energy efficiency and by a pollution factor in the case of emissions.

$$EE_z = \sum_{v \in M} \sum_{t} \left(D_{z,t,v} \times ff_v \right) \tag{11}$$

$$E_z = \sum_{v \in M} \sum_{t} \left(D_{z,t,v} \times fp_v \right) \tag{12}$$

5. Interventions to be Analyzed

As stated by Arampatzis et al. (2004) a characteristic of the decision support tool is that it pre-defines some "abstract" interventions and incorporates them as "methods" into the system, which are algorithms and procedures for each intervention type.

Seven benchmark interventions were taken into consideration to be applied in the scenarios using the proposed model: pick-up points; night/off-peak delivery; non-motorized delivery; unassisted delivery; joint delivery systems; redefinition of load/unload areas; and urban consolidation centers. The impacts of each intervention can be modeled by changing the functions incorporated in the system. Table 2 shows the relationship between the parameters and interventions.

In the case of the pick-up point intervention, the goal is to modify the transport mode in the last mile, replacing the freight transport by an individual transportation, which could potentially be non-motorized. The model includes some pick-up points based on density by zone and reallocates part of the e-commerce demand to this delivery option.

The night/off-peak delivery action concerns the change in the distribution operating hours. The idea is to avoid high traffic hours and difficult parking for unloading. This interference affects the vehicle travel speed and reduces the parking time, and consequentially makes the deliveries more efficient. Nevertheless, it may encounter difficulties in finding someone to receive the product, which can increase the unsuccessful deliveries and generate problems regarding noise and security concerns.

An interesting intervention discusses the use of non-motorized deliveries instead of motor vehicles deliveries. The deliveries can be implemented in the last mile using bicycles, cargo tricycles and other vehicle adaptations, as well as deliveries on foot.

The unassisted deliveries intervention seeks to achieve deliveries without the mandatory presence of a receiver. This intervention can be modeled simply by changing the success rate of the deliveries function and the average stop time.

A promising alternative to optimize the distribution of loads generated by e-commerce is the joint delivery system. In the model, this intervention simply changes the number of shippers operating in the logistics, which reduces the number of route origins and allows delivery consolidation to each delivery zone. From the city's point of view, the joint logistics operation provides the best use of employed vehicles; consequently it reduces traffic interferences and environmental impacts.

The interference of loading and unloading stops in urban areas is quite relevant, considering the lack of parking places available in the area. The absence of parking can generate greater movements of vehicles in search of vacancies, or even lead to the practice of using undue areas and traffic violations. The intervention establishes specific areas for loading and unloading within the urban area and is modeled by a change in the stop time per zone.

Table 2. Interventions characteristic and impacts

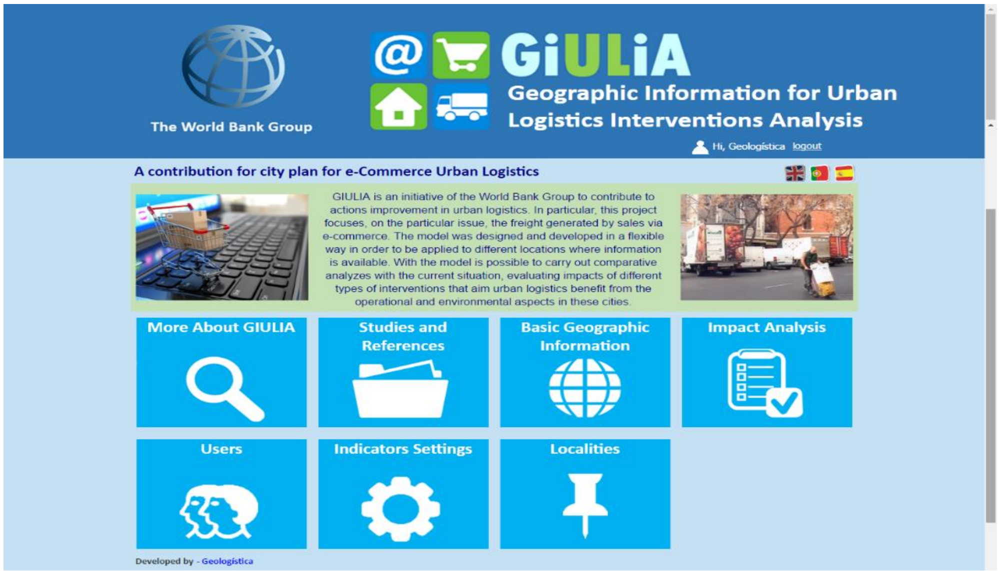
Parameters		Interventions Types						
		Pick-up points	Night/off-peak delivery	Non-motorized delivery	Unassisted deliveries	Joint delivery systems	Redefinition of load/unload areas	Urban consolidation centers
fsh	Deliver success factor at home		●		●			
fsc	Deliver success factor for commercial address		●		●			
fi	Pick-up points density	●						
fap	Pick-up point deliver factor	●						
fcp	Consolidation deliver factor (non-pick-up deliveries)							●
NC	Number of consolidation centers							●
NS	Number of shippers					●		
si	Supply interval (days)	●						●
fv	Vehicle usage factor		●	●				
ts	Average stop time		●	●	●		●	

The urban consolidation center is a smart solution with the aim of creating storage points or intermediate transshipment in the city where it can be switched from one type of vehicle to another smaller or even complete the operation at different times.

Furthermore, the model also predicts changes in demand, in order to give more flexibility to the proposed model, to tune the demand and allow simulating future scenarios in terms of population variation.

6. Visualization and Scenarios

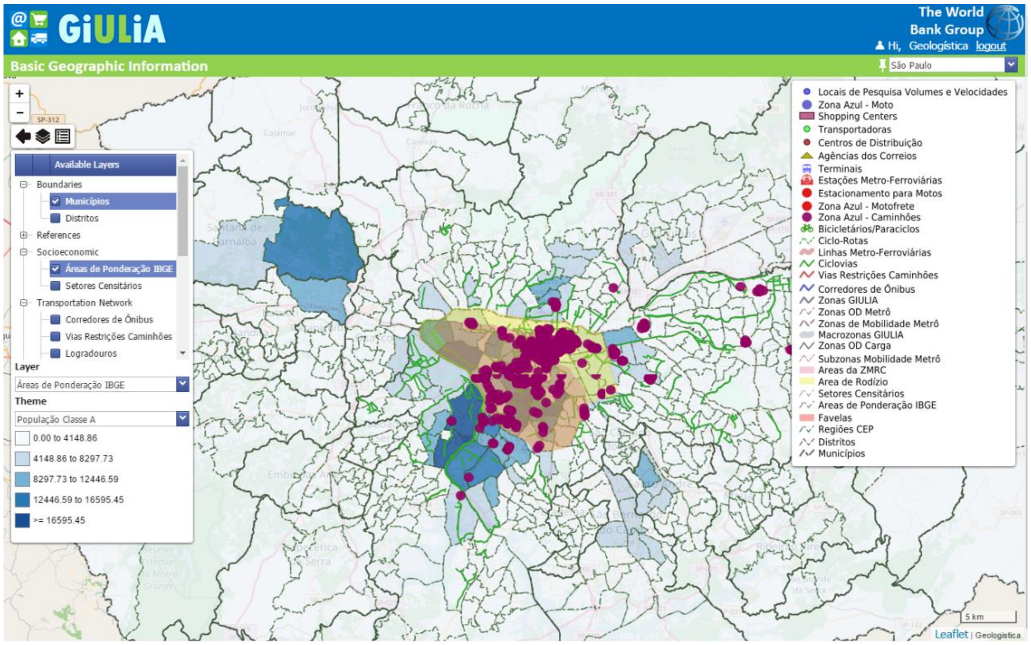
Figure 1 illustrates the initial page of the system, where the user has some initial information about GIULIA and has the main menu options. In Studies and References, the user can get extra information about the theme, where publications are available grouped according to the following themes: e-commerce, urban logistics, information systems and interventions experiences. Another option in the main menu is the consult of basic geographic information, socioeconomic and infrastructure characteristics and other available layers of interest (Figure 2). Each geographic information layer has a set of metadata information available for consultation.



A contribution for city plan for e-Commerce Urban Logistics

GIULIA is an initiative of the World Bank Group to contribute to actions improvement in urban logistics. In particular, this project focuses, on the particular issue, the freight generated by sales via e-commerce. The model was designed and developed in a flexible way in order to be applied to different locations where information is available. With the model is possible to carry out comparative analyzes with the current situation, evaluating impacts of different types of interventions that aim urban logistics benefit from the operational and environmental aspects in these cities.

Figure 1. Initial Page with Menu Options

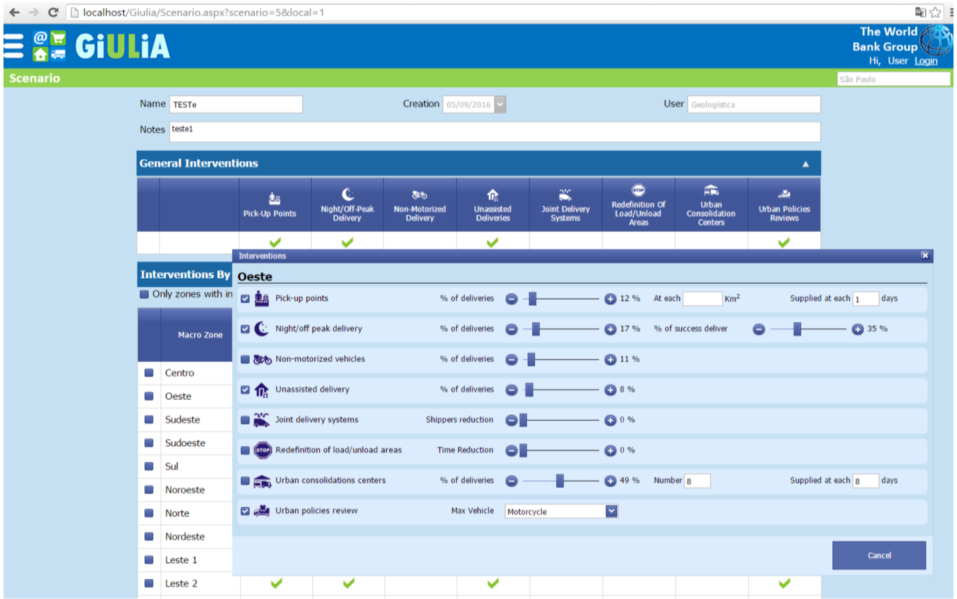


Layer: Áreas de Ponderação IBGE

Theme: População Classe A

Figure 2. Consult of Basic Geographic Information

As stated before, what differentiates a DSS is the model element that allows the user to construct alternative scenarios. In this aspect, the interface becomes an important tool as it allows the user to interact with the environment, manage scenarios, analyze and compare them. In the menu Impact Analysis, all the interventions considered in the model are listed to the user. The program has a baseline scenario for each locality that is always present and serves as base parameters settings for new scenarios creation and comparison. The management of scenarios is done by changing the proportion of deliveries to a specific intervention (Figure 3). The results come through thematic maps and graphics, which display a set of indicators (Figure 4).



Name TESTe | Creation 05/09/2016 | User Geologística

Notes teste1

General Interventions

Pick-Up Points	Night/Off-Peak Delivery	Non-Motorized Delivery	Unassisted Deliveries	Joint Delivery Systems	Redefinition Of Load/Unload Areas	Urban Consolidation Centers	Urban Policies Reviews
✓	✓		✓				✓

Interventions By

Only zones with in

Macro Zone
Centro
Oeste
Sudeste
Sudoeste
Sul
Noroeste
Norte
Nordeste
Leste 1
Leste 2

Interventions

Oeste

- ☑ Pick-up points — % of deliveries 12 % — At each Km² — Supplied at each 1 days
- ☑ Night/off peak delivery — % of deliveries 17 % — % of success deliver 35 %
- ■ Non-motorized vehicles — % of deliveries 11 %
- ☑ Unassisted delivery — % of deliveries 8 %
- ■ Joint delivery systems — Shippers reduction 0 %
- ■ Redefinition of load/unload areas — Time Reduction 0 %
- ■ Urban consolidations centers — % of deliveries 49 % — Number 8 — Supplied at each 8 days
- ☑ Urban policies review — Max Vehicle Motorcycle

Cancel

Figure 3. Page for alternative scenario construction.

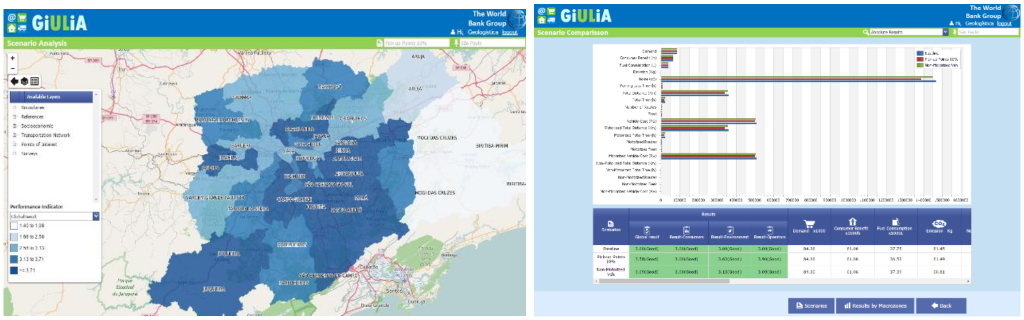


Figure 4. Example of impact analysis: map of scenario's indicator and graphic to compare scenario's results.

7. Conclusion

The aim of this paper was to introduce GIULIA, a spatial decision support system, designed to evaluate the relationship between online shopping and urban logistics, as well as the impact of interventions in urban logistics. The innovation brought from GIULIA was the modelling of a complex and important issue in urban areas with the objective to reach more sustainable cities.

Freight flows are fundamental in any city, but, usually, are associated with negative impacts on cities, such congestions, air pollution and noise. With the e-commerce consumption growth, there are expectations of an increase in logistical transport within cities. Nonetheless, freight transportation in urban areas is often missed in the public planning analysis, despite being a key factor to reach urban sustainability.

In light of this, the GIULIA system is an attempt to bring a more comprehensive analysis to the issue and help decision makers to analyze potential interventions. Not only may the system build urban logistics analysis from demographic data, but it may also spread good practices in urban logistics amongst its users, who will be mainly urban planners. As mentioned before, e-commerce is a fast-growing market in Brazil and well-prepared urban planners will make a difference in such complex scenario.

The software was built with the purpose to be applied to any city, requiring only the allocation of local data. This is perhaps the biggest challenge to its implementation. The problem in question requires a large amount of data and from different sources.

The next step it is the application of GIULIA in a real city to test its functionalities and the impact analysis. The city of São Paulo in Brazil was chosen for the first application, because it has the major e-commerce sales shares in the country and faces great problems in urban transportation.

8. Acknowledgements

This work was sponsored by the World Bank Group and the Multi–Donor Trust Fund for Sustainable Logistics (MDTF–SL). This Project is part of the MDTF-SL funds activities on strategic themes in sustainable logistics that benefit low income and developing countries.

References

Arampatzis, G., Kiranoudis, C.T., Scaloubacas, P., Assimacopoulos, D. (2004), “A GIS-based decision support system for planning urban transportation policies”. In European Journal of Operational Research, 152, pages 465-475.

Broft, A. D. (2015) “Modelling traditional city logistics in low and middle income countries: a method and tool as decision support for policy makers”. Eindhoven: Technische Universiteit Eindhoven.

Daganzo, C. F. (1984), "The distance traveled to visit N points with a maximum of C stops per vehicle: An analytic model and an application". In Transportation Science 18(4), pages 331-350

Densham, P. J. (1991), “Spatial decision support systems”, In: Geographical information systems: Principles and applications, ed. J. Maguire, M. S. Goodchild, and D. W. Rhind. London: Longman Publishing Group, chapter 26, pages 403-412.

E-bit/Buscapé Company. (2016), “WebShoppers Report”, 33rd Edition, http://portal.ebit.com.br/webshoppers, June.

Simon, H. A. (1955), “A behavioral model of rational choice”. In Quartely Journal of Economics, 69, pages 99-118.

Survey on the use of information and communication technologies in Brazilian households: ICT households 2014. Brazilian Internet Steering Committee – CGI.br, São Paulo, 2015.

Visser, J.; Nemoto, T.; Browne, M. (2014), “Home Delivery and the Impacts on Urban Freight Transport: A review”. Procedia – Social and Behavioral Science, 125, pages 15 – 27.

Warner, W. L. (1960), “Social class in America: an evaluation of status”. New York: Harper and Row.

Clustering Approaches and Ensembles Applied in the Delineation of Management Classes in Precision Agriculture

Eduardo A. Speranza[1], **Ricardo R. Ciferri**[2], **Cristina D. A. Ciferri**[3]

[1]Embrapa Agricultural Informatics – Brazilian Research Agricultural Corporation
13083-886 – Campinas – SP – Brazil

eduardo.speranza@embrapa.br

[2]Departament of Computer Science – Federal University of São Carlos
13.565-905 – São Carlos – SP – Brazil

ricardo@dc.ufscar.br

[3]Department of Computer Science – University of São Paulo at São Carlos
13.560-970 – São Carlos – SP – Brazil

cdac@icmc.usp.br

Abstract. *This paper describes an experiment performed using different approaches for spatial data clustering, aiming to assist the delineation of management classes in Precision Agriculture (PA). These approaches were established from the partitional clustering algorithm Fuzzy c-Means (FCM), traditionally used in this context, and from the hierarchical clustering algorithm HACC-Spatial, especially designed for this PA task. We also performed experiments using traditional ensembles approaches from the literature, evaluating their behavior to achieve consensus solutions from individual clusterings obtained from features splitting or running one of the abovementioned algorithms. Results showed some differences between FCM and HACC-Spatial, mainly for the visualization of management classes in the form of maps. Considering the consensus clusterings provided by ensembles, it became clear the attempt to achieve an agreement result that most closely matches the original clusterings, showing us some details that may go undetected when we analyse only the individual clusterings.*

1. Introduction

Precision Agriculture (PA) is an agricultural management system driven by spatio-temporal variability of soil and culture features of a crop. These parameters may be obtained from particular procedures and techniques based on information technology, remote sensing and Global Positioning System (GPS) [Molin 2003, Vendrusculo and Kaleita 2011]. Unlike conventional agriculture, where agricultural inputs and correctives are evenly applied across the cultivation area, PA enables its users to manage them in a site-specific way, aiming the maximization of profit cutting of yield limiting factors. Moreover, this system allows farmers to fit crop needs and supply of inputs, helping to reduce the environmental damage [Schwalbert et al. 2014]. Because of its highly dependency of the spatio-temporal variability built-in data collected on the field, the adoption of decision-making processes based on PA suggests data collection at high spatial resolutions. However, this usually is not possible for most farmers, because several factors such as the high cost of acquiring satellite images and gathering data on the field, beyond the need to acquire services and automated machinery able to perform variable rate interventions. In these cases, the delineation of subfields spatially internal to the crop area, which the internal spatial variability is so negligible as to allow for evenly distributed internal interventions, is a way to disseminate the adoption of PA even using accurate spatial resolutions (e.g. between 10 and 30 meters). These subfields, known as management classes, may be composed by one or many spatially contiguous areas in the coordinates space, known as management zones [Taylor et al. 2007]. Taking into account these concepts, it is really intuitive to relate the delineation of management classes with traditional clustering algorithms, such as Fuzzy c-Means (FCM) [Bezdek et al. 1984]. However, PA tasks produce complex and non-conventional data, composed by two distinct spaces: features, regarding the events occurring in the crop; and coordinates, regarding the spatial location where these events took place. Thereby, because of its complexity, the coordinates space must to be handled in different ways by clustering algorithms. With the purpose of solving this challenge, Ruß and Kruse 2011 developed an agglomerative hierarchical clustering algorithm, known as HACC-Spatial. The HACC-Spatial enables the delineation of management classes preserving the spatial contiguity as much as possible, in order to facilitate easy visual interpretation of the user while maintain the coherence of the clustering obtained by events related to soil and plants.

Using algorithms composed by different features and parameters, such as FCM and HACC-Spatial, to solve clustering problems present in any domain, can generate different results and hence questions regarding which of them is the best solution. In order to clarify such questions, several approaches enabling consensual and more robust clusterings have been emerged in the literature. These clusterings, known as ensembles, must be obtained from different ways, such as individual clusterings using different kinds of algorithms, parameters configurations or subsets of features at the same data set [Ghosh and Acharya 2011]. Our work described in this paper were aimed to evaluate, from internal clustering validation measures, the accuracy of clusterings representing management classes that were obtained individually using the FCM and HACC-Spatial algorithms, as well as using more robust and consensual clustering ensembles to consolidate individual results and feature space partitioning.

The remainder of the paper is structured as follows. In section 2, we briefly describe the FCM and HACC-Spatial algorithms and approaches commonly used to delineate management classes in PA, beyond the ensemble approach used in our work. In section 3, we present the methodology used for the experiments. In section 4, we present results for experiments using reald data. Finally, in section 5, we present our conclusions and provide suggestions for future work proposals.

2. Background and Related Work

Some clustering algorithms have been used to assist the delineation of management zones in PA. Nevertheless, most of the approaches available in the literature use the Fuzzy c-Means algorithm (FCM) as a basis for this task. Based on the standard clustering algorithm k-means [MacQueen et al. 1967], the Fuzzy c-Means algorithm (FCM) [Bezdek et al. 1984] calculates, at each iteration, the membership (ω_k) of each data sample with respect to each one of the desired clusters. This calculation takes into account the distance (d) from any particular data sample to each cluster centroid and a fuzzification parameter (*m*), defined by the user with default value of 2. At the end of each iteration, clusters centroids are recalculated taking into account all dataset samples and their membership values to each cluster. Instead of k-means, FCM convergence results not only to assign each sample to a unique cluster (hard clustering), but in a membership matrix with 0 to 1 values for each sample with respect to each cluster, known as fuzzy partition matrix (soft clustering). This matrix is one of the FCM advantages regarding hard clustering algorithms, providing better results for situations that have a difficult separation and overlapping datasets. However, like k-means, FCM centroids are randomly initialized, making the results susceptible to a local minima.

The main reason for using FCM in the context of this application is linked with the fact that abrupt changes do not occurs in soil and plant attributes in small enough parcels of the crop, causing input data and the obtained clusters to consider a membership degree. Over the years, several approaches in the literature using FCM and considering different types of these attributes have been developed. Brock et al. 2005 used FCM to delineate management zones considering historical yield data from corn-soybean rotation crops, indentifying the spatial association of the obtained maps with soil maps. Already Kitchen et al. 2005 used FCM to delineate management zones considering ratios of soil electrical conductivity (EC) in different depths (bulk of EC) and relief data, comparing them with yield zones obtained from historical yield data. As a result, it was found that the bulk of EC combined with relevant data are strong indications for management zones. Similar conclusions were obtained by Morari et al. 2009, including measures of soil and electrical resistivity data. The work of Li et al. 2007 used, in addition with abovementioned attributes, features indicating rates of organic matter and biomass. In this case, due to the large number of attributes, an intermediate phase of principal component analysis before getting the management zones by FCM was performed. High-resolution satellite images also appears as inputs to obtain management zones using the FCM, as in works of Song et al. 2009 and Zhang et al. 2010. More recently, Milne et al. 2012 used FCM to find management zones from smoothed spatial data obtained from three different methods. The results were compared with crop responses regarding the application of different nitrogen rates. The work of Scudiero et al. 2013 shows, using FCM to obtain management zones, that combined bare-soil and EC data can contribute to find spatial variability of a crop. The KM-sPC approach [Córdoba et al. 2013] allowed to show the importance of a principal component analysis considering the coordinate space to reduce the stratification provided by FCM when management zones are displayed in form of maps. This approach were used again in a pratical nitrogen management of wheat [Peralta et al. 2015]. The study of Chang et al. 2014 compared management zones generated by FCM using reflectance data regarding the soil properties and productivity, showing that it is feasible the use of an active canopy sensor for this PA application.

Despite the widespread use of FCM for this task, the coordinates space of PA datasets, composed by spatial coordinates variables (e.g., latitude and longitude), have been used only in preprocessing steps or to show the management classes provided by clustering in the form of maps. This fact does not prevent the use of these maps by automated machinery for variable rate interventions, but the reduction of spatial contiguity, causing stratification of management classes in too many areas, can confuse visual analysis by experts. In order to solve this problem, the HACC-Spatial hierarchical clustering algorithm were developed by Ruß and Kruse 2011. This approach takes into account spatial restrictions for clustering samples, and considers a preprocessing step to perform an initial tessellation of them in small spatial clusters, using the k-means algorithm at the coordinates space. Such subdivision aims to reduce computational costs by decreasing the number of steps of the construction of the hierarchical tree (or dendrogram) produced by the algorithm, regarding the geostatistics principle claiming that spatially very close samples tends to have close enough values in the features space [Matheron 1963]. As a result, a structure similar to a Voronoi diagram should be obtained by the preprocessing step. From this moment, each dendrogram step merges the most similar clusters, according to the feature space. First, only spatially adjacent clusters can be merged, providing the maintenance of spatial contiguity. However, when a user-defined contiguity threshold *cp* is reached, this restriction is switched off. This threshold is associated to the ratio of the average distances between the samples belonging to adjacent clusters and the average distances between samples belonging to non-adjacent clusters.

Because of the differing nature of FCM and HACC-Spatial (partitional and hierarchical, respectively) and the spatial restrictions used for one of them, are expected distinct clustering results for the same dataset, making it difficult for the user to choose the best approach. A feasible solution to solve this question can be achieved using ensembles. Ensembles are able to combine multiple sample clusterings in a unique and consolidated one, known as consensus solution. These kind of approach can be used to meet several requirements, such as: increase the quality of the solution, providing more robust clusterings; select models; reuse knowledge; find consensus between clusterings obtained from subsets of features or subsamples, among others [Ghosh and Acharya 2011].

The main aim of a clustering ensemble is to find a consensus solution composed by an unique clustering to share as much information as possible derived from original clusterings. This sharing can be measure by the average of normalized mutual information (ANMI), where the desired optimal value is ANMI equal to 1 [Strehl and Ghosh 2002]. The main goal of the three ensembles algorithms developed by Strehl and Ghosh 2002 is to build general approaches to obtain consensus from individual clusterings aiming at maximizing the ANMI value. These algorithms were evaluated by the authors in scenarios where individual clusterings were composed by distinct features, distinct subsamples or distinct clustering algorithms. The Cluster-based Similarity Partitioning Algorithm (CSPA) is the simplest and most obvious heuristic. It is based on the fact that two objects have a similarity of 1 if they are in the same cluster and 0 otherwise. Thus, a *n* x *n* binary matrix, where *n* is the number of samples, is created for each original clustering. To recluster these samples, a similarity-based clustering algorithm based on graph partitioning is used [Karypis and Kumar 1998]. The computational and storage complexity of this algorithm are both quadratic in *n*. The HyperGraph Partitioning Algorithm (HPGA) addresses the clustering ensemble as a hypergraph partitioning problem, where hyperedges represent the original given clusters as indications of strong bonds. To recluster the samples, a partitional hypergraph algorithm, cutting a minimal number of hyperedges is used [Han et al. 1997]. In this case, while CPSA only considers pairwise relationships, HPGA includes original clustering relationships. Finally, the Meta-Clustering Algorithm (MCLA) represent each cluster by a hyperedge, and then group and collapse related hyperedges (or clusters), attaching each sample to the collapsed hyperedge in which it belongs more activelly. At the end, a graph-based clustering of hyperedges is performed, indentifying consolidated ”clusters of clusters”. In contrast to CPSA, HPGA e MCLA have linear computational and storage complexity. Still according to [Strehl and Ghosh 2002], the MCLA tends to provide better ANMI values when the consensus solution were obtained from individual clusterings with low noise rates and diversity; and HPGA and CSPA are usually better were obtained from individual clusterings with high noise rates and diversity.

From the abovementioned algorithms, it were possible for us to prepare some experiments, described in section 3, combining distinct approaches that can be applied in the delineation of management classes in PA. Results of these experiments are presented in section 4.

3. Methodology

The methodology used in ours experiments follows the concepts of Knowledge Discovery in Databases (KDD). According to Fayyad et al. 1996 and Weiss and Indurkhya 1998, at least three main steps of KDD process should be taken into account when it will be used: preprocessing, data mining (or pattern extraction) and post processing. The planned activities for each one of these steps, in the context of management classes in PA, are described below.

3.1 Preprocessing

The preprocessing step comprises the changes that should be made in a raw dataset when it will be used by a KDD process, preparing it to the next steps. Regarding to spatial data, in addition to very common preprocessing activities, such as standardization, cleaning and feature selection, the spatial interpolation must be performed in order to accommodate data samples in a single and regular spatial grid [Vieira 2000]. This activity is required, because PA datasets are caught using different kinds of sensors and samples densities, usually at distinct spatial spots in the same area. Another important activities in this step are: verifying data distribution using probabilistic density functions, as a preassessment of possible distortions that can occur in clustering algorithms when using non-Gaussians distributed features; verifying features correlations, using methods such as Pearson's Coefficient Correlation [Benesty et al. 2009]; and data standardization, reducing the bias caused by features with highly predominant scales relative to the others.

3.2 Data Mining

The data mining step can be viewed as an iterative process, where should be used different solutions to improve the accuracy of the results. In the context of our work, due to the fact that datasets had no previous classification, clusterings tasks need to be considered. Therefore, the approaches to be used are classified as non-supervised machine learning algorithms [Mitchell 1997]. In this step, we used the HACC-Spatial and FCM algorithms in the traditional way and also combining results by ensembles. HACC-Spatial was run using non-spatial features of the whole dataset to calculate dissimilarity values at each step of dendrogram, and spatial features to build the initial tessellation and to support adjacency treatments at each step of dendrogram (Approach I). In the other hand, FCM was run in its traditional way, i.e., using only non-spatial features (Approach II). Regarding ensembles, it was created an approach to found consensus clusterings from individual results provided by Approach I and Approach II (Approach III); and another two approaches to found consensus clustering from individual results provided by non-spatial features subsets of soil, altimetry and yield using HACC-Spatial (Approach IV) and FCM (Approach V). The ensembles approaches was run using CSPA, HPGA and MCLA algorithms described in section 2, and the results with best values of ANMI were chosen as the best solution for each approach.

According to domain expert users, at least 2 and at most 5 management classes should be considered for a crop [Molin et al. 2015]. Thereby, the five abovementioned approaches were run using *k*=2 to 5 clusters for the experiments, when using FCM (partitional), and the same values for dendrogram cuts, when using HACC-Spatial (hierarchical). Regarding to dissimilarity measures, the Euclidean distance were used for all approaches. In relation to other parameters and customizations, for approaches using FCM, the standard fuzzification value *m*=2 was fixed, and samples were associated with the cluster where were achieved a higher membership degree. For approaches using HACC-Spatial, were used a binding criteria similar to average-linkage algorithm [Sokal 1958], because of its ability to handle data sets with presence of outliers. Other HACC-Spatial parameters, like initial tessellation number of clusters (*k*) and *cp*, were defined during the experiments.

3.3 Post Processing

Finally, in the post processing step, we used two internal validation criteria: the SD criteria and the silhouette width criteria. These criteria allow comparing and evaluating the effectiveness of the five approaches when they are run at the same number of clusters. The SD criteria [Halkidi et al. 2000, Halkidi and Vazirgiannis 2001] allows to verify, for each obtained clustering, how cohesive and well separated are the clusters, from average values of intra-cluster variance and distances between clusters centroids. In this case, optimal values should be closer to 0. The silhouette width criteria [Rousseeuw 1987] follows the same principles of SD, but using dissimilarity values of a sample regarding its associated cluster and the nearest neighbor cluster. In this case, values closer to 1 indicates that the sample has been allocated to the correct cluster; and values closer to -1 indicates that the sample could have been better allocated to the nearest neighbor cluster. According to Vendramin et al. 2010, the silhouette width criteria, in comparison to other internal criteria in the literature, can provide, in general, more effective assessments about the internal structure of the clusters.

4. Experiments

In this section, we present the results obtained from experiments using real data, following the methodology described in section 3. These data are composed by samples collected on an experimental crop field of sugarcane culture. This field has an area around 17 hectares belonging to Fazenda Aparecida, located in Mogi-Mirim, São Paulo state, Brazil, with central coordinates 7505136N (latitude) and 299621E (longitude), given the spatial reference system UTM Zone 23S. Figure 1 shows the contour shape and a cropped image of the experimental field.

The raw datasets used in our work comprises measures of soil electrical conductivity (EC), in milisiemens per meter; altimetry quota, in meters; and historical yield, in tons per hectare or culms per square meter. The samples were collected at different times and by different sensors or processes, providing us six conventional features associated with spatial coordinates: soil electrical conductivity at 30 e 90 cm deep in 2010

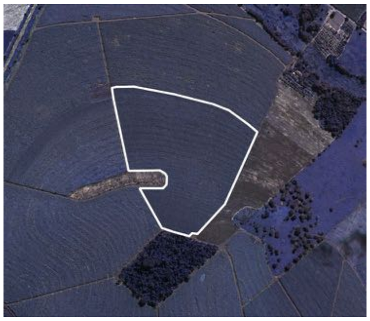
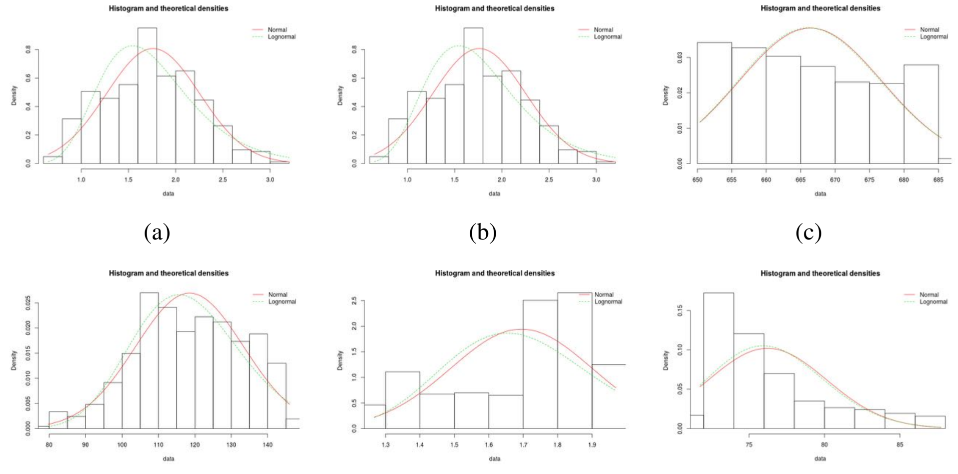


Figure 1. Experimental crop field of sugarcane (white contour) with a cropped image in the background provided by the World View 2 satellite (April 30, 2011).

(EC30 and EC90); altimetry quota (Quota); and historical yield in 2010 (Yield2010), 2012 (Yield2012) and 2013 (Yield2013). It is worth mentioning the need for historical yield data, because they could be considered susceptible to anthropic and climatic factors over the years. In addition, the rainfall data of the whole farm in the agricultural years should be considered to support some analysis: 1601 mm in 2010 (July 2009 to June 2010), 1538 mm in 2012 (July 2011 to June 2012) and 1599 mm in 2013 (July 2012 to June 2013). The probabilistic density distribution of EC30, EC90 and Yield2010 features could be described by Gaussians, with most values around the mean. On the other hand, the distributions of Yield2012 and Yield2013 indicates, respectively, predominance of higher and lower yield values, probably affected by the abovementioned factors. A special case occurs with the Quota feature, where average values are the minority because the experimental area has a slight slope and narrow in the central region. These distributions are shown in Figure 2.



(a) (b) (c)

(d) (e) (f)

Figure 2. Probabilistic density distributions of dataset features: (a) EC30; (b) EC90; (c) Quote; (d) Yield2010; (e) Yield2012; e (f) Yield2013.

Applying the Pearson's Coefficient Correlation between pairs of features, were verified that EC30 and EC90 hold the most positive correlation of the dataset. In general, the Quote feature was well correlated with all other features, and negatively (oppositely) correlated with Yield2010. Regarding to yield data, Yield2012 and Yield2013 features are highly correlated, and negatively correlated with Yeld2010 feature. The negative correlation of Yield2010 with other yield years could be influenced again by the anthropic and climatological factors.

Using the concepts of preprocessing described above, the dataset features were interpolated in a single regular spatial grid with spatial resolution of 20 meters. This value was calculated using the average coordinates spacing between samples for each one of the six features of the original data set. Simple algorithms, like the average of *k* nearest neighbors [Altman 1992], were used to interpolate features with higher sample densities. On the other hand, more sophisticated algorithms, like kriging [Matheron 1969], were used to interpolate features with smaller sample densities. After applying this process, each dataset feature were distributed in 415 samples spatially represented by points with latitude and longitude coordinates. Figure 3 shows raw samples of soil electrical conductivity (high density) and yield (medium density) and their respective interpolated samples in the same regular spatial grid. Lower values are represented by lighter colors, while higher values are represented by darker colors.

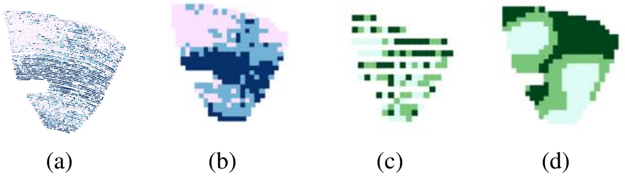


Figure 3. Example of raw and interpolated data in 3 classified intervals: (a) EC30 raw data (9046 samples); (b) EC30 interpolated data (415 samples); (c) Yield2010 raw data (111 samples); (d) Yield2010 interpolated data (415 samples).

Especially for the HACC-Spatial algorithm, when it was run in the context of approaches I, III and IV, the *cp* parameter was set to 0.5, according to the best results obtained by Ruß and Kruse 2011. Initial tessellation (*k*) was set to 200, after checking a significant increase in internal variance of the clusters for the following levels of the dendrogram.

Figures 4 and 5 show, respectively, linear charts containing values achieved by both SD and silhouette width criteria for the five proposed approaches, regarding *k* values between 2 and 5. Through these charts, we can observe better results for *k*=3, where we can found, in general, smaller values of SD and larger values of silhouette width.

By analyzing the results using ensembles, the charts of figures 4 and 5 show us that the approach IV, in the most of cases, achieved poor results regarding both the internal criteria. Therefore, we can conclude that the heuristic of HACC-Spatial algorithm, considering spatial relationships during the construction of the hierarchy, tends to be more consistent when using all features (approach I) than when using individual clusterings by features split to obtain a subsequent consensus by ensembles (approach IV). On the other hand, approach V achieved better results than approach IV, showing that in some cases consensus solutions from individual FCM clusterings by features division can be used to replace solutions provided by approach II. Finally, the approach III results shown, for all *k*

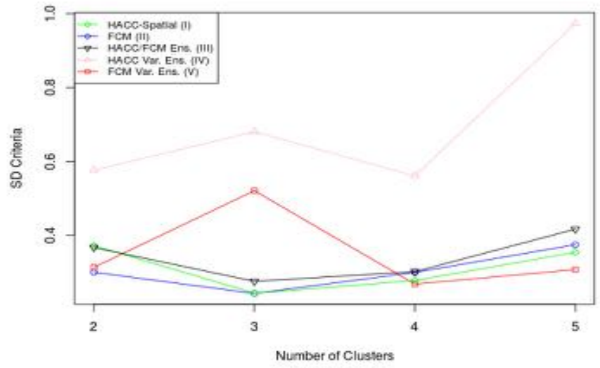


Figure 4. SD criteria values. Each line corresponds to values of SD achieved by the respective approach, considering *k*=2 to 5 clusters.

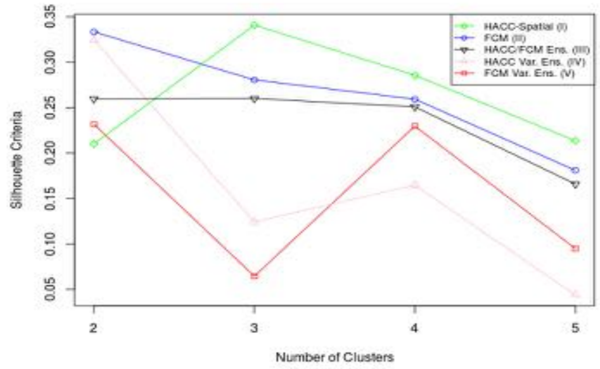


Figure 5. Silhouette width criteria values. Each line corresponds to values of SD achieved by the respective approach, considering *k*=2 to 5 clusters.

values, attempts to find consensus from clusterings obtained by approaches I and II, with slight variations in both the internal criteria values.

Beyond the analysis using internal criteria, we used the visualization of management classes in the form of maps to perform some observations. These analysis were performed from *k*=5 to 2 clusters, in order to observe some effects of agglomerative hierarchy provided by HACC-Spatial approaches. For *k*=5, management classes exhibited generally pronounced stratification, hindering the analysis and an accurate understanding by expert users. For *k*=4, were used, for comparsion with the clustering results, the interpolated dataset from each feature, classified in 4 classes of equal intervals (Figure 6). For each feature, lighter colors represent samples with higher values, while darker colors represent samples with lower values.

Figure 7 shows the results obtained by approaches I, II and III for *k*=4. As can be seen, the results are quite similar for management classes identified with the same color. We can observe the shaping of an isolated area on the top left of the map (blue),

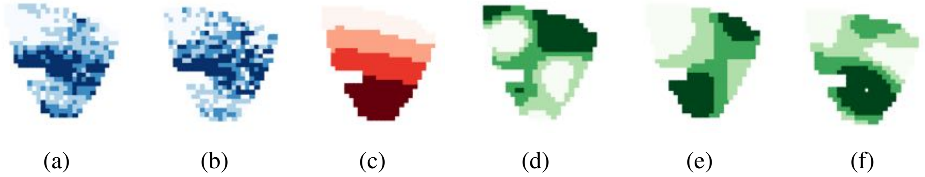


Figure 6. Interpolated data classified in 4 equal intervals: (a) EC30; (b) EC90; (c) Quote; (d) Yield2010; (e) Yield2012; e (f) Yield2013.

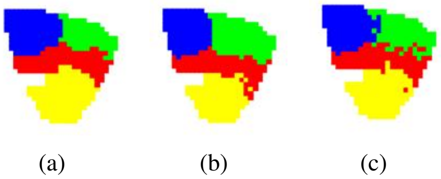


Figure 7. Results for *k*=4: (a) HACC-Spatial (I); (b) FCM (II); and (c) HACC/FCM Ensemble(III).

representing a low elevation region with lower rates of soil EC and historical yield. At the green area, also located in a low elevation region, can be observed medium values of yield and soil EC. Already at the red area, corresponding to a middle elevation region, can be observed a strong influence of extreme values of soil EC for its formation. Finally, the yellow area, located at a high elevation region, shows higher rates of yield.

Figure 8 shows the results obtained for *k*=3, regarding the approaches I, II and III, where were achieved the lowest value of SD criteria (approach II) and the highest value of silhouette width criteria (approach I) of the whole experiment. From this figure, can be observed that approaches I and II achieved very similar results. In both cases, the green and blue areas obtained for *k*=4 were practically kept. The main difference between these results is focused at the subdivision between green and red areas. While approach I is forced to merge two clusters because of the hierarchical caracteristics of HACC-Spatial, making the red area be composed by the most similar areas in *k*=4 (red and yellow), the approach II recalculates again which are the clusters where all samples should be assigned, promoting a greater amount of change. Neverthless, the differences observed between both approaches are quite small, which may still be noticed a strong influence of the low frequency of medium values of Quota in approach II, contributing for the user to clearly note the red region with higher values and blue and green regions with lower values of this feature. Regarding to ensemble approaches, Figure 8 (c) further reinforces that approach III, in turn, tried to find a consensus for these subdivision differences, turning the final map quite stratified.

Finally, for *k*=2 (Figure 9), we can verify many differences between approach I, that achieved the worst value of silhouette width criteria, and approach II, that achieved the best values for both internal criteria. While approach I strongly took into account low levels of historical yield in order to identify an isolated area at the top left region (green), merging clusters representing green and red classes for *k*=3, the approach II was affected again by the low frequency of average altitude values, clearly separating a low (green) from a high elevation region (red).

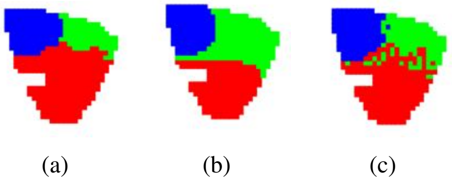


Figure 8. Results for *k*=3: (a) HACC-Spatial (I); (b) FCM (II); and (c) HACC/FCM Ens.(III).

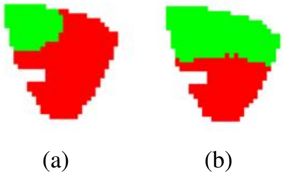


Figure 9. Results for *k*=2: (a) HACC-Spatial (I); e (b) FCM (II).

5. Conclusions and Future Work

If we take into account visual analysis and measures of cohesion and separation provided by SD criteria, approaches purely based on FCM (II e V) achieved, in general, better results in comparison to the approaches using the HACC-Spatial (I, III e IV) . Due to the fact that FCM is based in *k-means* algorithm, its bias is always performed to achieve the minimization of intracluster variance and maximization of intercluster dissimilarity. Because of this, internal criteria based on these measures, like SD and silhouette width, tends to provide suitable results for clusterings obtained by this algorithm. On the other hand, in some cases the visual perception of the expert user, of major importance in PA tasks, may be harmed. However, for the silhouette width criteria, these approaches achieved, in general, worst results in relation to those obtained by approach I, except for *k*=2. These results were likely influenced by intrinsic FCM fuzzy features, which can generate doubts if a sample was properly associated with a particular cluster or whether it will be better allocated to the nearest neighbor cluster.

Regarding to the use of ensembles, splitting of features (approaches IV and V) was important for clarifying some details that can get unnoticed in clusterings obtained using all features. However, the high stratification rates generated in the final maps can be very harmful to the users analysis. In the consensus approach between different kinds of algorithms (III), we can observe an increased stratification, causing damage to the visual user analysis. On the other hand, were observed slight variations in SD and silhouette width criteria for different values of *k*, indicating that this approach can be used as solution in some specific cases.

The ensembles approach used in this work is rather general and try to find consensus using only final clusterings obtained from splitting of features or from different algorithms. In an future work, could be used ensembles approaches that allow extracting the main features of each algorithm, making useful data like the membership values provided by FCM, might be used to obtain a better consensus solution.

6. Acknowledgement

Thanks to Fazenda Aparecida for the availability of the area for data collection that allowed the achievement of this study, and to researchers from Embrapa, Celia Grego and Luiz Vicente, responsible for collecting such data. We also thank the following Brazilian research agencies: CNPq, CAPES, FAPESP and FINEP. The second author have been supported by the grant 311868/2015-0 from CNPq.

References

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185.

Benesty, J., Chen, J., Huang, Y., and Cohen, I. (2009). Pearson Correlation Coefficient. In *Noise Reduction in Speech Processing SE - 5*, volume 2 of *Springer Topics in Signal Processing*, pages 1–4. Springer Berlin Heidelberg.

Bezdek, J. C., Ehrlich, R., and Full, W. (1984). FCM : The Fuzzy c-Means Clustering Algorithm. *Computer & Geosciences*, 10(2-3):191–203.

Brock, A., Brouder, S. M., Blumhoff, G., and Hofmann, B. S. (2005). Defining Yield-Based Management Zones for Corn-Soybean Rotations. *Agronomy Journal*, 97(4):1115–1128.

Chang, D., Zhang, J., Zhu, L., Ge, S. H., Li, P. Y., and Liu, G. S. (2014). Delineation of management zones using an active canopy sensor for a tobacco field. *Computers and Electronics in Agriculture*, 109:172–178.

Córdoba, M., Bruno, C., Costa, J., and Balzarini, M. (2013). Subfield management class delineation using cluster analysis from spatial principal components of soil variables. *Computers and Electronics in Agriculture*, 97:6–14.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). The KDD Process for Extracting Useful Knowledge from Volumes of Data. *Commun. ACM*, 39(11):27–34.

Ghosh, J. and Acharya, A. (2011). Cluster ensembles. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(4):305–315.

Halkidi, M. and Vazirgiannis, M. (2001). Clustering validity assessment: finding the optimal partitioning of a data set. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 187–194.

Halkidi, M., Vazirgiannis, M., and Batistakis, Y. (2000). Quality scheme assessment in the clustering process. In Zighed, D., Komorowski, J., and Zytkow, J., editors, *Principles of Data Mining and Knowledge Discovery*, volume 1910 of *Lecture Notes in Computer Science*, pages 265–276. Springer Berlin Heidelberg.

Han, E.-H., Karypis, G., Kumar, V., and Mobasher, B. (1997). Clustering based on association rule hypergraphs. In *DMKD*, page 0.

Karypis, G. and Kumar, V. (1998). Multilevelk-way partitioning scheme for irregular graphs. *Journal of Parallel and Distributed computing*, 48(1):96–129.

Kitchen, N., Sudduth, K., Myers, D., Drummond, S., and Hong, S. (2005). Delineating productivity zones on claypan soil fields using apparent soil electrical conductivity. *Computers and Electronics in Agriculture*, 46(1-3):285–308.

Li, Y., Shi, Z., Li, F., and Li, H.-Y. (2007). Delineation of site-specific management zones using fuzzy clustering analysis in a coastal saline land. *Computers and Electronics in Agriculture*, 56(2):174–186.

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Matheron, G. (1963). Principles of geostatistics. *Economic geology*, 58(8):1246–1266.

Matheron, G. (1969). Le krigeage universel.

Milne, A. E., Webster, R., Ginsburg, D., and Kindred, D. (2012). Spatial multivariate classification of an arable field into compact management zones based on past crop yields. *Computers and Electronics in Agriculture*, 80:17–30.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York; London.

Molin, J. P. (2003). Agricultura de Precisão: Situação atual e perspectivas. In Fancelli, A. L. and Neto, D. D., editors, *Milho: Estratégias de Manejo para Alta Produtividade*, pages 89–98. ESALQ/USP/LPV, Piracicaba.

Molin, J. P., do Amaral, L. R., and Colaço, A. (2015). *Agricultura de precisão*. Oficina de Textos.

Morari, F., Castrignanò, a., and Pagliarin, C. (2009). Application of multivariate geostatistics in delineating management zones within a gravelly vineyard using geo-electrical sensors. *Computers and Electronics in Agriculture*, 68(1):97–107.

Peralta, N. R., Costa, J. L., Balzarini, M., Castro Franco, M., C??rdoba, M., and Bullock, D. (2015). Delineation of management zones to improve nitrogen management of wheat. *Computers and Electronics in Agriculture*, 110:103–113.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20(0):53–65.

Ruß G. and Kruse, R. (2011). Exploratory hierarchical clustering for management zone delineation in precision agriculture. In Perner, P., editor, *Advances in Data Mining. Applications and Theoretical Aspects*, volume 6870 of *Lecture Notes in Computer Science*, pages 161–173. Springer Berlin Heidelberg.

Schwalbert, R. A., Amado, T. J. C., Gebert, F. H., Santi, A. L., and Tabaldi, F. (2014). Zonas de manejo: atributos de solo e planta visando a sua delimitação e aplicações na agricultura de precisão. *Revista Plantio Direto*, pages 21–32.

Scudiero, E., Teatini, P., Corwin, D. L., Deiana, R., Berti, A., and Morari, F. (2013). Delineation of site-specific management units in a saline region at the Venice Lagoon margin, Italy, using soil reflectance and apparent electrical conductivity. *Computers and Electronics in Agriculture*, 99:54–64.

Sokal, R. R. (1958). A statistical method for evaluating systematic relationships. *Univ Kans Sci Bull*, 38:1409–1438.

Song, X., Wang, J., Huang, W., Liu, L., Yan, G., and Pu, R. (2009). The delineation of agricultural management zones with high resolution remotely sensed data. *Precision Agriculture*, 10(6):471–487.

Strehl, A. and Ghosh, J. (2002). Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions. *Journal of Machine Learning Research*, 3:583–617.

Taylor, J. A., McBratney, A. B., and Whelan, B. M. (2007). Establishing Management Classes for Broadacre Agricultural Production. *Agronomy Journal*, 99(5):1366–1376.

Vendramin, L., Campello, R. J. G. B., and Hruschka, E. R. (2010). Relative clustering validity criteria: A comparative overview. *Statistical Analysis and Data Mining*, 3(4):209–235.

Vendrusculo, L. G. and Kaleita, A. L. (2011). Modeling zone management in precision agriculture through Fuzzy C-Means technique at spatial database. In *2011 Louisville, Kentucky, August 7 - August 10, 2011*, St. Joseph, MI. American Society of Agricultural and Biological Engineers.

Vieira, S. R. (2000). Geoestatistica em Estudos de Variabilidade Espacial do Solo. In Novais, R. F. and Alvarez, V H, S. G. R., editors, *Tópicos em ciência do solo*, pages 1–54. Sociedade Brasileira de Ciência do Solo, Viçosa, MG, 1 edition.

Weiss, S. M. and Indurkhya, N. (1998). *Predictive Data Mining: A Practical Guide*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Zhang, X., Shi, L., Jia, X., Seielstad, G., and Helgason, C. (2010). Zone mapping application for precision-farming: a decision support tool for variable rate application. *Precision Agriculture*, 11(2):103–114.

Web Services for Big Earth Observation Data

Lubia Vinhas[1], **Gilberto Ribeiro de Queiroz**[1],
Karine Reis Ferreira[1], **Gilberto Camara**[1]

[1]Instituto Nacional de Pesquisas Espaciais (INPE)
Av dos Astronautas, 1758, 12227-010, São José dos Campos, SP, Brazil

{lubia.vinhas, gilberto.queiroz}@inpe.br
{karine.ferreira, gilberto.camara}@inpe.br

Abstract. *The aim of geospatial web services is to enable users to share and process geospatial data on the Web, no matter what platform or protocol is used. In this paper, we investigate what are the design decisions needed to implement web services for big Earth observation (EO) data. The focus of the work is discussing what is unique about big EO data and why current standards are unsuitable for big EO data analytics. Instead, simpler APIs can be more effective for accessing big EO data than generic services such as WMS and WCS, specially for data analytics. We support this viewpoint by describing the WTSS Web Time Series Service that offers time series of remote sensing data using a simple API and is suited for big data analytics.*

1. Introduction

Earth observation (EO) satellites produce vast amounts of geospatial data. As an example, the archive of medium-resolution Landsat satellite at the US Geological Survey holds more than five million images of the Earth's surface, with about 1 PB of data. Most space agencies have adopted an open data policy, making unprecedented amounts of satellite data available for research and operational use. This data deluge has brought about a major challenge for Geoinformatics research: *How to design and build technologies that allow the EO community to use big data sets?*

EO satellites provide a continuous and consistent set of information about the Earth's land and oceans. Using big EO data sets, we can detect long-term changes and trends in the environment, and measure the impacts of climate change, urban and ocean pollution and land expansion for food production. To transform data in information, we need to analyse terabytes of data, with different spectral, temporal, and spatial resolutions, acquired by different sensors and stored at different places. Given the broad application areas of EO data, researchers need to consider how to store and organise them in a way that will facilitate data interoperability and data analytics.

The issue of handing big geospatial data has attracted much attention in recent literature. Vitolo et al. (2015) reviewed the web technologies dealing with big environmental data, concluding that domain-specific projects often require tailored solutions. Li et al. (2016) revisited the existing geospatial data handling methods and theories to determine if they are still capable of handling emerging geospatial big data. They concluded that traditional data handling approaches and methods are inadequate and new developments are needed. However, their conclusions are focused on the processing of discrete geographical data, as data captured by mobile devices and GPS based sensors. Amirian et al. (2014)

evaluated typical and modern database systems used for the management of big geospatial data, concluding that there is no single solution that meets all the user needs, therefore geospatial data handling is an open issue. Guo et al. (2014) presented an on-demand computing schema for remote sensing images based on a processing chain model that runs on a private cloud computing platform. However, their motivation is on the processing of timestamps of imagery, not fully exploiting the processing of time series of data.

There also have been recent technological developments for handling big geospatial data. Alternatives include MapReduce-based solutions such as Google Earth Engine (Gorelick, 2012), distributed multidimensional array databases such as SciDB (Stonebraker et al., 2013) and object-relational DBMS extensions such as RasDaMan (Baumann et al., 1998). Unlike the current state of object-relational databases for geospatial data, which have been standardised, each of these solutions is unique and incompatible with the others.

Given the incompatibility of the available architectures for big geospatial data, the question that naturally arises is whether we can still enjoy the benefits of interoperability provided by web services. Geospatial web services are important for interoperability, integration with applications and data sharing. In other words, *is it possible and viable to extend the current generation of geospatial web services to work with big geospatial data*?

To answer this question, we need to consider the standards proposed by the Open Geospatial Consortium (OGC). The OGC has played a crucial role in geospatial data interoperability by proposing web services standards for visualising, disseminating and processing geospatial data. These include the Web Map Service (WMS), Web Coverage Service (WCS), Sensor Observation Service (SOS) and Web Processing Service (WPS) standards. Some space agencies have adopted some OGC web services to deliver their EO products. However, OGC standards have not yet proved to be a consensus solution when it comes to process the big EO data sets available in the agencies repositories, to extract information in a timely and robust way. In this paper, we investigate if the implicit decisions taken by OGC when designing services such as WMS, WCS and WFS allow them to be used for big EO data. We also consider whether new kinds of web services are required to handle big EO data efficiently.

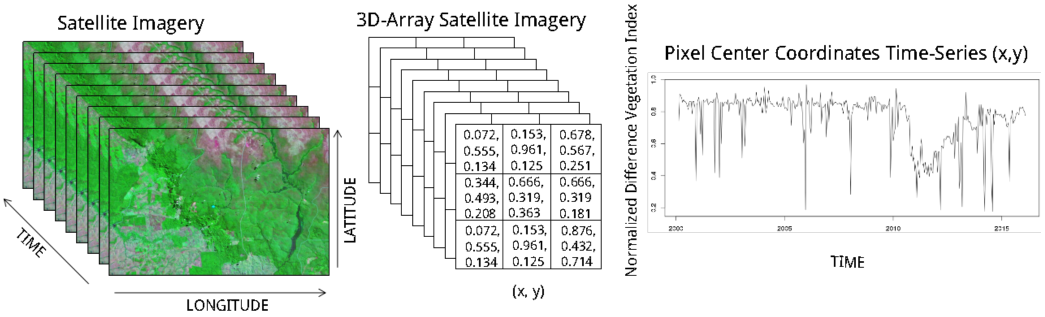
In what follows we will argue that OGC services such as WMS, WFS and WMS rely implicitly on a backend that stores EO data as as a stack of 2D images. We consider that their design is inadequate for scientists working with large sets of remote sensing time series. We also describe a new proposal for a Web Time Series Services (WTSS), which is better suited for accessing large sets of EO time series. We illustrate how WTSS works by presenting a case study. We conclude by looking ahead at the improvements required in WMS and WCS for efficient support for big EO data retrieval and processing.

2. Why current Web Coverage Services are not fit for big data

When designing an architecture for big EO data, one needs to consider the needs of the users. Researchers typically use web services such as WCS and WFS as data sources; they access them by an application, usually a desktop GIS. The typical access query is to retrieve a vector or raster 2D coverage. After retrieving the coverage, users will then apply spatial analysis or image processing algorithm to the data. This *modus operandi* still follows the traditional cartographic abstractions of maps, where each coverage is processed separately.

This working model breaks down when one wants to get access to a large EO data set. For example, the MODIS vegetation index archive for Brazil from 2002 to 2014 has 12.000 independent files. Using WCS to retrieve such a data set for analysis, one has to retrieve each file, run the desired algorithm and then move to the next file. Obviously, this way of working is not adequate. Researchers need to address the set of data as whole to run their algorithms.

To better understand the requirements of big EO data analytics, consider a conceptual view of the problem. Earth observation satellites revisit the same place at regular intervals. Thus measures can be calibrated so that observations of the same place in different times are comparable. These calibrated observations can be organised in regular intervals, so that each measure from sensor is mapped into a three dimensional multivariate array in space-time (see Figure 1). Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of remote sensing images which shows the same region at *n* consecutive times $T = \{t_1, t_2, \ldots, t_n\}$. Each location *[x, y, t]* of a pixel in an image (latitude, longitude, time) maps to a *[i, j, k]* position in a 3D array. Each array position *[i, j, k]* will be associated to a set of attributes values $A = \{a_1, a_2, \ldots, a_m\}$ which are the sensor measurements at each location in space-time (see Figure 1). For optical sensors, these observations are proportional to Earth's reflexion of the incoming solar radiation at different wavelengths of the electromagnetic spectrum. Therefore, a 3D array is a better appropriate conceptual model for big EO data than the 2D map metaphor used in GIS and in most OGC web services.


Satellite Imagery
TIME
LONGITUDE
LATITUDE
3D-Array Satellite Imagery
(x, y)
Pixel Center Coordinates Time-Series (x,y)
Normalized Difference Vegetation Index
1.0
0.8
0.6
0.4
0.2
2000
2005
2010
2015
TIME

0.072, 0.555, 0.134	0.153, 0.961, 0.125	0.678, 0.567, 0.251
0.344, 0.493, 0.208	0.666, 0.319, 0.363	0.666, 0.319 0.181
0.072, 0.555, 0.134	0.153, 0.961, 0.125	0.876, 0.432, 0.714

Figure 1. A Normalized Difference Vegetation Index (NDVI) time series.

Using the 3D array metaphor, scientists can approach the classification problem in different ways. They can first classify each image separately and then compare the resulting classifications in time (the *space-first, time-later* approach). Alternatively, they can first classify each time series and then join the results in space (the *time-first, space-later* approach). Combinations of these approaches are also possible. To enable researchers to develop innovative analytical methods that make good use of big EO data, the system architecture needs to support both approaches. The main question that arises is whether a standard such as WCS supports the 3D array metaphor.

In principle, the concept of WCS could apply both to 2D and 3D arrays. In practice, the current WCS 3D coverage specification is not fully supported by the implementations available, that serves only 2D coverages. The OGC WCS can easily handle multivariate coverages, but it is not easy to specify a multivariate, multitemporal coverage in WCS. Furthermore, coverages are still retrieved one by one, thus creating a stack of 2D maps on the client machine. In this way, WCS does not free the user from the burden of dealing with large numbers of individual files. Instead of dealing with a big EO data set as a sequence of maps, as is implicit in WFS and WCS, researchers need new tools. In this perspective, good geospatial web services for big data should be designed to serve the needs of information extraction algorithms. These methods will in general be different of those available in a typical GIS.

3. Web Time Series Service

In the previous sections, we argued that the current generation of OGC standards has been designed to match the cartographic metaphors used by today's GIS. We also argued that the 3D array metaphor is better suited for big EO data than the map metaphor. We now consider how to support an important kind of algorithms that extract information from remote sensing time series.

To get a remote sensing time series, one first organises a large set of EO data as a 3D array. From each pixel location in the array, one can extract a time series of one or more variables for a temporal interval. The benefits of remote sensing time series analysis arise when the temporal resolution of the big data set is sufficient to capture the most important changes. In this case, the temporal autocorrelation of the data can be stronger than the spatial autocorrelation. In other words, given data with adequate repeatability, a pixel will be more related to its temporal neighbours rather its spatial ones. In this case, *time-first, space-later* methods will give better results than the *space-first, time-later* approach.

Recent results in the literature show that analysis of satellite image times series enables extracting long-term trends of land change (Olsson et al., 2005; Adami et al., 2012; Sakamoto et al., 2005). Such information which would be harder to obtain by processing 2D images separately. These analysis are supported by algorithms such as TWDTW (Maus et al., 2016), TIMESTAT (Jönsson and Eklundh, 2004) and BFAST (Verbesselt et al., 2010). These algorithms process individual time-series and combine the results for selected periods to generate classified maps.

Motivated by the need to retrieve satellite image time series from large 3D arrays, we have designed and implemented the Web Time Series Service (WTSS). A WTSS server takes as input a 3D array. Each array has a spatial and a temporal dimension and can be multidimensional in terms of its attributes. For example, the dataset showed in Figure 1 represents a time series of the NDVI index extracted from a large 3D MODIS array.

The WTSS service is independent of the actual data architecture used for 3D array store. It can work with solutions such as flat files, MapReduce distributed datasets, array databases or object-relational databases. We have implemented the service using both a set of flat files and the SciDB array database management system (Stonebraker et al., 2013), with the same external interface. The WTSS interface provides three operations:

1. *list_coverages*: this operation allows clients to retrieve the capabilities provided by any server that implements WTSS. Or simply put, it returns a list of coverage names available in a server instance. The server response is a JSON[1] document. The names returned by this operation can be used in subsequent operations.

```
// request:
www.dpi.inpe.br/wtss/list_coverages

// response
 "coverages": [ "mod09q1", "mod13q1"]
```

2. *describe_coverage*: this operation returns the metadata for a given coverage identified by its name. It includes its range in the spatial and temporal dimensions. It also receives a JSON document as a response.

```
// request:
www.dpi.inpe.br/wtss/describe_coverage?name=mod09q1

// response:
{"name": "mod09q1",
  "description": "Surface Reflectance 8-Day L3 Global 250m",
  "detail": "https://lpdaac.usgs.gov/dataset_discovery/
             modis/modis_products_table/mod09q1",
  "dimensions":
  [ {"name": "col_id", "description": "column",
      "min_idx": 0, "max_idx": 172799, "pos": 0 },
    {"name": "row_id", "description": "row",
     "min_idx": 0, "max_idx": 86399, "pos": 1},
    {"name": "time_id", "description": "time",
     "min_idx": 0, "max_idx": 1024, "pos": 2  } ],
  "attributes":
  [ {"name": "red",
     "description": "250m Surface Reflectance Band 1",
     "datatype": "16-bit signed integer",
     "valid_range": {"min": -100, "max": 16000},
     "scale_factor": 0.0001, "missing_value": -28672} ]
    "geo_extent": {
      "spatial": {
        "extent": {"xmin": -180.0, "ymin": -90.0,
                   "xmax":  180.0, "ymax":   90.0},
        "resolution": {"x": 0.0020833333, "y": 0.0020833333},
        "srid": 4326},
      "temporal": {
        "interval": {"start": "2000-02-18", "end": "2014-08-13"},
        "resolution": 8,
        "unit": "day"}
```

[1] JavaScript Object Notation (JSON) is a lightweight data-interchange format (see json.org).

3. *time_series*: this operation requests the time series of values of a coverage attribute at a given location.

```
// request:
http://www.dpi.inpe.br/wtss/time_series?coverage=mod0q1&
    attributes=red&latitude=-12&longitude=-54&
    start=2000-02-18&end=2000-03-21

// response:
  "result": {
    "attributes": [
      {
        "attribute": "red",
        "values": [ 3726, 2834, 4886, 231, 1264 ]},
     ],
    "timeline": [ "2000-02-18", "2000-02-26",
                  "2000-03-05", "2000-03-13", "2000-03-21" ],
    "center_coordinates": {
      "latitude": -4.9989583328814176,
      "longitude": -54.000193143463676 } },
  "query": {
    "coverage": "mod09q1",
    "attributes": [ "red"],
    "latitude": -5,
    "longitude": -54}
```

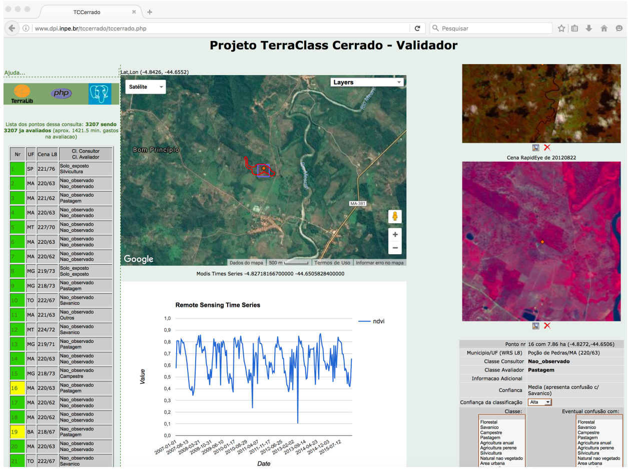
WTSS saves time when dealing with huge volumes of data because it closes the gap between big EO data and applications in a flexible and simple API. Syntactically, WTSS is different from OGC standards, since it uses the JSON format instead of XML to deliver complex responses. Writing lightweight web applications for data visualisation and retrieval is simpler with JSON. Its more cohesive and short notation helps users of the API to understand how the service works and simplifies decoding by client applications. OGC is considering using JSON based formats, as shown by the draft implementation of the OM-JSON service for Observation and Measurement Data (OGC, 2015a). This shows that OGC recognises the benefits of the JSON format for lightweight services.

WTSS time series response is also easy to be consumed by statistical computing languages such as **R**. We have developed an **R** package that implements the WTSS client to allows researchers to use **R** methods such as TWDTW (Maus et al., 2016) and BFAST (Verbesselt et al., 2010) to analyse large sets of satellite image time series.

4. Using the WTSS service

As an example of using the WTSS, consider an application that aims at validating thematic maps created from remote sensing imagery. Jensen (2009) points out that there are different types of errors introduced during the mapping process and stress the need for tools to assess the map accuracy, increasing the usefulness and credibility of the data. In a typical web-based validation tool, experts have to analyse different data, such as images with different spatial resolutions, and also the temporal dynamics at the locations represented by the EO time-series values.

Figure 2 shows a web-based validation tool, that includes a graphical display of MODIS/NDVI series. The user interacts with the application by clicking on the map, this will generate a location that will be used in a request to a WTSS server, using a JavaScript WTSS client (see Listing 1).



Projeto TerraClass Cerrado - Validador

Lista dos pontos dessa consulta: **3207 sendo 3207 ja avaliados** (aprox. 1421.5 min. gastos na avaliacao)

Nr	UF	Cena L8	Cl. Consultor Cl. Avaliador
1	SP	221/76	Solo_exposto Silvicultura
2	MA	220/63	Nao_observado Nao_observado
3	MA	221/62	Nao_observado Pastagem
4	MA	220/63	Nao_observado Nao_observado
5	MT	227/70	Nao_observado Nao_observado
6	MA	220/63	Nao_observado Nao_observado
7	MA	220/62	Nao_observado Nao_observado
8	MG	219/73	Solo_exposto Solo_exposto
9	MG	218/73	Nao_observado Pastagem
10	TO	222/67	Nao_observado Savanico
11	MA	221/63	Nao_observado Outros
12	MT	224/72	Nao_observado Savanico
13	MG	219/71	Nao_observado Pastagem
14	MA	220/63	Nao_observado Nao_observado
15	MG	218/73	Nao_observado Campestre
16	MA	220/63	Nao_observado Pastagem
17	MA	220/62	Nao_observado Nao_observado
18	MA	220/62	Nao_observado Nao_observado
19	BA	218/67	Nao_observado Pastagem
20	MA	220/63	Nao_observado Nao_observado
21	TO	222/67	Nao_observado Savanico

Modis Times Series -4.82718166700000 -44.6505828400000

Ponto nr	16 com 7.86 ha (-4.8272,-44.6506)
Municipio/UF (WRS L8)	Poção de Pedras/MA (220/63)
Classe Consultor	**Nao_observado**
Classe Avaliador	**Pastagem**
Informacao Adicional	
Confianca	Media (apresenta confusão c/ Savanico)
Confiança da classificação	Alta
Classe: Florestal Savanico Campestre Pastagem Agricultura anual Agricultura perene Silvicultura Natural nao vegetado Area urbana	Eventual confusão com: Florestal Savanico Campestre Pastagem Agricultura anual Agricultura perene Silvicultura Natural nao vegetado Area urbana

Figure 2. The TerraClass cerrado validating tool.

```
// connect to the WTSS server
var wtss_server = new wtss(’http://www.dpi.inpe.br/wtss’);

// for a given point (lat,long) return the series
wtss_server.time_series(
    {
        "coverage": "mod09q1",
        "attributes": ["red", "nir"],
        "longitude": long,
        "latitude": lat,
        "start"="2004-01-01",
        "end"="2004-05-01"
    }, fill_time_series);
```

Listing 1: The WTSS JavaScript client embbeded in the validation tool.

Figure 3 shows another web-based application that uses WTSS. In this application, again selecting a location with a click on the map, a user obtain graphical display of a MODIS/NDVI time series. But it also obtains the result of the algorithm TWDTW method for land use and land cover mapping (Maus et al., 2016) using a sequence of multi-band satellite images. This tool allows scientists to experiment the method in a single location before applying it to a given area. A **R** implementation of the TWDTW is available at `https://github.com/vwmaus/dtwSat`. Listing 2 shows a **R** code that consumes the MODIS/NDVI time series otained from a WTSS server, using a WTSS **R**-client.

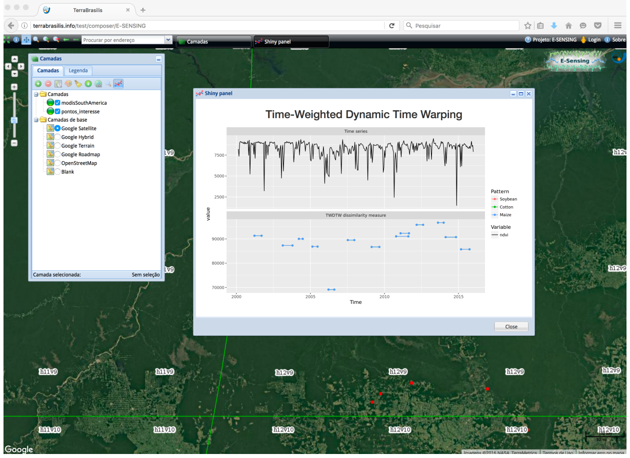


Figure 3. A web-based application with TWDTW algorithm.

```r
# installing and loading packages
library(dtwSat)
library(wtss.R)

# connect to the WTSS server and retrieve the time series
server = WTSS("http://www.dpi.inpe.br/tws/wtss")
coverages = listCoverages(server)
cv = describeCoverage(server, coverages[2])
attr <- c("ndvi", "evi",  "red",  "nir",  "blue", "mir" )

# get a time series
spatio_temporal = timeSeries(server, names(cv),
                  attributes=attr,
                  latitude=-10.408, longitude=-53.495,
                  start="2000-02-18", end="2016-01-01")

ts = twdtwTimeSeries(spatio_temporal[[
    names(spatio_temporal)]]$attributes)
plot(ts, type="timeseries")

patt = twdtwTimeSeries(patterns.list)

log_fun = logisticWeight(alpha=-0.1, beta=100)
matches = twdtwApply(x=ts, y=patt, weight.fun=log_fun, keep=TRUE)

plot(x = matches, type = "alignments")
```

Listing 2: The WTSS R client called in **R** implementation of the TWDTW algorithm.

5. Future Directions on Web Services for Big Data

In the previous sections, we have argued that the current design of WCS and WMS limits the use of these standards for dealing with big EO data sets. We have also presented WTSS, an example of a service designed to work with big data. In this section, we examine some future directions for web services suited for big EO data.

Given the burden of transferring large data sets over the internet, web services for big data should be concerned with performing server-side processing, rather than supporting client-side retrieval. Instead of downloading large data files, such web services would enable algorithms to be run on a remote server. This points to an emphasis on web processing services.

There are two OGC web services designed for data processing: the WPS Web Processing Service (OGC, 2015b) and the WCPS Web Coverage Processing Service (Baumann, 2010). WPS standardizes how clients can invoke the execution of a process. Thus it focus on the description of requests and reponses to inputs and outputs. It is not concerned on how to express the processing in the server side. Müller et al. (2010) build an architecture using WPS where client applications can move the code to the server in order to perform operations. However, WPS requires that the web server provides the needed operations, which is not currently the case with big EO data. Researchers are still in the stage of experimenting with new methods. Rather than providing a pre-defined set of algorithms, the server-side processing service should be flexible to allow researchers to run their own methods on big data sets.

The WCPS standard proposes a coverage processing language in an attempt to allow clients to express their computations for evaluation by the server. Although endorsed by OGC, this proposal has not been widely adopted. To the best of our knowledge there is only one implementation supporting it as a component of the RasDaMan project (Baumann et al., 1998). Furthermore, the WCPS standard is based on a sequence of pixel-by-pixel operations over a set of 2D coverages. The basic request structure of WCPS consists of a loop over a list of coverages, an optional filter predicate, and an expression indicating the desired processing of each coverage (Baumann, 2010). This way of working is not flexible enough to include algorithms that work with time-series and for spatio-temporal processing. This is illustrated by the work of Karantzalos et al. (2015), where the WCPS service was only used for data retrieval, and the classification algorithm was run in the client machine using open source libraries.

One of the challenges of designing a new language for coverage processing is the diversity of algorithms and applications already existing and the practices of the research community. Researchers are most productive when working on familiar computing environments. Scientists like to test new ideas on small and well-known data sets. Only after they are satisfied with the experiments, they are willing to work with big data. Therefore, a standard for big Earth observation data analytics should meet important needs of the research community:

1. Analytical scaling: support for the full cycle of research, allowing algorithms developed at the desktop to be run on big databases with minor or no modifications.
2. Software reuse: researchers should be able to easily adapt existing methods for big data, with minimal reworking.

3. Collaborative work: the results should be shareable with the scientific community, and different research teams should be able to build their own infrastructure.

Data scientists are conservative in their choice of tools. They prefer to work on tools with a simple software kernel where they can easily add new packages that encapsulate new analytical methods. A prime example is the **R** suite of statistical tools (R Core Team, 2015). **R** provides a wide variety of statistical and graphical tools, including spatial analysis, time-series analysis, classification, clustering, and data mining for many disciplines (e.g. hydrology, ecology, soil science, agronomy). It is extensible through high quality packages. It provides methods and tools in an open source environment and allows research reproducibility. Using **R** as their primary tool for big data analytics, researchers can thus scale up their methods, reuse previous work, and easily collaborate with their peers. Therefore, we consider that tailor-made, simpler, APIs that support **R** programming can be more effective for processing big EO data than requiring scientists to re-implement their methods on a new coverage processing language.

For future work, we envision a web service for processing big EO data as a simple and standardised interface that can be used with a few commands. It would let users encapsulate their algorithms for server-side processing. Such a service would go beyond what is currently offered by OGC standards such as WPS and WCPS to allow progress on big EO data analytics. Researchers would be able to develop and share new methods, working on a familiar **R** environment. They would be able to test their methods on their desktop before running experiments on big data sets. Such a service that would allow significant progress on big EO data analytics.

6. Conclusions

This paper addressed the problem of how to design and build technologies that allow the EO community to explore the unprecendented amounts of satellite data available for research and operational use. We investigate if the implicit decisions taken by OGC when designing services such as WMS, WCS and WFS allow them to be used for big EO data. We also consider whether new kinds of web services are required to handle big EO data efficiently.

We conclude that the current OGC standards, and their implementations, can not be considered to be the most appropriate to handle big data EO data sets since they are inspired by the traditional cartographic abstractions of maps, where each coverage is processed separately. The most innovative analytical methods that make good use of big EO data also require a conceptual view of ig EO data sets as 3D arrays of data, where each Each array position *[i, j, k]* will be associated to a set of attributes values $A = \{a_1, a_2, \ldots, a_m\}$ which are the sensor measurements at each location in space-time. This conceptual view allows the two approaches for data processing *space-first, time-later* and *time-first, space-later* shown in the paper.

We argue that tailor-made, simpler, APIs can be more effective than generic services such as WMS and WCS for accessing big EO data, specially aimed at data analytics on 3D arrays, where the temporal component is of crucial importance. As an example of such tailor-made service we describe the Web Time Series Service (WTSS). We show how WTSS closes the gap between big EO data and applications consuming time-series of data in a flexible and simple API that uses the JSON format to delivery complex responses.

This is the first step to realise a vision of a web service for processing big EO data as a simple and standardised interface that can be used with a few commands and let users encapsulate their algorithms for server-side processing.

Acknowledgements

This work is supported by FAPESP e-science program (grant 2014-08398-6). Gilberto Camara is also supported by CNPq (grant 3121512014-4).

We thank Ricardo Cartaxo de Souza for his valuable participation in the discussions about WTSS; Alber Sánchez, Luiz Fernando Assis, and Eduardo Llapa Rodríguez for their assistance with the deployment of WTSS server over SciDB and with the WTSS clients evaluation; Andre F. A. Carvalho, Luiz E. P. Maurano and Raul Castanhari for their support with the web-based applications shown in the case studies section.

References

Adami, M., Rudorff, B. F. T., Freitas, R. M., Aguiar, D. A., Sugawara, L. M., and Mello, M. P. (2012). Remote sensing time series to evaluate direct land use change of recent expanded sugarcane crop in brazil. *Sustainability*, 4(4):574–585.

Amirian, P., Basiri, A., and Winstanley, A. (2014). Evaluation of data management systems for geospatial big data. In Murgante, B. and Misra, S., editors, *14th International Conference in Computational Science and Its Applications – ICCSA 2014*, pages 678–690.

Baumann, P. (2010). The ogc web coverage processing service (wcps) standard. *GeoInformatica*, 14(4):447–479.

Baumann, P., Dehmel, A., Furtado, P., Ritsch, R., and Widmann, N. (1998). The multidimensional database system RasDaMan. *ACM SIGMOD Record*, 27(2):575–577.

Gorelick, N. (2012). Google earth engine. In *AGU Fall Meeting Abstracts*, volume 1, page 04.

Guo, W., She, B., and Zhu, X. (2014). Remote sensing image on-demand computing schema for the China ZY-3 satellite private cloud-computing platform. *Transactions in GIS*, 18:53–75.

Jensen, J. R. (2009). *Remote Sensing of the environment: na Earth resource perspective*. Prentice Hall, Chicago.

Jönsson, P. and Eklundh, L. (2004). Timesat–a program for analyzing time-series of satellite sensor data. *Computers & Geosciences*, 30(8):833 – 845.

Karantzalos, K., Bliziotis, D., and Karmas, A. (2015). A scalable geospatial web service for near real-time, high-resolution land cover mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(10):4665–4674.

Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., Pettit, C., Jiang, B., Haworth, J., Stein, A., et al. (2016). Geospatial big data handling theory and methods: A review and research challenges. *ISPRS journal of Photogrammetry and Remote Sensing*, 115:119–133.

Maus, V., Câmara, G., Cartaxo, R., Sanchez, A., Ramos, F. M., and de Queiroz, G. R. (2016). A time-weighted dynamic time warping method for land-use and land-cover mapping. *IEEE journal of Selected Topics in Applied Earth Observations and Remote Sensing*, PP(99):1–11.

Müller, M., Bernard, L., and Brauner, J. (2010). Moving code in spatial data infrastructures – web service based deployment of geoprocessing algorithms. *Transactions in GIS*, 14:101–118.

OGC (2015a). OGC Observations and Measurements – JSON implementation. Technical report, Open Geospatial Consortium.

OGC (2015b). OGC WPS 2.0 Interface Standard. Technical report, Open Geospatial Consortium.

Olsson, L., Eklundh, L., and Ardö, J. (2005). A recent greening of the sahel—trends, patterns and potential causes. *journal of Arid Environments*, 63(3):556–566.

R Core Team (2015). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Sakamoto, T., Yokozawa, M., Toritani, H., Shibayama, M., Ishitsuka, N., and Ohno, H. (2005). A crop phenology detection method using time-series MODIS data. *Remote Sensing of Environment*, 96(3-4):366–374.

Stonebraker, M., Brown, P., Zhang, D., and Becla, J. (2013). Scidb: A database management system for applications with complex analytics. *Computing in Science & Engineering*, 15(3):54–62.

Verbesselt, J., Hyndman, R., Newnham, G., and Culvenor, D. (2010). Detecting trend and seasonal changes in satellite image time series. *Remote Sensing of Environment*, 114(1):106–115.

Vitolo, C., Elkhatib, Y., Reusser, D., Macleod, C. J., and Buytaert, W. (2015). Web technologies for environmental big data. *Environmental Modelling & Software*, 63:185 – 198.

Spatiotemporal Data Representation in R

Lorena A. Santos[1], Karine R. Ferreira[1], Gilberto R. Queiroz [1], Lubia Vinhas[1]

[1]National Institute for Space Research
Av. dos Astronautas, 1758,
12227-010 - São José dos Campos (SP) - Brazil

{lorena.santos,karine.ferreira,gilberto.queiroz,lubia.vinhas}@inpe.br

Abstract. *Recent advances in devices that collect geospatial information have produced massive spatiotemporal data sets. Earth observation and GPS satellites, sensor networks and mobile gadgets are examples of technologies that have created large data sets with better spatial and temporal resolution than ever. This scenario brings a challenge for Geoinformatics: we need software tools to represent, process and analyze these large data sets efficiently. R is a environment widely used for data analysis. In this work, we present a study of spatiotemporal data representation in R. We evaluate R packages to access and create three spatiotemporal data types as different views on the same observation set: time series, trajectories and coverage.*

1. Introduction

Recently, the amount of devices that collect geospatial information has greatly increased. Earth observation and GPS satellites, sensor networks and mobile gadgets are examples of technologies that have created large data sets with better spatial and temporal resolution than ever. This technological advance brings many challenges for Geoinformatics. We need novel software tools to represent, process and analyze big spatiotemporal data sets efficiently.

In Geoinformatics, spatiotemporal data representation is an open issue. Spatial information is represented following well-established models and concepts. This includes the dichotomy between object-based and field-based models [Galton 2004]. Examples of long-standing concepts are vector and raster data structures, topological operators, spatial indexing, and spatial joins [RIGAUX et al. 2002]. Most existing GIS and spatial database systems, such as PostGIS and Oracle Spatial, are grounded on these concepts. However, there is no consensus on how to represent spatiotemporal information in computational systems.

Many existing proposals of spatiotemporal data models focus on representing the evolution of objects and fields over time. Some proposals are specific for discrete changes in objects [Worboys 1994] [Hornsby and Egenhofer 2000], others for moving objects [Guting and Schneider 2005] [ISO 2008] and still others for fields or coverage [Liu et al. 2008] [OGC 2006]. To properly capture changes in the world, representing evolution of objects and fields over time is not enough. We also need to represent events and relationships between events and objects explicitly [Worboys 2005]. Events are occurrents [Galton and Mizoguchi 2009]. They are individual happenings with definite beginnings and ends. The demand for models that describe events has encouraged recent research on spatiotemporal data modeling [Galton and Mizoguchi 2009].

R is a software tool widely used for data analysis [R Development Core Team 2011]. It provides a broad variety of statistical methods (time-series analysis, classification and clustering) and a high-level programming environment and language suitable for fast developing new algorithms. R is extended via packages. Although there are many packages for spatial data handling and analyzing, few of them can properly deal with the temporal dimension of spatial data.

This paper presents a study of spatiotemporal data handling in R. We evaluate R packages for spatiotemporal data access and representation. To guide this evaluation, we consider the spatiotemporal data types proposed by Ferreira et al. (2014). They propose a data model that represents objects and fields that change over time as well as events. Based on this model, we describe in this work how to load and create three spatiotemporal data types in R as different views on the same observation set: *time series*, *trajectory* and *coverage*.

2. An Observation-based Model for Spatiotemporal Data

Ferreira et al. (2014) propose a data model for spatiotemporal data and specify it using an algebraic formalism. Algebras describe data types and their operations in a formal way, independently of programming languages. The proposed algebra is extensible, defining data types as building blocks for other types. It takes observations as basic units for spatiotemporal data representation and allows users to create different views on the same observation set, meeting application needs.

Observations are our means to assess spatiotemporal phenomena in the real world. Recent research draws attention to the importance of using observations as a basis for designing geospatial applications [Kuhn 2009]. The proposed model defines three spatiotemporal data types as abstractions built on *observations*: *time series*, *trajectory* and *coverage*. A *time series* represents the variation of a property over time in a fixed location. A *trajectory* represents how locations or boundaries of an object change over time. A *coverage* represents the variation of a property in a spatial extent at a time. We also define an auxiliary type called *coverage series* that represents a time-ordered set of coverages that have the same boundary. Using these types, we can represent objects and fields that change over time as well as *events*.

2.1. Different Views on the Same Observation Set

Figure 1 shows an example of observations collected by five moving objects. Each observation is represented as a tuple in the form (id, x, y, t, p), where id is the object identification, x and y are spatial locations, t is time and p is a property value collected in the spatial local x and y and in time t. In this example, the property collected is air pollution.

On these observations, we can create different views depending on the kinds of analysis we want to perform on them. Each view is materialized as a data type. Figure 2 illustrates trajectory and coverage instances built on the same observation set shown in Figure 1. For example, to analyze how the objects move over time and space, we create an instance of the *trajectory* data type for each object. Each trajectory instance contains observations of an specific object. To analyze how the air pollution varies in a region, we create instances of the *coverage* data type. Each coverage instance contains observations in a specific period, mixing observations of different objects. To analyze how the air

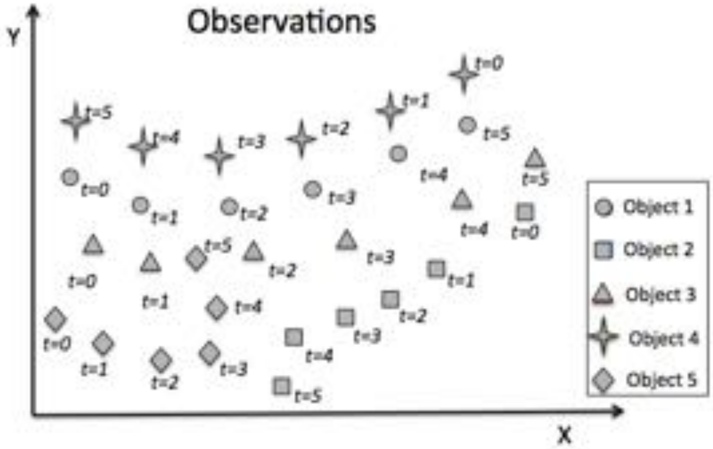


Figura 1. Spatiotemporal observations

pollution varies in a given spatial location over time, we can create an instance of *time series* data type.

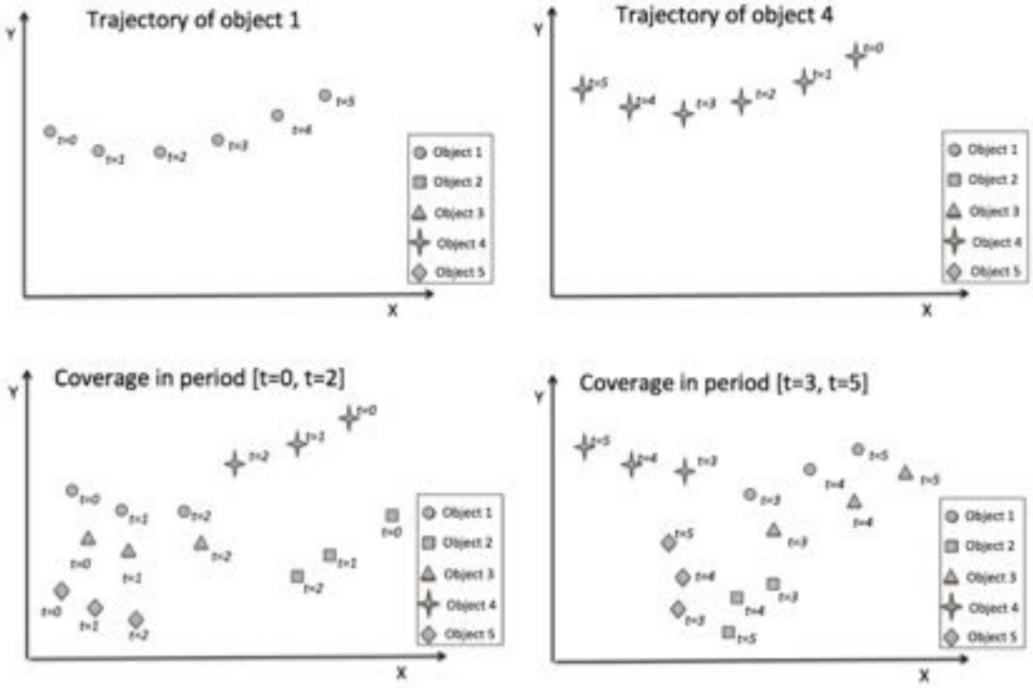


Figura 2. Different data types built on the same observation set

In the proposed model, each spatiotemporal data type instance, *time series*, *trajectory* and *coverage*, has an interpolation function, called interpolator, suitable for it. A time series has an interpolator that estimates property values in non-measured times. A trajectory has an interpolator able to estimate locations in non-observed times. A coverage has an interpolator to estimate property values in non-observed locations.

The capability of creating different views or data types on the same observation set is an important requirement for spatiotemporal data representation. In this work, we evaluate how to do this using R. We use equivalent R data types to represent observation sets and to create trajectory, time series and coverage from these sets.

3. Spatiotemporal Data Representation in R

In this section, we describe a set of R packages for spatiotemporal data representation and access. Packages for data access are `Rgdal` [Bivand et al. 2013a], `Rpostgres` [Conway et al. 2008] and `Rodbc` [Ripley and Lapsley 2016]. Packages with data types that can be used to represent spatiotemporal data are `spacetime` [Pebesma 2012], `xst` [Ryan and Ulrich 2012], `trajectories` [Pebesma and Klus 2015] and `raster` [Hijmans 2016]. Furthermore, we evaluate some R package that provide interpolation functions that are crucial for creating spatiotemporal data type, such as `gstat` [Pebesma and Graeler 2016].

Figure 3 shows a diagram with these packages and the relationships among them.

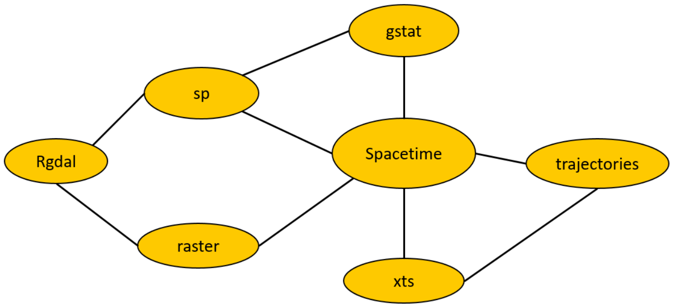


Figura 3. Relationships between packages

Spatiotemporal data can be obtained from different data source, such as, database (e.g. Postgis), data files (e.g. shapefiles and geotif raster files) and web services [Ferreira et al. 2015]. In R, there are packages that can access data from distinct type of source. However, these packages do not work directly with the concept of spatiotemporal data.

The `Rgdal` [Bivand et al. 2013a] package allows a broader range of spatial data sources [Lovelace and Cheshire 2014], such as shapefiles, raster data files and database systems. `RODBC` and `Rpostgres` packages allow to create SQL queries in R for accessing data in database systems, but they do not deal with spatial data.

The `spacetime` package contains a set of base classes for spatiotemporal data representation that are widely used for other R packages for spatiotemporal analyses. It is built upon the classes and methods for spatial data from package `sp` and for time series data from package `xts` [Pebesma 2012]. The `xts` package was chosen due to its support to represent several types of date and time. Moreover, it extends functionality of zoo package, that has good tools for aggregation over time [Zeileis and Grothendieck 2005].

Each spatial data type from sp package has two slots that contain bound box, a matrix of numerical coordinates and other slots that contain a CRS class object defining the coordinate reference system [Bivand et al. 2013b]. The spatial data can be a particular spatial point, line, polygon or set of polygons, or a pixel (grid or raster cell) [Pebesma 2012].

The spacetime package classes are shown in Figure 4. Spatial Full Grid (STF) and Sparse Grid (STS) have the same general layout, with observations on a space time grid. The main difference between both is that STF stores the full grid, all observations of all space time points, while STS only stores non-missing valued observations. Examples that can be represent by these classes are: time sequences of satellite imagery and measuring air quality every hour [Pebesma 2016].

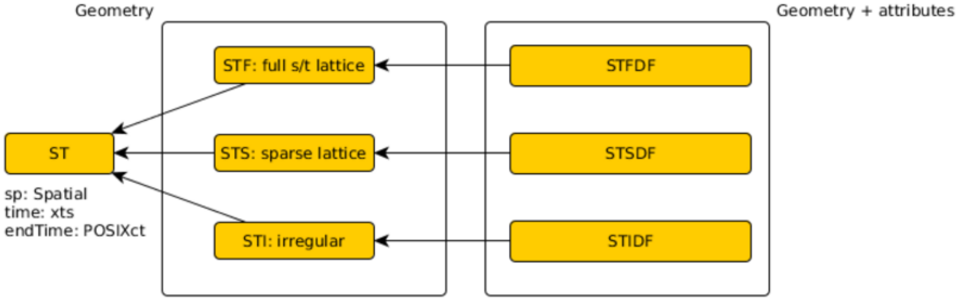


Figura 4. Classes for spatiotemporal data in package spacetime. [Pebesma 2016]

Irregular Grid (STI) represent a layout where for each spatial data a time point is stored. An example that can be represent for STI is measurement from mobile sensors. All classes presented derive from a base class,ST. This class is virtual, which is not represent actual data. The ST class derives two order classes: a spatio-temporal geometries and a augment class with actual data, in the form of data frame.

The objects from xts and sp packages are not used to store property values. For purely temporal information xts is used, and for purely spatial information sp objects is used. Then, it is necessary to use a data.frame to combine, space, time and property values. It represents the data as rectangle of rows containing observations on columns of property values [Bivand et al. 2013b].

The raster data set is composed by multiple layers, hence, the raster package has two classes for work with multi-layer data ,rasterStack and rasterBrick . The principal difference between these classes is that a rasterBrick can only be linked to a single file, while rasterStack can be formed from separate files and/or from few layers from a single file [Hijmans 2016]. Each raster layer in the stack or brick needs to be in the same projection, spatial extent and resolution. In other cases, such as, reading satellite images, we do not use Rgdal package, the raster package can be used directly using the function raster(). To represent spatiotemporal data raster, from a collection raster, we add a time for each layer using the setZ function from raster package, then we can coerce these data to spatiotemporal type from `spacetime` package.

`Trajectories` package provides three data types to represent trajectories, `Track`, `Tracks` and `TracksCollection`, based on the `STIDF` type. The class `Track` represents a single trajectory followed by a person, animal or object. `Tracks` embodies a collection of trajectories followed by a single person, animal or object. The class `TracksCollection` represents a collection of trajectories followed by different persons, animals or objects. Besides that, this package provides a set of operations over trajectories, such as computing of trajectories STBox and calculating distances between two tracks.

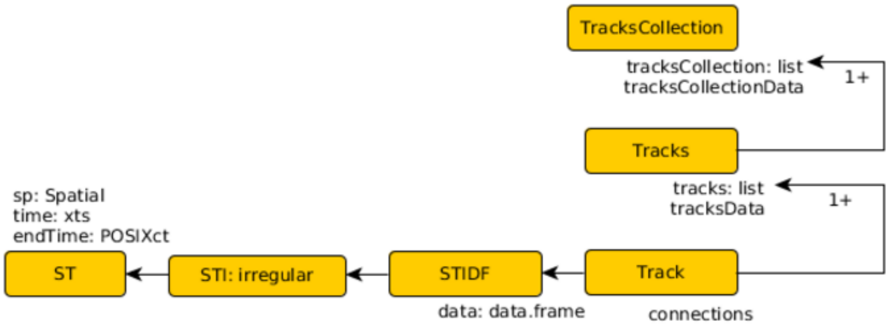


Figura 5. Classes for spatiotemporal data as trajectories. [Pebesma 2016]

The package `gstat` provides spatial and spatiotemporal interpolation functions. Types derived from `sp` and `spacetime` packages can be utilized in this package.

4. Case Study

In this section, we present the use of the R packages listed in the previous section approaching the data types proposed by [Ferreira et al. 2014]. We performed a case study using observations of vessels around the Brazilian coast. These observations are stored in a PostGIS database and contain trajectories of 993 vessels collected during 4 years, from 2008 to 2011. The Figure 6 presents the observations of all vessels.

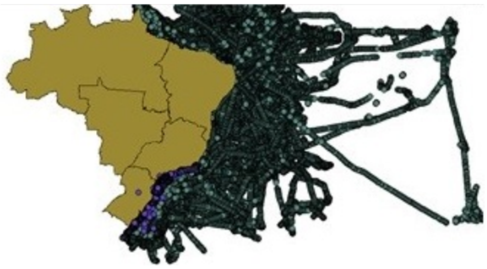


Figura 6. All Observation Set ploted against the map of Brazil

4.1. Observation Set

In this case study, we selected a small subset, containing trajectories of 166 vessels. We filtered data temporally obtaining trajectories only one day and spatially for locations in Rio de Janeiro State. Our subset is shown in Figure 7.

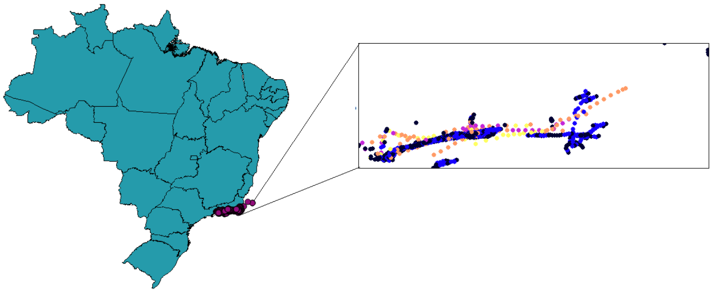
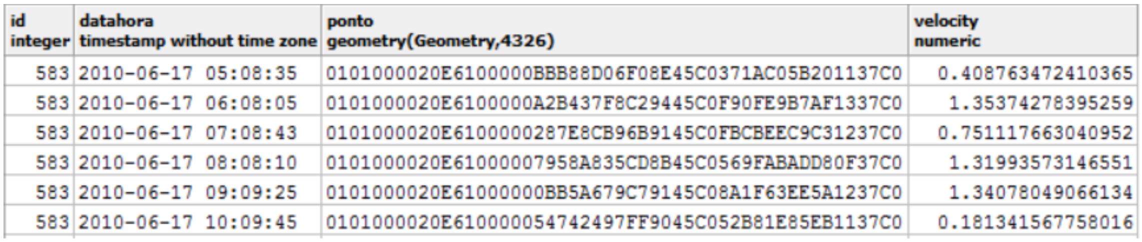


Figura 7. Filtered Data

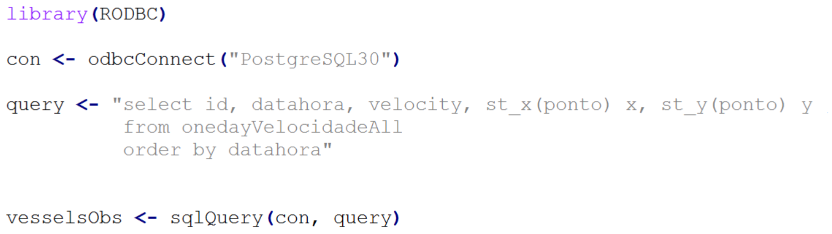
The filtered vessel observations are stored in table of PostGIS database, where each row contains an observation of a vessel. Each observations contains the vessel id (integer type), time (timestamp type), spatial location (geometry type) and velocity of each vessel (numeric type). The table format is shown in Figure 8.



id	datahora	ponto	velocity
integer	timestamp without time zone	geometry(Geometry,4326)	numeric
583	2010-06-17 05:08:35	0101000020E6100000BBB88D06F08E45C0371AC05B201137C0	0.408763472410365
583	2010-06-17 06:08:05	0101000020E6100000A2B437F8C29445C0F90FE9B7AF1337C0	1.35374278395259
583	2010-06-17 07:08:43	0101000020E6100000287E8CB96B9145C0FBCBEEC9C31237C0	0.751117663040952
583	2010-06-17 08:08:10	0101000020E61000007958A835CD8B45C0569FABADD80F37C0	1.31993573146551
583	2010-06-17 09:09:25	0101000020E61000000BB5A679C79145C08A1F63EE5A1237C0	1.34078049066134
583	2010-06-17 10:09:45	0101000020E610000054742497FF9045C052B81E85EB1137C0	0.181341567758016

Figura 8. Observation set

In this case study, we used RODBC to connect in the database and filter the observation data using SQL language inside R environment. This this step is shown in Figure 9.

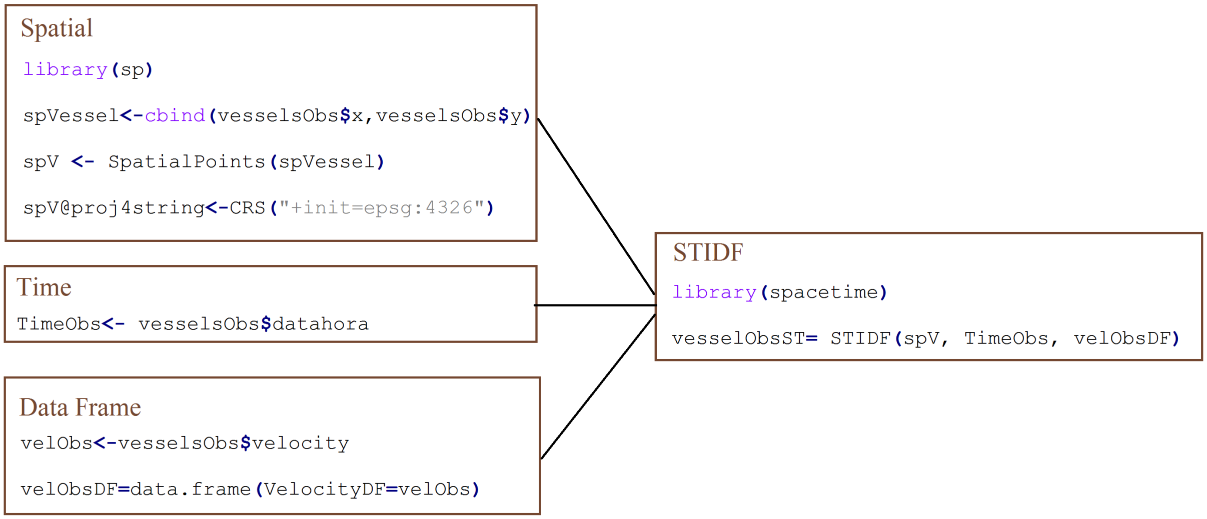


```
library(RODBC)

con <- odbcConnect("PostgreSQL30")

query <- "select id, datahora, velocity, st_x(ponto) x, st_y(ponto) y
         from onedayVelocidadeAll
         order by datahora"


vesselsObs <- sqlQuery(con, query)
```

Figura 9. Acessing data from Postgis

RODBC does not work with spatial data. Therefore, coordinates data is returned as numeric type and so must be converted to spatial type of R. A matrix is created with coordinate values and then we transform this matrix in a SpatialPoint in R format with a coordinate reference system. For temporal data, we do not need realize changes, because the data was returned in POSIXct type. This type is standard way of representing time in R [Wuertz et al. 2015], and also in spacetime package.

Finally we created a `data.frame` with only one column, containing velocity data, named `VelocityDF`. Combining the spatial, temporal and data frame objects we can apply in a spacetime class. In this case we apply `STIDF` , because our observations are irregular data. Figure 10 shows how these data are prepared in R.



Spatial

```
library(sp)

spVessel<-cbind(vesselsObs$x,vesselsObs$y)

spV <- SpatialPoints(spVessel)

spV@proj4string<-CRS("+init=epsg:4326")
```

Time

```
TimeObs<- vesselsObs$datahora
```

Data Frame

```
velObs<-vesselsObs$velocity

velObsDF=data.frame(VelocityDF=velObs)
```

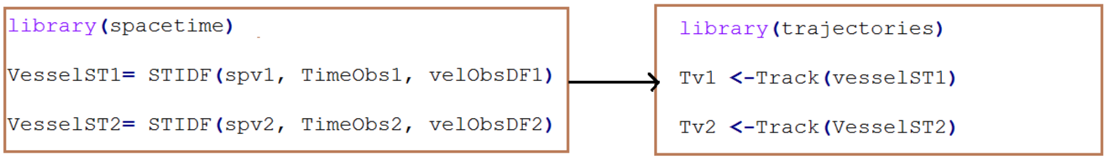
STIDF

```
library(spacetime)

vesselObsST= STIDF(spV, TimeObs, velObsDF)
```

Figura 10. Creating STIDF object

4.2. Trajectories and Time Series

Two vessels from the observation set was selected. We created two objects of type `STIDF`, one for each vessel, `vesselST1` and `vesselST2`. Then, we created two instances of `Track` type from `trajectories` package; each one to represent a trajectory of an object. The R code for this step is shown in Figure. 11 and the trajectories generate are show in Figure 12.



```
library(spacetime)

VesselST1= STIDF(spv1, TimeObs1, velObsDF1)

VesselST2= STIDF(spv2, TimeObs2, velObsDF2)
```

```
library(trajectories)

Tv1 <-Track(vesselST1)

Tv2 <-Track(VesselST2)
```

Figura 11. Creating Trajectories from STIDF

Here, we can also analyze how two vessels are moving together. The `trajectories` has a function to calculate distances between two tracks using the method `compare`. This function returns a object of type `difftrack`. From this object we can obtain the distance between trajectories over time and create a time series using the `xts` type. This time series represents the distance variation between two trajectories over time. The Figure 13 shows the time series and how it is generated.

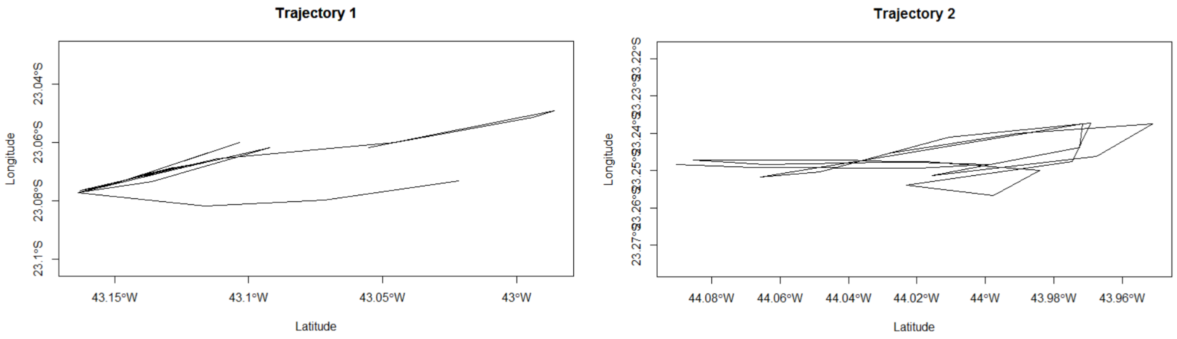


Figura 12. Visualizing spatiotemporal data as Trajectories

```
library(xts)

trajCmp<-compare(Tv1,Tv2)

timeCmp<-trajCmp@conns1$time

dist1<-trajCmp@conns1$dists

distTS<-xts(dist1,timeCmp)
```

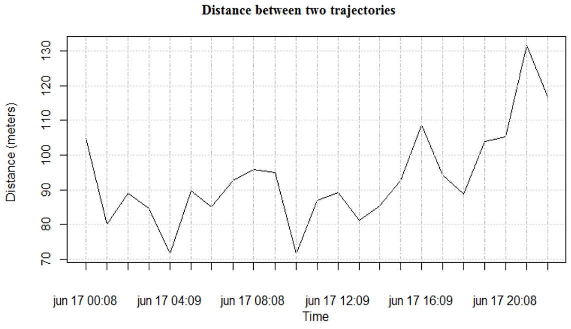


Figura 13. Time Series: distance variation over time

4.3. Coverage

To represent the velocity variation over time and space, we create coverages. We put together observations obtained by vessels during one day and produced a coverage that investigate how velocity varies within the boundary delimited by spatiotemporal bounding box object created previously. This kind of view on the vessel observations allows users to identify possible regions where vessels are fishing. In fishing areas, vessels have low velocity.

Our observations are discrete. They need to be combined with interpolation functions to estimate values in non-observed spatial locations [Ferreira et al. 2014]. We used a spatial interpolation and created a grid where every cell has a velocity value.

To create a grid, we constructed a `GridTopoloy` object using the bounding box of our subset region and a cell size of 0.01° in each direction. From the `GridTopoloy` object, we constructed the `SpatialGrid` object and associated a coordinate reference system (`CRS`) to it. Figure 14 shows the code used to create this grid.

We created a coverage for each hour of a day, using the Inverse Distance Weighted Interpolator (IDW) interpolation function. In this case, each coverage contains the observations of a specific hour, mixing observations of different vessels. The type returned by the IDW interpolator is `SpatialGridDataframe`. This type is typical for representation raster GIS [Bivand et al. 2013b]. Using `Rgdal`, each grid can be saved as Tif format. Figure 15 shows the code to obtain the time interval from `STIDF` object to be spatially interpolated over the grid created earlier. This step was realized 24 times, for each hour.

```
csN <- c(0.01, 0.01)

ccN <- vesselObsST@sp@bbox[,1] + (csN/2)

cdN <- ceiling(diff(t(vesselObsST@sp@bbox))/csN)

gridVessel <- SpatialGrid(GridTopology(cellcentre.offset = ccN,
                                       cellsize = csN, cells.dim = cdN))

proj4string(gridVessel)<-CRS("+init=epsg:4326")
```

Figura 14. Creating a spatiotemporal grid

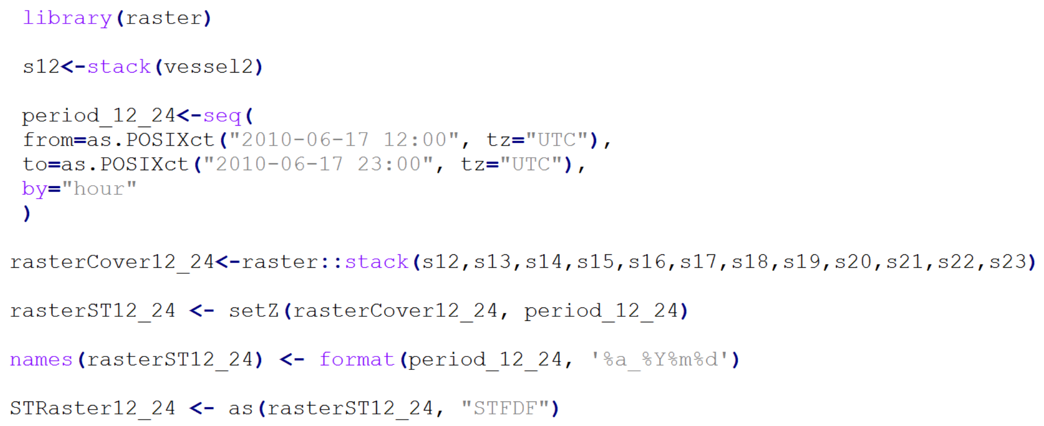
```
t0 <- coredata(naviosST@time['2010-06-17 00:00:00/2010-06-17 01:00:00'] )

vesselt0<- idw(naviosST@data[t0[1]:t0[length(t0)],]~ 1,
               naviosST@sp[t0[1]:t0[length(t0)],], gridNavio, idp = 2.5)
```

Figura 15. Inverse Distance Weighted Interpolator

Once all grids were created, we read the 24 grids using `stack()` function from `raster` package and put these raster grids together using `stack::raster` function. Two stacks are generated, one containing all raster grids during 00:00 until 12:00 and another containing all raster grids during 12:00pm until 23:59pm. Furthermore, each raster pushed in the stack was associated to time interval using `setZ()` function from `raster` package. These raster stacks contain spatial and temporal data. They can be converted to the spatiotemporal data type `STFDF` of the `spacetime` package. The Figure 16 shows the code that describes this step for one period.

The Figure 17 shows the spatiotemporal raster grids to represent coverages in R. These coverages represent how the velocity varies over time within a specific region of the ocean. Looking these coverages, we can visually identify areas where the velocity is low. Such areas can be possible regions where vessels are fishing. Thus, using coverage types built on the vessel observations, we can extract information about the variation of velocity over time and space, using spatiotemporal data mining techniques.

It is important to note that the interval between 12:00 and 15:00, figures 17, contains large regions where the vessel velocity is low. We contrast with red circles regions where velocity is lower than others. Visually, we observed that these regions have low velocity in all time intervals since 12:00 until 23:59 pm.



```
 library(raster)

 s12<-stack(vessel2)

 period_12_24<-seq(
 from=as.POSIXct("2010-06-17 12:00", tz="UTC"),
 to=as.POSIXct("2010-06-17 23:00", tz="UTC"),
 by="hour"
 )

rasterCover12_24<-raster::stack(s12,s13,s14,s15,s16,s17,s18,s19,s20,s21,s22,s23)

rasterST12_24 <- setZ(rasterCover12_24, period_12_24)

names(rasterST12_24) <- format(period_12_24, '%a_%Y%m%d')

STRaster12_24 <- as(rasterST12_24, "STFDF")
```

Figura 16. Creating spatiotemporal raster grids to represent coverages

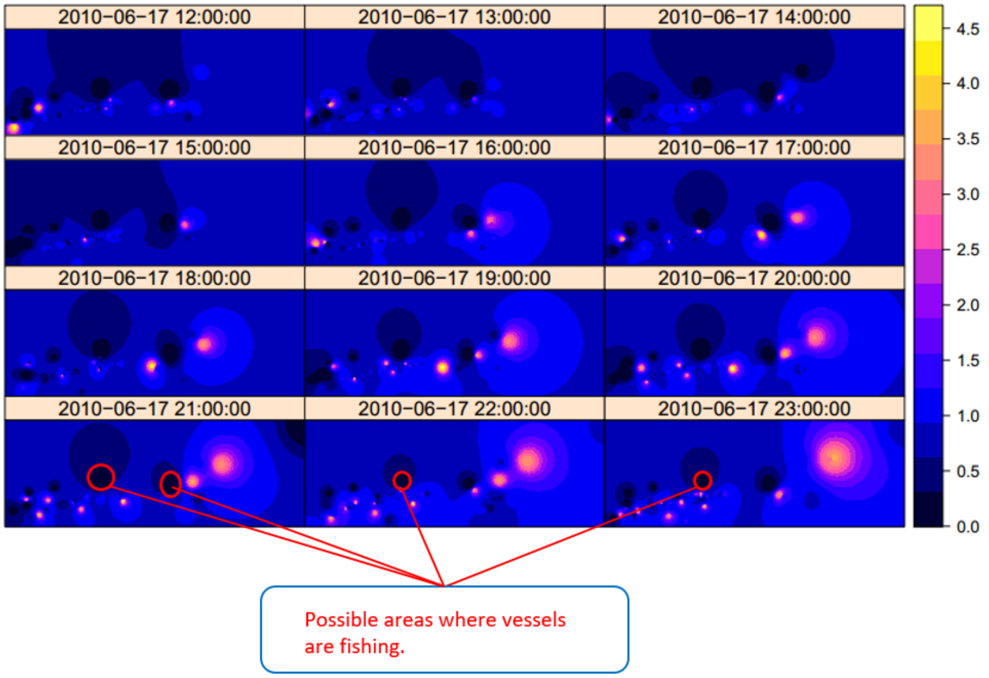


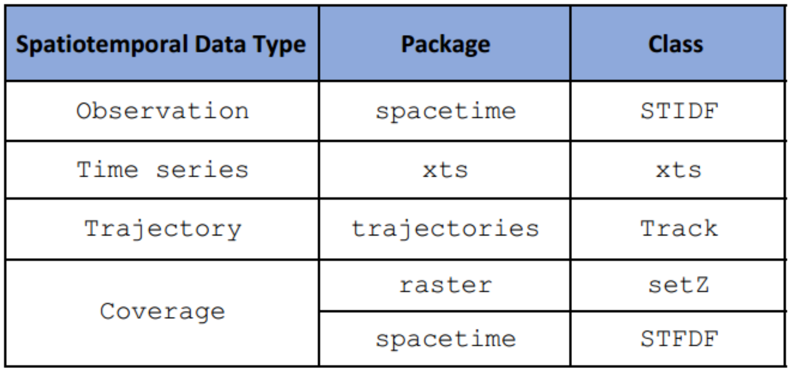
Figura 17. Velocity variation during 12:00 until 23:59

5. Evaluation and Final remarks

Although there are conceptual differences between the classes provided by R and the data types proposed by Ferreira et al. (2014), it is possible to use existing R classes to represent such data types. In summary, Table 1 lists the R packages and their classes that can be used to represent spatiotemporal data.

The data types proposed by Ferreira et al. (2014) have interpolation functions, called interpolator, associated to their instances. In R packages, the interpolation functions are not directly associated to their objects. Thus, we have to use interpolation functions provided by other packages to estimate values in non-observed locations and times. For example, to create the coverages shown Figure 16, we used the IDW interpolation function from gstat package.

We can conclude that the spacetime package plays a key role in spatiotemporal data representation in R. It provides base data types, STIDF, STFDF and STSDF, that are used for many other packages to represent and analyze spatiotemporal data. For example, the packages trajectories defines their classes to represent trajectories based on the STIDF type. The package gstat provides spatiotemporal interpolation functions, such as spatiotemporal kriging, based on the STFDF class.



Spatiotemporal Data Type	Package	Class
Observation	spacetime	STIDF
Time series	xts	xts
Trajectory	trajectories	Track
Coverage	raster	setZ
	spacetime	STFDF

Tabela 1. R Packages and their classes to represent spatiotemporal data

Using existing R packages, such RODBC and Rgdal, we can access data from different data sources. Furthermore we can load subsets of data using queries in the case of vector data or creating several stacks of raster data to be processed in batches avoid overhead in memory. However, they do not deal with spatiotemporal data.

When we load the observation set using the RODBC package, spatial data sets are loaded as textual or numerical types. Then, it is necessary to convert these data types to spatial type. Using the Rgdal package, we can access the observation set as spatial data types. Neither of them can access spatiotemporal data directly. It is necessary to prepare spatial and temporal data separately and then construct spatiotemporal data types.

A disadvantage of not having packages that directly access spatiotemporal data sets, such as trajectory and coverage, is that we can not filter such data sets properly. For example, we can not load from data sources to R classes only the trajectories whose spatiotemporal bounding boxes intersect a given box. Or, we can not load from data sources to R classes only a part of coverages based on a spatiotemporal restriction. These filters are important because R has limitations of memory on handling large objects. According to Kane et al. [Kane et al. 2013], R is not well-suited for working with data structures larger than about 10-20% of a computer RAM memory. Thus, it is necessary to handle data sets by parts in R, using filters to restrict the amount of data in memory.

As future work, we intend to use spatiotemporal data mining algorithms in R to analyze the coverages showed in Figure 16. The goal is to identify regions where vessel velocities are low, that is, regions where vessels are probably fishing.

Referências

Bivand, R., Keitt, T., and Rowlingson, B. (2013a). Rgdal: Bindings for the geospatial data abstraction library. r package version 0.8-10.

Bivand, R. S., Pebesma, E., and Gomez-Rubio, V. (2013b). *Applied spatial data analysis with R, Second edition*. Springer, NY.

Conway, J., Eddelbuettel, D., Nishiyama, T., Prayaga, S. K., and Tiffin, N. (2008). Rpostgresql: R interface to the postgresql database system.

Ferreira, K. R., Camara, G., and Monteiro, A. M. V. (2014). An algebra for spatiotemporal data from observations to events. *Transactions in GIS*, 18(2):253–269.

Ferreira, K. R., de Oliveira, A. G., Monteiro, A. M. V., and de Almeida, D. B. (2015). Temporal GIS and spatiotemporal data sources. In *XVI Brazilian Symposium on GeoInformatics(GEOINFO), Campos do Jordão, São Paulo, Brazil, November 29 - December 2, 2015.*, pages 1–13.

Galton, A. (2004). Fields and objects in space, time, and space-time. *Spatial cognition and computation*, 4(1):39–68.

Galton, A. and Mizoguchi, R. (2009). The water falls but the waterfall does not fall: New perspectives on objects, processes and events. *Applied Ontology*, 4(2):71–107.

Guting, R. H. and Schneider, M. (2005). *Moving Objects Databases*. Morgan Kaufmann.

Hijmans, R. J. (2016). Introduction to the raster package.

Hornsby, K. and Egenhofer, M. (2000). Identity-based change: a foundation for spatio-temporal knowledge representation. *International Journal of Geographical Information Science*, 14(3):207–224.

ISO (2008). Geographic information - schema for moving features. ISO 19141:2008, International Organization for Standardization, Geneva, Switzerland.

Kane, M. J., Emerson, J., and Weston, S. (2013). Scalable strategies for computing with massive data. *Journal of Statistical Software*, 55(14):1–19.

Kuhn, W. (2009). *A Functional Ontology of Observation and Measurement*, pages 26–43. Springer Berlin Heidelberg, Berlin, Heidelberg.

Liu, Y., Goodchild, M. F., Guo, Q., Tian, Y., and Wu, L. (2008). Towards a general field model and its order in GIS. *International Journal of Geographical Information Science*, 22(6):623–643.

Lovelace, R. and Cheshire, J. (2014). Introduction to visualising spatial data in R. *National Centre for Research Methods Working Papers*, 14(03).

OGC (2006). Opengis implementation specification for geographic information - simple feature access-part 1:common architecture. Technical Report 19141:2008, OPEN GEOSPATIAL CONSORTIUM, Geneva, Switzerland.

Pebesma, E. (2012). spacetime: Spatio-temporal data in R. *Journal of Statistical Software*, 51(7):1–30.

Pebesma, E. (2016). Handling and analyzing spatial, spatiotemporal and movement data.

Pebesma, E. and Graeler, B. (2016). gstat: Spatial and spatio-temporal geostatistical modelling, prediction and simulation.

Pebesma, E. and Klus, B. (2015). Analysing trajectory data in R.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

RIGAUX, P., SCHOLL, M., and VOISARD (2002). *Spatial Databases with Application to GIS*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Ripley, B. and Lapsley, M. (2016). Rodbc: Odbc database access.

Ryan, J. and Ulrich, J. (2012). xts: extensible time series. r package version 0.8-6.

Worboys, M. F. (1994). A unified model for spatial and temporal information. *Comput. J.*, 37(1):36–34.

Worboys, M. F. (2005). Event-oriented approaches to geographic phenomena. *International Journal of Geographical Information Science*, 19(1):1–28.

Wuertz, D., Setz, T., Chalabi, Y., and Byers, M. M. J. W. (2015). Package timedate.

Zeileis, A. and Grothendieck, G. (2005). zoo: S3 infrastructure for regular and irregular time series. *Journal of Statistical Software*, 14(6):1–27.

Integration of data sources on traffic accidents

Salatiel Ribeiro dos Santos[1], Clodoveu A. Davis Jr.[1], Rodrigo Smarzaro[1]

[1]Depto. de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)
Av. Presidente Antônio Carlos, 6627 – 31270-901 – Belo Horizonte – MG – Brasil

{salatiel.ribeiro,clodoveu,smarzaro}@dcc.ufmg.br

Abstract. *General interest geographic information is routinely produced by several public agencies. In cities, the use of smartphones and other mobile devices generates an increasing amount of unofficial georeferenced data. Although official data have usually higher reliability, it takes longer to be updated and made available, while the opposite occurs with unofficial data. This work explores the potential for integrating data from both official and unofficial sources. We present a case study with two traffic accident datasets in the city of Belo Horizonte, Brazil. We compare official traffic accident data to unofficial data collected from the mobile app Waze. We found that 7% of accidents reported by official sources have also been reported by users of Waze. Accidents reported only by official sources are concentrated in the central region, while those recorded by Waze are mostly on some major roads all over the city.*

1. Introduction

Among their institutional responsibilities, several public organizations produce geographic data of general interest. However, due to operational or technological difficulties, part of these data does not become accessible to the public, or is published in formats that preclude their dynamic integration to other data sources, thereby making it harder to accomplish more elaborate analyses. In Brazil, in spite of the approval of the Law on Information Access in 2011, most governmental data producers still have no clear open data policy or practice, nor do they implement technological resources such as APIs and service-based spatial data infrastructures (SDI) to foster easy access to data that are relevant to the society at large.

On the other hand, the intensive use of smartphones and other mobile devices generates a significant volume of data, since wherever people go their trajectories can be recorded, and there are many ways for them to express themselves on the visited places [Zheng et al. 2009]. Simultaneously, the interest in geographic data and geographic applications grows, as they enable users to locate themselves and to find events and other people based on their location [Quercia and Capra 2009, Quercia et al. 2010]. In this scenario, users are routinely issuing comments and opinions through social networks (e.g., Twitter and Facebook), commenting on tourist attractions and commercial venues (Foursquare, Yelp), publishing personal videos and photos (Instagram, Snapchat, Flickr) or sharing real-time information on traffic (Waze[1]). With a large number of contributors, such applications become important, albeit unofficial, sources of information.

This work intends to explore the potential for integrating official and unofficial sources of data on urban events, in order to verify to which degree data generated by

[1] http://www.waze.com

the usual work processes in public organizations can be confirmed or enhanced by data generated by volunteers, in crowdsourcing or crowdsensing systems. The possibilities for adding, expanding or replacing official data sources with unofficial ones is analyzed, with the additional challenge of treating the intrinsic heterogeneity of such data sources. Therefore, the main objective of this work is to integrate multiple and heterogeneous geographic data sources to support analyses that can be used in decision-making related to improvements in urban services. It includes a case study that uses official data and data obtained from Waze on traffic accidents in the city of Belo Horizonte, Brazil. Both sources contain geographic information on the location of accidents, but other descriptive attributes are quite distinct.

This paper is structured as follows. Section 2 describes related work. Section 3 presents a comparison between official and unofficial sources of traffic accident data. Section 4 describes our integration methodology. Results are presented and discussed in Section 5. Finally, Section 6 concludes the paper and presents ideas for future work.

2. Related Work

A study by the World Health Organization (WHO) [WHO 2015] indicates a serious worldwide traffic problem. According to that study, in 2013 alone 1.25 million deaths occurred due to road accidents in 180 countries, especially in those with low income. Another alarming information is that traffic accidents represent the main cause of death among people aged between 15 and 29 years. Also, Brazil is ranked in position 56 among the countries with the highest number of deaths caused by traffic accidents.

Overcoming this problem requires improvements in infrastructure and educational campaigns, thus the responsibility of changes is shared among politicians, managers, road designers, vehicle manufacturers and citizens that use the road system. Professional and amateur drivers have access to several technologies for routing in streets, roads and highways, including real-time information on accidents and traffic events. On the other hand, as citizens we perceive the lack of objective and useful information that allows drivers to take instantaneous notice of problems and adopt measures against accidents in critical points along urban streets and highways.

Data produced by governmental organizations, or data contributed voluntarily or unconsciously by common citizens, help in understanding city dynamics and user preferences in moving through urban space. Such information are instrumental in decision-making for improving infrastructure and can contribute to improvements in life quality [Corsar et al. 2015, Wolf and Fry 2013]. In the current context, in which solving traffic and transportation problems in large urban centers becomes even more urgent, it is important to promote the use of alternative transit so that the number of vehicles in the streets can be reduced. However, it is necessary to provide minimal security conditions for people that opt for the alternative transportation methods. Machado et al. (2015) identify points in which traffic accidents are concentrated for the city of São Paulo (Brazil) and Rome (Italy). The focus of the study are accidents involving individuals traveling by non-motorized means (on foot or by bicycle). From the results of the study, users of alternative transportation can be extra careful or avoid completely such dangerous regions. The article invokes the discussion on traffic accidents in Brazil.

Even though there are public data on accidents in Brazil, the absence of a nationally integrated database is remarkable. Bezerra et al. (2015) present a number of difficulties inherent to Brazilian sources that have an impact on the establishment of such an integrated database. Among these difficulties, authors highlight the way federative agents are structured and the distribution of responsibilities among them. Brazil has a mixture of federal highways, state highways and urban thoroughfares in charge of local governments. In the first case, accidents are recorded and followed up by many organizations, in particular the national highways department (DNIT - Departamento Nacional de Infraestrutura de Transportes), the Ministry of Transportation, the Ministry of Cities and the Ministry of Health, along with the Federal highway police. In each state, there is a transit department, a state highway police, an military police, and administrative organizations in charge of transit and transportation. As to municipalities, while the largest ones maintain transit and transportation engineering companies, the less populous ones usually have no means to record and analyze accidents.

The diversity of transit-related organizations would not be an important obstacle if there was a unified process for recording traffic events. Bezerra et al. (2015) highlight that there is not even a standardized police report form. Nevertheless, event reports are the largest source of information currently available. Undernotification of traffic accidents is also expected, since involved parties seek official reporting of the event mostly in case there are victims or the need to sustain insurance claims. Authors also indicate difficulties for data analysis in accident reports, due to incompleteness, coding errors, discontinuity and lack of elements with which to locate the accident.

In other countries, accident data are treated in a much different manner, leading to well-grounded analysis works. Morris et al. (2008) present the creation of a database on fatal traffic accidents in Europe. Many works use United States governmental sources to diagnose problems such as lack of attention while driving, hitting pedestrians, light vehicle crashes [Najm et al. 2003] and effects of driver population aging [Stamatiadis and Deacon 1995]. Such analyses are strongly hindered in Brazil due to the lack of a nation-wide system for recording traffic accidents.

Using Twitter data, Ribeiro et al. (2012) geolocate traffic-related events based on the content of posts. From that, traffic accidents, congestions and interruptions can be identified and mapped. Also using unofficial data, Silva et al. (2013) detect traffic conditions in urban roads using Waze data. They also discuss limitations related to the source, including its coverage.

Integrating the various sources of official data and combining them with unofficial sources is an important problem for initiatives related to Brazilian traffic problems. Such integration may help solving problems such as undernotification, and promoting unified access to official reports. A major goal is to obtain an integrated dataset that can be used in diagnosing and analyzing events such as those reported in the works listed in this section. The next section describes two datasets from the city of Belo Horizonte that are used in a case study for the integration of official and unofficial data. Following that, a methodology for integration is discussed.

3. Datasets

3.1. Official Data

Accidents are usually reported to the authorities in charge of traffic or public security, who, in turn, record the event in police reports, incident reports or similar documents. Such information are kept in databases by the authorities at the federal, state or local levels. In Brazil, accidents in urban thoroughfares are recorded by municipal authorities. Accidents in state or municipal highways are recorded by the respective administrative levels. Accidents in federal highways can be recorded by authorities at any level, depending on existing administrative agreements. Accidents on segments of federal highways that cross urban areas are typically recorded by the state's military police.

This work uses accident data for the city of Belo Horizonte, Brazil, in 2014, as supplied by the municipal transit company, BHTrans. These data are well structured and are reliable, since they originate in police reports, filed by the state's military police. However, these are the latest data available for analysis, since BHTrans is still unable to release the 2015 data at the time of this writing (September 2016).

3.2. Unofficial Data

We classify as unofficial those data that come from social networks, active or passive crowdsourcing or crowdsensing applications [Mateveli et al. 2015] or any other source that is not connected to governmental institutions. In this work, we use data from Waze, a GPS-based navigation application that is able to integrate data collected by users in order to guide others through traffic. Such collected data includes actively volunteered information on traffic congestion, police actions and accidents, as well as passively collected data on travel speeds.

The lack of an API for data collection is a serious shortcoming of Waze. Obtaining Waze data without an API requires monitoring the app's Web-based live map in small areas, and extracting relevant information from the underlying JSON files. As in the case of many crowdsourcing or volunteered geographic information (VGI) applications, Waze also suffers from lack of detail, questionable reliability of contributing users, and irregular spatial coverage. The validity of Waze information can partially be assessed by confirmation from other users. The main advantages of Waze are the timely access to accident data, which allows users to plan trips that avoid congested areas. This strongly contrasts with data publication policies by official transit authorities. Waze is also expected to record accidents that are not officially communicated to transit or police authorities, which is the case of less serious incidents or accidents involving uninsured vehicles.

4. Methods

The first step towards the implementation of this work is data acquisition. As mentioned, traffic accident reports for 2014 were provided by BHTrans. Besides geolocation, accident data includes date, time, type of accident and vehicles involved.

The unofficial data was obtained from Waze in 2014, as part of a data collection experiment in a project that intended to map frequently congested areas. Specifically, for this work we selected accident reports from the 2014 dataset. From each accident alert it is possible to obtain information that is similar to official BHTrans data, such as geolocation, date, time and type of accident.

Since Waze does not have an official API, data was collected through a GeoRSS file[1] generated by the Live Map at the application's Web site[2]. A JSON file is downloaded in regular intervals, containing real-time geographic features and locations of objects and, in this case, data on traffic congestion and alerts. Live Map can hide some alerts, depending on zoom settings. This is the main limitation on the data collection process, since it reduces data coverage. Furthermore, since JSON collection is re-executed frequently, the series of JSON files needs to be processed to eliminate duplicate incident reports.

The second step of the process is data processing: select data in the same time window (both between 2014-09-16 and 2014-11-06), group accidents reported more than once (by different users) on Waze and geocode records without geographical coordinates (some BHTrans records included the textual description of a location but not actual coordinates). In order to consolidate accidents reported more than once on Waze, the average position of the reports is calculated and the number of contributions that each of the accidents received is noted. The resulting datasets contain 1,434 and 1,543 accident reports, respectively in BHTrans ans Waze datasets.

Finally, the last step integrates data from the official and unofficial sources. In this step, data are checked for overlapping events. Furthermore, the characteristics of data coming specifically from either source are explored, thus verifying how complementary they might be. We established a set of matching criteria, by which two records, each one belonging to one of the data sources, refer to the same accident if they (1) were reported within 1 hour of each other and (2) occurred within 50 meters of each other or within 150 meters and on the same road. At the end of the process, a single integrated dataset on accidents is created, containing annotations about the origin of each data item.

The first matching criterion reflects that there might be a delay between the time the accident actually happened (informed by the involved parties in a police report) and the time the accident was reported on Waze. The second one was adopted because there might be situations in which the Waze user is passing by the accident location, but reports it to the app in a position that is away from where the accident actually happened.

We evaluated the accidents reported in both datasets, the notifications contained only in the official data and the accidents contained only in the unofficial data. Next section presents and discusses the results of the matching process.

5. Results and Discussion

5.1. Matched accidents

In order to verify which portion of Waze data corresponds to the data provided by BHTrans (that is, the number of accidents reported both officially and by Waze users), we apply comparisons following the criteria described on Section 4 (date/time frame and distance between reported positions). We found that only 7% of BHTrans reports matched accidents reported on Waze. Figure 1 shows the location of accidents reported on both official and unofficial datasets. Spatial distribution is sparse, with a few clusters of accidents in major thoroughfares.

[1] http://www.georss.org/

[2] https://www.waze.com/pt-BR/livemap

Since Waze allows a single event to be reported more than once, we consider only one of the multiple reports. Indeed, the consolidation step in the preparation of the Waze dataset shows that 38% of the accidents on Waze were reported by two or more users. A positive aspect on this repetition is that it makes the data more reliable, since many events are based on reports from more than one user. Notice, however, that Waze users that have already seen in their smartphone an accident report in their path, will probably select another route or consciously avoid reporting that same accident again.

Official data on traffic accidents are usually recorded in police reports, which are mandatory only in cases where the parts involved aim some kind of material compensation for damages, directly from the responsible for the accident or through an insurance claim. Considering that, it's possible that part of the accidents with lower severity are not officially reported. We therefore expect that a share of the events recorded in Waze are not officially reported, and thus the datasets are expected to be complementary. Likewise, a share of the accidents reported officially may not be recorded by Waze users, especially if they take place in locations where the impact on traffic is small, or at times and places where the presence of Waze users is low. Notice that Waze usage naturally tends to be concentrated around rush hours, in which knowing about traffic problems along one's path is of greater concern. We now proceed to analyze accidents that have not been matched at either dataset.

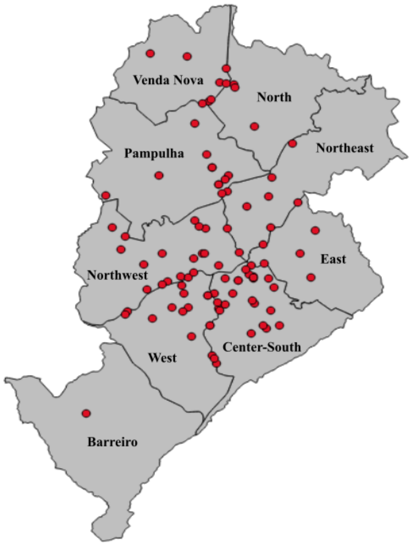


Figure 1. Matched accidents

5.2. Unmatched accidents

Analyzing the reports that appear exclusively on either Waze or BHTrans datasets, we can see a different distribution from the one seen on Figure 1.

Accidents that appear only in the BHTrans dataset (Figure 2a) are found mostly in the central regions of the city, which is understandable since this area has a more intense traffic flow with a high concentration of economic activities. Secondary commercial areas in a northern and in a southwestern regions of the city also concentrate many accidents.

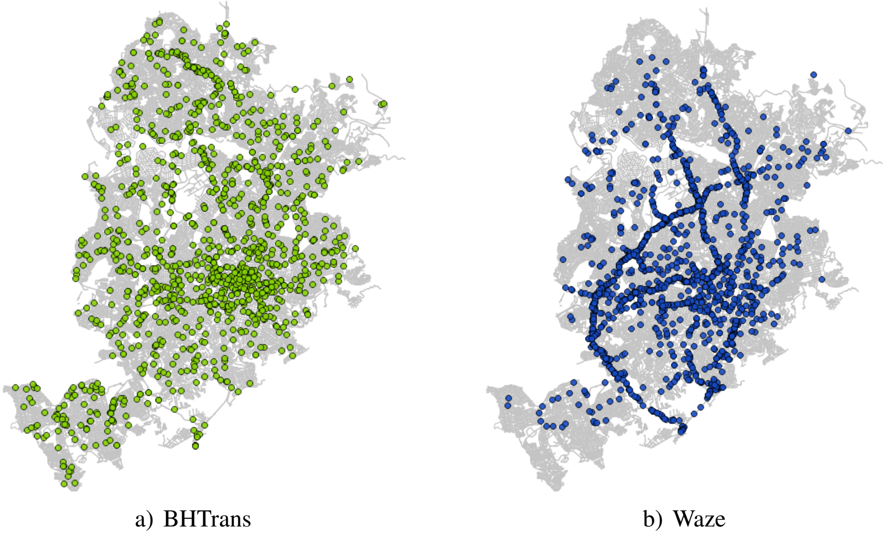


Figure 2. Unmatched accidents

Events reported exclusively by Waze (Figure 2b), on the other hand, show a distinct pattern. Accidents are concentrated along some large city thoroughfares and urban segments of highways. Figure 3 shows heatmaps based on the concentration of unmatched accidents from either source.

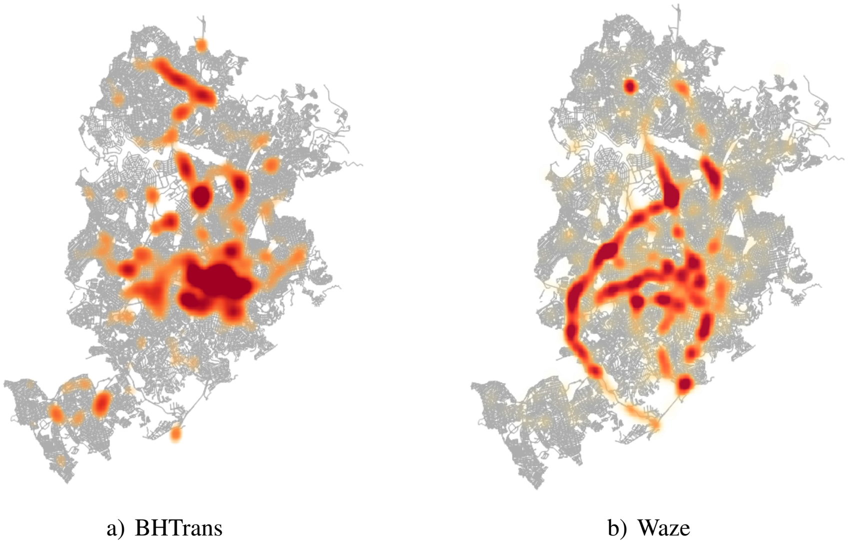


Figure 3. Heat map of unmatched accidents

5.3. Severity and types of Accidents

Official records include the types of traffic accidents. Most frequent types include side collisions, collisions involving pedestrians, rollovers, rear-end colisions, and even accidents where people are thrown out of the vehicle. The existence of victims (injuries, fatalities) is also indicated. However, the records do not consistently inform on the severity of each accident. For example, a collision can either cause light injuries or more serious ones, requiring hospital treatment, or even leading to death. However, such details are absent from official records.

Waze records include less details than official ones. Data about an accident are informed by users when they are nearby, often driving, and it can be difficult for them to obtain details. Waze uses two severity classifications for accidents: major and minor. A minor accident is described as "fender benders with minor or no injuries, also no fatalities" while major represents "major damages to vehicle, major injuries and possible fatalities" [Waze 2016].

The most frequent types of accident found in official dataset are "side collision with victims" (39%), followed by "rear-end collision with victims" (18%). In the Waze dataset, 56.1% of the accidents are classified as minor, and 21% as major. The severity of the remaining 23% is not reported. Figure 4 shows the six most frequent types from official data and Figure 5 shows the distribution of severity classification from Waze data.

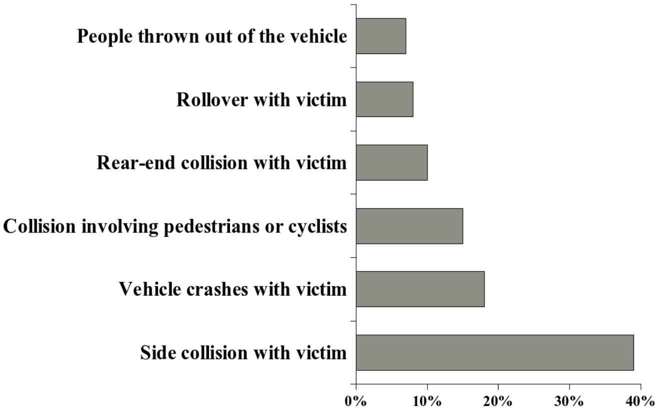


Figure 4. Main types of accidents from official sources

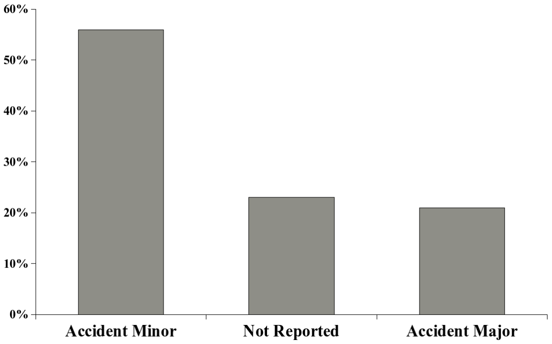


Figure 5. Severity of accidents from Waze

Comparing results from both datasets, most officially reported accidents can be understood as high severity (since they are classified as accidents with victims), while most accidents recorded by Waze are classified as minor severity. This discrepancy raises, at least, two hypotheses. First, it is difficult for Waze users to know detailed information about the accidents they report. If they witness the accident, they can overestimate its severity, while if they drive by the accident's location after some time their assessment can be underestimated. Second, Waze users may not be able to pay much attention on the correct severity classification when informing an accident.

Among matched accidents, Waze users only tend to classify as major severity accidents those officially classified as collisions involving pedestrians and rear-end collisions with victims (Table 1). In other cases, most accidents are deemed minor by Waze users. This indicates a semantic discrepancy between regular citizens (Waze users) and police or transit officials, which has to be investigated further. Also, for accidents reported by several different Waze users, the discrepancy among them must be assessed. However, as in any crowdsourcing process, accident classification by Waze users should be less reliable than official reports, since, as passers-by, Waze users do not have full access to the accident site.

Table 1. Classification of severity by Waze users for each type of accident from BHTrans (all numeric values are percentages)

Accident Types (BHTrans)	Severity (Waze - values are %)		
	Major	Minor	Not Informed
Side collision with victims	21.6	56.8	21.6
Vehicle collision with victims	31.1	46.6	22.3
Collision involving pedestrians without fatality	54.1	37.5	8.3
Vehicle crashes with victims	47.3	42.1	10.5
Rollover with victims	28.6	42.8	28.6
People thrown out of the vehicle	37.5	50.0	12.5

Regarding the distribution of accidents along the day, Waze reports concentrate on rush hours, either in the morning or in the afternoon (Figure 6). This is expected, since at those hours the impact of accidents on traffic is the greatest. On the other hand, official data, while also recording a high number of accidents at rush hours, contain reports of accidents that took place all through the day, including times at which circulation is lower.

5.4. Integrated dataset

After matching the two sources, an integrated dataset on accidents in Belo Horizonte was built. Table 1 shows its structure, indicating the attributes that were obtained from each individual source, plus an attribute that records the source of the original information. In the integrated dataset, there are 1,333 accidents that were reported exclusively to BHTrans, 1,442 gathered exclusively from Waze, and 101 that have been matched from both sources. Since the number of matching records is small, integrating Waze data to the official dataset represents a 100.5% increase in the overall number of accident reports.

6. Conclusions

The growing use of mobile apps and social networks generates a considerable amount of data, which can be used for several purposes. Information from these sources can be

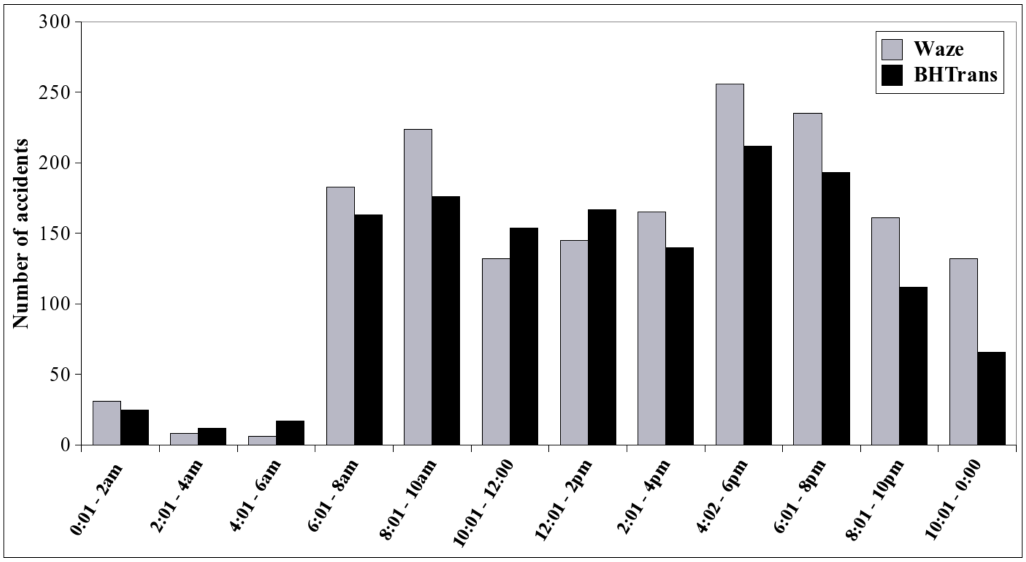


Figure 6. Distribution of accidents throughout the day (October 2014)

Table 2. Integrated accidents dataset

Integrated Attributes	Sources		Remarks
	BHTrans	Waze	
bhtrans id	police report id	—	—
waze id	—	alert id	—
date	date	date	—
street type	street type	—	—
street name	street name	street name	use BHTrans if available
number	number	—	—
neighborhood	neighborhood	—	—
region	region	—	—
city	—	city	—
country	—	country	—
geom	longitude, latitude	longitude, latitude	average position
type of accident	type of accident	—	—
severity	—	severity	—
number of victims	number of victims	—	—
number of deaths	number of deaths	—	—
vehicles involved	vehicles involved	—	number and type of vehicles
data source	—	—	BHTrans, Waze or both

used for identifying better routes, monitoring traffic conditions in real time and identifying areas with high car accident rates. Besides that, the data collected from this kind of source can complement the data extracted from official sources.

This work compared car accident records provided by BHTrans (official) and data collected from the mobile app Waze (unofficial). We found out that 7% of the car accidents officially reported were also reported through unofficial sources. Among these accidents, the main type was collision with victims, while most were classified as low severity according to Waze data. It's important to stress, however, that the classification used by Waze is not totally reliable, given that users usually do not have direct access to the accident site.

Accidents reported only by BHTrans were concentrated on the central region of Belo Horizonte, while the ones reported by Waze were mostly on highway segments, like Rodovia MG-10 and Anel Rodoviário. These patterns endorse the idea that the datasets are complementary, since coverages are quite distinct.

We verified that most of the accidents reported by Waze but not by BHTrans were classified as having lower severity. This happens possibly due to the fact that police reports are not mandatory, which probably implies that most of the lower severity accidents are not officially reported. Thus, Waze data can be used to fill the gap of undernotification in the case of traffic accidents, providing authorities with a broader view of the problem.

The official data has the advantage of being more reliable and detailed. However, it is harder to access due to government-imposed limitations and time constraints. On the other hand, the unofficial registers are easy to access, but poor in details. The data from these two sources can be integrated in order to obtain a dataset with broader coverage. This could compensate the deficiencies of each of the sources, taken individually. However, official data should be made available more readily, possibly using technologies such as spatial data infrastructures or APIs dedicated to the task of providing unrestricted access to traffic accident data.

The next step for this work is to collect data from a wider range of official sources regarding car accidents, expanding the analysis to other cities besides Belo Horizonte. Still regarding the official data, we aim to consider Brazilian federal highways accidents database, provided by the National Transportation Infrastructure Department (DNIT). We also intend to get data from other unofficial sources, such as Twitter. With multiple sources of heterogeneous data, integration methods need to evolve accordingly.

7. Acknowledgments

Authors acknowledge the support of CNPq and FAPEMIG, Brazilian agencies in charge of fostering research and development. We also thank BHTrans for the official dataset, Patrick Prado for the Waze dataset, and professors Cláudia A. Soares Machado and José Alberto Quintanilha for their important contributions in the development of this work.

References

Bezerra, B. S., Cunto, F. C., Barbosa, H. M., Davis, C., and Lança, J. F. d. A. (2015). Main stumbling blocks for a good traffic accident database system – evidences from Brazil. *Latin American J. Management for Sustainable Development*, 2(2):112–123.

Corsar, D., Markovic, M., Edwards, P., and Nelson, J. D. (2015). The Transport Disruption Ontology. In *The Semantic Web - ISCW 2015: 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part II*, volume 9367 of *Lecture Notes in Computer Science*, pages 329–336. Springer International Publishing.

Machado, C., Giannotti, M., Neto, F., Tripodi, A., Persia, L., and Quintanilha, J. (2015). Characterization of Black Spot Zones for Vulnerable Road Users in São Paulo (Brazil) and Rome (Italy). *ISPRS International Journal of Geo-Information*, 4(2):858–882.

Mateveli, G. V., Machado, N. G., Moro, M. M., and Davis Jr., C. A. (2015). Taxonomia e desafios de recomendação para coleta de dados geográficos por cidadãos. In Braganholo, V., editor, *XXX Simpósio Brasileiro de Banco de Dados - Short Papers, Petrópolis, Rio de Janeiro, Brasil, October 13-16, 2015.*, pages 105–110. SBC.

Morris, A., Brace, C., Reed, S., Fagerlind, H., Bjorkman, K., Jaensch, M., Otte, D., Vallet, G., Cant, L., Giustiniani, G., Parkkari, K., Verschragen, E., and Hoogvelt, B. (2010). The development of a european fatal accident database. *International Journal of Crashworthiness*, 15(2):201–209.

Najm, W. G., Sen, B., Smith, J. D., and Campbell, B. N. (2003). Analysis of Light Vehicle Crashes and Pre-Crash Scenarios Based on the 2000 General Estimates System. *The National Academies of Sciences, Engineering, and Medicine*, page 80 p.

Quercia, D. and Capra, L. (2009). Friendsensing: Recommending friends using mobile phones. In *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys '09, pages 273–276, New York, NY, USA. ACM.

Quercia, D., Lathia, N., Calabrese, F., Di Lorenzo, G., and Crowcroft, J. (2010). Recommending Social Events from Mobile Phone Location Data. In *2010 IEEE International Conference on Data Mining*, pages 971–976. Institute of Electrical & Electronics Engineers (IEEE).

Ribeiro Jr., S. S., Rennó, D., Gonçalves, T. S., Davis, C., Meira Jr., W., and Pappa, G. L. (2012). Observatório do Trânsito: sistema para detecção e localização de eventos de trânsito no Twitter. *Simpósio Brasileiro de Bancos de Dados*, pages 81–88.

Silva, T. H., Vaz De Melo, P. O. S., Viana, A. C., Almeida, J. M., Salles, J., and Loureiro, A. a. F. (2013). Traffic condition is more than colored lines on a map: Characterization of Waze alerts. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8238 LNCS:309–318.

Stamatiadis, N. and Deacon, J. A. (1995). Trends in highway safety: Effects of an aging population on accident propensity. *Accident Analysis and Prevention*, 27(4):443–459.

Waze (2016). *Manual do Usuário versão 3.5*. Waze. Available: https://wiki.waze.com/wiki/Como_Alertar [Accessed 15 August 2016].

WHO (2015). Global status report on road safety 2015. Technical report, World Health Organization. Available: http://www.who.int/violence_injury_prevention/road_safety_status/2015/en/ [Accessed 03 August 2016].

Wolf, K. and Fry, J. (2013). Benchmarking performance data. In Goldstein, B. and Dyson, L., editors, *Beyond Transparency: Open Data and the Future of Civic Innocation*, chapter 18, pages 233–252. Code for America Press.

Zheng, Y., Chen, Y., Xie, X., and Ma, W.-Y. (2009). GeoLife2.0: A location-based social networking service. In *2009 Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*. Institute of Electrical & Electronics Engineers (IEEE).

Assessment of texture features for Brazilian savanna classification: a case study in Brasilia National Park

Cesare Di Girolamo Neto[1], Leila Maria Garcia Fonseca[1], Thales Sehn Körting[1]

[1]Instituto Nacional de Pesquisas Espaciais – INPE
Caixa Postal 515 - 12227-010, São José dos Campos – SP – Brazil

{cesare.neto;leila.fonseca; thales.korting}@inpe.br

Abstract. *Distinguishing Brazilian savanna physiognomies is an essential task to better evaluate carbon storage and potential emissions of greenhouse gases. In this study, we propose to evaluate the potential of texture features to improve the discrimination among five physiognomies in the Brazilian savanna: Open Grasslands, Shrubby Grassland, Shrubby Savanna, Savanna Woodland and Gallery Forest. Texture features extracted from RapidEye images and also from Spectral Linear Mixture Model components and Vegetation Index are evaluated in this study. Results showed that texture features based on GLCM can reduce misclassification for Open Grasslands, Shrubby Grasslands and Shrubby Savanna classes.*

1. Introduction

Brazilian savanna, also known as Cerrado, occupies an area of approximately two million square Kilometers on the Brazilian territory, mainly in the central part of Brazil (MMA, 2015). Cerrado is one of the richest biomes in the world and it contains more than 160.000 species of plants, animals and fungi (Ferreira *et al*., 2003). Besides that, Cerrado is responsible for storing about 5.9 billion tons of carbon in vegetation and 23.8 billion tons in the ground (MMA, 2014).

The loss of natural vegetation in Cerrado reached 45.5% of its original area by 2013 (MMA, 2015). The loss of biodiversity can lead to problems such as: soil erosion, water pollution, carbon cycle of instability, microclimate changes and also biome fragmentation (Klink & Machado, 2005). Considering these negative effects on biodiversity, it is essential to promote strategies to monitor the Cerrado biome.

Mapping of heterogeneous tropical areas, such as Cerrado, should be carried out considering biological, climatic and topographical information. The major natural formations in Cerrado are Grasslands, Shrublands and Woodlands (Figure 1). Their mapping has been the subject of several studies. Sano *et al*. (2009) performed visual interpretation of satellite images to produce maps of Cerrado. This process was very time consuming and difficult to discriminate Grasslands.

The difficulty to map Cerrado patterns is even greater when considering more formations than those mentioned above. For example, the system proposed by Ribeiro & Walter (2008) splits these major formations into 14 physiognomies. Identifying these physiognomies is important to evaluate carbon storage and potential emissions of greenhouse gases for each type of land cover.

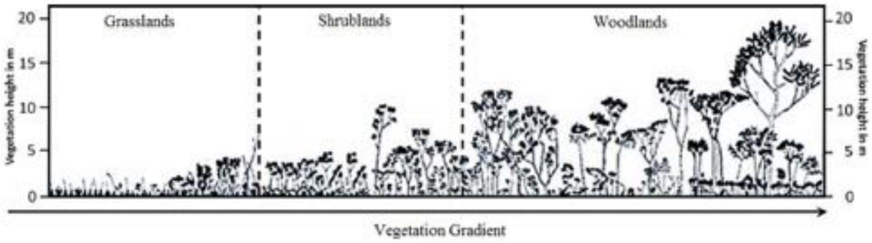


Figure 1. Major Cerrado formations describing the vegetation gradient (adapted from Schwieder *et al.*, 2016).

Most studies aiming to classify vegetation types in the Cerrado biome rely on the use of spectral information from remote sensing imagery. The NDVI (Normalized Difference Vegetation Index) has been tested to discriminate Cerrado physiognomies (Liesenberg *et al*., 2007; Oliveira *et al*., 2007; Costa *et al*., 2014). However, there is still difficulty to discriminate different grassland physiognomies using only NDVI. Spectral Linear Mixture Model (SLMM) has also been used to classify physiognomies on a protected area of Cerrado in Distrito Federal State, Brazil (Ferreira *et al*., 2007). The SLMM reduced the classification error between Grasslands and Shrublands, but it was not enough to fully automate the classification.

Differently, Carvalho *et al*. (2010) used texture features to map the vegetation cover in the Cerrado. In this paper, an initial classification was performed using data based on NDVI, SLMM and spectral features. Afterwards, they included texture information into the dataset and noticed an increase in the classification accuracy. Peneque-Galvez *et al*. (2013) also used texture features to classify Cerrado physiognomies in Bolivia. In this case, Woodlands were classified with high accuracy, but some errors confusion occurred in the discrimination between Grasslands and Shrublands. However, some texture features increased this error, and reduced the overall classification accuracy. Therefore, it is necessary to investigate whether these features may be really effective for Cerrado classification.

In order to better analyze texture features in the Cerrado classification, we could calculate texture features from NDVI instead of calculating it from original image. The NDVI texture has been used in other applications such as urban studies (Nussbaum & Menz, 2008). It has also been used to detect bushfire prone areas (Chen *et al*., 2001) as well as to identify different types of forest and spatial patterns of vegetation structure (Ning *et al*., 2011).

Therefore, we propose in this work to evaluate the potential use of texture information extracted from RapidEye original images, and also from vegetation index and SLMM components to classify the following physiognomies in the Brazilian Cerrado: Open Grasslands, Shrubby Grassland, Shrubby Savanna, Savanna Woodland and Gallery Forest.

2. Methodology

Figure 2 presents the methodology flowchart proposed to classify vegetation cover in Cerrado. Each processing step is detailed in the following sections.

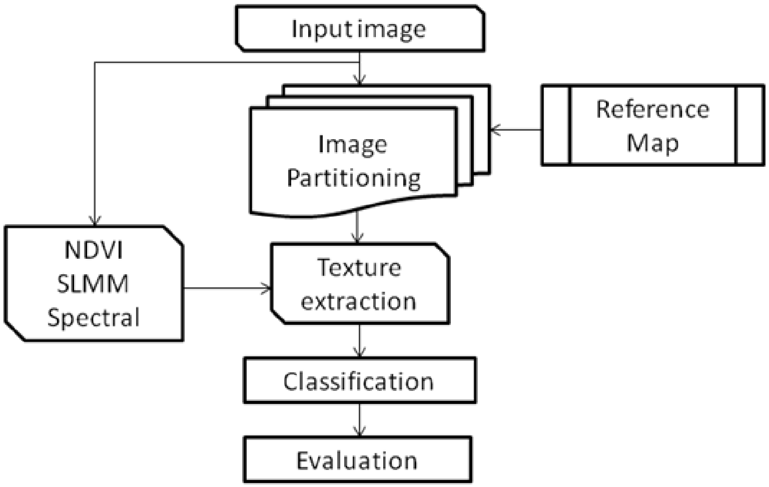


Figure 2: Methodology flowchart.

2.1. Study Area and Reference Map

The study area is located in the Brasília National Park (PNB), which has approximately 30.000 ha of preserved natural Cerrado vegetation. Figure 3 shows the major part of the park, in which a red line highlights the study area. For the experiments, we used a RapidEye image in the path-row 1-318 tile 2331801 of the RapidEye Earth Imaging System (REIS). This image was acquired in 05/30/2014 and processed in level 3A product (Blackbridge, 2015).

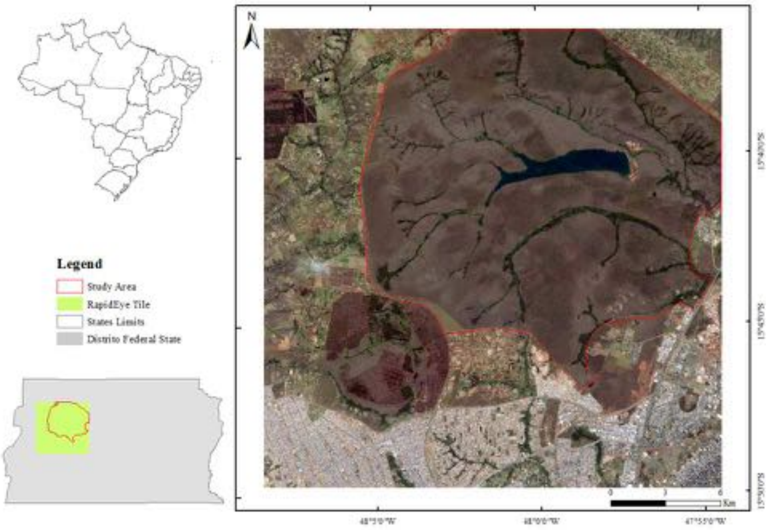


Figure 3: Study area in the Brasilia National Park.

We also used as reference the map of PNB that was produced by Ferreira *et al*. (2007). The authors used the system proposed by Ribeiro & Walter (1998) to classify 5 Cerrado physiognomies described in Table 1. Other classes such as Water Bodies, Marsh, Reforestation, Bare Soil and Constructed Area were removed from the dataset.

Table 1. Cerrado physiognomies characteristics (Adapted from Ferreira *et al.*, 2007 and Ribeiro & Walter, 2008)

Physiognomy name	*Vegetation description*	*Tree cover (%)*	*Tree height (m)*
Open Grassland (OG)	Grasses	0	-
Shrubby Grassland (SG)	Grasses and Shrubs	0-5	-
Shrubby Savanna (SS)	Shrubs and a few trees	5-20	2-3
Wooded Savanna (WS)	Trees and a few Shrubs	20-50	3-6
Gallery Forest (GF)	Trees	70-95	15-30

2.2. Image Partitioning

In order to extract texture features, the image was partitioned into square objects of size ***s*** by ***s*** pixels. The use of square objects instead of polygons extracted from segmentation algorithms based on similarity allows us to evaluate texture features that capture the natural heterogeneity in the image. This procedure prevents detecting texture as a possible rule in the classification process, once we intend to evaluate texture potential for classifying Cerrado vegetation cover.

Following, the image was linked with the reference map to identify the square objects represented within each class in the map. The larger is the square object size, the fewer number of objects can be extracted from the image. We tested different object sizes ranging from 10 to 35, with a step of 5. The maximum value 35 was chosen because the number of samples for Gallery Forest reached almost zero. Figure 4 illustrates the influence of object size in relation to the number of samples for Gallery Forest.

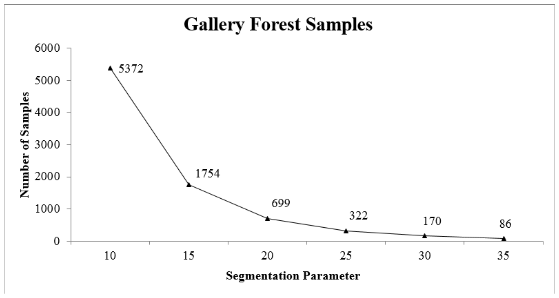


Figure 4. The influence of objects size in relation to the number of samples of Gallery Forest.

After partitioning process, some segments presented two or more classes, which can lead to misclassification (Ferreira *et al*., 2007, Oliveira *et al*., 2007 and Carvalho *et al*., 2010). To reduce this problem these elements were removed from the dataset. Figure 6 illustrates this procedure of cleaning up.

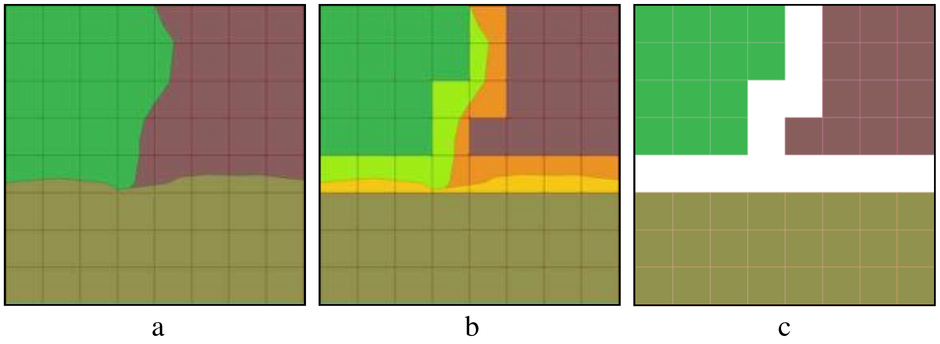


Figure 6. Clean up process. a) original map with 3 different classes; b) highlighted objects for removal; c) map of samples after cleanup process.

2.3. Feature Extraction

Spectral features were obtained from Digital Numbers (DN) of RapidEye (RE) bands. The description the spectral features is presented in Table 2.

Table 2. Spectral Features.

Feature	Description
Band_1	DN from band 1 (Blue 440 – 510 μm)
Band_2	DN from band 2 (Green 520 – 590 μm)
Band_3	DN from band 3 (Red 630 – 685 μm)
Band_4	DN from band 4 (Red Edge 690 – 730 μm)
Band_5	DN from band 5 (NIR 760 – 510 μm)
Brightness (BT)	Average of the sum of means for bands 1-5
Maximum Difference (MD)	Maximum of the difference between bands
NDVI	Normalized Difference Vegetation Index
SLMM_*soil*	SLMM component of soil
SLMM_*shadow*	SLMM component of shadow
SLMM_*vegetation*	SLMM component of vegetation

NDVI and SLMM components (*soil*, *shadow* and *vegetation*) were computed according to Tucker (1979) and Shimabukuro & Smith (1991), respectively. Texture features were computed from Gray Level Co-occurrence Matrix (GLCM) as shown in Table 3. GLCM is a second order histogram in which each entry reports the join probability of finding a set of two grey levels at a certain distance and direction from each other over some pre-defined window (Haralick *et al*., 1973). Additionally, some texture measures were computed from Gray Level Difference Vector (GLDV), as shown in Table 3. GLDV indicates occurrence of the absolute difference between a reference pixel and its neighbor. It can be calculated for 4 different directions (0°, 45°, 90° and 135°). In this study, we used only direction 0°.

Texture features were also extracted from image bands, from NDVI, and from each SLMM component (*soil, shadow and vegetation*). Therefore, 9 texture features (Table 3) were extracted from five images (vector of bands, NDVI, SLMM), which produced a total of 45 features. The features of BT and MD were not used for extracting texture.

Table 3. Textural features based on (Haralick *et al.*, 1973). $P_{i,j}$ is the normalized co-occurrence matrix, N is the number of rows or columns, σ_i and σ_j are standard deviation of row i and column j, μ_i and μ_j are means of row i and column j, V_k is the normalized gray level difference vector, and $k = |i-j|$.

Feature	Formula	Feature	Formula
GLCM Entropy	$\sum_{i,j=0}^{N-1} P_{i,j}\,(-\ln P_{i,j}\,)$	GLCM Dissimilarity	$\sum_{i,j=0}^{N-1} P_{i,j}\,\lvert i-j\rvert$
GLDV Entropy	$\sum_{k=0}^{N-1} V_k\,(-\ln V_k\,)$	GLCM Homogeneity	$\sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+\,(i-j)^2}$
GLCM Contrast	$\sum_{i,j=0}^{N-1} P_{i,j}\,(i-j)^2$	GLCM Mean	$\mu_i = \sum_{i,j=0}^{N-1} i\,(P_{i,j})$
GLDV Contrast	$\sum_{k=0}^{N-1} V_k\,(k^2)$	GLDV Mean	$\mu_i = \sum_{i,j=0}^{N-1} V_k\,(k)$
GLCM Correlation	$\sum_{i,j=0}^{N-1} P_{i,j}\left[\frac{(1-\,\mu_i)(1-\mu_j)}{\sqrt{(\sigma_i)^2\,(\sigma_j)^2}}\right]$		

2.4. Classification

In the classification phase we performed two experiments. In the first experiment, we used four datasets in the classification phase, as shown in Table 4. This was the baseline to evaluate the classification accuracy gain by including each feature into the datasets one at a time. The idea of this experiment is to evaluate classification accuracy for each spectral feature and also the spectral texture, which has been pointed out by Carvalho *et al*. (2010) and Peneque-Galvez *et al*. (2013) as features that improve Cerrado classification. In Table 4, Spectral Texture means texture features extracted from RapidEye bands only.

Table 4: Combination of groups of features to evaluate the best subset for classification.

	RE bands, BT, MD	NDVI	SLMM	Spectral Texture
Dataset 1	X			
Dataset 2	X	X		
Dataset 3	X	X	X	
Dataset 4	X	X	X	X

In the experiment 2, we evaluated all 45 texture features, adding one at a time in each one of the datasets 1-3. The idea was to evaluate the improvement or not in the classification accuracy for each texture feature.

For comparison, we established the same classification algorithm and parameters for all tests. We used Random Forest classification algorithm (Breiman, 2001), implemented in Weka software (Hall *et al*., 2009), and set the number of trees to 100 in order to construct each forest.

2.5. Texture Feature Evaluation

The experiments were carried out using a 10-fold cross validation. For the validation process, we used Global Accuracy, Precision and Recall values to summarize the confusion matrix in the experiments:

$$Global\ Accuracy = \frac{TP + TN}{n} \tag{1}$$

$$Precision = \frac{TP}{(TP + FP)} \tag{2}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{3}$$

in which TP = True Positive, FP = False Positive, TN = True Negative, FN = False Negative and n = number of samples. Recall and Precision mean, respectively, percentage of instances of one class that are correctly classified and the map accuracy.

We proposed a ranking system to evaluate the inclusion of textural features into the datasets. It is based on accuracy percentage gain (or loss), *Acc*, when a certain feature is added into the dataset:

$$Acc = \frac{Acc_f}{Acc_i} \tag{3}$$

in which *Acc* is the accuracy percentage gain, Acc_i is the initial accuracy and Acc_f is the accuracy when a texture feature is included in the set of features.

Each one of the 45 features were ranked from 1-45, being 1 the feature that presented more percentage gain and so on. This was performed for datasets 1-3 and segmentation of size equal to 30. A final rank considered the average performance. Table 5 shows an example of this ranking for 3 hypothetical features.

Table 5: Ranking system example for 3 hypothetical features.

Dataset #	1	2	3	Average rank	Final Rank
Feature 1 Rank	1st	1st	1st	1,00	1st
Feature 2 Rank	2nd	2nd	3rd	2,33	2nd
Feature 3 Rank	3rd	3rd	2nd	2,66	3rd

3. Results

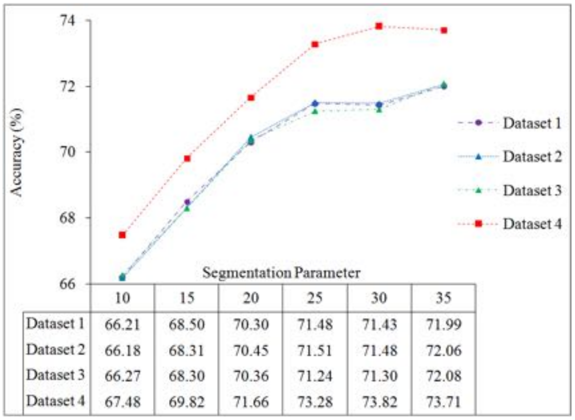
This section presents results obtained from two experiments mentioned in section 2.4.

3.1. Experiment 1: spectral texture features analysis

Figure 7 presents the accuracy classification for all 4 datasets (Table 3) in relation to the segmentation parameter s. We observe that for datasets 1, 2 and 3, the classification values did not presented meaningful difference. That is, the addition of NDVI and SLMM features into the feature set did not improve the classification result. Nevertheless, inclusion of spectral texture features (dataset 4) improved the classification result, which corroborates with Carvalho *et al*. (2010) and

Peneque-Galvez *et al.* (2013). Another observation is that the classification for segmentation parameter equals to 30 presented better result than the others (Figure 7).

In order to better investigate this result, we evaluated Precision and Recall values for each class for the classification for the segmentation parameter of 30 (Table 7). We can observe that Shrubby Savanna (SS) and Shrubby Grassland (SG) classes presented the worst classification. Open Grassland (OG) and Wooded Savanna (WS) presented better precision values, but not as good as the ones for Gallery Forest (GF) class. GF class is the only physiognomy with forest structure and it was expected that it would present better classification accuracy than the others.



	10	15	20	25	30	35
Dataset 1	66.21	68.50	70.30	71.48	71.43	71.99
Dataset 2	66.18	68.31	70.45	71.51	71.48	72.06
Dataset 3	66.27	68.30	70.36	71.24	71.30	72.08
Dataset 4	67.48	69.82	71.66	73.28	73.82	73.71

Figure 7: Accuracy values (%) for datasets 1-4 according to the segmentation parameter.

When spectral texture is added, we noticed an increase in the recall values for all classes, except for GF. The SG class presented the highest precision gains when spectral texture was added. Oliveira *et al*. (2007) pointed out that discrimination between OG and SG classes is difficult. Costa *et al*. (2014) even suggested merging both classes to decrease classification error. Ferreira *et al*. (2007) also reported confusion between SG and SS classes. However, our results show that the use of spectral texture can improve considerably their discrimination.

Table 7. Precision (P) and Recall (R) values for each class from dataset 1 and 4 for the segmentation parameter of 30.

	Open Grassland		Shrubby Grassland		Shrub Savanna		Wooded Savanna		Gallery Forest		
	P	R	P	R	P	R	P	R	P	R	Accuracy
Dataset1	0,761	0,829	0,519	0,411	0,620	0,510	0,761	0,827	0,947	0,941	71,43
Dataset4	0,757	0,866	0,578	0,486	0,666	0,507	0,781	0,842	0,970	0,935	73,82

3.2. Experiment 2: texture features analyses

In this part, we analyze the texture potential of improving the classification in relation to the 5 cerrado classes and s = 30. As mentioned on section Section 2.4, each of the 45 texture features were added separately on datasets 1-3. Final rank of the best features is presented in Table 8.

Table 8. Ranking the 10 best texture features. *Vegetation*, *shadow* and *soil* represent the features obtained from SLMM images.

Feature Name	Average Rank	Final Rank
GLCM Entropy *vegetation*	1,3	1st
GLCM Entropy NDVI	2,3	2nd
GLCM Entropy *shadow*	2,6	3rd
GLCM Entropy *soil*	4,6	4th
GLDV Entropy spectral	9,6	5th
GLCM Contrast *shadow*	10,0	6th
GLCM Contrast spectral	11,0	7th
GLDV Contrast *shadow*	11,0	
GLCM Dissimilarity spectral	11,0	
GLCM Corre *shadow*	11,6	10th

Considering SLMM components and NDVI, the best ranks were achieved by 'GLCM Entropy' features. Homogeneous objects have high entropy values while heterogeneous ones have low entropy. In Cerrado, tree density and canopy formation are responsible for making an object more or less homogeneous.

Figure 8 shows the mean values for the "GLCM Entropy *vegetation*" for the five Cerrado physiognomies. It shows us that, SG and SS classes presented the lowest mean values for 'GLCM Entropy *vegetation*'. Although SG does not have continuous canopy it presents bushes more frequently when compared to OG. This makes SG less homogeneous than OG and, therefore, producing lower entropy vegetation than OG. Regarding to SS class, Ribeiro and Walter (2008) stated that there is a canopy formation, however it is much sparser and with a lower tree cover percentage than the WS class. These vegetation patterns were captured by features such as "GLCM Entropy *vegetation"* and "GLCM Entropy NDVI", which achieved the best rankings (1st and 2nd, respectively).

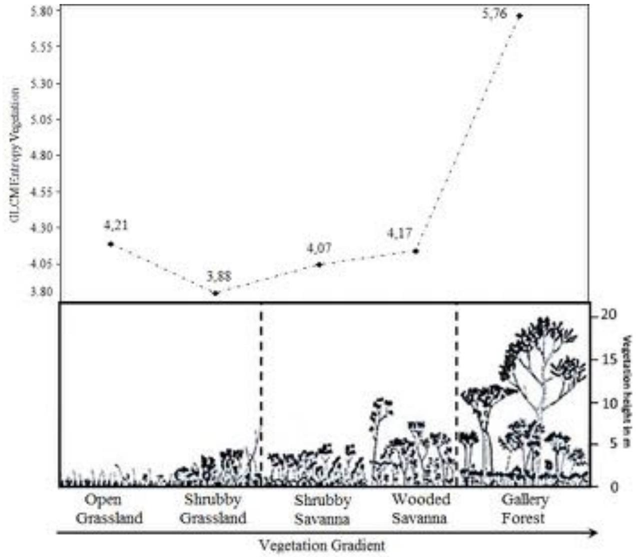


Figure 8: GLCM Entropy *vegetation* mean values for Cerrado physiognomies (adapted from Schwieder *et al.*, 2016).

Table 10 presents Recall and Precision values when "GLCM Entropy *vegetation"*, first feature in the ranking, was added to dataset 3. We noticed a slightly classification improvement for all classes, except for GF Precision. The SG and OG classes presented a little increase in the Recall values. The use of features such as 'GLCM Entropy *vegetation*' and 'GLCM Entropy NDVI' improved the discrimination of both classes, as can be noticed in Recall values. We also observed a little improvement of Recall values for SS class.

Table 10. Precision (P) and Recall (R) for each class with addition of textural entropy.

	Open Grassland		Shrubby Grassland		Shrub Savanna		Wooded Savanna		Gallery Forest	
	P	R	P	R	P	P	R	P	R	P
Dataset3	0,760	0,824	0,516	0,450	0,611	0,504	0,763	0,824	0,958	0,941
+ GLCM Entropy vegetation	0,782	0,864	0,567	0,488	0,649	0,529	0,791	0,846	0,936	0,953

Figure 9 shows an example of how important is to correctly choose the best features to improve the classification accuracy. GLCM Entropy features were much more consistent than the others in all classifications, obtaining always the best ranking. Using some texture features may not really improve the classification results as mentioned by Peneque-Galvez *et al*. (2013).

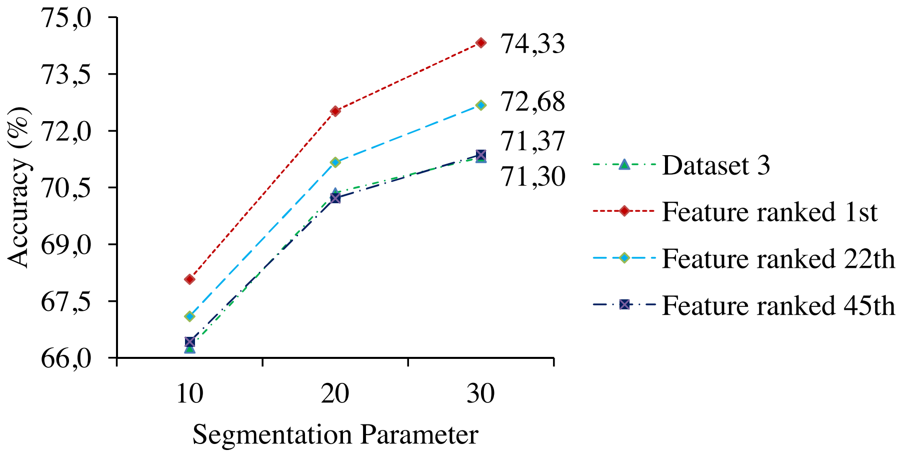


Figure 9. Influence of some features in the final classification accuracy. Feature ranked as 1st is "GLCM Entropy *vegetation"*, 22th is "GLDV Contras NDVI" and 45th is "GLCM Mean *spectral"*.

4. Conclusion

In this study, we presented the assessment of texture features (spectral, NDVI and SLMM) to improve the discrimination of Cerrado physiognomies. Considering only spectral features, the initial accuracy was about 71.3%. The spectral texture improved the classification accuracy to 73.8%. Spectral texture was responsible for reducing the misclassification between grassland physiognomies (Open Grassland and Shrubby Grassland). However, the texture based on GLCM entropy extracted from NDVI and SLMM components, especially vegetation, improved even more the classification accuracy reaching 74.3%. They not only reduced the confusion between grassland physiognomies mentioned before but also increased the discrimination of Shrubby Grassland and Shrubby Savanna. Gallery Forests had high accuracy on all cases. As future works, we suggest using temporal data analysis and combining spectral texture with NDVI and SLMM textures.

5 References

Blackbridge – **Satellite Imagery Product Specifications**, 2015. Available at: www.e-geos.it/images/Satellite_data/RAPIDEYE/RE_Product_Specifications_ENG.pdf. Accessed aug 04. 2016.

Breiman, L. Random forests. **Machine Learning Journal**, v.45, p.5-32, 2001.

Carvalho, L.; Rahman, M.; Hay, G.; Yackel, J. Optical and SAR imagery for mapping vegetation gradients in Brazilian savannas: Synergy between pixel-based and object-based approaches. In: International Conference of Geographic Object-Based Image, 38, 2010, Ghent, Belgium. **Proceedings...** 2010, p.1-7.

Chen, K.; Jacobson, C.; Blong, R. Using NDVI image texture analysis for bushfire-prone landscape assessment. In: Asian Conference on Remote Sensing, 22, 2001, Venue, Singapore. **Proceedings…** 2010. p.9.

Costa, W.S.; Fonseca, L.M.G.; Kosting, T.S. Mapping Grasslands Formations and Cultivated Pastures in the Brazilian Cerrado Using Data Mining. In: GeoProcessing - International Conference on Advanced Geographic Information Systems and Applications, 6, 2014, Barcelona, Spain, P**rocedings…** 2014. p.138-141.

Ferreira, L.G.; Yoshioka, H.; Hueta, A.; Sano, E. E. Seasonal landscape and spectral vegetation index dynamics in the Brazilian Cerrado: An analysis within the Large-Scale Biosphere–Atmosphere Experiment in Amazônia (LBA). **Remote Sensing of Environment**, v.87, n.4, p.534-550, 2003.

Ferreira, M.E.; Ferreira, L.G.; Sano, E. E.; Shimabukuro, Y.E. Spectral linear mixture modeling approaches for land cover mapping of tropical savanna areas in Brazil. **International Journal of Remote Sensing**, v.2, n.28, p.413-429, 2007.

Hall, M.A.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. The WEKA Data Mining Software: An Update. **SIGKDD Explorations**. New York, v.11, n.1, p.10-18, 2009.

Haralick, R.M.; Shanmugam, K.; Dinstein, I. textural features for image classification. **IEEE Transactions on systems, man, and cybernetics**, n.6, p.610-621, 1973.

Klink, C.; Machado, R. Conservation of the Brazilian Cerrado. **Conservation Biology**, v.19, n.3, p.707-713, 2005.

Liesenberg, V.; Ponzoni, F.J.; Galvão, L.S. Análise da dinâmica sazonal e separabilidade espectral de algumas fitofisionomias do Cerrado com índices de vegetação dos sensores Modis Terra e Aqua. **Revista Árvore**, v.31, n.2, p.295-305, 2007.

Ministério do Meio Ambiente (MMA) – **PPCerrado – Plano de ação para prevenção e controle do desmatametno e das queimadas no cerrado: 2ª fase (2014-2015)**,

2014. Avaliable at: http://www.florestal.gov.br/snif/images/Publicacoes/ppcerrado_2fase.pdf. Accessed aug. 04. 2016.

Ministério do Meio Ambiente (MMA) – **Mapeamento do uso e cobertura do Cerrado: Projeto TerraClass Cerrado 2013**, 2015. Avaliable at: http://www.mma.gov.br/publicacoes/biomas/category/62-cerrado. Accessed aug. 05. 2016.

Ning, H.A.N.; Jing, W.U.; Tahmassebi, A.R.S.; XU, H.W.; Ke, W.A.N.G. NDVI-based lacunarity texture for improving identification of torreya using object-oriented method. **Agricultural Sciences in China**, v.10, n.9, p.1431-1444, 2011.

Nussbaum, S.; Menz, G. **Object-based image analysis and treaty verification:** new approaches in remote sensing-applied to nuclear facilities in Iran. Springer Science & Business Media, 2008. 172 p.

Oliveira, L.T.; Oliveira, T.C.A.; Carvalho, L.M.T.; Lacerda, W.S.; Campos, S.R.S; Martinhago, A.Z.. Comparison of machine learning algorithms for mapping Phytophysiognomies of the Brazilian Cerrado. In: Brazilian Symposium on GeoInformatics, 9, 2007, Campos do Jordão, SP. **Anais…**, 2007. p.195-205.

Peneque-Galvez, J.; Mas, J.F.; Moré, G.; Cristobál, J.; Orta-Martinez, M.; Luz, A.C.; Guéze, M.; Macía, M.J.; Reyes-Garcia, V. Enhanced land use/cover classification of heterogeneous tropical landscapes using support vector machines and textural homogeneity. **International Journal of Applied Earth Observation and Geoinformation**, v.23, n.1, p.372-383, 2013.

Ribeiro, J.F.; Walter, B.M.T. As principais fitofisionomias do Bioma Cerrado. In: Sano, S.M.; Almeida, S.P.; Ribeiro, J.F. **Cerrado:** ecologia e flora. Brasília: EMBRAPA, 2008. p.152-212.

Sano, E.E.; Rosa, R.; Brito, J.L.S.; Ferreira, L.G.; Bezerra, H.S. Mapeamento da cobertura vegetal natural e antrópica do bioma Cerrado por meio de imagens Landsat ETM+ In: Simpósio Brasileiro de Sensoriamento Remoto, 14, 2009, Natal, RN. **Anais...**, 2009. p.1199-1206.

Schwieder, M.; Leitão, P.J.; Bustamante, M.M.C.; Ferreira, L.G.; Rabe, A.; Hostert, P. Mapping Brazilian savanna vegetation gradients with Landsat time series. **International Journal of Applied Earth Observation and Geoinformation**, v.52, p.361-370, 2016.

Shimabukuro, Y.E.; Smith, J.A. The least-squares mixing models to generate fractio images derived from remote sensing multispectral data, **IEEE Transactions on Geoscience and Remote Sensing**, v.29, n.1, p.16-20, 1991.

Tucker, C.J. Red and photographic infrared linear combinations for monitoring vegetation. **Remote Sensing of Environment**, v.8, n.2, p.127-150, 1979.

Bayesian network model to predict areas for sugarcane expansion in Brazilian Cerrado

Alexsandro C. de O. Silva, Leila M. G. Fonseca, Thales S. Körting

National Institute for Space Research (INPE)
Post Office Box: 515 – 12227-010 – São José dos Campos – SP – Brazil

{alexsandro.silva, leila.fonseca, thales.korting}@inpe.br

***Abstract**: The growing demand for ethanol has powered the shift of the sugarcane frontier into the Brazilian Cerrado, mainly in the states of Goiás and Mato Grosso do Sul. Therefore, this study aims to propose Bayesian Network models for identifying potential areas for sugarcane expansion in Goiás and Mato Grosso do Sul states. The models take into account constraint factors in relation to the sugarcane expansion such as topography, soil aptitude, climate conditions, and available infrastructures. Results showed that Bayesian Network models proposed in this study were able to represent the tendency of sugarcane expansion.*

1. Introduction

The intense demand for ethanol in the last years has stimulated the sugarcane expansion in new areas, which intensify the competition with other agricultural areas, increase the land prices and reduce the options for further expansions [Adami et al. 2012; Granco et al. 2015]. The saturation of traditional producing areas such as the state of São Paulo [Castro et al. 2010; Shikida 2013] along with the higher production costs for the industry has motivated the producers to search potential new areas for production of sugarcane.

The ethanol industry identified in the Brazilian Cerrado opportunities for investment, especially in the states of Goiás and Mato Grosso do Sul – the Cerrado Biome covers 97% of Goias and 61% of Mato Grosso do Sul [BRASIL 2009]. However, even though favorable climate and soil conditions to the sugarcane cultivation [Shikida 2013] and affordable land prices, few mills were operating in the region. Until 2005, only 22 mills had been established in Goiás and Mato Grosso do Sul, which was a limiting factor for farmers to start planting sugarcane, once the crop needs to be promptly processed after harvesting [Granco et al. 2015].

To enhance the attractiveness for the sugarcane industry the government provided support through fiscal incentives, credit lines and investments in transportation infrastructure. As the result, the number of mills and areas planted to sugarcane in Goias and Mato Grosso do Sul increased approximately three times from 2005 to 2015 [Granco et al. 2015]. Consequently, theses states, which previously had an economy centered on cattle ranching and grains (soybean and corn), witnessed a strong sugarcane expansion [Shikida 2013].

The rapid sugarcane expansion and the eventual land cover changes in the Cerrado led the Brazilian government to implement the Sugarcane Agroecological Zoning to regulate the expansion and sustainable sugarcane production in Brazil, in 2009 [Manzatto et al. 2009]. To identify potential areas for sugarcane crop in the

Cerrado Biome, Ribeiro et al. (2015) carried out Boolean spatial analysis. However, Boolean analysis results only two classes (favorable and non-favorable), which do not adequately represent the spatial phenomena. Other spatial inference method for spatial data integration, such as Bayesian inference, can numerically express the potential of sugarcane areas from 0 to 1, which allows obtaining a decision surface (Moreira et al. 2000).

Within this context, this study aims to propose a Bayesian inference model for identifying potential areas for sugarcane expansion in the states of Goiás and Mato Grosso do Sul. We used a Bayesian Networks approach: the enhanced Bayesian Network for Raster Data (e-BayNeRD) method [Silva et al. 2014], which is an enhanced version of the BayNeRD algorithm [Mello et al. 2013]. The e-BayNeRD method is a probabilistic approach based on raster data observations and it is able to incorporate experts' knowledge for analysis. In the next section, we present a brief description about the theory of Bayesian Networks and the e-BayNeRD method employed in this study.

2. Bayesian Network Model

Bayesian Networks (BN) are defined in terms of two components: (i) qualitative component – a Directed Acyclic Graph (DAG), in which the nodes represent the variables in the model and the statistical dependence between pair wise variables is indicated by directed arrows that start in a parent node and end in a child node, as illustrate on Figure 1; and (ii) quantitative component – probability functions associated to each variable denoting the strengths of the links in the BN model [Aguilera et al. 2011; Landuyt et al. 2013].

Figure 1 shows an example of BN graphical model, in which *Suitable Area* variable is statistically dependent of *Soil Aptitude* variable and both are statistically dependent of *Terrain Slope* variable. Prior probability is assigned to variable without parent (e.g.: *Terrain Slope*), whereas conditional probability is assigned to descendant ones (e.g.: *Soil Aptitude* and *Suitable Area*).

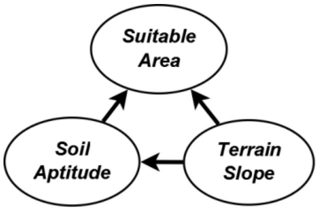


Figure 1: Example of Bayesian Network graphical model.

The prior knowledge of an event is updated taking into account new evidence through the Bayes' theorem:

$$P(A = a \mid B = b) = \frac{P(B = b \mid A = a)P(A = a)}{P(B = b)},$$

in which $P(A = a)$ is the prior probability of the event A; $P(B = b \mid A = a)$ is the likelihood function and $P(A = a \mid B = b)$ is the posterior probability; and $P(B = b)$ is a normalizing constant [Neapolitan 2004]. Upper-case letters denote the variables and the same but lower-case letters denote the state or value of the variable. This ability to compute posterior probabilities given new evidence is called inference.

Bayesian Networks Model has been used in many applications employing Bayesian inference to develop a plausible reasoning and to describe the occurrence probability of a phenomenon. Aguilera et al. (2011) argue that the BN have mainly been used as a technique for inference in Environmental Sciences. Dlamini (2010) presented a BN approach to estimate the probability of wildfire occurrence based on satellite-detected wildfires data and a set of social, physical, environmental and climatic factors. McCloskey et al. (2011) proposed a BN to identify suitable sites for the economic development and landscape conservation, while Gonzalez-Redin et al. (2016) used BN approach to find areas for a sustainable timber production and biodiversity conservation. Mello et al. (2013) developed a BN model for raster data analysis to study soybean mapping based on remote sensing variables.

However, Aguilera et al. (2011) reported that few studies have used methods based on BN in agriculture applications. Therefore, we expected that this study contribute to show the potentiality of BN models, which are used to infer potential areas for sugarcane expansion in this study.

2.1. e-BayNeRD algorithm

To identify the appropriate areas for sugarcane expansion, we used e-BayNeRD method [Silva et al. 2014], which is briefly described below. Figure 2 shows the e-BayNeRD's workflow.

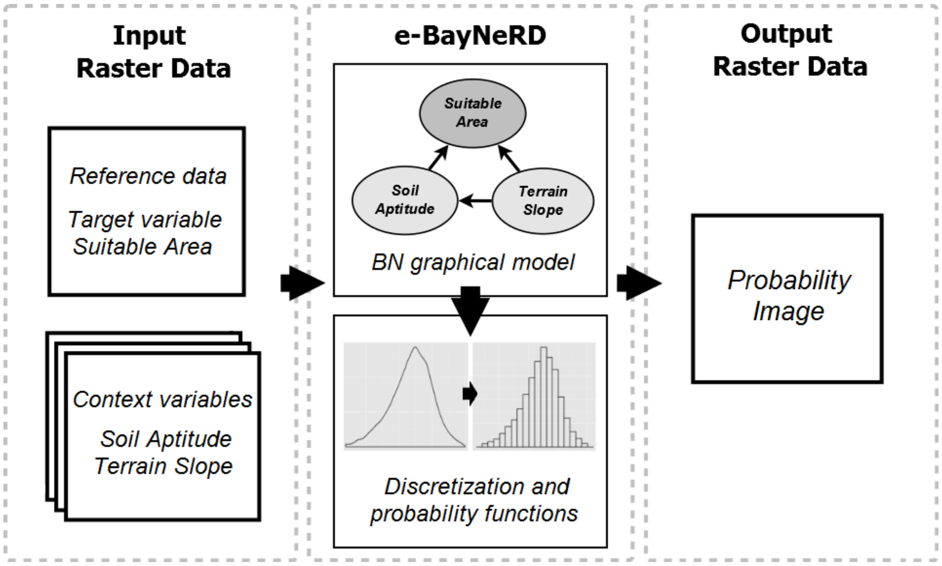
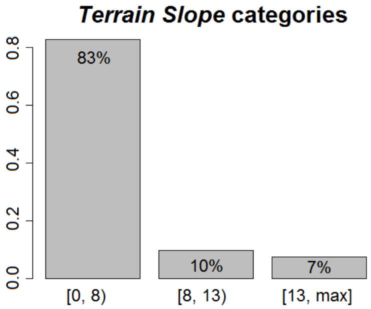


Figure 2. e-BayNeRD's Workflow.

The method deals with raster data in GeoTiff format, in which each GeoTiff corresponds to a variable (node) in the BN model. The variable that represents the studied phenomenon is called *target variable* and its GeoTiff must contain the reference data for training. To illustrate, suppose that *Suitable Area* is the target variable in the BN model exemplified in Figure 1. The others variables in the model are called *context variables,* and they can have some relation with the target one and they can have some interrelation. After entering all the raster data, the user designs the BN graphical model (i.e., the DAG) by defining the relations among all variables.

Once the BN graphical model is defined, user needs to convert continuous context variables into categorical ones. In the discretization phase, the range of observed values for each variable is divided into intervals according to the lower and upper limits chosen by the user. Each interval is one category; therefore, a variable will have n categories if user discretizes it into n intervals. Figure 3 illustrates the *Terrain Slope* variable discretized in three categories: [0, 8), [8, 13), [13, max]. Values into the bars indicate the probability to find a pixel within the defined interval limits. The limits of the intervals for each variable should be appropriately chosen to describe as best as possible the variable according to the phenomenon studied.



Conditional probability table of *Soil Aptitude*

Soil Aptitude	*Terrain Slope* (in %)		
	[0, 8)	**[8, 13)**	**[13, max]**
High	0.7135	0.5241	0.4972
Moderate	0.0269	0.0737	0.0695
Low	0.2596	0.4022	0.4333

Figure 3. Example of the discretization phase and computed conditional probability table.

In the e-BayNeRD algorithm, probability functions are computed based on pixel counting according to the dependence relations among variables and their categories. Prior probability is assigned to those variables without parents (e.g.: *Terrain Slope* variable), whereas conditional probability is assigned to descendant ones. Figure 3 illustrates the conditional probability table associate to the *Soil Aptitude* variable, in which *High*, *Moderate* and *Low* are categories for *Soil Aptitude*. Values in table are conditional probabilities $P(Soil\ Aptitude\,|\,Terrain\ Slope)$, which indicate the probability of finding a pixel equal to some *Soil Aptitude* category given that this pixel is equal to some *Terrain Slope* category.

After compute the probability functions associated with each variable, e-BayNeRD is able to calculate the probability of target presence given the values observed in the context variables. When the probability is computed for each pixel in the study area, the output, called Probability Image, is formed.

3. Study area

The study area covers Goiás and Mato Grosso do Sul states in the Brazilian Cerrado biome excluding some regions of no interest for sugarcane expansion, as showed in Figure 4. Some areas excluded in study are: urban areas, water bodies, areas under environmental protection laws (i.e. conservation units, indigenous lands, the Upper Paraguay River basin, Pantanal Biome), and areas where it is not allowed commercial sugarcane production such as agrarian settlements or quilombo communities.

Figure 4 shows also cultivated sugarcane areas in Goiás and Mato Grosso do Sul for 2008/2009 crop year (dark green color). These regions were excluded from the study area because they were also excluded in the Sugarcane Agroecological Zoning, which was used as a reference data for training and assessing the model for infer potential areas for sugarcane expansion proposed in this work.

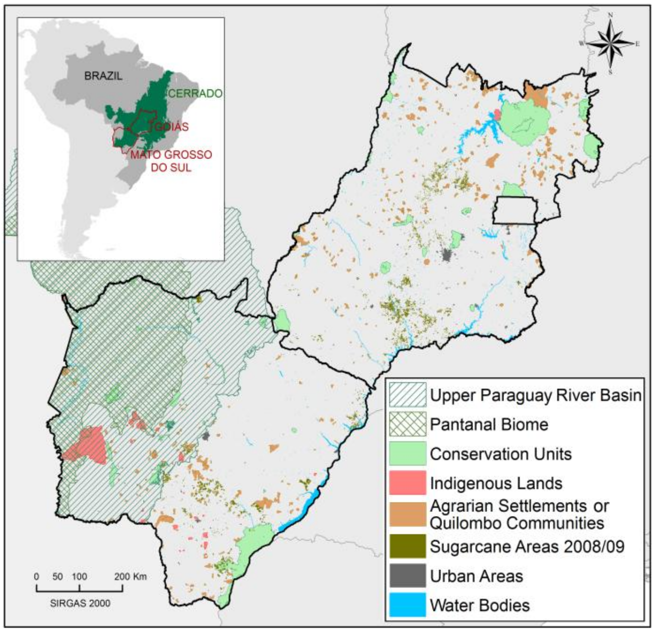


Figure 4. States of Goiás and Mato Grosso do Sul and masked out areas from the study.

4. Selected variables to build the BN model

This section presents the variables chosen to compose the BN model. The *target variable* represents appropriate areas for sugarcane expansion and the *context variables* are constraining factors that exhibit any kind of relation with the sugarcane expansion.

4.1. Target variable

Most of the suitable areas for sugarcane expansion in Brazil are located in Goiás and Mato Grosso do Sul states [Manzatto et al. 2009]. Figure 5 shows the *target variable Suitable Area* for sugarcane expansion according to the Sugarcane Agroecological Zoning (Source: http://geo.cnpma.embrapa.br/). Suitable and non-suitable areas are denoted by green and yellow colors, respectively. About 70% of the pixels in each class were randomly selected to compose the reference data for training. The remaining 30% were used for accuracy assessment.

4.2. Context variables

Considering that Goiás and Mato Grosso do Sul states have topography adequate for sugarcane expansion [Shikida 2013], we selected *Terrain Slope* variable as a context variable in the model. *Terrain Slope* variable was settled as parent of *Suitable Area* variable in the BN model. (*Terrain Slope* variable source: http://srtm.csi.cgiar.org/).

Soil in our study area presents favorable conditions for sugarcane cultivation although there are some poor-nutrient areas as stated by [Shikida 2013]. As soil conditions influence sugarcane cultivation, we set *Soil Aptitude* variable as parent of *Suitable Area* variable. Considering that terrain slope is an important factor to determine the aptitude to some crop cultivation, we set *Soil Aptitude* variable as descendant of *Terrain Slope* variable. (*Soil Aptitude* variable source: http://mapas.mma.gov.br/i3geo/datadownload.htm).

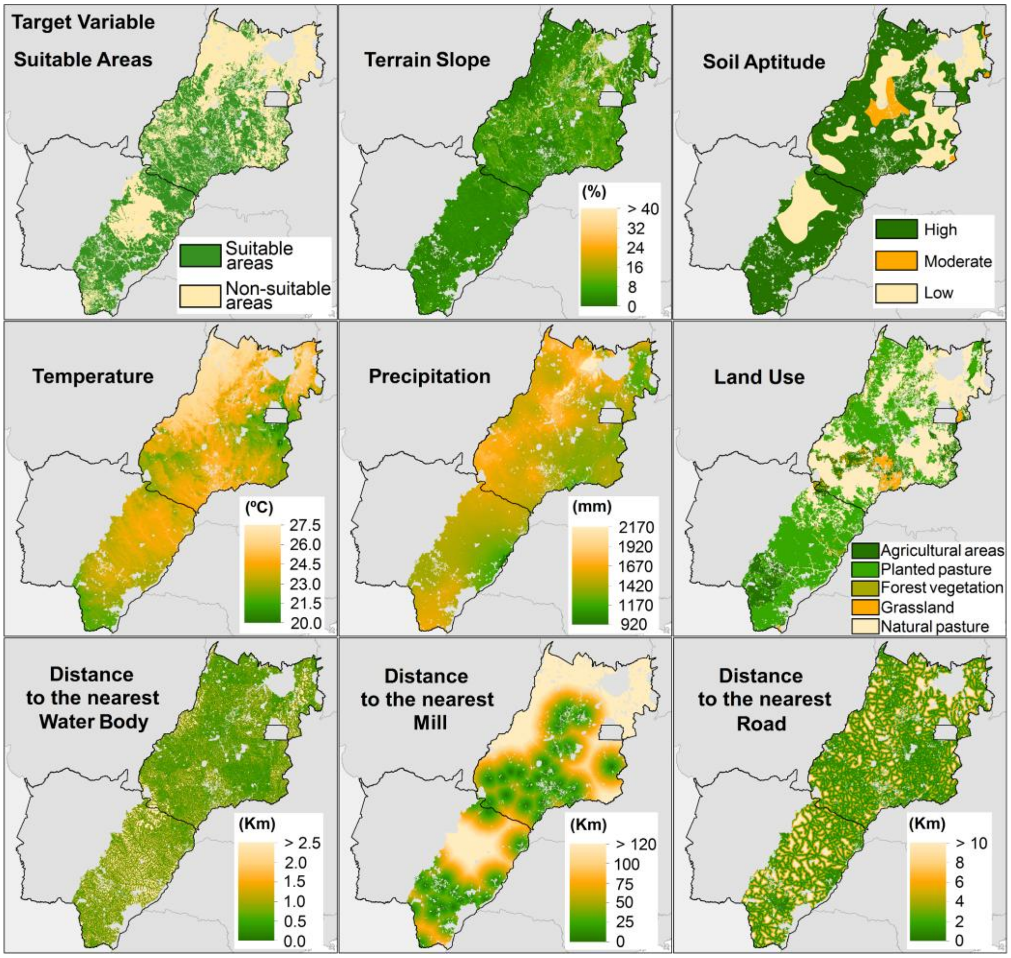


Figure 5. GeoTiff data of target and context variables.

Furthermore, it is important to consider climate risks to cultivate sugarcane [Brunini et al. 2008]. In this sense, the climate data *annual mean Temperature* and *annual Precipitation* were also selected as context variables. As these variables indicate suitable climate conditions regions for sugarcane cultivation, we set them as parent of *Suitable Area* variable in the BN model (*Temperature* and *Precipitation* variables source: http://www.worldclim.org/).

Sugarcane expansion in the South-central region of Brazil has occurred mainly on pasturelands and eventually on annual crops [Adami et al. 2012; Castro et al. 2010]. Taking this account, the land use maps produced by IBGE (12 classes) were rasterized to represent the *Land Use* variable. Land use classes were grouped in five classes: agriculture; planted pasture; forest; grassland; and natural pasture. Agriculture, planted and natural pasture represent 11%, 45% and 40% of the land use, respectively. Considering that each land use class sets limits to the sugarcane expansion, *Land Use* variable is considered as *Suitable Area* variable's parent. We define *Land Use* variable as descendent of *Terrain Slope* and *Soil Aptitude* variables (*Land Use* variable source: http://portaldemapas.ibge.gov.br/).

Another context variable is the *Distance to the nearest Water Body*, which was computed using the drainage network. This variable was selected to take into account conservation of natural vegetation around water bodies/rivers. This suggests that sugarcane cultivation does not occur in areas close to water bodies. Therefore, *Distance to the nearest Water Body* variable is set as parent of *Suitable Area* variable. It is noteworthy that small distances from water bodies are generally associated with higher slope. Therefore, *Distance to the nearest Water Body* variable is also descendent of *Terrain Slope* variable (Drainage network source: http://portaldemapas.ibge.gov.br/).

Installation of new mills was also a driver for sugarcane expansion in Goiás and Mato Grosso do Sul states, usually located close to the roads to facilitate the production transport [Granço et al. 2015]. Hence, the *Distance to the nearest Mill* and *Distance to the nearest Road* were selected as context variables, and they were built from highways network and points of mills location, respectively. For logistical and economic reasons, it is expected that the sugarcane expansion occurs near mills and roads. Thus, the *Distance to the nearest Mil* and *Distance to the nearest Road* variables can be defined as parents of the *Suitable Area* variable in BN model (Highways network source: http://portaldemapas.ibge.gov.br/ and points of mills location source: http://ctbe.cnpem.br/pesquisa/producao-biomassa/cana-info/).

4.3. BN graphical models and discretization phase

Two BN models were created. The first one identifies sustainable areas for sugarcane expansion taking into account the variables *Terrain Slope*, *Soil Aptitude*, *Precipitation*, *Temperature*, *Land Use* and *Distance to the nearest Water Body*. The second one specifies logistically appropriate areas as well, considering the variables *Distance to the nearest Mill* and *Distance to the nearest Road*. Additionally, the *Probability Image* produced by the first BN model was incorporated into the second BN model as a context variable. Figure 6 shows the first and second BN models.

The limits of the intervals should be appropriately chosen to describe as best as possible the context variable according to the target variable *Suitable Area*. Table 1 presents the interval limits and the categories defined for context variables, which were chosen based on our knowledge and experience.

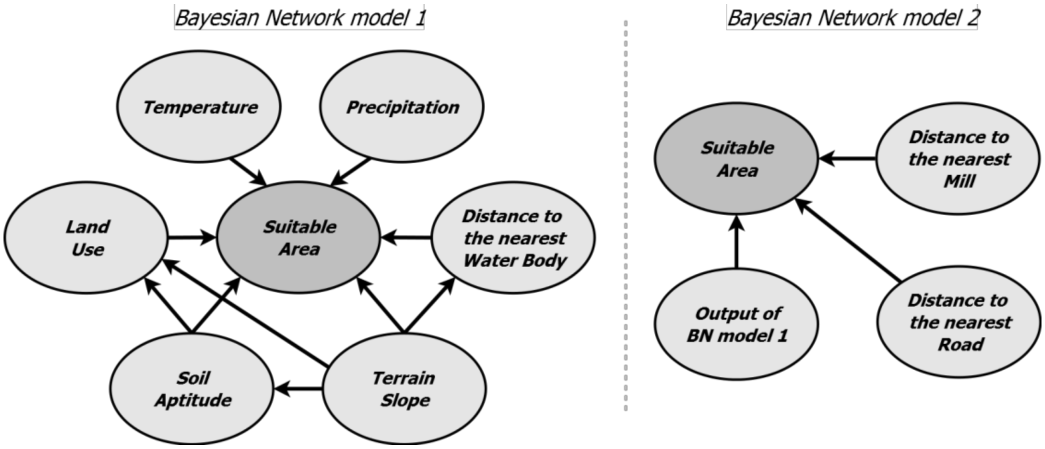


Figure 6. BN model 1 (left): identify sustainable areas; BN model 2 (right): identify sustainable and logistically appropriate areas.

Table 1. Interval limits and categories defined for each context variable.

Context variable	*Terrain Slope* (%)	*Soil Aptitude*	*Land Use*	*Temperature* (ºC)
	[0, 8)	High	Agriculture	[20, 23.6)
	[8, 13)	Moderate	Planted pasture	[23.6, 24.9)
	[13, max]	Low	Forest	[24.9, max]
	-	-	Grassland	-
	-	-	Natural pasture	-
Context variable	*Precipitation* (mm)	*Distance to the nearest Water Body* (km)	*Distance to the nearest Mill* (km)	*Distance to the nearest Road* (km)
	[920, 1600)	[0, 0.25)	[0, 25)	[0, 1)
	[1600, max]	[0.25, 0.5)	[25, 50)	[1, 2)
	-	[0.5, 1)	[50, max]	[2, 5)
	-	[1, max]	-	[5, max]

5. Results and Discussion

Probability Image is the main e-BayNeRD outcome, in which each pixel value represents the probability of such area be a sugarcane growing area given the observed context variables values. The Probability Images produced by both BN models are shown in Figures 7.

Green colored pixels in Figure 7 represent areas with high probability for sugarcane expansion. High probability values were achieved when context variables exhibited good conditions for sugarcane plantation: *Terrain Slope* < 8%; high *Soil Aptitude*; *Temperature* < 24.9 ºC; *Distance to the nearest Water Body* > 1km; agriculture or grassland as *Land Use*; *Distance to the nearest Mill* < 50km; *Distance to the nearest Road* < 5km; and *Precipitation* had no much influence. This result is similar to those obtained by Ribeiro et al. (2015). This result can be explained by two factors: agricultural aptitude and logistical issues.

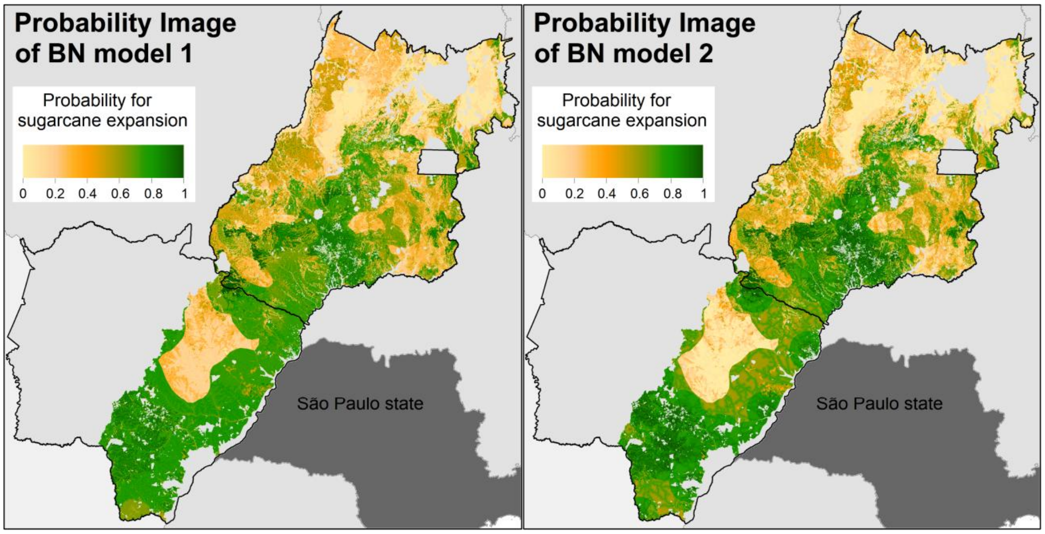

Probability Image of BN model 1
Probability for sugarcane expansion
0 0.2 0.4 0.6 0.8 1
São Paulo state
Probability Image of BN model 2
Probability for sugarcane expansion
0 0.2 0.4 0.6 0.8 1
São Paulo state

Figure 7. Probability Images of: BN model 1 (left) and BN model 2 (right). Green colored pixels represent areas with high probability for sugarcane expansion.

In according to Castro et al. (2010) and Silva and Miziara (2011), there is sugarcane expansion tendency in the south-central of Goiás state. Areas in the south of Goiás present appropriate conditions for sugarcane cultivation. The Sugarcane Agroecological Zoning [Manzatto et al. 2009] pointed out that most areas in Goiás have conditions for sugarcane expansion. Our BN models showed high probabilities for sugarcane expansion in the south-central of Goiás. Silva and Miziara (2011) argue that sugarcane spatial dynamics is directly related to installation of new mills. Indeed, our results showed a similar spatial distribution between the mills locations (context variable *Distance to the nearest Mill* in Figure 5) and the potential areas for sugarcane expansion. This distribution is known as the south-central expansion axis [Castro et al. 2010], which can be observed in our results showed in Figure 7.

On the other hand, sugarcane expansion is more restricted in the north of Goiás, where *Terrain Slope* (above 13%) and *Soil Aptitude* (low) are not favorable. Results also presented low probability values in the west and east of Goiás. In this case, the limiting factor in the modeling was high *annual Temperature* in the west and high *Terrain Slope* and low *Soil Aptitude* in east.

Regarding the state of Mato Grosso do Sul, Probability Images in Figure 7 show that more suitable regions for sugarcane expansion are located in the eastern and southern regions, which are near to São Paulo state, the main domestic market for ethanol. This result is in according to those presented by Manzatto et al. (2009) and Ribeiro et al. (2015). Mato Grosso do Sul state presents favorable agricultural and environmental conditions for sugarcane cultivation. However, low probability values in the central region of the state occurred mainly due to low *Soil Aptitude* in this region. Pasture represents most land use in Mato Grosso do Sul, but the highest probability values in Mato Grosso do Sul were achieved in agriculture areas in the southern of the state.

In addition to the good agriculture and climate conditions in the Mato Grosso do Sul state, it is important to observe that most mills are concentrated in the southern state (context variable *Distance to the nearest Mill* in Figure 5). It is possible to note that potential areas for sugarcane expansion stated more specific in the result of second BN model, which considered the *Distance to the nearest Mill* and *Distance to the nearest Road* context variables. Mills distribution in the southern of Mato Grosso do Sul facilitates the logistics and transportation of ethanol, since this region has better road infrastructure connecting it with São Paulo state [Granco et al., 2015].

5.1. BN models assessment

We analyzed the Probability Images (Figure 7) for sugarcane areas only in 2012/2013 crop year, which correspond to 955.100 hectares approximately, according to Canasat project [Rudorff et al. 2010]. Sugarcane was monitored in the South-central region of Brazil by the Canasat Project from 2005 to 2013 [Rudorff et al. 2010].

Probability Images can be used to indicate the best regions for sugarcane expansion. Based on these probabilities, we sliced the range of probability values in aptitude categories for sugarcane expansion, as illustrated in Figure 8. We verified that the expansion has mostly occurred in areas in which BN models considered as high probability of sugarcane cultivation. For BN model 1, about 75% of sugarcane areas have probability ≥ 0.7; for BN model 2, about 85% of sugarcane areas have probability

≥ 0.7. These results indicates that second BN model, which infer about sustainable and logistically appropriate areas for sugarcane expansion, it is better than the first BN model.

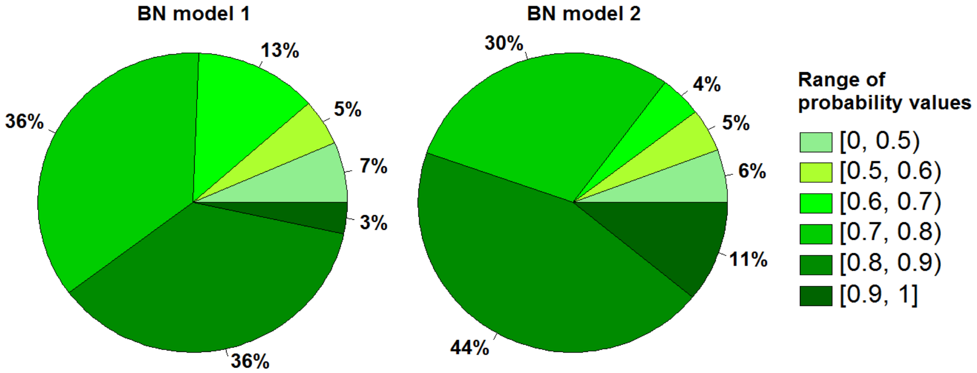


Figure 8: Percentage of sugarcane areas in the 2012/2013 crop year according to the range of probability values for each BN model.

Two complementary indices are commonly used in binary classifications: sensitivity, which indicates true positives rate (target areas labeled as target), and specificity, which indicates true negatives rate (non-target areas labeled as non-target). As BN models output a predicted probability, several thematic maps were produced by varying the probability threshold from 0 to 1. For example, if threshold = 0, all pixels are classified as suitable area. If all pixels that are suitable for sugarcane cultivation are correctly classified, then the sensitivity value is equal to 100%. On the other hand, pixels that are not suitable but are misclassified lead to specificity value equals to 0%.

The Receiver Operating Characteristic (ROC) curve [Hanley and McNeil 1982] is a graph that shows performance of a binary classifier in terms of two indices. ROC curve graph is plotted with sensitivity (Y-axis) versus 1-specificity (X-axis). To generate the ROC curve it is necessary to plot sensitivity versus 1-specificity for all possible classification thresholds. The upper left corner is the best point, where both indexes are equal to 100%.

Figure 9 shows ROC curve for both BN models. In ROC curve, points plotted above diagonal represent a classification better than random (random guess). This means that BN models did a good job separating suitable areas and non-suitable areas classes. In the Probability Image for BN model 1, threshold = 0.45 resulted in a point closest to the upper left corner with sensibility = 84% and specificity = 67%. In the BN model 2, threshold = 0.51 resulted in a point with sensibility = 85% and specificity = 72%.

Area under the curve (AUC) is used to quantify BN model performance. AUC is the percentage of area under the ROC curve [Fawcett 2006]. Therefore, points in the ROC curve closest as possible to the upper left corner represent the best accuracy. AUC for BN model 1 and BN model 2 was 82% and 84%, respectively. This indicates that the BN models are able to differentiate suitable area and non-suitable area classes.

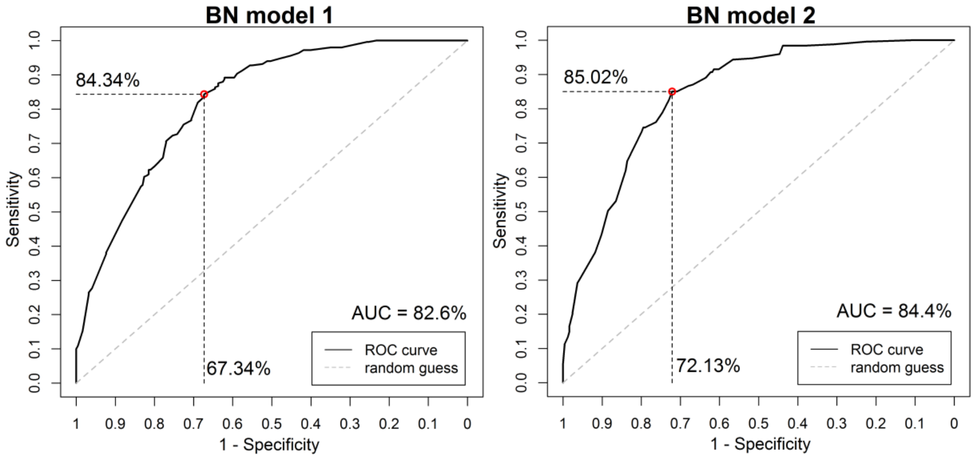


Figure 9. ROC curve of the first (left) and second (right) BN model.

6. Conclusion

This work used Bayesian Network method based on raster data observations to identify adequate areas for sugarcane expansion in the Brazilian Cerrado in the states of Goiás and Mato Grosso do Sul. Most of new sugarcane areas in recent crop year occurred in regions in which the model assigned to them high probability values (> 70%). This indicates that the Probability Image output of BN models can be used to direct the sugarcane expansion to most suitable areas. Bayesian Network model was able to distinguish suitable and non-suitable areas for sugarcane expansion and it can be used as a planning decision tool.

In general, the study area presented favorable agro-environmental conditions for sugarcane plantation. Among all constraining factors, *Terrain Slope* and *Soil Aptitude* variables were the main drivers for sugarcane expansion. The highest probability values for expansion were achieved in agricultural areas and grassland. Future research should consider only pasture areas for sugarcane expansion, since there is a concern about competition with other agricultural areas; and also compare the e-BayNeRD method with others similar algorithms.

Acknowledgements

The authors thank to CAPES for financial support; and the reviewers for their valuable comments and inputs.

References

Adami, M. et al. 2012. “Remote Sensing Time Series to Evaluate Direct Land Use Change of Recent Expanded Sugarcane Crop in Brazil.” *Sustainability* 4(12): 574–85.

Aguilera, P. A. et al. 2011. “Bayesian Networks in Environmental Modelling.” *Environmental Modelling & Software* 26(12): 1376–88.

BRASIL, Meio Ambiente. 2009. “Conheça Os Biomas Brasileiros.” http://www.brasil.gov.br/meio-ambiente/2009/10/biomas-brasileiros (August 10, 2016).

Brunini, O. et al. 2008. *Zoneamento de Culturas Bioenergéticas No Estado de São Paulo: Aptidão Edafoclimática Da Cultura Da Cana-de-Açúcar*.

Castro, S. S. et al. 2010. "A Expansão Da Cana-de-Açúcar No Cerrado E No Estado de Goiás: Elementos Para Uma Análise Espacial Do Processo." *Boletim Goiano de Geografia* 30(1): 171–91.

Dlamini, W. M. 2010. "A Bayesian Belief Network Analysis of Factors Influencing Wildfire Occurrence in Swaziland." *Environmental Modelling & Software* 25(2): 199–208.

Fawcett, T. 2006. "An Introduction to ROC Analysis." *Pattern Recognition Letters* 27(8): 861–74.

Gonzalez-Redin, J. et al. 2016. "Spatial Bayesian Belief Networks as a Planning Decision Tool for Mapping Ecosystem Services Trade-Offs on Forested Landscapes." *Environmental Research* 144: 15–26.

Granco, G. et al. 2015. "Exploring the Policy and Social Factors Fueling the Expansion and Shift of Sugarcane Production in the Brazilian Cerrado." *GeoJournal*.

Hanley, J a, and B J McNeil. 1982. "The Meaning and Use of the Area under a Receiver Operating (ROC) Curvel Characteristic." *Radiology* 143(1): 29–36. http://www.ncbi.nlm.nih.gov/pubmed/7063747.

Landuyt, D. et al. 2013. "A Review of Bayesian Belief Networks in Ecosystem Service Modelling." *Environmental Modelling & Software* 46: 1–11.

Manzatto, C. V. et al. 2009. Embrapa Solos *Zoneamento Agroecológico Da Cana-de-Açúcar*. Rio de Janeiro.

McCloskey, J. T. et al. 2011. "Using Bayesian Belief Networks to Identify Potential Compatibilities and Conflicts between Development and Landscape Conservation." *Landscape and Urban Planning* 101(2): 190–203.

Mello, M. et al. 2013. "Bayesian Networks for Raster Data (BayNeRD): Plausible Reasoning from Observations." *Remote Sensing* 5(11): 5999–6025.

Moreira, F. R. et al. 2000. "Inferência Geográfica E Suporte À Decisão." In *Introdução À Ciência Da Geoinformação*, eds. G. Câmara, C. A. Davis Junior, and A. M. V. Monteiro. São José dos Campos, 345. http://www.dpi.inpe.br/gilberto/livro/introd/cap9-inferencia.pdf.

Neapolitan, R. 2004. *Learning Bayesian Networks*. 2nd ed. New Jersey: Person Prentice Hall.

Ribeiro, N. V. et al. 2015. "Padrões E Impactos Ambientais Da Expansão Atual Do Cultivo Da Cana-de-Açúcar : Uma Proposta Para O Seu Ordenamento No Bioma Cerrado." *Ateliê Geográfico* 9(2): 99–113.

Rudorff, B. F. T. et al. 2010. "Studies on the Rapid Expansion of Sugarcane for Ethanol Production in São Paulo State (Brazil) Using Landsat Data." *Remote Sensing* 2(4): 1057–76.

Shikida, P. F. A. 2013. "Expansão Canavieira No Centro-Oeste: Limites e Potencialidades." *Revista de Política Agrícola* XXII(2): 122–37.

Silva, A. A. and Miziara, F. 2011. "Avanço Do Setor Sucroalcooleiro E Expansão Da Fronteira Agrícola Em Goiás." *Pesquisa Agropecuária Tropical* 41(3): 399–407.

Silva, A. C. O. et al. 2014. "Enhancements to the Bayesian Network for Raster Data (BayNeRD)." In *Proceedings of XV Brazilian Symposium on Geoinformatics*, eds. Clodoveu Augusto Davis Jr and Karine Reis Ferreira. Campos do Jordão: Proceedings of the XV Symposium on GeoInformatics - GEOINFO, 73–82.

Big data streaming for remote sensing time series analytics using MapReduce

Luiz Fernando Assis[1], Gilberto Ribeiro[1], Karine Reis Ferreira[1], Lúbia Vinhas[1], Eduardo Llapa[1], Alber Sanchez[1], Victor Maus[1], Gilberto Câmara[1]

[1]Image Processing Department
INPE - National Institute for Space Research
Av dos Astronautas 1758
Caixa Postal Sao Jose dos Campos – SP – Brazil

{luizffga,gribeiro,karine,lubia,edullapa,alber.ipia,gilberto}@dpi.inpe.br

Abstract. *Governmental agencies provide a large and open set of satellite imagery which can be used to track changes in geographic features over time. The current available analysis methods are complex and they are very demanding in terms of computing capabilities. Hence, scientist cannot reproduce analytic results because of lack of computing infrastructure. Therefore, we propose a combination of streaming and map-reduce for time series analysis of time series data. We tested our proposal by applying the classification algorithm BFAST to MODIS imagery. Then, we evaluated account computing performance and requirements quality attributes. Our results revealed that the combination between Hadoop and* ***R*** *can handle complex analysis of remote sensing time series.*

1. Introduction

Currently, there is huge amount of remote sensing images openly available, since many space agencies have adopted open access policies to their repositories. This large data sets are a good chance to broaden the scope of scientific research that uses Earth observation (EO) data. To support this research, scientists need platforms where they can run algorithms that analysis big Earth observation data sets. Since most scientists are not data experts, they need data management solutions that are flexible and adaptable.

To work with big EO, we need to develop and deploy innovative knowledge platforms. When users want to work with hundreds or thousands of images to do their analysis, it is not practical to work with individual files at their local disks. Innovative platforms should allow scientists to perform data analysis directly on big data servers. Scientists will be then able to develop completely new algorithms that can seamlessly span partitions in space, time, and spectral dimensions. Thus, we share the vision for big scientific data computing expressed by the late database researcher Jim Gray: *"Petascale data sets require a new work style. Today the typical scientist copies files to a local server and operates on the data sets using his own resources. Increasingly, the data sets are so large, and the application programs are so complex, that it is much more economical to move the end-user's programs to the data and only communicate questions and answers rather than moving the source data and its applications to the user's local system"* [Gray et al. 2005].

For instance, the standard for land use and land cover monitoring includes to select and download a set of images, processing of each one using visual interpretation or semi-automatic classification methods, to delineate the areas of interest. This approach is ineffective when there are too much data, or for example, when working on large extensions of land using high spatio-temporal resolution. In contrast to analyzing one image at a time, time-series analysis had become a valuable alternative in land use/land cover monitoring, including early warning of deforestation [Verbesselt et al. 2012a]. Although, we lack environments for validating and reproducing the analysis results of large remote sensing data [Lu et al. 2016, Maus et al. 2016]. To avoid this problem, streaming analytics have emerged as a solution by combining fast access, scalable storage and easy deployment for complex analysis. This approach is able to analyze data in near real-time with low latency and to point to events in regional and global scales without overhead.

Sensor and location-based social networks are common data sources analysis of spatial data in near real-time. Since these network users generate petabytes of data, they are provided through streaming APIs which have several applications, including the analysis the occurrence of events [Assis et al. 2015, Schnebele et al. 2014]. Unlike these streaming APIs, parallel streaming processing plug-ins deal with I/O interpreters in a more intuitively by allowing a powerful and flexible way to analyze data. Hadoop[1] and SciDB streaming [2] are APIs that gather large amounts of data from a file system and multidimensional database such as Hadoop and SciDB respectively. Specifically Hadoop streaming has the advantage of using a standard processing model called MapReduce, which optimized for specific features with different degrees of conformance to the model [Urbani et al. 2014, Dede et al. 2014].

However, most of the MapReduce-based approaches only provide an image library [Sweeney et al. 2011] by means of a customization, which is limiting for analysis. Besides only a small variety of analysis methods are provided at a instance and new complex algorithms are costly to develop and reproduce [Almeer 2012]. Furthermore, most of the available methods extract land use and land cover information using region-based classifications, even though they may cause loss of information [Giachetta and Fekete 2015]. For these reasons, a flexible, generic and broad solution is required to reuse remote sensing time series analysis methods, avoiding the burden of development and adaptation according to the scientific needs.

Therefore, we propose a combination of distributed file systems and complex analysis environments in a MapReduce streaming processing analytics. It is implemented as <*key*, *values*> pairs, where *key* is an image pixel location and *values* is the time series associated to that given location. We evaluated this approach, using the BFAST algorithm that iteratively estimates the time and number of abrupt changes within time series, and characterizes change by its magnitude and direction [Verbesselt et al. 2010]. We use BFAST to detect and characterize changes in time series of MODIS (Moderate Resolution Imaging Spectroradiometer) data [Rudorff 2007]. Briefly, the main contributions of this work are:

1. To present a time series-based streaming processing analytics using MapReduce;
2. To discuss the learned lessons from a case study to evaluate our approach in terms of performance and quality requirements;

[1] https://hadoop.apache.org/docs/r1.2.1/streaming.html

[2] https://github.com/Paradigm4/streaming

The remainder of this paper is structured as follows. Section 2 presents a discussion about the time-first, space-later vs space-first, time-later analysis. Section 3 describes the related works while Section 4 outlines our approach using MapReduce for remote sensing time series. Section 5 depicts the evaluation of our approach and its results. Section 6 concludes this paper with recommendations for future works.

2. Time-first, Space-later vs Space-first, Time-later

Scientists have analyzed time series of remote sensing imagery, to detect changes, in three different ways: 1) process each image independently and compare the results for different time instances, 2) build time series of each pixel and process them independently and 3) develop algorithms that process multiple pixels at multiple time instances . The first type of analysis will be called hereinafter as *space-first, time-later* approach. This type of analysis aims to evaluate and compare the results of a pixel classification independently in time. For example, if more than one method of an image classification based on forest cover percentage (see Figure 1) are applied, a pixel may be classified in distinct land cover types. The error resulted in one of them can lead the results to a classification inconsistency when analyzing the pixels of each scene separately. Also, this inconsistency may also increase with the number of scenes and leading to an analysis mistake depending on the application.

Due to this limitation, scientists have used an alternative approach in which the methods are based on what we define as *time-first, space-later* approach. The key is to consider the temporal auto-correlation of the data instead of the spatial auto-correlation [Eklundha and Jönssonb 2012], which is really important for remote sensing time series analysis. In this case, scientists analyze each pixel independently taking into consideration all the values of the pixel along the time (see Figure 2).

For example, given a set S = $\{s_1, s_2, ..., s_n\}$ of remote sensing satellite imagery that depicts the same region at n-consecutive times, we can define them as a 3-D-dimensional array in space-time. For each digital image $s_i \in S$, millions of pixels are associated with their respective spatial location (*latitude*, *longitude*), which corresponds to the (x, y, z) position in a 3D matrix. The z-component of the matrix corresponds to the time axis in the satellite imagery. Each pixel location (x, y, z) contains a set A = $\{a_1, a_2, ...a_m\}$ of attributes values, represented by spectral bands of the set of images. These attributes can provide land-use and land-cover information as each kind of target (forest, water, soil, among others) on the ground has a different spectral reflectance signatures based on the wavelength.

Time-first, space-later approach is more suitable, for example, to detect deforestation or forest degradation from time series of remote sensing imagery. Supposing that we are working with images that have an spectral attribute a that is associated to the forest cover. We can think of a situation in which an area was a prestine forest until 2000, it was cut out in 2001 and started to regenerate in 2010. If we follow the value of a along the time, using the time-series complex analytics, we can monitor this dynamics. If we consider large databases of imagery, with high spatial and temporal resolutions and covering large extensions we will need the best and robust methods to deal with the big EO data. The streaming processing analytics approach presented in this paper, is a contribution to fulfill this demand.

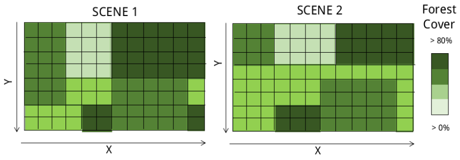


Figure 1. Space First, Time Later

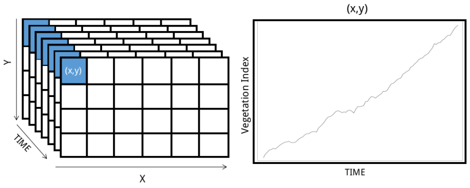


Figure 2. Time First, Space Later

3. Related Works

Due to the increasing interest on EO applications, a set of additional mechanisms have emerged to load, process and analyze remote sensing imagery. These mechanisms aim to convert the images into different data formats since storage components sometimes only accepts a specific representation. Analytic algorithms have been built to enrich existing storage components with more statistical and mathematical operations, but they still lag far behind statistical software packages such as those presented in the CRAN repository. In order to reduce the data movement and the communication overhead between storage and analysis, integrating these storage components and **R** by letting each do what they do best is still a better approach. This combination aims to scale for analytic methods over massive datasets by exploiting the parallelism of storage components in an analyst-friendly environment [Integrating 2011]. The problem about this integration is that a sophisticated understanding of their particular characteristics are mandatory and functionalities need to be re-implemented. For these reasons, data should be acquired, processed and analyzed continuously in an easily and flexible manner in near real-time.

For this, location-based social networks streams analytics have been emerged as the most common approaches in the literature provided by means of APIs. Most of the existing studies that use these streamings aim to provide location-based eventful visualization, statistical analysis and graphing capabilities [Schnebele et al. 2014]. They also aim to explore the spatial information involved in social networks messages. For example, social network messages can be used to detect events in near real-time such as floods and elections [Assis et al. 2015, Song and Kim 2013]. The challenge here is in the combination of different data flows and data formats to support the analysis of high value social network messages in near real-time. In distributed parallel processing, streaming APIs[3][4] have been mainly used to perform an arbitrary set of independent tasks that can be broken into parts, and run separately in another environment with a reusable code. It takes into consideration input/reading and output/writing commands by using stdin and stdout.

Hadoop Streaming is an exemplary API that has an advantage of using MapReduce, a standard processing model, to process in near real-time by customizing how input and output are splitted into key/value pairs. One of the most important features of this open implementation is that Hadoop is fault-tolerant. Its main goal is to support the execution of tasks using a scalable cluster of computing nodes [Rusu and Cheng 2013]. Hadoop-GIS, MD-HBase and SpatialHadoop are exemplary GIS tools that require an extra overhead for more flexible functions [Aji et al. 2013, Nishimura et al. 2013,

[3] https://hadoop.apache.org/docs/r1.2.1/streaming.html

[4] https://github.com/Paradigm4/streaming

Eldawy and Mokbel 2015]. Unlike dedicated proprietary services such as Google Earth Engine that offer minimal standards for scientific collaboration, alternative interfaces of Hadoop can abstract highly technical details for image processing from the point of view of computer vision [Sweeney et al. 2011].

However, when a large amount of analytics algorithms are necessary, these approaches burden the developers and scientists since there is a clearly limitation of available operations and functions, mainly regarding remote sensing time series analysis. Furthermore, existing studies address this approach with a more spatial focus in image classification algorithms [Almeer 2012, Giachetta and Fekete 2015], which result in more loss of information. For these reasons, the high technical complexities involved in developing new applications should be hide from them, and consequently, a more flexible and generic approach is required.

4. Streaming Processing Analytics using MapReduce

Since remote sensing time series analytics require dealing with a large amount of satellite imagery of the same place at different times, it is necessary to build an approach that provides a fast access, a scalable storage and more flexible complex analysis methods. This makes easier to other scientists to reproduce and validate scientific research on this topic. With this in mind, we propose an approach that combines a streaming processing mechanism based on MapReduce with a complex statistical analysis environment. These choices were made based on the flexibility offered by the existing streaming processing that allows the implementation of algorithms in different languages, as well the several analysis components provided by these environments with specific purpose. At first, we stored all the images in a distributed file system so that they are processed by means of two methods (Mapper and Reducer) aiming to build the timeline values and analyze them calling a complex algorithm.

The main advantage of using a standard processing model such as MapReduce is in the fact that both methods receive and transmit data as <*key*, *values*> pairs, giving to the scientists more interoperability and clear capacity of processing data. In our approach, the Mapper input is a <*key*, *values*> pair, in which the *key* is an image identifier and the *values* are all of the desired pixel locations (x,y), that is, the image content itself. The Mapper is responsible for extracting the features from the images for each desired pixel, transforming them into a time series data and emit them to the Reducer. The Mapper output is a <*key*, *values*> pair, in which the *key* is a pixel location (x,y) and the *values* are time series data (e.g., x = 10, y = 45, values = ”0.5 0.7 0.4 0.6” are represented as a <(10, 45), (0.5 0.7 0.4 0.6)> pair). As the Mapper output is the Reducer input, the Reducer receives the combination of pixel and time series values, and analyze them by means of a complex method. The result in this case is stored in the distributed file system. A high level architecture of this time series-based streaming processing analytics for remote sensing data can be seen in Figure 3.

4.1. Data Model and Storage

As a distributed file system is able to store any data type and format without any restriction, its schema-on-read approach offers a more adequate design for our case. Unlike schema-on-write approaches such as database management systems that require a predefined schema to store and query the data, schema-on-read approaches lead to load raw

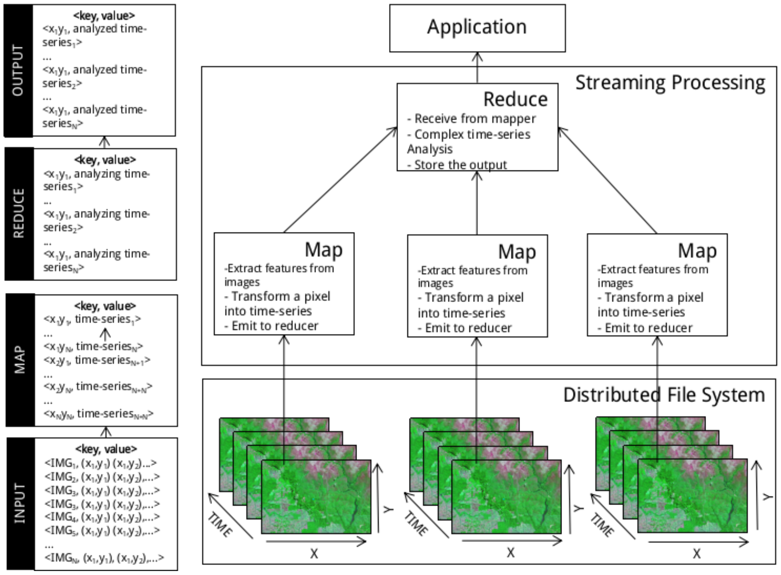

OUTPUT
<key, value>
<x_1y_1, analyzed time-series$_1$>
...
<x_1y_1, analyzed time-series$_2$>
...
<x_1y_1, analyzed time-series$_N$>
REDUCE
<key, value>
<x_1y_1, analyzing time-series$_1$>
...
<x_1y_1, analyzing time-series$_2$>
...
<x_1y_1, analyzing time-series$_N$>
MAP
<key, value>
<x_1y_1, time-series$_1$>
...
<x_1y_N, time-series$_N$>
<x_2y_1, time-series$_{N+1}$>
...
<x_2y_N, time-series$_{N+N}$>
...
<x_Ny_N, time-series$_{N+N}$>
INPUT
<key, value>
<IMG$_1$, (x_1,y_1) (x_1,y_2)...>
<IMG$_2$, (x_1,y_1) (x_1,y_2),...>
<IMG$_3$, (x_1,y_1) (x_1,y_2),...>
<IMG$_3$, (x_1,y_1) (x_1,y_2),...>
<IMG$_4$, (x_1,y_1) (x_1,y_2),...>
<IMG$_5$, (x_1,y_1) (x_1,y_2),...>
...
<IMG$_N$, (x_1,y_1), (x_1,y_2),...>
Application
Streaming Processing
Reduce
- Receive from mapper
- Complex time-series Analysis
- Store the output
Map
-Extract features from images
- Transform a pixel into time-series
- Emit to reducer
Map
-Extract features from images
- Transform a pixel into time-series
- Emit to reducer
Map
-Extract features from images
- Transform a pixel into time-series
- Emit to reducer
Distributed File System
TIME
Y
X
TIME
Y
X
TIME
Y
X

Figure 3. MapReduce Streaming Analytics Processing

and unprocessed data with a structure based on a versatile processing according to the applications requirements. As a result, data not previously accessible are interpreted as it is read, that is, scientists learn the data over time in near real-time. The distributed file system enables the storage of binary files such as raster and shapefiles. Additional tools can help scientists organizing the data either defining a structure or not around their data. In our case, the images gathered by the satellites are stored into years in a sequence it was processed by the provider so that it makes easier to build the time series.

4.2. MapReduce Programming Model

The MapReduce programming model consists of two methods responsible for extracting the features from the images and processing the complex algorithms for remote sensing time series applications in a independently and reusable manner. Both Mapper and Reducer methods receive their input and output by means of standard input (*stdin*) and standard output *stdout* as <key, values> pairs. Unlike other approaches, the <key, values> pairs are line oriented and processed as it arrives, since the Mapper and Reducer controls the processing. In this work, the Mapper performs the filtering and sorting of both pixel and the attributes values into lines, while Reducer performs the complex analysis and stores the result.

An informal high-level description of Mapper can be seen in the Algorithm 1. At first, the Mapper get the dataset names for standardized stored images before creating raster layer objects for them according to the spectral band id chosen by the scientists. The input is a *<IMG, (x_1,y_1), (x_1,y_2), ..., $(x_{(}n),y_{(}n)$)>* pair, where *IMG* is an identifier for each image and the latter is a list of pixel coordinates to be analyzed. At second, the Mapper builds the time series by getting the values for each pixel. In this part, the scientist define the pixel interval and get the values for each pixel of them. For example, for an entire image, the scientist would define the interval from 1 to 23040000 (4800x4800 - MODIS data resolution). At third, the Mapper calculate the pixel by ceiling the number of the pixel divided by the image resolution for the row and getting the remainder for the col. Lastly, the Mapper emit the time series built to the Reducer.

Algorithm 1 Transform <key, values> input into a intermediate <key, values>

```
procedure MAPPER
    connection ← openFile(”stdin”, open ← ”readbynary”)
    while length(path ← readLines(connection) do
        files ← insert(files, openDirectory(path))
    end while
    closeFile(connection)
    for i←1 to length(files) do
        r[i] ← raster(getDatasets(files[i])[bandId])
    end for
    for pixel←beginInterval to endInterval do
        initialize(values)
        for j←1 to length(files) do
            values ← concatenate(values, getValues(r[j], row←ceiling(j/imageRes), col←remainder(j/imageRes))
        end for
        emit(”stdout”, pixel, values)
    end for
end procedure
```

On the other hand, the Reducer receives each <(x,y), *time series*> pair as an input, so that (x, y) is a pixel coordinate and the *time series* are the attributes found in a pixel of an image for a spectral band defined. Similar to the Mapper, the Reducer get the dataset names for standardized files before creating the *time series*. Then, it adapts the time series format as an input for the complex analysis. Finally, the Reducer emit the output as the result of the complex analysis by storing them into the distributed file system (see Algorithm 2).

Algorithm 2 Transform <key, values> from Mapper into output <key, values>

```
procedure REDUCER
    connection ← openFile(”stdin”, open ← ”readbynary”)
    while length(line ← readLines(connection)) do
        timeseries ← getTimeSeries(line)
        ts ← preProcess(timeseries)
        analysis ← complexAnalysis(ts)
        emit(”stdout”, pixel, analysis)
    end while
    closeFile(connection)
end procedure
```

5. Evaluation and Results

5.1. Experimental Setup

Runtime Environment: The experiments were run on a single-node computer with Intel(R) Core(TM) i7-5500U CPU @ 2.40GHz and 16GiB GB RAM memory running Ubuntu 14.04.4 LTS (64 bit).

***Dataset*:** The MODIS scientific instruments launched in the Earth's orbit by NASA in 1999 were used in our experiments since they are able to capture 36 spectral bands ranging in wavelength from 0.4 μ m to 14.4 μ m. They are designed to provide measures description of the land, oceans and the atmosphere that can be used for studies of processes on local to global scales. In our case, we considered the MOD13Q1 Normalized Difference Vegetation Index (NDVI) due to the large amount of remote sensing studies that have focused on time series analysis using this index [Verbesselt et al. 2010, Grogan et al. 2016]. Since MODIS data are provided every 16 days at 250-meter spatial resolution in the Sinusoidal projection and has more than 18,000 satellite images covering Brazil from 2000 to 2016, we built a time series only using a fraction of these data regarding time and space (92 images with 21 Giga Bytes in total).

5.2. Application Case Study: Deforestation Detection

For handling remote sensing imagery as MODIS time series, at first we organized the MODIS data into years. This organization enables us to build an infrastructure able to extract, transform and load all the images by converting them into standard input for the desired methods. In this work, we considered a method, that is part of an **R** package called BFAST, that aims to detect iteratively breaks in seasonal and trend components of a time series [Verbesselt et al. 2011]. This package is not only helpful for deforestation and phenological change detection, but also for forest health monitoring [Verbesselt et al. 2012b]. After running BFAST for a specific pixel (latitude=-10.408, longitude=-53.495), we obtained a breakpoint in 01-17-2011 (see Figure 4). As this processing can be performed for a large amount of other pixels, we are not considering here to check the accuracy of such algorithm. Our focus in this work is on presenting how these kind of analysis can be validate by using a high variety of systems. For example, the deforestation detection in this pixel situated in the state of Mato Grosso in Brazil (see Figure 5) can be seen in DETER[5], a system for deforestation detection in near real-time. The problem here is in the distinct date of breakpoint found when using both sources (BFAST and DETER).

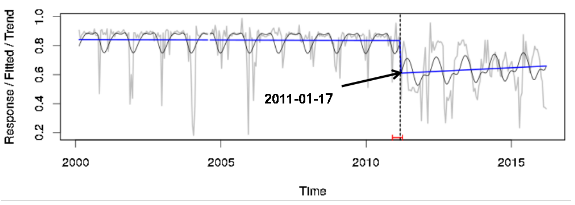


Figure 4. BFAST for a NDVI time series (latitude=-10.408, longitude=-53.495).

In our approach, we decided to integrate Hadoop and **R** since we were able to take the best of massively scalable capabilities and research-friendly programming environment of complex analytics. For evaluating this integration, we performed a set of experiments by using BFAST and other R packages to see how this integration behaves in terms of processing time and scalability (varying the amount of pixel and images). Our tests also allowed us to see how the overhead of these tools affected this kind of processing. The results are shown in Figure 6 for four different amount of images consisting of

[5] http://www.obt.inpe.br/deter/

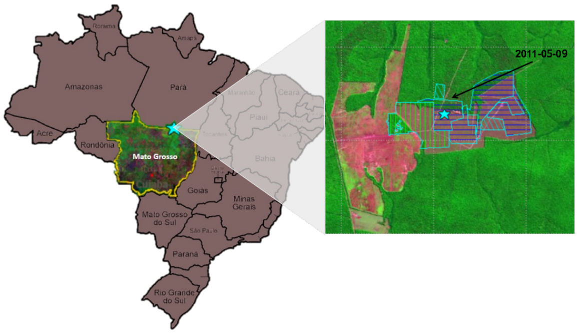


Figure 5. Deforested Area in the state of Mato Grosso in Brazil (latitude=-10.408, longitude=-53.495).

one, two, three and four year MODIS time series data. As we can see, the integration between Hadoop and **R** has a stable, adequate and linear performance even when the amount of information increase with the time. The limitation of the performance is upon to the hardware infrastructure, that is, an extension of the hardware capabilities would provide a better performance in terms of storage and computation power. By comparison, for each thousand of pixels, an amount of 6000 seconds is necessary to analyze using a complex algorithm such as BFAST. The flexibility of running complex algorithms using the familiarity of an **R** script overcome the high cost related to the learning curve of Hadoop. The reason is that in **R** is easy to install and load new packages and a high variety of complex algorithms can be easily deployed.

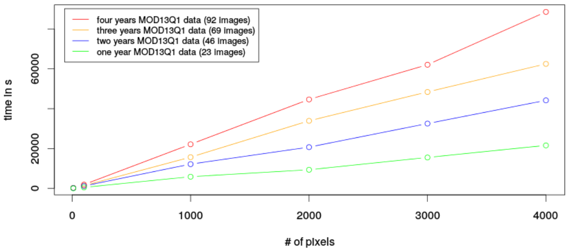


Figure 6. Processing Time to apply BFAST to different amount MOD13Q1 images using MapReduce.

We also calculated the output size files in bytes produced by BFAST in the MapReduce programming model (see Table 1). As we can see, the variation of the image amount change few the size of the output using an algorithm such as BFAST. On the other hand, as the amount of pixel increase the size of the output increase proportionally. The output files contain the timestamps when the break of the time series were detected for each pixel.

In addition, we deployed similar packages in R aiming to detect breaks in time series since they can also be applied to remote sensing time series applications. We considered R packages that help to perform behavioral change point analysis (*bcpa*), change point detection methods (*changepoint*), structural changes detection in regression models (*strucchange*) and behavioral change detection in several other applications (*BreakoutDetection*). The processing time spent for each algorithm is almost the same and can be seen

Table 1. Size Files in Bytes of MapReduce output to apply BFAST

	23 images	46 images	69 images	92 images
10 pixels	171	171	171	171
100 pixels	1792	1792	1783	1750
1000 pixels	18881	18868	18820	18662
2000 pixels	38863	38859	38771	38290
3000 pixels	58849	58844	58722	58107
4000 pixels	78827	78819	78675	77694

in Figure 7. In this experiment, we vary the amount of pixel to a smaller scale compared to the previous one.

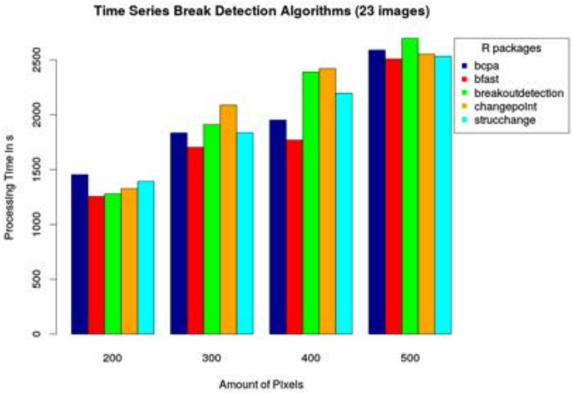


Figure 7. Processing Time to apply several other R packages aiming to detect breaks using 23 images.

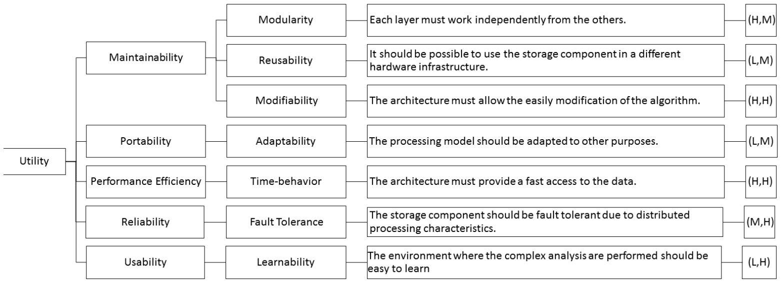
5.3. Quality Architectural Requirements

According to [Pressman 2005], external quality architectural requirements correspond to the attributes of the systems that can be recognized by users and are important for design evaluation, which includes performance, flexibility, portability, reusability, interoperability, etc. In this work, we aim to use a qualitative evaluation of these attributes with the main purpose of generating results that can respond whether the designed system meets the architecture quality requirements of domain specialists. For example, decide whether the performance of the software fail or not to compromise the previously planned information processing time.

The chosen method is an adaptation of the most used scenario-based evaluation by industry, also known as Architecture Trade-off Analysis Method (ATAM). ATAM considers how the goals interact with each other in an achieved balance between desirable and compatible features aiming to provide an adequate detail about architectural documents [Nord et al. 2003]. This method guide all the stakeholders to search for conflicts in the architecture, and consequently, solve them. In Table 2 we list the quality attributes found in each architectural decisions. In Figure 8 is depicted the quality attributes in terms of ISO/IEC 25010. We also aim to highlight the level of how hard is to implement each of them and how important they are to the application domain (H: high; M: medium; L: low).

Table 2. List of architectural decisions

Id	Architectural Decision	Quality Attributes	Description
D1	Distributed File System	Performance Fault-Tolerance Reusability	The file system provide fast access to unstructured data in a properly, continuously and reusable operating manner
D2	MapReduce processing model	Modifiability Adaptability	The programming model is easily modifiable for different purposes
D3	Multilayered Architectural	Modularity	The storage, processing and analysis occur in several layers by means of decoupling
D4	Complex Analysis Environment	Learnability	The complex analysis environment should be easy to learn



Each layer must work independently from the others. (H,M)

It should be possible to use the storage component in a different hardware infrastructure. (L,M)

The architecture must allow the easily modification of the algorithm. (H,H)

The processing model should be adapted to other purposes. (L,M)

The architecture must provide a fast access to the data. (H,H)

The storage component should be fault tolerant due to distributed processing characteristics. (M,H)

The environment where the complex analysis are performed should be easy to learn (L,H)

Figure 8. Utility tree.

6. Conclusions

Complying with the memory limitations of the **R**, data scientists often have to restrict their analysis only to a subset of the data. Integrating technologies such as Hadoop with **R** language offer not only a strategy to overcome its memory challenges of large data sets, but also provides a more flexibility programming of complex analysis in storage components. This paper presents an approach for analyzing big remote sensing time series in near real-time using a processing model known as MapReduce.

Our results guide the processing analytics streaming approaches as a more generic way in terms of performance and capacity. They highlighted that for different amount of pixels, and MODIS time series (one, two, three and four years), the processing time was linear for complex algorithms such as those found in deforestation detection applications. Exemplary situations in which such algorithms are important were demonstrated for a specific region in Brazil. Future works will comprise studies about alternative approaches that perform streaming analytics processing in other sources of information such as SciDB, a multidimensional array database. We also plan to evaluate this approach in a multi-node cluster experiment focusing more on data, memory and CPU intensive tests. The Spark framework is also a promising and efficient approach to be tested in our approach.

7. Acknowledgments

Gilberto Camara, Luiz Fernando Assis and Alber Sanchez are supported by São Paulo Research Foundation (FAPESP) e-science program (grants 2014-08398-6, 201515/19540-0 and 2016-03397-7). Gilberto is also supported by CNPq (grant 312151-2014-4). Eduardo Llapa is also supported by a BNDES Fundo Amazonia grant.

References

Aji, A., Wang, F., Vo, H., Lee, R., Liu, Q., Zhang, X., and Saltz, J. (2013). Hadoop GIS: A High Performance Spatial Data Warehousing System over Mapreduce. *Proc. VLDB Endow.*, 6(11):1009–1020.

Almeer, M. H. (2012). Hadoop mapreduce for remote sensing image analysis. *International Journal of Emerging Technology and Advanced Engineering*, 2(4):443–451.

Assis, L. F. F. G., Herfort, B., Steiger, E., Horita, F. E. A., and ao Porto de Albuquerque, J. (2015). Geographical prioritization of social network messages in near real-time using sensor data streams: an application to floods. In *XVI Brazilian Symposium on Geoinformatics (GEOINFO)*.

Dede, E., Fadika, Z., Govindaraju, M., and Ramakrishnan, L. (2014). Benchmarking mapreduce implementations under different application scenarios. *Future Generation Computer Systems*, 36:389–399.

Eklundha, L. and Jönssonb, P. (2012). Timesat 3.1 software manual. Technical report, Lund University.

Eldawy, A. and Mokbel, M. F. (2015). SpatialHadoop: A MapReduce framework for spatial data. *Data Engineering (ICDE), 2015 IEEE 31st International Conference on*, 1:1352–1363.

Giachetta, R. and Fekete, I. (2015). A case study of advancing remote sensing image analysis. *Acta Cybernetica*, 22:57–79.

Gray, J., Liu, D. T., Nieto-Santisteban, M., Szalay, A., DeWitt, D. J., and Heber, G. (2005). Scientific data management in the coming decade. *SIGMOD Rec.*, 34(4):34–41.

Grogan, K., Pflugmacher, D., Hostert, P., Verbesselt, J., and Fensholt, R. (2016). Mapping clearances in tropical dry forests using breakpoints, trend, and seasonal components from modis time series: Does forest type matter? *Remote Sensing*, 8(8):657.

Integrating, R. (2011). Bridging two worlds with rice. *Proceedings of the VLDB Endowment*, 4(12).

Lu, M., Pebesma, E., Sanchez, A., and Verbesselt, J. (2016). Spatio-temporal change detection from multidimensional arrays: Detecting deforestation from {MODIS} time series. *{ISPRS} Journal of Photogrammetry and Remote Sensing*, 117:227–236.

Maus, V., Câmara, G., Cartaxo, R., Sanchez, A., Ramos, F. M., and de Queiroz, G. R. (2016). A time-weighted dynamic time warping method for land-use and land-cover mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*.

Nishimura, S., Das, S., Agrawal, D., and El Abbadi, A. (2013). Md-hbase: design and implementation of an elastic data infrastructure for cloud-scale location services. *Distributed and Parallel Databases*, 31(2):289–319.

Nord, R. L., Barbacci, M. R., Clements, P., Kazman, R., and Klein, M. (2003). Integrating the architecture tradeoff analysis method (atam) with the cost benefit analysis method (cbam). Technical report, DTIC Document.

Pressman, R. S. (2005). *Software engineering: a practitioner's approach*. Palgrave Macmillan.

Rudorff, B. F. R. (2007). *Sensor Modis e Suas Aplicações Ambientas no Brasil*. Editora Parêntese.

Rusu, F. and Cheng, Y. (2013). A survey on array storage, query languages, and systems. *arXiv preprint arXiv:1302.0103*.

Schnebele, E., Cervone, G., Kumar, S., and Waters, N. (2014). Real time estimation of the calgary floods using limited remote sensing data. *Water*, 6(2):381–398.

Song, M. and Kim, M. C. (2013). Rtˆ2m: Real-time twitter trend mining system. In *Proceedings of the 2013 International Conference on Social Intelligence and Technology*, pages 64–71.

Sweeney, C., Liu, L., Arietta, S., and Lawrence, J. (2011). Hipi: A hadoop image processing interface for image-based mapreduce tasks. Technical report, University of Virginia.

Urbani, J., Margara, A., Jacobs, C., Voulgaris, S., and Bal, H. (2014). Ajira: a lightweight distributed middleware for mapreduce and stream processing. In *Distributed Computing Systems (ICDCS), 2014 IEEE 34th International Conference on*, pages 545–554. IEEE.

Verbesselt, J., Hyndman, R., Zeileis, A., and Culvenor, D. (2010). Phenological change detection while accounting for abrupt and gradual trends in satellite image time series. *Remote Sensing of Environment*, 114(12):2970–2980.

Verbesselt, J., Zeileis, A., and Herold, M. (2011). Near real-time disturbance detection in terrestrial ecosystems using satellite image time series: Drought detection in somalia. Technical report, Faculty of Economics and Statistics, University of Innsbruck.

Verbesselt, J., Zeileis, A., and Herold, M. (2012a). Near real-time disturbance detection using satellite image time series. *Remote Sensing of Environment*, 123:98–108.

Verbesselt, J., Zeileis, A., Hyndman, R., and Verbesselt, M. J. (2012b). Package 'bfast'.

Assessment of a Multi-Sensor Approach for Noise Removal on Landsat-8 OLI Time Series Using CBERS-4 MUX Data to Improve Crop Classification Based on Phenological Features

Hugo N. Bendini[1], Leila M. G. Fonseca[2], Thales S. Körting[2], Rennan F. B. Marujo[2], Ieda D. Sanches[1], Jeferson S. Arcanjo[2]

[1]Divisão de Sensoriamento Remoto – Instituto Nacional de Pesquisas Espaciais (INPE)
[2]Divisão de Processamento de Imagens – INPE
Caixa Postal 515 – 12.227-010 – São José dos Campos – SP – Brazil

hbendini@dsr.inpe.br, {leila.fonseca,thales.korting, ieda.sanches}
@inpe.br, jeferson@dpi.inpe.br

Abstract. *We investigated a method for noise removal on Landsat-8 OLI time-series using CBERS-4 MUX data to improve crop classification. An algorithm was built to look to the nearest MUX image for each Landsat image, based on user defined time span. The algorithm checks for cloud contaminated pixels on the Landsat time series using Fmask and replaces them with CBERS-4 MUX to build the integrated time series (Landsat-8 OLI+CBERS-4 MUX). Phenological features were extracted from the time series samples for each method (EVI and NDVI original time series and multi-sensor time series, with and without filtering) and subjected to data mining using Random Forest classification. In general, we observed a slight increase in the classification accuracy when using the proposed method. The best result was observed with the EVI integrated filtered time series (78%), followed by the filtered Landsat EVI time series (76%).*

1. Introduction

Given the large availability of arable land, and the growing demand for food in the world, Brazil has been consolidated as a big player on the global agricultural scene. Remote sensing is an important tool used within agriculture, regarding its ability to generate information on a large scale in a cost-effective way. In this way, agricultural mapping has become strategic enabling to provide better understanding of the distribution of croplands, and its impact on the environment. With advances in data processing and storage technologies as well as the availability of consistent and continuous long-term image series, remote sensing is undergoing a paradigm shift. Time series techniques stand out for allowing seasonal variation accounts of the analyzed target. Although the use of time series for cropland classification has been well explored using MODIS data (Sakamoto et al., 2005; Arvor et al., 2011; Körting, 2012; Risso et al., 2012; Borges & Sano, 2014; Neves et al., 2016), there is still a demand for more detailed maps, which are made possible from time series with finer spatial resolutions, such as Landsat-like images (Zheng et al., 2015; Peña et al., 2015; Pan et al., 2015; Bendini et al., 2016). As the temporal resolution of Landsat-like satellites it is still low (16 days, generally), an open question in the scientific literature is about how to deal with the noise in the time series. The noise is characterized by negative outliers, which are possibly a result of factors such as cloud cover, cloud shadow contamination and atmospheric scattering. To deal with this, there are some approaches which include cloud and cloud shadow flags generated from the Automated Cloud Cover Assessment (ACCA) algorithm (Irish et al., 2006) and Fmask algorithm (Zhu & Woodcock, 2012). However, both ACCA and Fmask sometimes fail to detect thin clouds i.e. cirrus and the edges of cumulus clouds (Lymburner et al., 2016) and thus sometimes can be followed by methods based on the use of thresholds (Hamunyela et al., 2013; Bendini et al., 2016; Lymburner et al., 2016) or on the use of smoothers (Pan et al., 2015). There is also the possibility to take advantage of multi-sensor data, considering the large amount of available remote sensing data. In a previous investigation, we show the potential use of higher temporal resolution Landsat-like images for crop mapping (Bendini et al., 2016). Recently the China Brazil Earth Resources Satellite (CBERS) program launched the CBERS-4 that carries in the payload module, among others, the Multispectral Camera (MUX). In this study, we investigated a method for noise removal on Landsat-8 OLI time-series using CBERS-4 MUX data to improve a crop classification method based on phenological features.

2. Materials and Methods

2.1. Study area

The study area is situated in Sao Paulo state (southeast of Brazil), in a region located into the Cerrado biome (Figure 1). As the focus is on croplands, we selected a region of interest where main land cover is agriculture, silviculture and pasture. In this region, farmers grow a variety of crops throughout the year. Major field crops in this area are sugarcane, corn, bean, potato, soybean, sugar beet and onions. There is also production of mango, avocado and eucalyptus. Farmers grow crops in double cropping systems and even in triple cropping systems, mainly within the irrigated areas. The usual planting for summer crops occurs from October to December and harvesting from February to April. We also observed the planting of crops in late fall (May – July) and harvesting in the next spring, especially within the irrigated areas.

2.2. Remote Sensing Data

A total of 23 scenes of Landsat-8 OLI (WRS 2 – Worldwide Reference System2, Path/Row 219/75) between August 2015 and August 2016 were processed to Level 1 Terrain Corrected (L1T). These were corrected for atmospheric conditions to identify and mask cloud and cloud shadows by the USGS EROS Science Processing Architecture (ESPA) (DeVries et al. 2015; DeVries et al. 2015a). Landsat-8 data were corrected using L8SR, a newly developed algorithm that takes advantage of some of Landsat-8's new sensor characteristics (U.S. Geological Survey, 2015; Vermote, 2016). Cloud (pixel value 4), cloud shadow (pixel value 2), snow (pixel value 3), water (pixel value 1) and clear (pixel value 0) masks were provided for Landsat-8 data using Cfmask, a C implementation of the Fmask algorithm (Zhu & Woodcock, 2012; Zhu, Wang, & Woodcock, 2015). The CBERS 4 MUX imagery has been provided by the National Institute for Space Research (INPE).

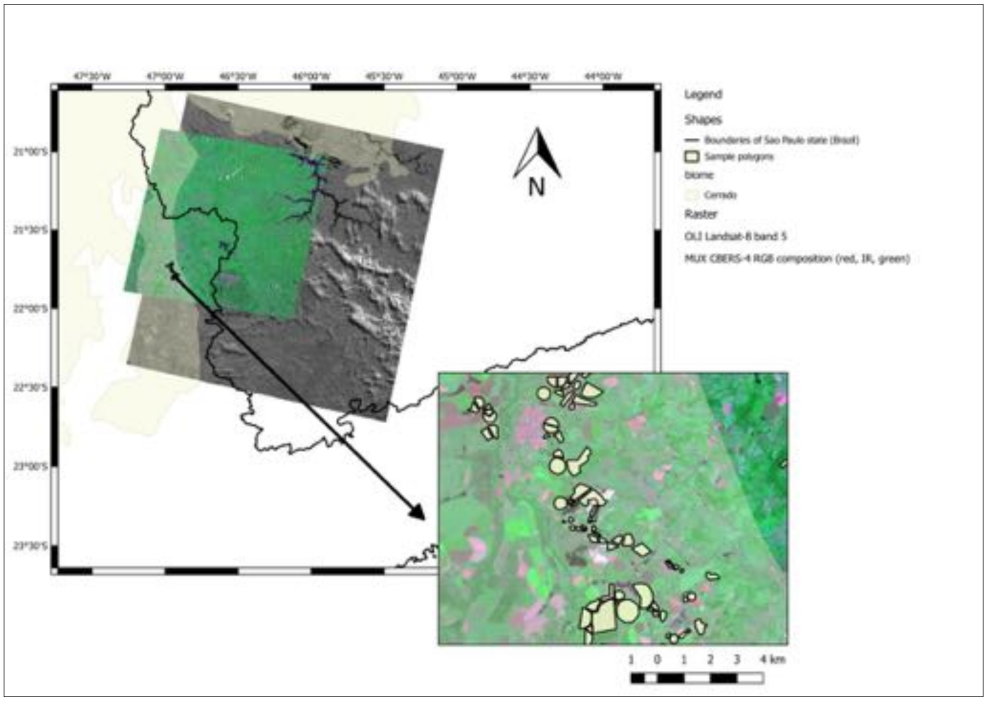


Figure 1. Location of the study area in Sao Paulo state, Brazil.

A total of 11 scenes of CBERS-4 MUX (CBERS WRS Path/Row 155/124) were acquired in the same period (August 2015 and August 2016). They were radiometric and geometrically corrected, adjusted and refined by using control points and the SRTM 30mv. 2.1 digital elevation model (DEM) (Level 4) and corrected for atmospheric conditions using the 6S model (Second Simulation of a Satellite Signal in the Solar Spectrum) (Vermote et al. 1997). Table 1 shows the availability of images from August 2015 to August 2016.

Table 1. Availability of Landsat-8 (Path/Row 219/75) and CBERS-4 (Path/Row 155/124) imagery from August 2015 to August 2016.

Month/Year	Sensor	Acquisition dates (day of year)	Number of scenes
Aug - Dec/2015	OLI	218, 234, 250, 266, 282, 298, 314, 330, 346, 362	10
	MUX	215, 241, 267, 345	4
Jan - Aug/2016	OLI	13, 29, 45, 61, 77, 93, 109, 125, 141, 157, 173, 189, 205, 237	14
	MUX	32, 110, 162, 188, 240	5

For the MUX imagery, we visually assessed the cloud cover for the region of interest for this study. The spectral band specifications for Landsat-8 OLI and CBERS-4 MUX can be seen on Table 2.

Table 2. Spectral band specifications for Landsat-8 OLI and CBERS-4 MUX.

Band	Landsat-8 OLI (µm)	CBERS 4 MUX (µm)
Blue	Band 2: 0.45 - 0.51	Band 5: 0.45 - 0.52
Green	Band 3: 0.53 - 0.59	Band 6 0.52 - 0.59
Red	Band 4: 0.64 - 0.67	Band 7: 0.63 - 0.69
Near Infrared (NIR)	Band 5: 0.85 - 0.88	Band 8: 0.77 - 0.89

The greatest difference in spectral bandwidths between the two sensors are on the NIR band, but there are also significant differences in spectral response function (SRF) profiles between corresponding CBERS-4MUX and Landsat-8 OLI spectral bands (Pinto et al., 2016).

2.3. Correlations Analysis between Landsat-8 OLI and CBERS-4 MUX

First we selected a pair of MUX and OLI images, considering the time proximity between them. The characteristics of the two images are shown in Table 3.

Table 3. Characteristics of the pair of MUX and OLI images used for correlation analysis.

Satellite/Sensor	Date	Acquisition Time (UTC)	Path/Row	Sun elevation	Sun azimuth	Look Angle
CBERS-4 MUX	04 August 2015	13:26:11	155/124	43.37°	36.05°	NADIR
Landsat-8 OLI	06 August 2015	13:03:18	219/75	40.61°	41.58°	NADIR

Considering the difference of spatial resolution between the images (30 meters for OLI and 20 meters for MUX), we resampled the MUX images to 30 meters, using a nearest neighbor approach. To deal with cloud contamination problems, we used the Fmask image to crop a free cloud region on both OLI and MUX surface reflectance images (Figure 2).

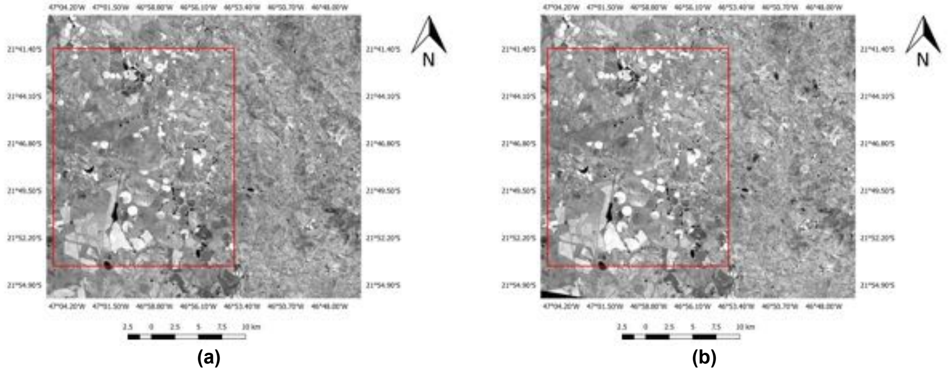


Figure 2. Cropped images used on the correlation analysis. (a) Landsat-8 OLI EVI (06 August 2015) and (b) CBERS-4 MUX EVI (August 4th, 2015).

We analyzed the correlations between the cropped images, for each selected vegetation index (EVI and NDVI). In order to determine an equation to predict OLI reflectance from MUX reflectance, linear regressions were constructed.

2.4. Building the multi-sensor time series

An algorithm was built to look to the nearest MUX image for each Landsat image, based on a user defined time span. Here, we used the time span of 8 days. Figure 3 shows a general scheme of the proposed method.

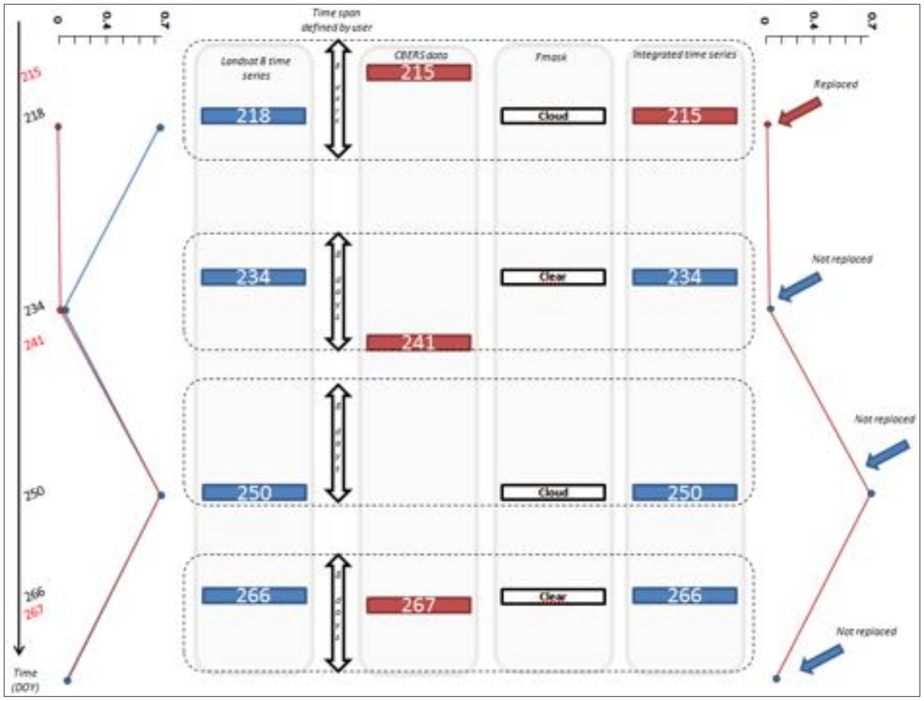


Figure 3. General scheme of the methodology used to build the integrated time series. On the left, a time series of EVI (the red line is the predicted time series using the equation to predict OLI reflectance from MUX and the blue line is the original Landsat time series); on the right is the integrated time series, with marks to illustrate the positions where the replacement has occurred.

After detecting the nearest MUX images for each Landsat images, the algorithm checks for cloud and cloud shadow contaminated pixels on the Landsat time series, by a conditional expression using Fmask images. When a contaminated pixel is detected in the time series, it is replaced by a value calculated from the equation to predict OLI reflectance from MUX, if it is within the time window.

2.5. Filtering the time series

We also applied a combined filtering approach for noise removal on the Landsat time series in order to access the improvement of the classification results compared to the integrated time series. The approach was put forth by interpolating the noise values with the average between the nearest neighbors in time, considering the Fmask quality data (Equation 1) and negative outliers based on a threshold as recommended by Hamunyela et al. (2013) (Equation 2).

$$x_t = \frac{x_{t-1}+x_{t+1}}{2}\{if\ fmask_t = 2\ OR\ fmask_t = 4\} \qquad (1)$$

$$x_t = \frac{x_{t-1}+x_{t+1}}{2}\{if\ x_t - x_{t-1} < -0.01x_{t-1} \&\ x_t - x_{t+1} < -0.01x_{t+1}\} \qquad (2)$$

where x_t is an observation of the time series at time t, x_{t-1} is the observation in the time series at time t-1, and $x_{t+1}$is the observation at time t+1. Observation x_t is replaced as an outlier with the average of x_{t-1} and x_{t+1} if the difference between x_t and $x_{t-1}$is less than -1% of x_{t-1}, and the difference between x_t and x_{t+1} is less than -1% of x_{t+1}. This method, however, is not capable of removing consecutive outliers. Figure 4 shows an example of how local outliers were removed from the NDVI and EVI time series.

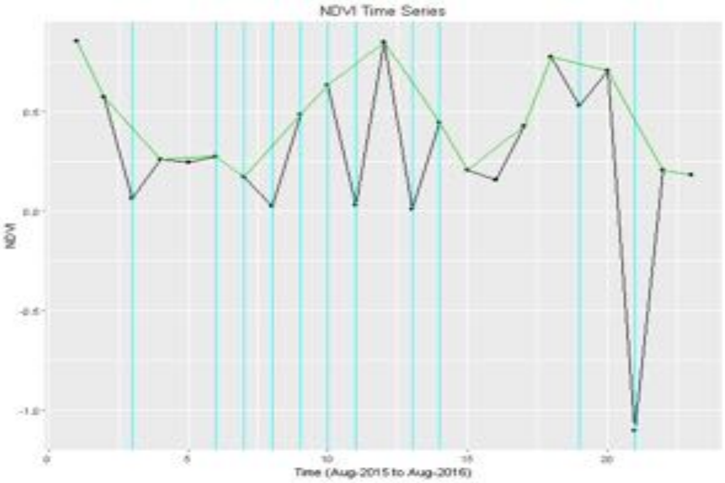


Figure 4. Example of how local outliers were removed from the NDVI time series. The cyan lines are the positions where cloud and cloud shadow were detected by Fmask. The black line is the integrated time series and the green line is the filtered integrated time series using Equation 2.

2.5. Extracting phenological features for classification

We selected 100 well-known polygon samples in the study area, considering the classes of annual agriculture (potato, corn, sugar beet, onion, bean and soybean), perennial agriculture (avocado and mango), semi-perennial agriculture (sugarcane), grassland and native forest.

We extracted NDVI and EVI time series of pixels from each sample polygon in the study area. Phenological metrics in time series were obtained by the TIMESAT v3.2 software (Jönsson; Eklundh, 2004), where seasonal data are extracted for each of the growing seasons of the central year (Figure 5). During a period of n years there may be n – 1 full seasons together with two fractions of a season in the beginning and end of the time series. So, to extract seasonality parameters from one year of data, the time series has been duplicated to span three years, as recommended by Jönsson and Eklundh (2015). For the phenological metrics extraction, the time series was smoothed considering the double logistic filter (Zhang et al., 2003; Jönsson; Eklundh, 2004). This function is recommended for smoothing image time series on cropland areas in the Brazilian Cerrado (Borges & Sano, 2014).

Figure 5 illustrates the schema of the seasonality parameters generated by TIMESAT. In this study, we assume that the seasonality parameters are the same of the phenological metrics. The time for the beginning of season (a), or start of the season (sos), and the end of season (eos) (b) is the time for which the left and right edge, respectively, has increased to a defined level (often a certain fraction of the seasonal amplitude) measured from the minimum level on the corresponding side. The length of the season (c) is the time from the start to the end of the season. Base value (d) is given as the average of the left and right minimum values. The middle of season (e) is computed as the mean value of the times for which, respectively, the left edge has increased to the 80 % level and the right edge has decreased to the 80 % level.

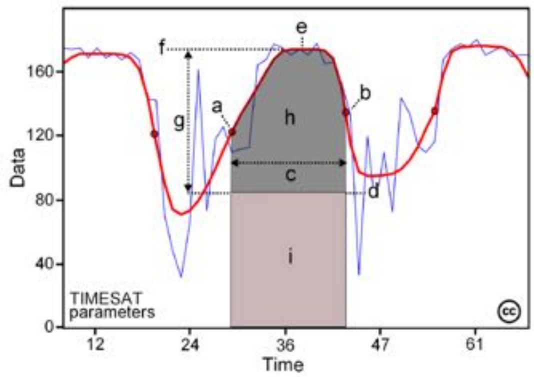


Figure 5. Some of the seasonality parameters generated by TIMESAT: (a) beginning of season, (b) end of season, (c) length of season, (d) base value, (e) time of middle of season, (f) maximum value, (g) amplitude, (h) small integrated value, (h+i) large integrated value. The red and blue lines represent the filtered and the original data, respectively.

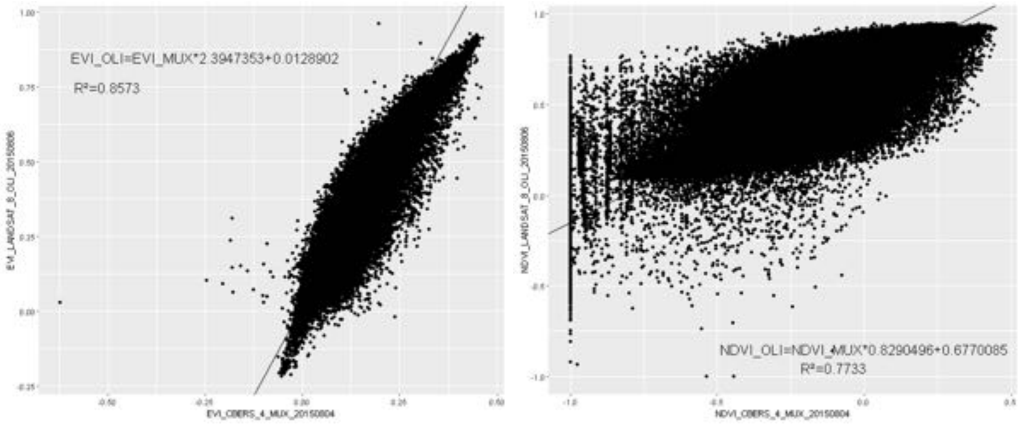
The maximum value (f), or the peak of the phenological cycle, is the largest data value for the fitted function during the season. The seasonal amplitude (g) is the difference between the maximum value and the base level. The left derivative is calculated as the ratio of the difference between the left 20% and 80% levels and the corresponding time difference. The right derivative (i.e. the rate of decrease at the end of the season) is the absolute value of the ratio of the difference between the right 20% and 80% levels and the corresponding time difference. The rate of decrease is thus given as a positive quantity. Large seasonal integral (h+i) is integral of the function describing the season from start to end. The small seasonal integral (h) is the integral of the difference between the function describing the season and the base level from start to end of the season (Jönsson and Eklundh, 2015). For more details see Jönsson and Eklundh (2002; 2004).

We subject the phenological metrics obtained on TIMESAT to data mining using the Random Forest (RF) algorithm (Breiman, 2001) considering each method:1) Original Landsat EVI time series; 2) Filtered Landsat EVI time series; 3) Integrated EVI time series; 4) Filtered Integrated EVI time series; 5) Original Landsat NDVI time series, 6) Filtered Landsat NDVI time series, 7) Integrated NDVI time series and 8) Filtered

Integrated NDVI time series. This RF algorithm is a classification technique in which the data set is randomly divided into several subsets of smaller size by means of applying bootstrap, and from each subset a decision tree is developed. All trees contribute to the classification of the object under study, by voting on which class the target attribute must belong. Random Forest algorithm has been widely used in remote sensing (Müller et al, 2015; Peña et al, 2015) because of its advantages in efficiently handling large databases, providing estimates on the most relevant variables, and allowing the identification of outliers (Rodriguez-Galiano et al., 2012). There were a total of 31 training pixels for the annual agriculture classes, 15 pixels for perennial agriculture class, 26 pixels for semi-perennial agriculture, 14 pixels for grassland class and 14 pixels for native forest. The results were evaluated by the confusion matrix index, global accuracy (Witten; Frank; Hall, 2011). The models were executed considering a 10-fold cross validation method. The classification results were obtained using the software package WEKA (Hall et al., 2009).

3. Results and Discussion

The results of the correlation analysis between the cropped images are shown in Figure 6, for each selected vegetation index: a) EVI and b) NDVI.



(a) (b)

Figure 6. Scatterplot of the pair of cropped images used to determine the linear regressions equations to predict OLI reflectance from MUX reflectance. (a) EVI and (b) NDVI.

The linear regressions equations to predict OLI reflectance from MUX reflectance are also shown. The regression coefficients for EVI and NDVI are respectively 0.8573 and 0.7733. Figure 7 shows the results of different approaches for noise removal on an EVI time series.

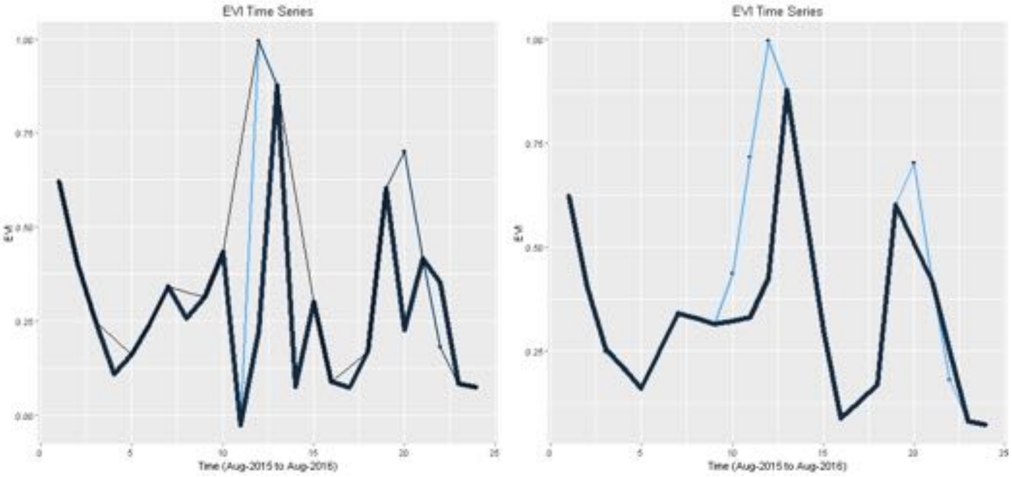


Figure 7. Results of different approaches for noise removal on an EVI time series. On the left the black line is the original Landsat-8 time series, the blue line is the integrated time series and the black thin line is filtered integrated time series. On the right, the blue line is the filtered integrated and the black line is the Landsat-8 filtered time series.

As we can see in Figure 7, the integrated time series can deal with noise, replacing cloud and cloud shadow contaminated pixels withclear pixels of MUX images, and allowing improvement of the time series according to the phenological behavior of the vegetation, which is significant regarding the capability of TIMESAT on extracting the features. We found that concerning the 100 analysed pixels time series, an average of 11.96% of cloud and cloud shadow contaminated observations were replaced using CBERS-4MUX images.

A 10 fold cross-validation technique was applied within the different training sets (Original Landsat EVI time series; Filtered Landsat EVI time series; Integrated EVI time series; Filtered Integrated EVI time series; Original Landsat NDVI time series, Filtered Landsat NDVI time series, Integrated NDVI time series and Filtered Integrated NDVI time series). The accuracy of the different data set classifications are presented in Table 4.

Table 4. Accuracy of classification for the different data set classifications.

Time series Data sets	NDVI	EVI
Integrated	68%	73%
Filtered Integrated	64%	78%
Filtered Landsat	70%	76%
Original Landsat	60%	70%

We found that concerning the NDVI time series, the multi-sensor approach accuracy was 64% when combined with the filtering approach (Equation 2), against 68% without the filtering step. When using only Landsat-8 data, the accuracy was 60%. But when combining the filtering approaches of Equation 1 and 2, the accuracy of the classification results with Landsat-8 time series was 70%.

The results using the EVI time series showed that when the multi-sensor approach was used, the accuracy was higher than when using the original Landsat-8 time series (respectively, 73% and 70%), as well as when combined to the filtering approaches. The classification accuracy using the filtered integrated time series was better than using the Landsat-8 time series (respectively, 78% and 76%). The best result was observed with the filtered integrated EVI time series.

We can derive a hypothesis by that which was observed by Holden et al. (2016), whereby the effect of combining data from the two sensors (L7 ETM+ and L8 OLI), once L7 ETM+ has the same spectral bandwidths of CBERS-4 MUX. NDVI relies on the contrasting relationship between the near infrared band and the red band. They observed that there is a strong and consistent positive bias in NDVI, with Landsat-8 having much higher NDVI. The EVI differs from NDVI by utilizing the blue band as an additional normalizing factor that corrects the red band for atmospheric influences. It appears that the bias in the blue band between Landsat-8 and Landsat-7 nullifies the bias in the red and near infrared band, resulting in a similar EVI across sensors (Holden et al., 2016).This is probably the reason explaining why the results with EVI, when using the integrated time series are better. We can see that small differences on the time series values leads to changes in the results of the smoothers improved by TIMESAT. Furthermore, differences on the extracted parameters can modify the results of classification. As the MUX NDVI values tend to be higher, it modifies the amplitude of the signal, resulting in significant changes on the smooth time series. We can also see in Figure 6 that the regression coefficient between the Landsat NDVI and MUX NDVI are significantly lower than in respect EVI. As observed by Pinto et al. (2016), the greatest difference in spectral bandwidths between the sensors are on the NIR band, but there are also significant differences in SRF profiles between corresponding CBERS-4MUX and Landsat-8 OLI spectral bands (Pinto et al., 2016).

Thus, we can suggest that normalizing the SRF between the sensors would improve the results. We can also infer that the different methods of atmospheric correction may be affecting the results; as well problems of misregistration between the images and resampling can also be a source of errors. More studies are needed to better comprehend the effects of the different filtering approaches, as well to understanding these effects on the smooth improved by TIMESAT with double logistic functions. It is also suggested to test the other smooth approaches implemented by TIMESAT as the Asymmetric Gaussian functions and Savitzky-Golay.

5. Acknowledgements

This work is partially supported by FAPESP e-science program (grant 2014-08398-6 and 2016-08719-2).

References

Arvor, D., Jonathan, M., Meirelles, M. S. O. P., Dubreuil, V., Durieux, L., 2011. Classification of MODIS EVI time-series for crop mapping in the state of Mato Grosso, Brazil. International Journal of Remote Sensing, 32 (22), pp. 7847-7871.

Bendini, H., Sanches, I. D., Körting, T. S., Fonseca, L. M. G., Luiz, A. J. B., and Formaggio, A. R.: Using Landsat 8 Image Time Series For Crop Mapping In A Region Of Cerrado, Brazil, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLI-B8, 845-850, doi:10.5194/isprs-archives-XLI-B8-845-2016, 2016.

Borges, E.F., Sano, E. E., 2014. Séries temporais de EVI do MODIS para o mapeamento de uso e cobertura vegetal do oeste da Bahia. Boletim de CiênciasGeodésicas, 20 (3), pp. 526-547.

Breiman, L., 2001. Random Forests.Machine Learning. Vol 45, pp. 5-32.

Hall, M. A.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I. H.2009. The WEKA Data Mining Software: An Update; SIGKDD Explorations. New York, v.11, n.1, p. 10-18.

Hamunyela, E., Verbesselt, J., Roerink, G., &Herold,M. (2013). Trends in spring phenology ofWestern European deciduous forests.Remote Sensing, 5(12), 6159–6179 http://doi.org/10.3390/rs5126159.

Holden, E. C., Curtis, E. W., (in press) 2016.An analysis of Landsat 7 and Landsat 8 underflight data and the implications for time series investigations.Remote Sensing of Environment.

Irish, R.R. ,Barker,J.L., Goward, S.N., Arvidson, T. (2006), Characterization of the Landsat-7 ETM +automated cloud-cover assessment (ACCA) algorithm. Photogrammetric Engineering and Remote Sensing, 72, pp. 1179–1188

Jönsson, P., Eklundh, L. 2002. Seasonality extraction by function fitting to time-series of satellite sensor data.IEEE Transactions on Geoscience and Remote Sensing. 40 (8), pp. 1824-1831.

Jönsson, P., Eklundh, L. 2004. TIMESAT – a program for analyzing time-series of satellite sensor data.Computers & Geosciences. 30 (8), pp. 833-845.

Jönsson, P., Eklundh, L., 2015. TIMESAT 3.2 with Parallel Processing Software Manual. Lund University, Sweden. pp. 22-24.

Korting, T. S., 2012. Geodma: A Toolbox Integrating Data Mining with Object-Based and Multi-Temporal Analysis of Satellite Remotely Sensed Imagery. PhD Thesis, National Institute for Space Research (INPE). 97p.

Lymburner, L., et al. (2016), Landsat 8: Providing continuity and increased precision for measuring multi-decadal time series of total suspended matter, Remote Sensing of Environment, http://dx.doi.org/10.1016/j.rse.2016.04.011

Müller, H., Rufin, P., Griffiths, P., Siqueira, A. J. B., Hostert, P., 2015. Mining dense Landsat time-series for separating cropland and pasture in a heterogeneous Brazilian savanna landscape. Remote Sensing of Environment, 156, pp. 490-499.

Neves, A. K., Bendini, H. N., Körting, T. S., Fonseca, L. M. G., (in press) 2016. Combining Time Series Features and Data Mining to Detect Land Cover Patterns: A Case Study in Northern Mato Grosso State, Brazil. Revista Brasileira de Cartografia.

Pan, Z.; Huang, J., Zhou, Q., Wang, L., Cheng, Y., Zhang, H., Blackburn, G. A.; Yan, J., Liu, J., 2015. Mapping crop phenology using NDVI time-series derived from HJ-

1A/B data. International Journal of Applied Earth Observation Geoinformation, 34, pp. 188-197.

Peña, M.A., Brenning, A., 2015. Assessing fruit-tree crop classification from Landsat-8 time-series for the Maipo Valley, Chile. Remote Sensing of Environment, 171, pp. 234-244.

Pinto, C., Ponzoni, F., Castro, R., Leigh, L., Mishra, N., Aaron, D., Helder, D. (2016)First in-Flight Radiometric Calibration of MUX andWFI on-Board CBERS-4.Remote Sens., 8, 405; doi:10.3390/rs8050405

Risso, J.,Rudorff, B. F. T., Adami, M., Aguiar, A. P. D., Freitas, R. M, 2012. MODIS Time Series for Land Use Change Detection in Fields of the Amazon Soy Moratorium. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Melbourne, Australia, Vol. 23, pp. 339-344.

Rodriguez-Galiano, V. F., Ghimire, B., Rogan, J., Chica-Olmo, M., Rigol-Sanchez, J. P., 2012. An assessment of the effectiveness of a random forest classifier for land-cover classification. ISPRS Journal of Photogrammetry and Remote Sensing, 67, pp. 93-104.

Sakamoto, T., Yokozawa, M., Toritani, T., Shibayama, M., Ishitsuka, N., Ohno, H., 2005.A crop phenology detection method using time-series MODIS data.Remote Sensing of Environment, 96 (3-4), pp. 366-374.

U.S. Geological Survey, 2015.U.S. Geological SurveyPRODUCT GUIDE: Landsat Surface Reflectance products courtesy of the U.S. Geological Survey(2015), pp. 1–27

Vermote, E.; Tanre, D.; Deuze, J.; Herman, M.; Morcette, J.J. (1997).Second simulation of thesatellite signal in the solar spectrum, 6S: An overview. IEEE Trans. Geosci. RemoteSens., 35, 675–686.

Vermote, E. (2016). Placeholder: Landsat 8 surface reflectance correction. Remote Sensing of Environment (PLACEHOLDER (PLACEHOLDER), 0–0).

Witten, I. H.; Frank, E.; Hall, M. A. 2011.Data mining: practical machine learning tools and techniques. 3ed. San Francisco: Morgan Kaufmann.

Zhang, X., Friedl, M. A., Schaaf, C. B., 2003.Monitoring vegetation phenology using MODIS.Remote Sensing of Environment. 84, pp. 471-475.

Zheng, B.; Myint, S. W. Thenkabail, P. S.; Aggarwal, R. M. 2015.A support vector machine to identify irrigated crop types usingtime-series Landsat NDVI data.International Journal of Applied Earth Observation and Geoinformation. 34, pp. 103-112.

Zhu and Woodcock, 2012. Z. Zhu, C.E. Woodcock. Object-based cloud and cloud shadow detection in Landsat imagery. Remote Sensing of Environment, 118 (2012), pp. 83–94 (URL: http://dx.doi.org/10.1016/j.rse.2011.10.028)

Zhu, Z., Wang, S., & Woodcock, C. E. (2015b). Improvement and expansion of the Fmask algorithm: cloud, cloud shadow, and snow detection for Landsats 4–7, 8, and Sentinel 2 images. Remote Sensing of Environment (URL: http://dx.doi.org/10.1016/j.rse.2014. 12.014).

PostGIS-T: towards a spatiotemporal *PostgreSQL* database extension

Rolf E. O. Simoes[1], **Gilberto Ribeiro de Queiroz**[1], **Karine Reis Ferreira**[1], **Lubia Vinhas**[1], **Gilberto Camara**[1]

[1]Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 15.064 – 91.501-970 – São José dos Campos – SP – Brazil

{rolf.simoes, gilberto.queiroz, karine.ferreira,

lubia.vinhas}@inpe.br

Abstract. *The temporal dimension of spatial data has been the subject of discussion in the literature for a long time. While there are numerous Database Management System (DBMS) solutions only for spatial dimension, we did not observe the same situation for* spatiotemporal *data. Considering this gap, our purpose is to design and implement an extension to the DBMS PostgreSQL that is based on a formal spatiotemporal algebra in order to incorporate representations of spatiotemporal data within the DBMS. The proposed extension can be used in a large range of applications. We intend that this extension be a reasonable framework to store and handling observational remote sensing data usually present in applications like animal migration researches, wildfires monitoring, vessel tracking for monitoring fishing, and the like. In this work, we show how to apply it in a case study based on spatiotemporal data collected from drifting buoys belonging to the NOOA's Global Drifter Program.*

1. Introduction

Earth Observation data generation has been increased since the last decades. This phenomenon occurs, considering that a great amount of data are daily collected by different missions such as *CBERS*[1] in Brazil/China, *Landsat*[2] in the USA and *Sentinel*[3] in Europe. The development of mobile positioning technologies and its low costs are also factors that enable spatial data gathering through time.

These different data sources are associated with temporal dimension, mainly by allowing the monitoring of spatially located objects in time, either by allowing the time analysis by increasing the temporal resolution of the observations. The collection, representation and processing of this data have been largely facilitated by database management systems (DBMS) and their spatial extensions, which are based on international standards such as the OGC Simple Feature Specification [Herring 2011] and ISO geographic information standards [Kresse and Fadaie 2004]. Furthermore, while there are numerous DBMS solutions supporting the spatial dimension, we do not observe the same situation for spatiotemporal data.

[1] http://www.cbers.inpe.br/

[2] http://landsat.usgs.gov/

[3] https://sentinel.esa.int/

The temporal dimension of spatial data has been the subject of discussion in the literature for a long time. Additionally, some conceptual systems regarding to representation of spatiotemporal data have been proposed [Camara et al. 2014, Ferreira et al. 2014, Erwig et al. 1999]. One of these systems, particularly discussed in [Ferreira et al. 2014], is structured around the concept of observations, the basic unit of data acquisition of a spatial temporal phenomenon. From observations, it is possible to generate three types of spatiotemporal data: *time series*, *trajectories* and *coverages*. With these three types, it is possible to represent the geo-ontological concepts of object and field and to define a space-time algebra. The authors have implemented their work as a C++ library, that works as middleware between the actual sources of data (databases of flat files, for example) and their conceptual model.

We have observed a number of solutions for processing spatiotemporal data as client side solutions. This approach may presents as disadvantage the introduction of an overhead by transferring data between processes or even between machines when those data are managed over a network. Differently from that, we propose a model that works inside the database system.

Besides that, this approach can avoid memory issues for huge volume of data since this is a solved subject in the context of relational DBMS such as PostgreSQL. Moreover, spatiotemporal data usually consumes a higher volume of memory space. Considering this drawback, our goal is to implement an extension to the DBMS PostgreSQL to provide support for spatiotemporal types within the DBMS. The PostGIS *geometry* type is used as a basic spatial representation for the extension. Furthermore, temporal dimension was integrated to get our spatiotemporal type.

Here, we propose a spatialtemporal database extension. Based on the work of [Ferreira et al. 2014], we introduce three new spatiotemporal data types for the DBMS PostreSQL. Moreover, we implemented some algebra functions of those data. In order to demonstrate how our extension can manipulate real data observations, we conducted a case study based on the Global Drifter Program (GDP) database.

2. Background

According to [Sinton 1978], space, time and theme (or a quantity measure) are the three dimensions of geographical structure and observation. In such a way, it is possible to observe by fixing one dimension, controlling another and measuring the other. Hence, six types or structure of observation can be produced. Proceeding in this manner [Ferreira et al. 2014] claim that we can capture all kind of spatiotemporal phenomena exhaustively with three of them:

1. **time series**: fix time, control space and measuring theme;
2. **trajectory**: fix time, control theme and measuring space;
3. **coverage**: fix space, control time and measuring theme;

Time series and trajectory play an important role considering the necessity to analyze data along time. The main difference between them remains in space dimension: in case of trajectory we are interested in measuring the space locations of an given observed phenomena, whereas in time series the measured data is gathered from a fixed space.

It is easy to see that the fix operation defines a domain from which the measured data must be filtered in. In this manner, a geo-located time series are sequences of ob-

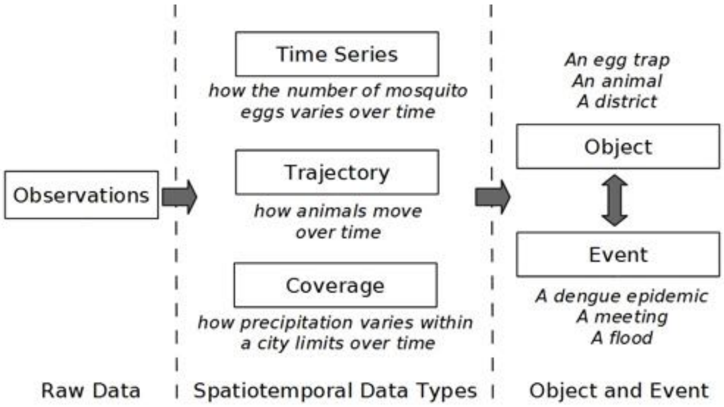

Observations
Time Series
how the number of mosquito eggs varies over time
Trajectory
how animals move over time
Coverage
how precipitation varies within a city limits over time
An egg trap
An animal
A district
Object
Event
A dengue epidemic
A meeting
A flood
Raw Data
Spatiotemporal Data Types
Object and Event

Figure 1. The proposed model. Source: [Ferreira et al. 2014, p. 258]

servations over time of a measured phenomena that takes place in a given space domain. For instance, Landsat series spans over 40 years and is a very informative temporal record of radiance of geo-located sites represented by pixels [Roy et al. 2014]. This database can provide significant data about each pixel spectral response time series that can be further analysed to give us information about land use and land cover change over time [Maus 2016].

On the other hand, trajectories can be seen as a sequence in time dimension of geo-located observed geometries (points, lines, polygons or volumes) that are associated to a given theme. For instance, database monitoring oceanic buoys give us a rich data about its positions, water temperature and salinity. One can be interested on tracking buoys positions over time to capture surface oceanic chains [Lumpkin and Pazos 2007].

Here, the time dimension organizes different space locations of a fixed theme (the buoy). Another example, the policy for preventing Dengue epidemic may depend on monitoring of egg traps over time [Regis et al. 2009]. In this case, one may be interested to know all locations over time that presents a given higher level of captured mosquito eggs (threshold theme). The main difference between these two examples of trajectories is the kind of fixed theme. In the first one, the fixed theme was an identifiable object entity, a buoy. In the second one, we used a measured data not related to a specific object entity but a condition that may involve a set of objects. This condition refers to object's properties and can denote events. For example, if we are interested in flood monitoring in some urban area, we may relate the event *flood* to a condition of water precipitation metered by a set of pluviometers.

Objects and events play a key role on the interpretation of a spatiotemporal data [Ferreira et al. 2014, Worboys 2005]. In this approach, an object is any identifiable entity over time and an event is an episode on the time that may relates to one or more objects. Episodes have a definite begin and an end. Events may be distinguished by punctual occurrence, if it occurs instantaneously, or durative, if it takes some time [Galton 2004]. An event does not change over time and we can derive them from conditions of spatial and non-spatial properties of objects, as we can see from the example above. Figure 1 summarizes this model.

Spatiotemporal structures can be implemented on computer systems as data types. A data type is a set of values over which we can define some operations. To formalize these ideas, [Ferreira et al. 2014] follow [Guttag and Horning 1978] and propose an algebraic specification that consists of (1) definitions of type names, their domains, ranges, and operations; (2) a set of axioms that expresses truth relations among those operations. PostgreSQL works similarly: all data has a type with an explicit or implicit ranges; all functions operates over some types and returning new data. To comply with [Ferreira et al. 2014] abstract model, we just need to guarantee that PostGIS-T functions be implemented in such a way that its behavior be consistent with axioms specification. In what follows, we describe how our extension implements and represents spatiotemporal data.

3. PostGIS-T model

PostGIS-T introduces the `SPATIOTEMPORAL` type, a composite PostgreSQL type that plays the same role as `SpatioTemporal` type in [Ferreira et al. 2014]. The main difference between them is that unlike `SpatioTemporal`, `SPATIOTEMPORAL` type can represent both `TimeSeries`, `Trajectory` and, `Coverage` and does not work with the idea of abstract type that is specialized later.

In order to access and manipulate the data, we must define functions. To achieve the same level of specification, we have implemented each declared operator in [Ferreira et al. 2014] model definition. We list all function signature in Table 1.

PostGIS-T stores observations as tuples in one or more relations that can be further queried to instantiate `SPATIOTEMPORAL` data. To get a new `SPATIOTEMPORAL` data, we use `TST_SPATIOTEMPORAL()` aggregate function.

As we can see, PostGIS-T was built over PostgreSQL and PostGIS. For example, spatial representations are `GEOMETRY` values, a type introduced by PostGIS. As a first prototype we have limited measure values as `NUMERIC` type, so that an observation must be a triple (`TIMESTAMP`, `GEOMETRY`, `NUMERIC`). In order to instantiate `SPATIOTEMPORAL` data we must call `TST_SPATIOTEMPORAL()` inside a query informing where to find these values.

Spatiotemporal data may be interpreted differently depending on the underlying phenomena it represents. As previously discussed, we could conceive an observation as taking place continuously or instantaneously. Our data type was defined to accommodate all spatiotemporal data phenomena without a specific semantic. For example, we does not know in advance if we should conceive an observation as an occurrent or a durative one, or if the sample observation is a time series or a trajectory. To overcome this limitation, we have implemented some functions that can be combined in order to get the right phenomena interpretation. These functions are `TST_ESTIMATE_MEASURE()`, `TST_ESTIMATE_LOCATION()`, `TST_RESAMPLE_TIME()`, and `TST_COVERAGE()`.

If we are interested in how to get an approximate measure between two empirical observed phenomenon we must take into account its underlying nature, that is, if it refers to an object or to an event. For example, we took observations of a drifter floating on the ocean at times `10:00AM` and `11:00AM`, and we are interested to know its location at `10:30AM`. We may assume that a linear interpolation would give us a good approximation and so we call TST_ESTIMATE_LOCATION() function informing the SPATIOTEMPORAL data, the time of interest to calculate the interpolation (in this case 10:30AM), and the interpolation method name 'LINEAR' as parameters. For now, we have implemented a small set of interpolation methods that we can use, 'LINEAR', 'LAST' or 'NEAREST', meaning, respectively, simple linear interpolation, last registered location or measure before or equal a given time, and the closest time registered location or measure of a given time. The process to estimate a measure from a SPATIOTEMPORAL is analogous.

Other useful application of interpolators are re-sampling data. PostGIS-T is able to re-sampling time observations between a time range at regular time resolution with the function TST_RESAMPLE_TIME(). This function returns a regular time spaced SPATIOTEMPORAL data whose locations and measures were estimated according to a given interpolation method (see more details in section 4).

Furthermore, if we are interested to re-sampling our observations through space in order to produce what [Ferreira et al. 2014] calls *coverage*, we use the function TST_COVERAGE(). This function returns a SPATIOTEMPORAL data whose observations refer to a regular extent over which measures are aggregated into an unique value according to a given aggregate strategy (e.g. 'COUNT', 'AVG', 'MIN', 'MAX' and 'AREA'). The result is a time flattened spatiotemporal data where no gap and no overlapping area exists between two adjacent extents. In [Ferreira et al. 2014] model, this represents a Coverage.

Other functions are related to operations that retrieve SPATIOTEMPORAL properties or subset of the spatiotemporal data. For instance, the functions TST_BEGINS() and TST_ENDS() indicates the start and the end times for the sampling, whereas TST_HULL() gives its convex hull polygon where observations took place. Finally, to get the max and the minimum values of the measured data, we must use the functions TST_MIN() and TST_MAX(), respectivelly. On other hand, the TST_DURING() function returns all observations that have been made in a given time range. Likewise, to get only a given location samples, we use the function ST_INTERSECTION() passing to it the area or point of interest.

A complete and comprehensive documentation can be found in the extension's webpage https://gitlab.dpi.inpe.br/postgis-t. In the following section, we demonstrate an application of how to use PostGIS-T.

4. The Global Drifter Program: a case study with real spatiotemporal data

Regarding to spatiotemporal data and its complexity, we chose the satellite-tracked surface drifting buoy (drifter) data to evaluate and get experienced with the extension implementation details in PostgreSQL environment. The Global Drifter Program (GDP) is a branch of the National Oceanic and Atmospheric Administration (NOAA). It aims to maintain a global satellite-tracked surface drifting buoys and to provide the data set for scientific purposes, as climate predictions and climate research and monitoring. In this manner, GDP produces observations from most areas of the world's oceans at sufficient density to map the mean currents at one degree resolution [Lumpkin and Pazos 2007].

Drifter is a surface buoy connected with a subsurface drogue. Its observations have been largely used in oceanographic and climate researches. The main use of this data is to

Function signature	Return type
TST_SPATIOTEMPORAL(TIMESTAMP, GEOMETRY, NUMERIC)	SPATIOTEMPORAL
TST_ESTIMATE_MEASURE(SPATIOTEMPORAL, TIMESTAMP, TEXT)	NUMERIC
TST_ESTIMATE_LOCATION(SPATIOTEMPORAL, TIMESTAMP, TEXT)	GEOMETRY
TST_RESAMPLE_TIME(SPATIOTEMPORAL, TSRANGE, INTEGER, TEXT)	SPATIOTEMPORAL
TST_COVERAGE(SPATIOTEMPORAL, INTEGER, INTEGER, TEXT)	SPATIOTEMPORAL
TST_BEGINS(SPATIOTEMPORAL), TST_ENDS(SPATIOTEMPORAL)	TIMESTAMP
TST_HULL(SPATIOTEMPORAL)	GEOMETRY
TST_AFTER(SPATIOTEMPORAL, TIMESTAMP)	SPATIOTEMPORAL
TST_BEFORE(SPATIOTEMPORAL, TIMESTAMP)	SPATIOTEMPORAL
TST_DURING(SPATIOTEMPORAL, TSRANGE)	SPATIOTEMPORAL
TST_INTERSECTION(SPATIOTEMPORAL, GEOMETRY)	SPATIOTEMPORAL
TST_DIFFERENCE(SPATIOTEMPORAL, GEOMETRY)	SPATIOTEMPORAL
TST_MEASURE(SPATIOTEMPORAL, TIMESTAMP)	NUMERIC
TST_MEASURE(SPATIOTEMPORAL, GEOMETRY, TEXT)	NUMERIC
TST_MIN(SPATIOTEMPORAL), TST_MAX(SPATIOTEMPORAL)	NUMERIC
TST_LESS(SPATIOTEMPORAL, NUMERIC)	SPATIOTEMPORAL
TST_GREATER(SPATIOTEMPORAL, NUMERIC)	SPATIOTEMPORAL
TST_BETWEEN(SPATIOTEMPORAL, NUMRANGE)	SPATIOTEMPORAL
TST_LOCATION(SPATIOTEMPORAL, TIMESTAMP)	GEOMETRY
TST_EQUALS(SPATIOTEMPORAL, SPATIOTEMPORAL)	BOOLEAN
TST_INTERPOLATOR(SPATIOTEMPORAL)	TEXT
TST_SETINTERPOLATOR(SPATIOTEMPORAL, TEXT)	SPATIOTEMPORAL
TST_OBSERVATIONS(SPATIOTEMPORAL)	INTEGER

Table 1. List of all defined PostGIS-T functions

map oceanic surface currents of different seas and oceanic regions of the planet. Also, the data can be used to calibrate satellite sensors. Each drifter has an unique identifier code and is equipped with sensors that periodically measure properties such as salinity and surface temperature of the water. All these data are subsequently transmitted to satellites. The drifter's position and velocity are usually inferred by *Doppler shift*, which occurs during the transmission step. The positioning system, also known as *Argos*, provides drifter locations with $O\left(100m\right)$ errors. The raw data are then assembled and normalized by the Drifter Data Assembly Center of the Atlantic Oceanographic and Meteorological Laboratory (DAC/AOML).

The GDP drifter database used contains $2,263,842$ collected locations with respective zonal and meridional velocities observations for 408 drifters worldwide. The time resolution of the observation is one hour. More information about how this database was collected can be see in [Elipot et al. 2016].

All data sample were loaded in a table of observations as defined in the Listing 1. Subsequently, we proceeded with data instantiation by calling the aggregate function `TST_SPATIOTEMPORAL(TIMESTAMP, GEOMETRY, NUMERIC)`. All the following queries uses this table.

Sometimes it is useful to change the amount of observations of a given

```
CREATE TABLE buoy_obs_st (
  buoy_id     INTEGER PRIMARY KEY,
  spatiotemp SPATIOTEMPORAL
);
```

Listing 1: Definition of the spatiotemporal table buoy_obs_st.

data. In PostGIS-T, we can obtain a new sample of a trajectory by the function TST_RESAMPLE_TIME(). This function receives as parameter the spatiotemporal data, the time interval that we are want re-sampling, the number of observations to be re-sampled, and the interpolation method. Note that the extension makes no assumption about the observation continuity or duration. In this regard, an appropriate interpolation method must be informed by the user. The queries in Listing 2 shows how do we re-sample observations. A graphical result is presented in Figure 2.

```
SELECT buoy_id,
       TST_RESAMPLE_TIME(
           spatiotemp,
           TST_OBSERVATIONS(spatiotemp) / 10,
           'LINEAR')
  FROM buoy_obs_st;
```

```
SELECT buoy_id,
       TST_RESAMPLE_TIME(
           spatiotemp,
           TST_OBSERVATIONS(spatiotemp) * 2,
           'LINEAR')
  FROM buoy_obs_st;
```

Listing 2: Re-sampling on time. Query on the top (bottom) reduces (increase) the time resolution.

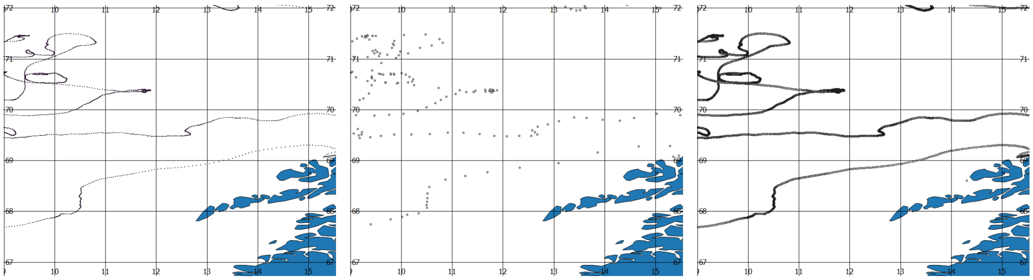


Figure 2. Trajectory re-sampling. From left to right: original data, re-sampling for every 10 hours, re-sampling for every 30 minutes.

Re-sampling technique may be employed to synchronize data observations, in order to speed-up queries that performs sequential time calculations. Comparing trajectories (a) and (b) from the Figure 2, we can note that some trajectory sections can be well approximated by few interpolated observations. These sections mainly resembles straight segments.

Another useful application of these data is to estimates the mean velocity of ocean currents over different regions and a given period of year. This is a very important question in related climate researches. Velocity may be represented by a vector that informs us about the direction and speed magnitude of the moving entity. Here, re-sampling technique may be useful if we would like to measure the mean direction of currents.

Suppose now we are interested to measure the mean velocity magnitude of a drifter between a small time range (for the sake of simplicity). How can we proceed in PostGIS-T? First we need a function that returns the observations of a given time interval. This function is TST_DURING() which returns a SPATIOTEMPORAL data. From this result, TST_COVERAGE() creates a regular grid whose extent is the same as observations location bounding box.

The dimensions of that grid is given as parameters to the function. A new measure is then calculated for each cell grid according to aggregate strategy. Thereafter, a new SPATIOTEMPORAL data is instantiated and returned. The corresponding query is presented in Listing 3 and a graphical representation is showed in Figure 3.

```
SELECT buoy_id,
       TST_COVERAGE(
           TST_DURING(
               spatiotemp,
               TSRANGE(
                   '2015-05-18 14:00:00',
                   '2015-05-20 14:00:00', '[]')
           ), 7, 13, 'AVG')
  FROM buoy_obs_st
 WHERE buoy_id = 132470;
```

Listing 3: Coverage.

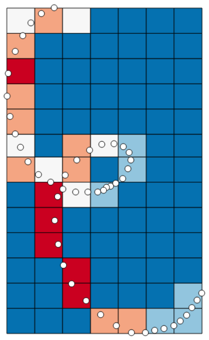


Figure 3. Coverage calculated from drifter mean velocity. Blue-Red colors denotes lower-higher velocities.

From Figure 3 we can see that not all cells grid contains a drifter observation. Only those regions that have at least one register has a measure value equals to that of velocity average. In order to fill those cells grid we would need a spatial interpolator. The process to get a coverage of velocity magnitudes is similar.

5. Conclusions

Here, we have proposed a spatiotemporal database extension based on conceptual model of [Ferreira et al. 2014]. We have implemented this model with some adaptations for the relational database environment. However this adaptation was not conceptual but operational. For example, in this preliminary version we did not provide a way to extend the base of interpolation methods used as a parameter of function like `TSTRESAMPLETIME`. In spite of that, the implementation shows us that the spatiotemporal model proposed in [Ferreira et al. 2014] is feasible in an relational DBMS context.

Our first approach suggest that this model may be more indicated to applications of sparse geo-referenced data like movable objects (e.g. drifters, ship trajectories) and observed events (e.g. wildfires, disease occurrences). This applications should work better over snapshots of the original data as the task of packing a huge spatiotemporal tuples is expensive. However, it is too early to note some processing improvement from our data type columnar design.

Further research and development consist of: (a) designing a compact and efficient disk storage layout for values of `SPATIOTEMPORAL` type; (b) introduce the notion of subtypes of `SPATIOTEMPORAL` as type modifiers (`TIMESERIES`, `TRAJECTORY` and `COVERAGE`), which will give more constraint about the data and the result of operations; (c) how to include spatiotemporal indexes that could take advantage of approximations of spatiotemporal data; (d) explore the extension with big spatiotemporal data applications in a database cluster environment.

As a first approach, we have prototyped the PostGIS-T extension with the high level SQL and PL/pgSQL languages. The next version of this extension will be developed in the C programming language and it will include a *view* named `spatiotemporal_columns` with the same purpose of PostGIS `geometry_columns`. Moreover, we will provide a larger number of functions to deal with different spatiotemporal data manipulation demands.

References

Camara, G., Egenhofer, M. J., Ferreira, K., Andrade, P., Queiroz, G., Sanchez, A., Jones, J., and Vinhas, L. (2014). Fields as a generic data type for big spatial data. In *International Conference on Geographic Information Science*, pages 159–172. Springer.

Campbell, J. B. and Wynne, R. H. (2011). *Introduction to remote sensing*. Guilford Press.

Elipot, S., Lumpkin, R., Perez, R. C., Lilly, J. M., Early, J. J., and Sykulski, A. M. (2016). A global surface drifter data set at hourly resolution. *Journal of Geophysical Research: Oceans*.

Erwig, M., Gu, R. H., Schneider, M., Vazirgiannis, M., et al. (1999). Spatio-temporal data types: An approach to modeling and querying moving objects in databases. *GeoInformatica*, 3(3):269–296.

Ferreira, K. R., Camara, G., and Monteiro, A. M. V. (2014). An algebra for spatiotemporal data: From observations to events. *Transactions in GIS*, 18(2):253–269.

Galton, A. (2004). Fields and objects in space, time, and space-time. *Spatial cognition and computation*, 4(1):39–68.

Goncalves, M., Netto, M., Costa, J., and Zullo Junior, J. (2008). An unsupervised method of classifying remotely sensed images using kohonen self-organizing maps and agglomerative hierarchical clustering methods. *International Journal of Remote Sensing*, 29(11):3171–3207.

Guttag, J. V. and Horning, J. J. (1978). The algebraic specification of abstract data types. *Acta informatica*, 10(1):27–52.

Herring, J. (2011). Opengis implementation standard for geographic information-simple feature access-part 1: Common architecture. *OGC Document*, 4(21):122–127.

Kemp, K. K. (1996). Fields as a framework for integrating gis and environmental process models. part 1: Representing spatial continuity. *Transactions in GIS*, 1(3):219–234.

Kresse, W. and Fadaie, K. (2004). *ISO standards for geographic information*. Springer Science & Business Media.

Lillesand, T., Kiefer, R. W., and Chipman, J. (2014). *Remote sensing and image interpretation*. John Wiley & Sons.

Lumpkin, R. and Pazos, M. (2007). Measuring surface currents with surface velocity program drifters: the instrument, its data, and some recent results. In A. Griffa, A. D. Kirwan, A. J. M. T. O. and Rossby, T., editors, *Lagrangian analysis and prediction of coastal and ocean dynamics*, chapter 2, pages 39–67. Cambridge University Press New York, NY.

Maus, V. W. (2016). *Land Use and Land Cover Monitoring Using Remote Sensing Image Time Series*. PhD thesis, Instituto Nacional de Pesquisas Espaciais, São José dos Campos.

Melton, J. and Eisenberg, A. (2001). Sql multimedia and application packages (sql/mm). *ACM SIGMOD Record*, 30(4):97–102.

OGC (2010). OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 2: SQL option. Technical report, Open Geospatial Consortium.

OGC (2011). OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture. Technical report, Open Geospatial Consortium.

Regis, L., Souza, W. V., Furtado, A. F., Fonseca, C. D., Silveira Jr, J. C., Ribeiro Jr, P. J., Melo-Santos, M. A. V., Carvalho, M. S., and Monteiro, A. (2009). An entomological surveillance system based on open spatial information for participative dengue control. *Anais da Academia Brasileira de Ciências*, 81(4):655–662.

Richards, J. A. and Richards, J. (1999). *Remote sensing digital image analysis*, volume 3. Springer.

Roy, D. P., Wulder, M., Loveland, T., Woodcock, C., Allen, R., Anderson, M., Helder, D., Irons, J., Johnson, D., Kennedy, R., et al. (2014). Landsat-8: Science and product vision for terrestrial global change research. *Remote Sensing of Environment*, 145:154–172.

Sinton, D. (1978). The inherent structure of information as a constraint to analysis: mapped thematic data as a case study. In Dutton, G., editor, *Harvard Papers on Geographic Information Systems*, volume 6, pages 1–17. Harvard University Cambridge, MA.

Worboys, M. (2005). Event-oriented approaches to geographic phenomena. *International Journal of Geographical Information Science*, 19(1):1–28.

The Performance Relation of Spatial Indexing on Hard Disk Drives and Solid State Drives

Anderson Chaves Carniel[1], **Ricardo Rodrigues Ciferri**[2], **Cristina Dutra de Aguiar Ciferri**[1]

[1]Department of Computer Science – University of São Paulo
13.560-970 – São Carlos – SP – Brazil

accarniel@gmail.com, cdac@icmc.usp.br

[2]Department of Computer Science – Federal University of São Carlos
15.565-905 – São Carlos – SP – Brazil

ricardo@dc.ufscar.br

Abstract. *Spatial indexing is a core aspect in spatial databases and Geographic Information Systems. Commonly, spatial indices like the R-tree and the R*-tree consider Hard Disk Drives (HDDs) as the main storage device in data management. On the other hand, flash memories in the form of Solid State Drives (SSDs) have widely been adopted in data servers. Due to their unique characteristics like the erase-before-update operation and the asymmetry between read and write costs, the impact of spatial indexing on SSDs needs to be studied. In this paper, we conduct an experimental evaluation in order to analyze the performance relation of spatial indexing on HDDs and SSDs. As a result, we show experimentally that spatial indices originally designed for HDDs should be redesigned for SSDs in order to take into account the unique characteristics of SSDs. We also propose guidelines to improve the performance of spatial indexing on SSDs by considering these characteristics.*

1. Introduction

Several advanced applications like agriculture systems, urban planning, and public transportation planning make use of geometric or spatial information in order to represent spatial phenomena. These applications commonly employ specialized systems to manage, analyze, and store a spatial phenomenon, such as *spatial database systems* and *Geographic Information Systems* (GIS). In order to aid in decision making, *spatial queries* return a set of spatial objects that satisfy some *topological predicate* (e.g., overlap, disjoint, inside) according to a given object [Gaede and Günther 1998]. For instance, a *spatial selection* that finds all rivers intersecting the Sao Paulo state. To speed up the processing of spatial queries, spatial indices are employed, such as the R-tree [Guttman 1984] and the R*-tree [Beckmann et al. 1990].

In general, these indices manage spatial objects stored in *Hard Disk Drives* (HDD), and thus take into account the slow mechanical access and the cost of search and rotational delay of magnetic disks. On the other hand, *flash memories* have widely been utilized in many applications since they are increasingly being used as the main storage device in mobile phones and laptops [Mittal and Vetter 2015, Suzuki and Swanson 2015]. *Solid State Drives* (SSD) are robust forms of flash memories that have also been popular in data centers and data servers. Flash memories have many positive characteristics compared to HDDs, such as (i) smaller size, (ii) lighter weight, (iii) lower power consumption, (iv) better shock resistance, and (v) faster reads and writes.

Although the positive characteristics of flash memories, these memories have other unique characteristics that may affect the performance of many applications [Chen et al. 2009, Mittal and Vetter 2015]. The main unique characteristic is the asymmetry between read and write costs, where a write requires more time and power consumption than a read. Another characteristic is that a random write is slower than a sequential write since the flash memory is block-oriented. Therefore, an evaluation of the performance of disk-based spatial indices (i.e., originally designed for HDDs) on flash memories is needed.

There are few approaches [Emrich et al. 2010, Fevgas and Bozanis 2015] that conduct an experimental evaluation of spatial indexing on flash memories. There are also approaches [Wu et al. 2003, Lv et al. 2011, Sarwat et al. 2013] that make use of a buffer in the main memory, which stores the modifications of the spatial indices and thus, avoid random writes to the flash memory. When the buffer is full, a flushing operation is performed by applying sequential writes to the flash memory. However, these approaches face several problems, such as the lack of an analysis between the performance relation of spatial indexing on HDDs and flash memories, the conduction of a limited experimental evaluation that do not focus on the spatial query processing, and the lack of analysis of the parameterization impact of spatial indices on flash memories.

The goal of this paper is threefold. The first goal aims to check the relative performance results of an index on a SSD and a HDD, that is, if a spatial index that show the best results on the HDD also shows the best results on the SSD and vice-versa. For this purpose, we conducted an extensive experimental evaluation that varied several parameters of different spatial indices. The second goal aims to verify the impact of parameters on flushing operations. For instance, we analyzed if the size of bytes in a flushing operation (i.e., a flushing unit) has relation with the page size used in a spatial index. Finally, the third goal aims to analyze if the parameterization that could benefit the flushing operation also guarantee a good performance in the spatial query processing.

This paper is organized as follows. Section 2 surveys related work. Section 3 summarizes underlying concepts from spatial indexing. Section 4 briefly describes the unique characteristics of flash memories. Section 5 details the conducted experimental evaluation. Section 6 concludes the paper and presents future work.

2. Related Work

There are few approaches [Emrich et al. 2010, Fevgas and Bozanis 2015] that conduct an experimental evaluation of spatial indexing on flash memories. In addition, there also approaches [Wu et al. 2003, Lv et al. 2011, Sarwat et al. 2013] that propose new spatial indices for flash memories based on the R-tree. We classify these approaches according to the following characteristics: (i) the analysis of the performance of spatial indexing on HDDs and SSDs, (ii) the variety of spatial indices used in experiments, (iii) the parameters considered in the indices, and (iv) the focus of the experiments.

Regarding the first characteristic, almost all the approaches do not evaluate the performance of spatial indexing on HDDs and SSDs. This kind of evaluation is important to study and check if the performance of well-known indices (e.g., the R-tree and the R*-tree) are the same on HDDs and SSDs due to the different characteristic of these storage systems. Only [Emrich et al. 2010] conducts a performance evaluation considering both storage systems, HDDs and SSDs.

Regarding the second characteristic, the majority of the approaches [Wu et al. 2003, Lv et al. 2011, Sarwat et al. 2013] employs the R-tree as baseline, while the remaining approaches [Emrich et al. 2010, Fevgas and Bozanis 2015] employ the R*-tree as baseline. However, it is important to consider both the R-tree and the R*-tree in a same experiment due to their good performance reported in the literature. In addition, many approaches [Wu et al. 2003, Lv et al. 2011, Sarwat et al. 2013, Fevgas and Bozanis 2015] also propose new spatial indices for flash memories. These approaches conduct experimental evaluations to check the performance behavior of their indices. But, the lack of studies about the performance of well-known spatial indices on flash memories can ignore other characteristics that could improve their performance.

Regarding the third characteristic, the main parameter analyzed in the approaches is the buffer size in the main memory. Another parameter analyzed in [Sarwat et al. 2013] is the type of the flushing policy. However, parameterization plays an important role in spatial indexing and the variation of specific parameters of a spatial index could impact its performance (see Section 3). For instance, there is a lack of analysis of the relation between the page size considered in a spatial index on the HDD and SSD.

Regarding the fourth characteristic, the main focus of the experiments conducted in the approaches is insertion and deletion operations. The reason is that the flash memories introduce challenges in the maintenance of a spatial index since a random write is an expensive operation. But, it is also important to examine the spatial query processing since this is a very common operation in spatial databases. As a result, there is a lack of studies verifying the impact of query selectivity. For instance, since the flash memories provide fast reads, an open question is how to adapt the spatial organization to take into account this characteristic.

In this paper we conduct a performance evaluation considering different spatial indices with different parameters on both storage systems, HDDs and SSDs. For this purpose, we consider disk-based spatial indices (e.g., the R-tree) and flash-aware spatial indices (e.g., the FAST), which are summarized in the next section. Further, we analyze the performance results for creating and querying spatial indices.

3. Spatial Indexing

In general, hierarchical structures are employed to index spatial objects by using their Minimum Boundary Rectangles (MBR). The most known spatial index is the R-tree [Guttman 1984], which is composed of internal and leaf nodes where indexed spatial objects are stored in leaf nodes. The R*-tree [Beckmann et al. 1990] is a well-known variant of the R-tree that employs other aspects to organize the spatial objects in the nodes, such as the overlapping area among the entries, redistribution to maximize the storage utilization, and the margin of the nodes. Hence, the R*-tree modifies the insert algorithm of the R-tree in order to consider these aspects. Further, it applies a policy of reinsertion

Table 1. Comparison of the HDD[1] and the SSD[2] used in the experiments.

	4KB Random Transfers[3]		Power Consumption[4]		Endurance[5]
	Read	Write	Read	Write	
HDD	0.185MB/s	0.441MB/s	4.1W	4.1W	$> 10^{15}$
SSD	285.156MB/s	109.375MB/s	1.423W	2.052W	$10^4 - 10^5$

in order to decrease the number of split operations. Since these indices consider MBRs, a spatial query is composed of the filter and refinement steps [Gaede and Günther 1998].

With the increasing use of flash memories in applications, new spatial indices for flash memories were proposed [Wu et al. 2003, Lv et al. 2011, Sarwat et al. 2013, Fevgas and Bozanis 2015]. Despite the particular characteristic of each index, in general they make use of a buffer in the main memory that stores the most recent modifications of nodes of a spatial index instead of applying directly them to the flash memory. The goal is to avoid random writes, such as those that occur in splits. A *flushing operation* composed of sequential writes is performed when the buffer is full. Almost all the indices consider all modifications in the flushing operation, which can introduce overhead in write operations. On the other hand, [Sarwat et al. 2013] uses a refined *flushing policy* that chooses a set of nodes with modifications to be flushed. This set form a *flushing unit*, which has a fixed number of nodes to be written.

Parameterization plays an important role in spatial indexing [Gaede and Günther 1998]. Page (node) size as well as minimum and maximum number of entries of leaf and internal nodes are examples of typical parameters used by hierarchical structures. In addition, each index may include specific parameters according to its design. For instance, we are able to vary the reinsertion percentage of the R*-tree. Another example is to vary the buffer and flushing unit sizes of flash-aware spatial indices, which will impact directly on the performance of flushing operations. Therefore, there is a significant performance impact if *different* parameters are used in a spatial index in *different* datasets under *different* storage systems. We mainly conduct an empirical study regarding to it by analyzing the unique characteristics of flash memories, which are summaried in Section 4.

4. Flash Memories

Flash memories in the form of SSDs have been very popular in many applications [Suzuki and Swanson 2015, Mittal and Vetter 2015]. Table 1 shows a comparison of the SSD and HDD used in our experimental evaluation (Section 5). Flash memories have unique characteristics that need to be taken into account in the development of applications for them. Flash memories are block-oriented. This means that a fixed number of flash pages composes a flash block. Commonly, the flash page and block sizes of a SSD are 4KB and 256KB, respectively [Chen et al. 2009, Mittal and Vetter 2015].

[1] `https://support.wdc.com/product.aspx?ID=608&lang=en`

[2] `https://www.kingston.com/us/ssd/consumer/sv300s3`

[3] Measured by Iometer (`http://www.iometer.org/`).

[4] According to the manufactures[1 2].

[5] According to [Mittal and Vetter 2015].

Flash memories support the following operations: erase, read, and write (program) [Chen et al. 2009]. An erase is a block level operation that changes the bits from 0 to 1 of all pages contained in the block, and thus, this is the most expensive operation. The read and write are page level operations with asymmetric costs. A read requires much less time and power consumption than a write (see Table 1). A write is only able to change the bits from 1 to 0 in an erased block. Hence, a write operation in a previously written block requires an *erase-before-update operation* that leads to a very time-consuming operation. On the other hand, a write operation in a previously erased block is a lower latency operation and is denominated as a *sequential write*. Another important characteristic of flash memories is their lower endurance than HDDs (see Table 1). Endurance refers to the maximum number of writes and erases in a block before its unreliableness.

In order to avoid erase-before-update operations and improve the endurance of flash memories, the Flash Translation Layer (FTL) [Wu et al. 2009, Chung et al. 2009] is employed. For this purpose, FTL provides an interface that allows operation systems to use flash memories as a virtual disk and only enables reads and writes for application layers. Further, FTL maps physical page addresses of a flash memory into logical page addresses, which are effectively used by application layers. A logical page is marked as either free, valid, or invalid. A free logical page is able to store data. A valid logical page contains data previously written. A logical page is marked as invalid if a write is performed on a valid logical page; and its new content is stored in another free logical page. This operation is termed as an *out-of-place update* and avoids an erase-before-update operation. A *garbage collection* is needed when space is required and there is no sufficient free logical pages. This operation selects a set of blocks to apply erase operations causing erase-before-update operations. Hence, this is the most expensive operation performed by the FTL. The algorithms of the garbage collection and out-of-place update also consider a *wear leveling* in order to improve the endurance of flash blocks. For a survey of FTLs, see [Chung et al. 2009].

5. Performance Evaluation

We conduct an experimental evaluation in order to analyze the performance relation between the spatial indexing on HDDs and SSDs. Section 5.1 details the experimental setup used in the experiments, while Section 5.2 discusses the obtained results.

5.1. Experimental Setup

We used a real dataset extracted from the OpenStreetMap[6], which consisted of 534.926 complex regions with holes. This dataset represents the buildings of Brazil, such as hospitals, schools, universities, houses, stadiums, and so on. We used the PostgreSQL database management system with the PostGIS extension to store this dataset in a relational table.

In order to conduct our experiments, we employed FESTIval [Carniel et al. 2016]. FESTIval is a PostgreSQL extension that enables the performance comparison of different spatial indices with different parameters under different storage systems by using a unique environment. We used it to compare the performance of the following spatial indices designed for HDDs: the R-tree and the R*-tree. Further, we implemented the

[6] http://www.openstreetmap.org/

Table 2. Configurations of the spatial indices and their corresponding specific parameters used in the experiments.

Configuration Name	Spatial Index	Specific Parameters
Linear R-tree	R-tree	Split: Linear
Quadratic R-tree	R-tree	Split: Quadratic
R-tree 20%*	R*-tree	RP: 20%
R-tree 30%*	R*-tree	RP: 30%
R-tree 40%*	R*-tree	RP: 40%
FAST Linear R-tree	FAST R-tree	Split: Linear; FP: FAST*
FAST Quadratic R-tree	FAST R-tree	Split: Quadratic; FP: FAST*
FAST R-tree 20%*	FAST R*-tree	RP: 20%; FP: FAST*
FAST R-tree 30%*	FAST R*-tree	RP: 30%; FP: FAST*
FAST R-tree 40%*	FAST R*-tree	RP: 40%; FP: FAST*

Table 3. Generic parameters used in the experiments.

Page Size	Minimum Occupancy	Maximum Occupancy
2KB	28	56
4KB	57	113
8KB	114	227
16KB	228	455
32KB	455	910
64KB	910	1820

FAST [Sarwat et al. 2013], which is an approach that adapts a spatial index to be efficiently used in flash memories. Note that FAST does not change the index structure but only changes the way in which the nodes are written in the flash memory. We applied the FAST to be used together with the R-tree and R*-tree, and thus, we formed the FAST R-tree and the FAST R*-tree, respectively.

FESTIval also enables the configuration of a spatial index by using specific and generic parameters. Specific parameters only determine the configuration of a specific spatial index. For the R-tree, we varied its split algorithm. For the R*-tree, we varied the reinsertion percentage (RP) since it impacts on the number of writes in the structure. Further, based on the recommendations for the R*-tree [Beckmann et al. 1990], we considered the close reinsert and fixed the number of elements to be examined in the insertion algorithm as 32. For the FAST-based indices, we considered the FAST* flushing policy (FP) due to its advantages over other flushing policies [Sarwat et al. 2013]. In addition, we studied the effect of the variation of the size of the buffer and the flushing unit (Section 5.2.1). Based on that, we employed the configurations depicted in Table 2.

We also varied generic parameters, which can be applied for any spatial index. For instance, the page size (i.e., the size of a node) and the minimum and maximum number of entries of a node. Further, we considered the DIRECT I/O to avoid operational system caching in read and write operations. Table 3 shows the generic parameters employed for all the configurations in Table 2.

We executed two workloads defined as follows. The first workload focused on the *index construction*, and thus, we collected the time processing in seconds for creating a spatial index (Section 5.2.1). The creation of a spatial index was performed by inserting element by element according to the original insert algorithm of the respective index.

The second workload focused on the *spatial query processing* (Section 5.2.2). Due to its utilization in many applications, we executed *intersection range queries* (IRQ) [Gaede and Günther 1998]. This kind of query returns from a dataset D, a set of spatial objects R that intersects a given query window QW, i.e., $R = \{o|o \in D \wedge intersects(o, QW) = true\}$. Here, we employed rectangular-shaped objects as query windows. We synthetically generated three sets of query windows. Each set was composed of 100 query windows, which had a $x\%$ of area of the bounding box of Brazil. The first set had 0.1%, the second set had 0.5%, and the third set had 1%. They correspond to query windows with low, medium, and high selectivities, respectively.

We collected the total elapsed time in seconds taken to execute the 100 IRQs of each set of query windows. The total elapsed time was calculated as follows. For a specific set of query windows, we executed each IRQ 10 times, collected the average elapsed time of the execution, and then calculated the sum of the average elapsed times of the 100 IRQs. We performed the tests locally to avoid network latency and flushed the system cache after the execution of each IRQ.

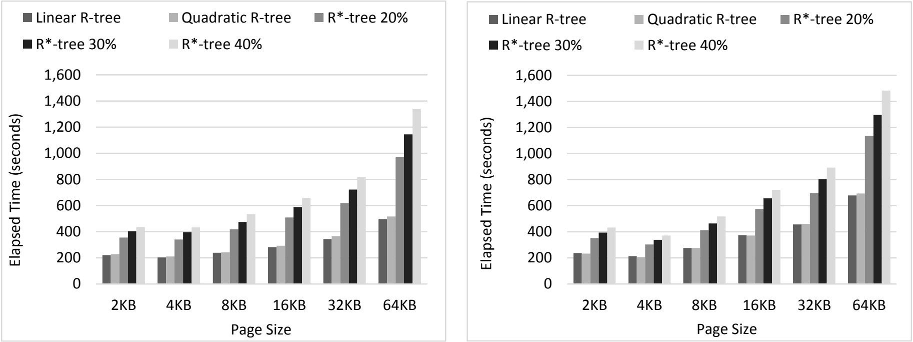
The experiments were conducted on a computer with an Intel® Core™ i7-4770 with frequency of 3.40GHz and 32GB of main memory. For the experiments with HDD we used a 2TB Western Digital with 7200RPM, while for the experiments with SSD we used a 480GB Kingston V300. We employed the Ubuntu Server 14.04 64 bits, PostgreSQL 9.5, PostGIS 2.2.0, and GEOS 3.5.2. GEOS is used by PostGIS for the computation of topological predicates.

5.2. Performance Results

The obtained results related to the spatial index construction and spatial query processing are discussed in Sections 5.2.1 and 5.2.2, respectively.

5.2.1. Spatial Index Construction

Figure 1 depicts the elapsed time for creating disk-based spatial indices according to Tables 2 and 3. We obtained the best performance results for all spatial indices on both storage systems by using the page size equal to 4KB, indicating that this page size should be used for creating spatial indices. Considering the page size equal to 4KB, the performance gain of the indices on the SSD overcame the HDD in at most 14.45% since 4KB is the page size of the SSD. A performance gain is the percentage that shows how much one configuration is more efficient than another configuration. On the other hand, for other page sizes (i.e., 2KB, 8KB, 16KB, 32KB, and 64KB) we obtained best performance results on the HDD with performance gains between 1.88% and 27.05%, which increased as the page size also increased. This case happens since an index construction mix many random writes and reads, and thus, it can degenerate the performance on flash memories (also discussed in [Lee and Moon 2007] and [Chen et al. 2009]).



(performed on the HDD) (performed on the SSD)

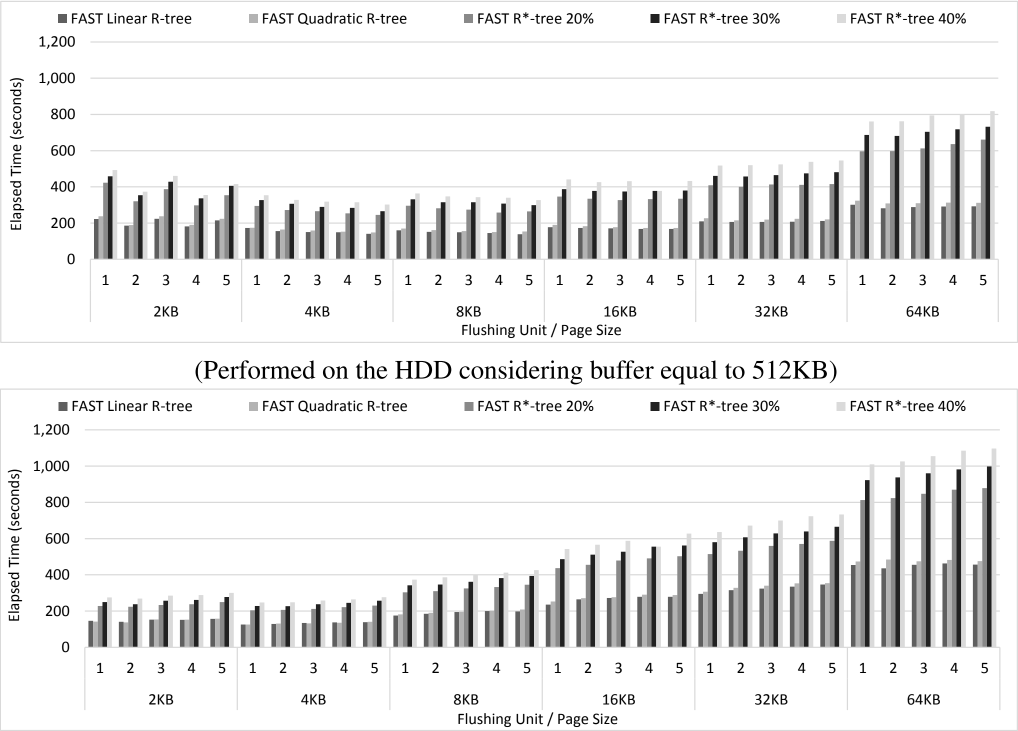
Figure 1. Performance results for creating disk-based spatial indices.

The results demonstrated that spatial indices originally designed for HDD should be redesigned for flash memories in order to take into account the unique characteristics of these memories. FAST is an approach that adapts disk-based spatial indices to be used efficiently on flash memories. In order to evaluate the performance of FAST-based spatial indices, we varied the flushing unit size from 1 to 5. For instance, for configurations with page size equal to 16KB, the flushing unit size equal to 3 performs writes of 48KB in each flushing operation. The flushing unit sizes were combined with each page size, which allowed to verify the impact of this parameter on the index construction. We further considered three different buffer sizes: 128KB, 256KB, and 512KB. In Figure 2, we only report the performance results for creating FAST-based spatial indices by using the buffer size equal to 512KB since it had the same behavior than the other buffer sizes.

The performance results showed that FAST-based indices improved the performance of their counterparts on both storage systems. For instance, the *FAST R*-tree 30%* (Figure 2) configuration with 4KB performed on the HDD imposed a performance gain from 17.23% to 32.84% compared to the *R*-tree 30%* with 4KB (Figure 1) on the HDD. This also occurred for the SSD, where the performance gains were yet more expressive, varying from 23.85% to 32.86%. Further, we clearly note that we did not obtain the same performance behavior on the storage systems for constructing FAST-based indices. We emphasize two main differences. The first difference was that the spatial indices showed best performance results on the HDD by utilizing the page size equal to 4KB and the flushing unit size equal to 5. On the other hand, the best performance results on the SSD utilized the page size equal to 4KB and the flushing unit equal to 1. The second difference was that with the increase of page and flushing unit sizes in the index construction on the SSD, the time processing also increased. This was even much slower than the construction performed on the HDD. For instance, for the page size equal to 64KB, the results demonstrated that the performance gains on the HDD were higher than the SSD because big writes turned out problematic on the SSD.

5.2.2. Spatial Query Processing

Figures 3, 4, and 5 depict the obtained results for processing spatial queries by employing IRQs with 0.1%, 0.5%, and 1%, respectively. Note that we use a different scale to report



(Performed on the HDD considering buffer equal to 512KB)

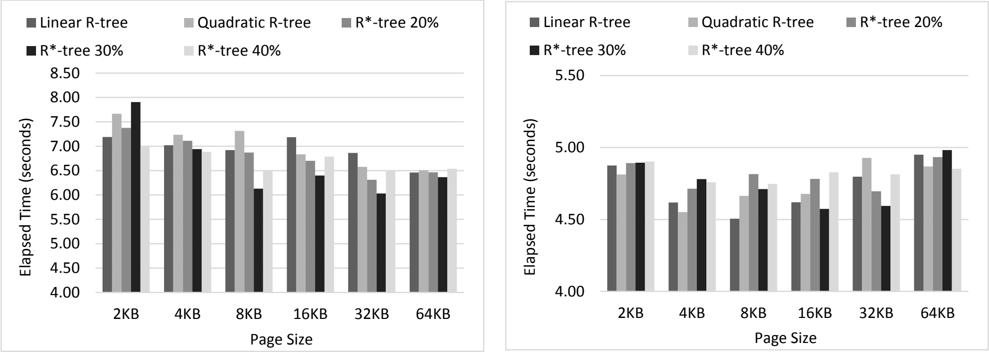
(Performed on the SSD considering buffer equal to 512KB)

Figure 2. Performance results for creating FAST-based spatial indices.

the results for the SSD in order to better compare the performance results. We considered only configurations based on the R-tree and the R*-tree (Table 2) since the FAST does not change the structure of a spatial index (e.g., the *Linear R-tree* using 4KB as page size has the same structure than the *FAST Linear R-tree* using 4KB as page size). Clearly, the performance of spatial query processing on the SSD overcame the HDD in all experiments because of its faster random reads.

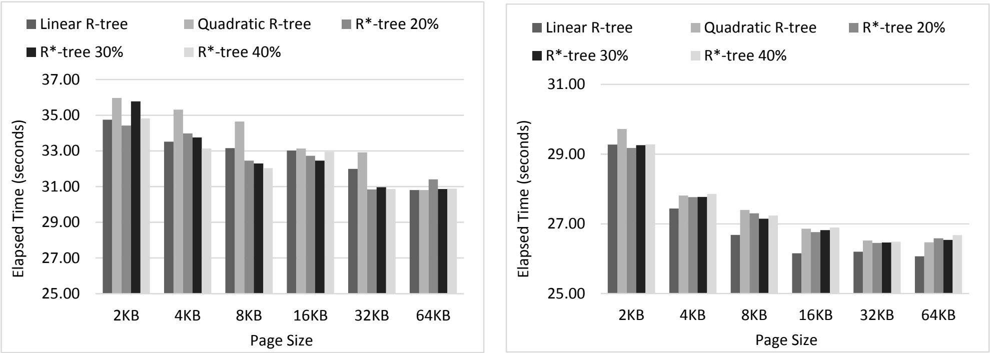
With respect to the execution on the HDD, almost all the configurations improved their performance by increasing page sizes. The reason is that if we use large page sizes, we are able to retrieve more elements from the disk with few reads. The best results were obtained by using the *R*-tree 30%* configuration. For the query windows with 0.1% (Figure 3), the page size equal to 32KB provided the best results since the number of performed reads in the tree required for answering the queries was lower than the other IRQs due to the low selectivity. Hence, for the IRQs with medium and high selectivity (Figures 4 and 5), the use of the page size equal to 64KB improved the elapsed time.

With respect to the execution on the SSD, the performance behavior was slightly different than the HDD. For the IRQs with low selectivity (Figure 3), we obtained the best result by using the *Linear R-tree* configuration with the page size equal to 8KB. For this kind of selectivity, the number of traversed paths on the index to answer the query impacted on the elapsed time. It led to perform sequential reads, which is a very low latency operation [Chen et al. 2009]. Further, for the majority of the cases, the performance



(performed on the HDD) (performed on the SSD)

Figure 3. Performance results of spatial queries considering IRQs with 0.1%.



(performed on the HDD) (performed on the SSD)

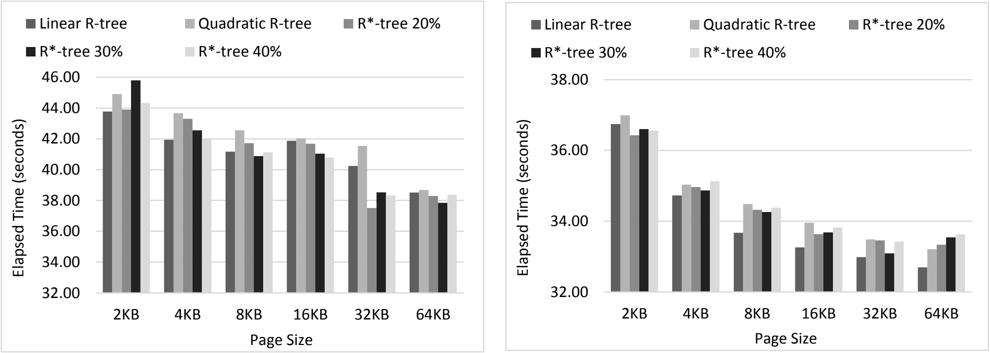
Figure 4. Performance results of spatial queries considering IRQs with 0.5%.

deteriorated as the page size increased. On the other hand, by increasing the page size for medium and high selectivities (Figures 4 and 5), we guaranteed more efficient time processing. This is due to the fact that small page sizes introduce much more random reads.

The results of this experiment demonstrated that the spatial organization for an efficient spatial query processing on SSDs tends to be different than on HDDs. A main finding is that we can exploit the good performance of random reads by using larger page sizes for higher selectivities. Conversely, we can use smaller page sizes for spatial query processing with lower selectivities.

6. Conclusions and Future Work

In this paper, we conducted an extensive experimental evaluation to check the performance relation of spatial indexing on HDDs and SSDs. We considered the disk-based spatial indices R-tree and R*-tree due to their positive characteristics known in the literature. We also considered flash-aware spatial indices based on the FAST due to its positive features, such as the support of flushing policies and flushing units. For all these indices, we varied several parameters and as a result, at least 180 distinct configurations of spatial indices were analyzed in our experiments.



(performed on the HDD) (performed on the SSD)

Figure 5. Performance results of spatial queries considering IRQs with 1%.

As main conclusions we can cite the following performance behaviors, which can lead us to a set of guidelines to further improve the performance of spatial indexing on SSDs. Firstly, in all experiments we see that the direct employment of disk-based spatial indices that showed good performance results on HDDs did not guarantee the best performance results on SSDs. For instance, the R-trees provided better performance on the SSD in the spatial query processing than the R*-trees; but the R*-trees led to best performances on HDDs. This means that the spatial indexing should consider other aspects in order to explore the positive characteristics of flash memories. Based on that, the experiments showed that FAST-based indices improved the elapsed time for the creation of spatial indices on the SSD and even on the HDD since it uses a buffer in the main memory. In general, the performance evaluation showed that employing the FAST for the R-tree with page sizes varying from 4KB to 16KB did not require much time for creating the indices and provided a good performance in the spatial query processing. While the use of the page size equal to 4KB should be recommended for queries with low selectivity, the page size equal to 16KB should be recommended for queries with higher selectivities.

Future work will consider new workloads by including insertions, deletions, and updates of spatial objects in order to analyze the performance behavior for maintaining spatial indices. We will also extend the experiments by considering other spatial indices, like the Hilbert R-tree [Kamel and Faloutsos 1994].

Acknowledgments

This work has been supported by the Brazilian research agencies CAPES, CNPq, and São Paulo Research Foundation (FAPESP). The first author has been supported by the grant #2015/26687-8, FAPESP. The second author has been supported by the grant #311868/2015-0, CNPq.

References

Beckmann, N., Kriegel, H.-P., Schneider, R., and Seeger, B. (1990). The R*-tree: An efficient and robust access method for points and rectangles. *SIGMOD Record*, 19(2):322–331.

Carniel, A. C., Ciferri, R. R., and Ciferri, C. D. A. (2016). Experimental evaluation of spatial indices with FESTIval. In *Proceedings of the Brazilian Symposium on Databases - Demonstration Track*, pages 123–128.

Chen, F., Koufaty, D. A., and Zhang, X. (2009). Understanding intrinsic characteristics and system implications of flash memory based solid state drives. *SIGMETRICS Perform. Eval. Rev.*, 37(1):181–192.

Chung, T.-S., Park, D.-J., Park, S., Lee, D.-H., Lee, S.-W., and Song, H.-J. (2009). A survey of flash translation layer. *Journal of Systems Architecture: the EUROMICRO Journal*, 55(5-6):332–343.

Emrich, T., Graf, F., Kriegel, H.-P., Schubert, M., and Thoma, M. (2010). On the impact of flash SSDs on spatial indexing. In *Int. Workshop on Data Management on New Hardware*, pages 3–8.

Fevgas, A. and Bozanis, P. (2015). Grid-file: Towards to a flash efficient multi-dimensional index. In *Proceedings of the International Conference on Database and Expert Systems Applications*, pages 285–294. Springer International Publishing.

Gaede, V. and Günther, O. (1998). Multidimensional access methods. *ACM Computing Surveys*, 30(2):170–231.

Guttman, A. (1984). R-trees: A dynamic index structure for spatial searching. *SIGMOD Record*, 14(2):47–57.

Kamel, I. and Faloutsos, C. (1994). Hilbert R-tree: An improved R-tree using fractals. In *Int. Conf. on Very Large Data Bases*, pages 500–509.

Lee, S.-W. and Moon, B. (2007). Design of flash-based DBMS: An in-page logging approach. In *ACM SIGMOD Int. Conf. on Management of Data*, pages 55–66.

Lv, Y., Li, J., Cui, B., and Chen, X. (2011). Log-Compact R-tree: An efficient spatial index for SSD. In *Int. Conf. on Database Systems for Advanced Applications*, pages 202–213.

Mittal, S. and Vetter, J. (2015). A survey of software techniques for using non-volatile memories for storage and main memory systems. *IEEE Trans. on Parallel and Distributed Systems*, PP(99):1–14.

Sarwat, M., Mokbel, M. F., Zhou, X., and Nath, S. (2013). Generic and efficient framework for search trees on flash memory storage systems. *GeoInformatica*, 17(3):417–448.

Suzuki, K. and Swanson, S. (2015). A survey of trends in non-volatile memory technologies: 2000-2014. In *IEEE Int. Memory Workshop*, pages 1–4.

Wu, C.-H., Chang, L.-P., and Kuo, T.-W. (2003). An efficient R-tree implementation over flash-memory storage systems. In *ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems*, pages 17–24.

Wu, P.-L., Chang, Y.-H., and Kuo, T.-W. (2009). A file-system-aware FTL design for flash-memory storage systems. In *Conf. on Design, Automation and Test in Europe*, pages 393–398.

Gerenciamento de nuvem de pontos em SGBD: avaliando a extensão PointCloud para PostgreSQL

Vitor C. F. Gomes[1], **Luciane Y. Sato**[2],
Gilberto Ribeiro de Queiroz[2], **Lúbia Vinhas**[2], **Karine Reis Ferreira**[2]

[1]Instituto de Estudos Avançados – Departamento de Ciência e Tecnologia Aeroespacial
CEP 12.228-001 – São José dos Campos – SP – Brasil

[2] Coordenação de Observação da Terra – Instituto Nacional de Pesquisas Espaciais
Caixa Postal 515 – CEP 12.227-010 – São José dos Campos – SP – Brasil

vitor@ieav.cta.br, lusato@dsr.inpe.br
{gilberto.queiroz, lubia.vinhas, karine.ferreira}@inpe.br

Abstract. *The potential observed in various applications and the advancement of technologies for point cloud data acquisition has increased the availability of this type of geospatial data. With increasing volume, new challenges arise in the efficient management of this information. Recently, the Database Management System PostgreSQL has supported this data type through the extension PointCloud. This paper explores this extension, through the analysis of its capabilities and its use in an experimental set of LiDAR data. Performance tests and disk usage metrics are collected for different compression methods. In addition, we present performance testing and disk usage metrics collected for different compression methods.*

Resumo. *O potencial observado em diversas aplicações e o avanço das tecnologias de aquisição de nuvem de pontos têm aumentado a disponibilidade desse tipo de dado geoespacial. Com o aumento do volume, novos desafios surgem quanto ao gerenciamento eficiente dessas informações. Recentemente, o Sistema Gerenciador de Banco de Dados PostgreSQL tem gerenciado esse tipo de dado através da extensão PointCloud. Este trabalho explora essa extensão, através da análise de suas capacidades e do seu uso em um conjunto experimental de dados LiDAR. Além disso, apresentamos testes de desempenho e métricas de utilização de disco coletadas para diferentes métodos de compressão.*

1. Introdução

Nuvens de pontos 3D representam uma categoria essencial de dados geoespaciais usados em uma variedade de aplicações e sistemas de geoinformação. O uso desse tipo de dado tem crescido ao longo da última década, com aplicações em diversas áreas do conhecimento, como em modelagens de objetos e construções, sítios arqueológicos, mapeamentos topográficos, aplicações florestais entre outros [Richter et al. 2015, Martinez-Rubi et al. 2016].

Tecnologias modernas de aquisição e processamento de nuvens de pontos 3D, como ecobatímetros multi-feixe, sistemas de acompanhamento de dados sísmicos e LiDAR (*Light Detection And Ranging*) em plataformas fixas ou móveis ou aeroembarcados, podem produzir de milhares a trilhões de pontos. Esses pontos, além dos dados de posição, podem conter vários atributos, como intensidade, frequência, número de retornos, horário de coleta, entre outros [van Oosterom et al. 2015].

Dentre as tecnologias que produzem nuvens de pontos, o LiDAR tem se destacado nos últimos anos devido aos avanços na técnica de obtenção dos dados. Se por um lado essa tecnologia tem permitido a obtenção de dados com alta precisão e de grandes extensões de áreas, o armazenamento, o gerenciamento e a utilização desses dados representa um desafio devido a sua distribuição irregular, densidade e a quantidade de informações que possuem [Sabo et al. 2014].

O grande volume de dados dessas nuvens dificulta o tratamento de forma eficiente através de infraestruturas de informações e comunicação tradicionalmente utilizadas em aplicações geoespaciais. A maioria das soluções atuais de gestão desses dados apresenta limitações quanto a quantidade de dados que podem ser manipulados de forma eficiente em termos de carga de dados, armazenamento e tempo de recuperação [van Oosterom et al. 2015, Martinez-Rubi et al. 2015].

Recentemente, estruturas de dados escaláveis e estratégias para gerenciamento eficiente de nuvens de pontos têm sido exploradas com diferentes abordagens [Sabo et al. 2014, Rieg et al. 2014, Martinez-Rubi et al. 2015, van Oosterom et al. 2015, Martinez-Rubi et al. 2016]. A eficiência do sistema de gerenciamento de nuvem de pontos depende do tamanho do conjunto de dados, do *hardware* disponível e das funcionalidades requeridas pelos usuários [van Oosterom et al. 2015]. Ferramentas baseadas em processamento de arquivos, como a PDAL [Butler and Gerlek 2016] e a LAStools [Rapidlasso GmbH 2016], são amplamente utilizadas e fornecem funcionalidades para processamento em lote (*batch*) a partir da linha de comando [Martinez-Rubi et al. 2015]. Recentemente, Sistemas Gerenciadores de Banco de Dados (SGBDs), como o Oracle e PostgreSQL têm dado suporte a nuvens de pontos através de tipos específicos de dados ou extensões [Oracle 2016, Ramsey 2016].

A extensão *PointCloud* tem se destacado como uma solução para fornecer ao PostgreSQL suporte a nuvens de pontos. Resultados recentes [van Oosterom et al. 2015] mostraram bom desempenho dessa extensão em relação a solução da Oracle quando utilizando mesmo ambiente de teste.

Neste contexto, este trabalho tem como objetivo explorar as funcionalidades da extensão *PointCloud*, como as capacidades de carga de dados, seleção de pontos e seleção de estatísticas, a fim de identificar suas capacidades e avaliar seu desempenho quanto à ocupação em disco e aos tempos de carga, recuperação e consulta de dados.

2. Gerenciamento de nuvens de pontos

Em geral, há diversas possibilidades para o gerenciamento de grandes volumes de nuvens de pontos. As principais abordagens são as baseadas na manipulação de arquivos, uso de SGBDs e soluções híbridas [Rieg et al. 2014].

As soluções baseadas em arquivos armazenam os dados em arquivos ASCII ou em formatos binários, como o formato LAS [Sabo et al. 2014]. Esse formato tem se tornado um padrão de fato entre os usuários de dados LiDAR e inclui, além das coordenadas tridimensionais, atributos do retorno do laser e parâmetros de voo. Nessa abordagem, um conjunto de ferramentas é utilizado para a seleção, análise, manipulação e visualização dos dados [van Oosterom et al. 2015]. A automatização dessas tarefas normalmente é realizada através de *scripts* para processamento em lote.

Na solução baseada em arquivos, ferramentas como a PDAL e a LAStools são frequentemente utilizadas. A PDAL (*Point Data Abstraction Library*) é uma biblioteca de licença livre com suporte a leitura e escrita dos principais formatos utilizados para nuvens de pontos, como LAS, LAZ, BPF, ILVIS2, entre outros, além de realizar a interface com servidores, como Oracle, PostgreSQL e Greyhound [Butler and Gerlek 2016]. A LAStools é um conjunto de ferramentas desenvolvidas pela *rapidlassso GmbH* e disponibilizada gratuitamente com limitações ou completa através da licença paga. Esse conjunto de ferramentas é amplamente utilizado no setor comercial e governamental [Martinez-Rubi et al. 2015, Rapidlasso GmbH 2016].

Na abordagem baseda em SGBDs, busca-se aproveitar, principalmente, os benefícios de segurança, acesso concorrente, gerenciamento de acesso, escalabilidade, indexação, particionamento, controle de versão, facilidade do uso da linguagem SQL, integração com outros formatos armazenados (vetor e raster) e integração com vasta gama de aplicações já existentes [Sabo et al. 2014]. Três métodos podem ser considerados para o armazenamento de nuvem de pontos em SGBDs: por ponto (também referenciado na literatura por *single-point* [Sabo et al. 2014] ou por *flat-model* [van Oosterom et al. 2015, Martinez-Rubi et al. 2015]), multiponto (*multipoint*) ou por bloco (também encontrado na literatura por *tile-method* [Sabo et al. 2014] ou por *block-model* [van Oosterom et al. 2015]).

No método **por ponto**, cada ponto da nuvem é armazenado em um registro no banco de dados e cada atributo representa um campo na tabela. Essa é a solução mais simples de ser implementada, pois não requer estruturas espefícias. As limitações desse método tratam-se da necessidade de indexar um grande volume de pontos, o que requer grande espaço de armazenamento, e do tempo necessário para atualizar o índice a cada atualização realizada [Sabo et al. 2014].

O método **multiponto** consiste no armazenamento de conjuntos de pontos em blocos em formato binário. Nesta abordagem, os atributos e informações geométricas são armazenados separadamente. Informações sobre a geometria do conjunto de pontos também são armazenadas para facilitar as operações espaciais sobre os dados. A limitação desse método é quanto a realização de consultas que envolvem dados espaciais e atributos, os quais estão em diferentes blocos, podendo causar problemas de desempenho. Outro fator que pode aumentar o tempo das consultas é o acesso arbitrário a pontos dentro dos blocos, os quais não são indexados [Sabo et al. 2014].

No método **por blocos**, os dados são subdivididos em pequenos blocos, onde são armazenados atributos e dados espaciais em formato binário. Neste método, também são armazenados dados sobre o envoltório do bloco de dados para auxiliar na indexação. Este método também herda as limitações da abordagem multiponto quanto ao acesso arbitrário a pontos dentro do bloco. Por conta disso, os blocos devem ter tamanho reduzido, o que gera o aumento do número de blocos e, como consequência, possíveis problemas de desempenho na indexação do grande número de registros [Sabo et al. 2014].

Na solução híbrida, SGBDs são utilizados para armazenar os envoltórios espaciais dos arquivos, e um conjunto de ferramentas é utilizado para manipular diretamente arquivos de nuvem de pontos. Nesta abordagem, consultas ao banco de dados é utilizada para evitar processamento desnecessário em arquivos que não interceptam as consultas realizadas [van Oosterom et al. 2015].

As vantagens e desvantagens de cada solução são dependentes do volume de dados a serem armazenados e das operações a serem realizadas pelos usuários. Recentes trabalhos têm explorado esse tema [van Oosterom et al. 2014, Rieg et al. 2014, Sabo et al. 2014, van Oosterom et al. 2015, Martinez-Rubi et al. 2015], propondo novas estruturas de dados, algoritmos de ordenação dos dados e execução paralela de consultas.

3. Extensão PointCloud

A *PointCloud* é uma extensão para o SGBD PostgreSQL desenvolvida para lidar com o armazenamento e análise de nuvens de pontos, motivada pela necessidade de tratamento dos grandes volumes de dados gerados por sensores LiDAR. Esta extensão introduz dois novos tipos de dados: `PcPoint` e `PcPatch`. O tipo `PcPoint` é um tipo básico que representa um ponto no espaço multidimensional. O tipo `PcPatch` é uma estrutura que agrupa objetos do tipo `PcPoint` para otimizar o armazenamento dos dados. Essa estrutura permite reduzir o número de linhas necessárias para o armazenamento de uma grande quantidade de pontos. Um melhor desempenho é obtido quando os objetos do tipo `PcPoint` são agrupados por proximidade e não há sobreposição espacial entre os objetos do tipo `PcPatch` [Ramsey 2016]. Cada `PcPatch`, além de armazenar um *array* com os pontos, armazena também o retângulo envolvente dos pontos, estatísticas sobre cada dimensão e o total de pontos do *patch* [van Oosterom et al. 2015, Ramsey 2016].

Assim como a extensão PostGIS, a *PointCloud* define uma tabela e uma visão para metadados específicas do domínio de nuvens de pontos. A tabela `pointcloud_formats` é utilizada para armazenar a especificação dos atributos dos pontos que serão armazenados. A visão `pointcloud_columns` fornece a lista de todas as colunas do tipo `point_cloud`.

Muita da complexidade que existe em armazenar, gerenciar e recuperar nuvens de pontos LiDAR vem da necessidade de manipular múltiplas variáveis por ponto, as quais podem ter diferentes características, dependendo do tipo do sensor ou do modo de captura desses dados [Ramsey 2016]. Para tratar essa variabilidade, a extensão *PointCloud* utiliza um esquema XML para descrever o conteúdo dos pontos, que segue o mesmo padrão adotado pela biblioteca PDAL [Butler and Gerlek 2016]. Neste esquema, cada atributo dos pontos, chamado de dimensão, é descrito quanto a sua posição na estrutura de armazenamento, tamanho em *bytes*, tipo de dado, nome, descrição e escala. Além disso, este esquema contém informações sobre a compressão utilizada para armazenar os pontos. Atualmente a *PointCloud* permite a utilização de quatro tipos de compressão [Ramsey 2016]:

- **None**: sem nenhum tipo de compressão;
- **Dimensional**: agrupa em *arrays* cada dimensão dos pontos para compressão. Esse modo é indicado para grupos de dados homogêneos;
- **GHT** (**GeoHashTree**): armazena os pontos em uma árvore onde cada nó armazena os valores comuns compartilhados por todos os nós descendentes [Sabo et al. 2014]. Esse modo de compressão é indicado para grupos com grande quantidade de pontos;

- **LAZ**: utiliza o sistema de compressão LASZip, que é um sistema de compressão aberto e gratuito desenvolvido pela *rapidlasso GmbH* [Rapidlasso GmbH 2016].

Atualmente, a *PointCloud* fornece 28 funções para a manipulação e extração de estatísticas sobre os tipos `PcPoint` e `PcPatch`. Outra característica importante da *PointCloud* é sua integração com a extensão PostGIS, desenvolvida com uma extensão própria chamada *PointCloud PostGIS*. Através dessa integração é possível, por exemplo, realizar consultas para obter o conjunto de *patches* que interceptam uma dada geometria (`PC_Intersection(pcpatch, geometry)`) ou verificar se uma dada geometria intercepta um *patch* (`PC_Intersects(pcpatch, geometry)`).

4. Avaliação da Extensão PointCloud

A fim de avaliar as capacidades da extensão *PointCloud* e seu desempenho quanto aos diferentes modos de compressão frente a tarefas como a carga e recuperação de dados e consultas sobre uma nuvem de pontos, dois conjuntos de testes foram realizados. No primeiro, foi avaliado o espaço ocupado em disco e tempo necessário para a carga e recuperação de uma nuvem de pontos. Neste teste, foram obtidos os tempos médios de carga e recuperação para as quatro opções de compressão disponíveis na extensão *PointCloud*. Para o segundo conjundo de testes, foram avaliados os tempos necessários para a execução de consultas frequentemente utilizadas por usuários de dados desta natureza [Suijker et al. 2014, van Oosterom et al. 2015]. As consultas escolhidas para o teste foram:

- Seleção de pontos que interceptam um polígono retangular (C1);
- Seleção de pontos que interceptam um polígono irregular (C2);
- Seleção de atributo (C3);
- Seleção de estatísticas por *patch* (C4); e
- Seleção de estatísticas globais (C5).

As consultas em SQL de cada seleção são apresentadas na Listagem 1.

Listing 1. Consultas utilizadas nos testes de seleção de pontos

```
-- Selecao de pontos que interceptam um poligono retangular (C1)
CREATE TABLE result1 AS(
  SELECT 1 as id,
    PC_Union(PC_Intersection(l.pa, r.geom)) AS pa
   FROM lidar_tlb l, recorte r WHERE r.gid = 1);

-- Selecao de pontos que interceptam um poligono irregular (C2)
CREATE TABLE result2 AS(
  SELECT 1 as id,
    PC_Union(PC_Intersection(l.pa, r.geom)) AS pa
   FROM lidar_tlb l, recorte r WHERE r.gid = 2);

-- Selecao de atributos (C3)
SELECT PC_Get(PC_Explode(pa), 'Z') FROM lidar_tlb;

-- Extracao de estatisticas por \textit{patch} (C4)
SELECT PC_Summary(pa) FROM lidar_tlb;
```

```
-- Extracao de estatisticas globais (C5)
SELECT PC_Summary(PC_Union(pa)) FROM lidar_tlb WHERE id <= 20000;
```

Para a realização dos testes, foram utilizados dois arquivos LAS com dados coletados por um instrumento LiDAR modelo *Optec Orion*. O primeiro arquivo (Arquivo 1) ocupa 79 MB, possui 2.933.330 pontos, foi adquirido sobre uma área de 46.168 m^2 e possui em média 63,5 pontos por metro quadrado. O segundo arquivo (Arquivo 2) utilizado para os testes possui 1,8 GB, 65.452.394 pontos, representa uma área sobrevoada de 850.291 m^2 e possui em média 76,98 pontos por metro quadrado. Os arquivos possuem 13 dimensões, incluindo as dimensões X, Y e Z, e utilizam o sistema de referência SIRGAS 2000 / UTM zona 20S (`EPSG:31980`).

Para cada arquivo, foi criada uma base de dados com cada um dos 4 tipos de compressão disponíveis na extensão *PointCloud*. Em cada base de dados foi criada uma tabela com duas colunas. Um identificador (id) sequencial do tipo *integer* e um campo para o armazenamento dos `PcPatch` (pa).

A plataforma utilizada para os testes consiste de um servidor com processador i7-5820K 3.30 GHz (6 cores, 2 threads por core), 15MB Cache e 16 GB de memória RAM. Para o armazenamento dos dados, utilizou-se um disco SSD. O sistema operacional utilizado foi o Ubuntu 14.04 64-bit. A extensão *PointCloud* foi instalada em um servidor PostgreSQL versão 9.5. Para a carga dos dados, foi utilizada a biblioteca PDAL versão 1.2.0 [Butler and Gerlek 2016].

Para ambos os arquivos criados, foram utilizados *patches* com até 400 pontos, criados através do filtro `filters.chipper` da PDAL. Para a carga dos dados utilizando o modo ght, foi necessário converter os dados para um sistema com coordenadas geográficas, pois o algoritmo somente aceita coordenadas entre os valores -180/180 e -90/90. Para isso, foi utilizado o filtro `filters.reprojection` da PDAL para converter do sistema de referência SIRGAS 2000 para o WGS 84 (`EPSG:4326`). Para todas as cargas, foi utilizado o esquema XML gerado pela PDAL e gravado automaticamente na tabela `pointcloud_formats` durante a importação. A exceção foi com o modo de compressão LAZ, que por padrão gerou um esquema para o modo sem compressão, diferente do informado no arquivo de configuração. Para garantir o uso do modo de compressão LAZ, o esquema foi criado manualmente e informado à PDAL durante a carga dos dados.

A partir da definição dos testes, foram elaborados *scripts* e consultas SQLs para serem executados em cada uma das base de dados. Para a coleta do tempo de carga e recuperação dos dados foi utilizada a ferramenta `time` do terminal do Linux. Para as consultas em SQL, foi utilizado o comando `\timing` do `psql`. O tamanho ocupado por cada base de dados foi obtido através do comando `\l+` do `psql`.

Para o segundo conjunto de testes, cada consulta foi executada 6 vezes, sendo descartado o tempo da primeira execução, e calculado o tempo médio das 5 restantes. Esse procedimento foi adotado para evitar que dados em *cache* afetassem seletivamente algumas das consultas em análise. Antes da realização das consultas, a base foi otimizada através do comando `VACUUM FULL`.

5. Resultados

Os resultados quanto ao espaço ocupado em disco e tempo de carga e recuperação dos dados dos dois arquivos são apresentados na Tabela 1. Os valores ausentes na tabela indicam erros ocorridos na extensão *PointCloud* durante a execução dos testes. Para ambos os conjuntos de testes ocorreram erros associados aos limites no tamanho da consulta retornada, os quais são específicos para cada modo de compressão.

Com os resultados, é possível observar que há pouca variação na velocidade de carga e recuperação quando comparamos os diferentes modos de compressão. As versões sem compressão (none) e LAZ possuem velocidades equivalentes e melhores em relação as demais (dimensional e ght) para ambos os tamanhos de arquivos. Cabe destaque a variação de espaço necessário para armazenar os dados nos diferentes modos de compressão. O modo de compressão LAZ foi o que apresentou menor tamanho final da base de dados, seguido pelo dimensional. Como esperado, o método sem compressão foi o modo que requisitou mais espaço em disco para armazenar os pontos.

Tabela 1. Tempos médios dos testes de carga e recuperação de dados

	Arquivo 1 (79 MB)				Arquivo 2 (1,8 GB)			
Atributo / Compressão	none	dim.	ght	LAZ	none	dim.	ght	LAZ
Tamanho (MB)	157	51	138	20	3257	886	2630	158
Tempo carga (s)	45,83	49,39	53,89	45,98	1015,23	1090,40	1328,58	1034,31
Vel. carga (MB/s)	1,72	1,60	1,47	1,72	1,81	1,68	1,38	1,81
Tempo recuperação. (s)	29,90	29,76	35,42	27,81	651,02	647,35	853,98	-
Vel. recuperação (MB/s)	3,21	3,23	3,71	3,45	3,42	3,44	2,61	-

ght: GeoHashTree; dim: dimensional

Os tempos das consultas realizadas são apresentados na Tabela 2. Para a consulta C5 aplicada às tabelas com dados do Arquivo 2, foi necessário restringir a seleção em 20.000 *patches*, devido a limitação de memória a ser utilizada pelo retorno da consulta. Atualmente existe uma *Issue* (#46) aberta que trata desse problema no repositório da extensão *PointCloud*.

Tabela 2. Tempos médios das consultas

	Arquivo 1 (79 MB)				Arquivo 2 (1,8 GB)			
Consulta / Compressão	none	dim.	ght	LAZ	none	dim.	ght	LAZ
C1 (s)	6,52	11,34	12,73	6,34	113,64	132,47	-	138,73
C2 (s)	6,19	9,22	12,70	6,36	114,46	132,39	-	138,72
C3 (s)	4,39	5,16	11,23	8,68	95,46	113,06	249,82	209,05
C4 (s)	0,264	0,290	0,258	0,237	4,41	4,79	4,25	3,44
C5 (s)	0,713	22,11	20,68	2,18	1,64*	57,10*	57,07*	5,79*

ght: GeoHashTree; dim: dimensional;
*: consulta limitada aos 20.000 *patches*.

De maneira geral, observa-se que sem compressão as consultas são realizadas em menor tempo. Este comportamento era esperado, uma vez que no modo sem compressão, os dados podem ser acessados sem a necessidade de serem descompactados. Para o Arquivo 1, outro modo de compressão que se destaca é o LAZ, sendo o segundo melhor tempo em quase todas as consultas. O desempenho observado pode ser justificado pelo fato do modo de compressão LAZ permitir que os dados compactados possam ser acessados diretamente pela aplicação, sem a necessidade de serem descompactados previamente em disco [Isenburg 2011]. A exceção é observada na consulta C3, onde o modo de compressão dimensional tem o segundo melhor tempo de resposta. A consulta C3 trata da seleção de um atributo específico dos pontos, o que pode justificar o melhor desempenho, uma vez que neste modo de compressão as dimensões são armazenadas separadamente.

Ainda para o Arquivo 1, observa-se um pobre desempenho para a operação da extração de estatística globais para os modos de compressão dimensional e ght. Esse mesmo comportamento é visto para o Arquivo 2. Nesta consulta, todos os *patches* são agrupados em um único `PcPatch` para, em seguida, ser realizado o cálculo das estatísticas.

Para o Arquivo 2, destaca-se o modo de compressão dimensional para as consultas C1, C2 e C3. Para as consultas C4 e C5, o modo LAZ teve melhor desempenho.

6. Considerações Finais

Este trabalho apresenta o estado da arte das ferramentas de software livre existentes para o gerenciamento de nuvens de pontos. Além disso, faz uma avaliação da extensão *PointCloud* através de testes em 4 modos de compressão para avaliar o seu funcionamento, o uso de espaço em disco e o tempo necessário para a realização de consultas frequentemente utilizadas.

Os resultados obtidos indicam que os modos de compressão afetam significativamente o espaço ocupado pelos dados em um SGBD e o tempo necessário para a realização das consultas. Verifica-se que de fato a decisão sobre qual modo utilizar depende do volume dos dados a serem geridos e das operações que serão realizadas sobre eles, como também é observado no trabalho de [van Oosterom et al. 2015].

As limitações e erros encontrados com o uso dos métodos de compressão ght e LAZ indicam ainda baixa maturidade dessas implementações na extensão *PointCloud*. Pelos testes realizados neste trabalho, acreditamos que o modo de compressão *dimensional* ou sem compressão sejam os mais adequados para uso no momento.

Uma das vantagens da extensão *PointCloud* é ter disponível uma biblioteca como a PDAL e as ferramentas associadas para realizar a carga e recuperação dos dados.

Os resultados obtidos neste trabalho nos motivam a estabelecer uma continuidade. Planejamos avaliar um grupo mais amplo de funções e tamanhos diferentes de *patches* em um conjunto maior de dados LiDAR, compatível com levantamentos que estão sendo realizados por projetos como o Paisagens Sustentáveis Brasil - EMBRAPA/USFS e o Monitoramento Ambiental por Satélite no Bioma Amazônia - INPE. Além disso, planeja-se estruturar e realizar experimentos que comparem a utilização de nuvens de pontos utilizando a abordagem por arquivos e a baseada em SGBD em uma aplicação real dos dados.

Agradecimentos

Os dados LiDAR utilizados neste trabalho foram adquiridos pelo Projeto Paisagens Sustentáveis Brasil, apoiado no Brasil pela Embrapa e pelo Serviço Florestal, Agência para o Desenvolvimento Internacional e Departamento de Estado dos Estados Unidos.

Referências

Butler, H. and Gerlek, M. (2016). PDAL - Point Data Abstraction Library. `http://www.pdal.io`. Acessed: 04 sep 2016.

Isenburg, M. (2011). Laszip: lossless compression of lidar data. `http://lastools.org/download/laszip.pdf`. Acessed: 29 oct 2016.

Martinez-Rubi, O., de Kleijn, M., Verhoeven, S., Drost, N., Attema, J., van Meersbergen, M., van Nieuwpoort, R., de Hond, R., Dias, E., and Svetachov, P. (2016). Using modular 3d digital earth applications based on point clouds for the study of complex sites. *International Journal of Digital Earth*, pages 1–18.

Martinez-Rubi, O., van Oosterom, P., Gonçalves, R., Tijssen, T., Ivanova, M., Kersten, M. L., and Alvanaki, F. (2015). Benchmarking and improving point cloud data management in monetdb. *SIGSPATIAL Special*, 6(2):11–18.

Oracle (2016). Point cloud-related object types. `http://docs.oracle.com/database/121/SPATL/GUID-D347DC78-8782-4BB5-995D-0315B3DD2AB4.htm`. Acessed: 04 sep 2016.

Ramsey, P. (2016). Pointcloud: a PostgreSQL extension for storing point cloud (LIDAR) data. `https://github.com/pgpointcloud/pointcloud`. Acessed: 04 sep 2016.

Rapidlasso GmbH (2016). Lastools. `https://rapidlasso.com`. Acessed: 04 sep 2016.

Richter, R., Discher, S., and Döllner, J. (2015). Out-of-core visualization of classified 3d point clouds. In *Lecture Notes in Geoinformation and Cartography*, The Selected Papers of the 3D GeoInfo 2014, pages 1863–2246. Springer International Publishing.

Rieg, L., Wichmann, V., Rutzinger, M., Sailer, R., Geist, T., and Stötter, J. (2014). Data infrastructure for multitemporal airborne lidar point cloud analysis – examples from physical geography in high mountain environments. *Computers, Environment and Urban Systems*, 45:137 – 146.

Sabo, N., Beaulieu, A., Belanger, D., Belzile, Y., and B., P. (2014). The geohashtree: a multi-resolution data structure for the management of point clouds. Technical notes 4, Minister of Natural Resources Canada.

Suijker, P., Alkemade, I., Kodde, M. P., and Nonhebel, A. (2014). User requirements massive point clouds for esciences (wp1). Technical report, DelftUniversityof Technology.

van Oosterom, P., Martinez-Rubi, O., Ivanova, M., Horhammer, M., Geringer, D., Ravada, S., Tijssen, T., Kodde, M., and Gonçalves, R. (2015). Massive point cloud data management: Design, implementation and execution of a point cloud benchmark. *Computers & Graphics*, 49:92 – 125.

van Oosterom, P., Ravada, S., Horhammer, M., Rubi, O. M., Ivanova, M., Kodde, M., and Tijssen, T. (2014). Point cloud data management. *IQmulus Workshop on Processing Large Geospatial*.

Novo Método de Enquadramento de Objetos Espaciais Complexos em Histogramas Espaciais

Isabella de Freitas Nunes[1], **Thiago Borges de Oliveira**[1]

[1]Instituto de Ciências Exatas e Tecnológicas (ICET)
Universidade Federal de Goiás, Regional Jataí (UFG)
BR 364, KM 195, 3800 – 75.801-615 – Jataí – GO – Brasil

idfn.ufg@gmail.com, thborges@ufg.br

Abstract. *The selectivity estimate is an important metric when selecting efficient execution plans on spatial databases. However, little effort was dedicated to enhance the methods and data structures which support the calculations of these estimations. In this paper we proposed an enhancement in the method used to make a multidimensional grid histogram. The proposed method reduced the error in the estimation up to 30.16%, when estimating the cardinality of spatial window queries, compared to the grid histogram construction method originally proposed.*

Resumo. *A estimativa de seletividade é uma importante métrica para escolha de planos de execução eficientes em banco de dados espaciais. No entanto, pouco estudo foi dedicado ao aprimoramento dos métodos e estruturas de dados que formam a base para o cálculo desta estimativa. Neste artigo propomos um aprimoramento no método de construção dos histogramas multidimensionais de grade, que resultou numa redução do erro de estimativa de até 30.16%, ao estimar a cardinalidade do conjunto resultante de consultas espaciais de janela, comparado com a técnica original de histograma de grade para dados espaciais.*

1. Introdução

Os dados espaciais, representam elementos do mundo real, descrevendo aspectos geográficos e espaciais de fenômenos da superfície terrestre (edificações, ruas, rios, áreas de vegetação, acidentes geográficos e outros) [Rigaux et al. 2002], sendo muitas vezes utilizados para auxiliar a tomada de decisões em larga escala e de grandes organizações. A fim de armazenar, recuperar, combinar, realizar análises variadas acerca de uma região, além de confeccionar materiais cartográficos e outros, estes dados são processados em SDBMS (*Spatial Database Management System*) [Campbell and Shin 2012].

Ao realizar uma consulta, um SDBMS deve ser capaz de definir qual o melhor plano de execução, a fim de conduzir a execução das consultas de forma eficiente. Esta escolha é realizada através de métricas registradas nos metadados do SDBMS. Uma métrica bastante empregada para este fim é a estimativa de seletividade. De acordo com [An et al. 2001], estimar a seletividade de consultas é crucial em um otimizador de consultas, a fim de que o melhor plano de execução seja escolhido.

Uma das técnicas propostas para estimar a seletividade de consultas espaciais é o uso do histograma multidimensional, uma estrutura de dados cuja principal característica é a divisão da extensão espacial do *dataset* (base de dados) em *buckets* (células), que registram a quantidade de objetos espaciais no fragmento do espaço do *dataset* [Mamoulis and Papadias 2001]. Um tipo de histograma multidimensional é o histograma de grade, onde os *buckets* possuem tamanho fixo. Para [Acharya et al. 1999], o uso de histogramas tornou-se popular devido a sua construção ser simples, bem como a pouca utilização de espaço de armazenamento, além de não necessitar que a distribuição da entrada seja conhecida previamente.

Como o histograma é uma técnica de aproximação de um *dataset*, a geometria dos objetos espaciais (linhas, polígonos, e pontos) é também aproximada através de algumas estruturas, que mantém propriedades geométricas essenciais (estruturas conservativas) [Brinkhoff et al. 1994]. De acordo com [Gatti 2000], a estrutura mais utilizada é o mínimo retângulo envolvente (MBR - *minimum bounding rectangle*), ou seja, o menor retângulo com lados paralelos aos eixos das dimensões, que envolve todo o objeto espacial. Suas principais vantagens são o pouco espaço de armazenamento que ocupa e o baixo custo computacional da avaliação inicial dos predicados espaciais (etapa de filtragem).

No entanto, devido ser uma aproximação bem simples de um objeto espacial, o MBR causa problemas em estruturas de dados, devido aos erros de sua aproximação. O principal problema do MBR é a área morta (*dead space*), ou seja, o espaço livre que não é preenchido pelo objeto original. Para um *dataset* grande e composto de muitos objetos complexos, do tipo linha, a área morta pode interferir significativamente na construção das estruturas de dados. Como os histogramas de grade usam o MBR dos objetos, isso pode tornar a estimativa de seletividade das consultas espaciais bem divergente da realidade, e levar à escolha de um plano de execução ineficaz para uma consulta.

Alguns métodos alternativos para aproximar objetos espaciais foram propostos em [Brinkhoff and Kriegel 1994], [Brinkhoff et al. 1994], e [Lee et al. 1996], visando diminuir a área morta das aproximações. Este artigo é um resultado parcial de um projeto de pesquisa, cujo objetivo é investigar como os métodos de aproximação existentes interferem na estimativa de seletividade das consultas espaciais e escolher ou propor um novo método com o intuito de melhorar tal estimativa. Neste artigo, apresentamos o resultado de experimentos que comprovam que o uso de melhores aproximações dos objetos espaciais podem resultar numa melhora significativa da estimativa de seletividade.

O restante do texto está organizado da seguinte forma: na Seção 2 é apresentado um referencial teórico com conceitos relevantes para o entendimento do método proposto, a Seção 3 apresenta os trabalhos relacionados, a Seção 4 apresenta o método inicial proposto, que realiza o enquadramento dos objetos espaciais complexos, a Seção 5 apresenta os resultados iniciais dos experimentos e por fim, a Seção 6 apresenta a conclusão e trabalhos futuros.

2. Referencial Teórico

2.1. Histograma Multidimensional

Uma das técnicas utilizadas para estimar a seletividade de consultas espaciais é o histograma multidimensional, uma estrutura de dados utilizada para simplificar um *dataset* real, cuja principal característica é a divisão da extensão espacial deste *dataset* em uma

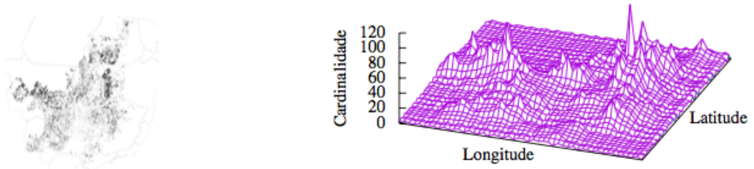


Figura 1. Dataset de alertas de desmatamento e histograma de grade [de Oliveira et al. 2015].

quantidade de *buckets*. De acordo com [Ioannidis and Poosala 1999], o histograma é uma técnica simples e de custo relativamente baixo, porém requer esforço computacional para calcular o número necessário de *buckets* e identificar os valores dos atributos que devem ser associados a eles, a fim de obter uma boa estimativa. A Figura 1 apresenta um dataset espacial de alertas de desmatamento do cerrado brasileiro e um gráfico em 3D, ilustrando os valores das cardinalidades para cada *bucket* em um histograma multidimensional de grade.

2.2. Aproximações de Objetos Espaciais

Os objetos espaciais também são aproximados no momento da construção do histograma. Trabalhar com a geometria real dos objetos é algo complexo, e por tal motivo é necessário utilizar representações simplificadas destes objetos espaciais [Gatti 2000]. Uma das técnicas mais utilizadas para representar objetos espaciais em um histograma é o MBR (*Minimum bounding rectangle* - mínimo retângulo envolvente). Tais aproximações conservam propriedades geométricas essenciais, tais como a posição no espaço e a extensão em cada dimensão [Teotônio 2008]. As duas maiores vantagens de se utilizar o MBR são o pouco espaço utilizado em seu armazenamento e a facilidade na avaliação de predicados espaciais.

3. Trabalhos Relacionados

O método proposto por [Mamoulis and Papadias 2001] enquadra os objetos espaciais na grade do histograma usando o centro do MBR, ignorando a extensão do objeto. Devido a isso, células do histograma, que contém o objeto originalmente, podem não registrar este fato, o que provoca erro na estimativa da cardinalidade do resultado das consultas.

De forma a aprimorar o método anterior, [de Oliveira et al. 2016] fez um estudo e propôs o método de sobreposição parcial, que calcula a sobreposição parcial do MBR em cada célula do histograma sobreposta, adicionando a fração obtida na contagem de objetos que é registrada em cada célula.

Apesar de proverem boas aproximações, conforme apresentado pelos autores, ambos os métodos possuem problemas com objetos do tipo linha, nos quais a extensão geográfica provoca o aumento do erro de seletividade. Um objeto do tipo linha naturalmente sobrepõe várias células do histograma e este fato deve ser registrado para se obter uma estimativa mais precisa. No entanto, o MBR de objetos do tipo linha sobrepõe muitas células do histograma erroneamente, devido a área morta. A próxima seção apresenta um método para lidar com estes dois problemas simultaneamente.

4. Método de Enquadramento de Objetos Espaciais Complexos

Nossa proposta baseia-se no método de sobreposição fracionada, proposto em [de Oliveira et al. 2016] e consiste no uso de uma aproximação mais refinada, para enquadrar melhor o objeto espacial nas células do histograma.

A Figura 2 ilustra esta operação. O método proposto consiste no seguinte procedimento: dado um objeto espacial do tipo linha, divide-se o mesmo em duas partes, gerando dois MBR's parciais. A área coberta pelos MBR's parciais é, frequentemente, menor que a área total do MBR original. Procura-se, então, o melhor ponto para dividir o objeto, de forma que a divisão minimize a área coberta pelos dois MBR's parciais (ou maximize a área morta, destacada na figura). O par de MBR's resultante é então usado para enquadrar o objeto no histograma. As células não sobrepostas pelos MBR's parciais não serão alteradas pelo enquadramento do objeto, e portanto, reduzirão o erro de estimativa sobre o histograma resultante.

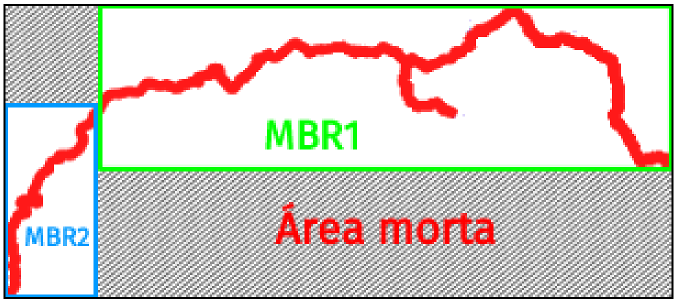


Figura 2. Ilustração do método de enquadramento proposto

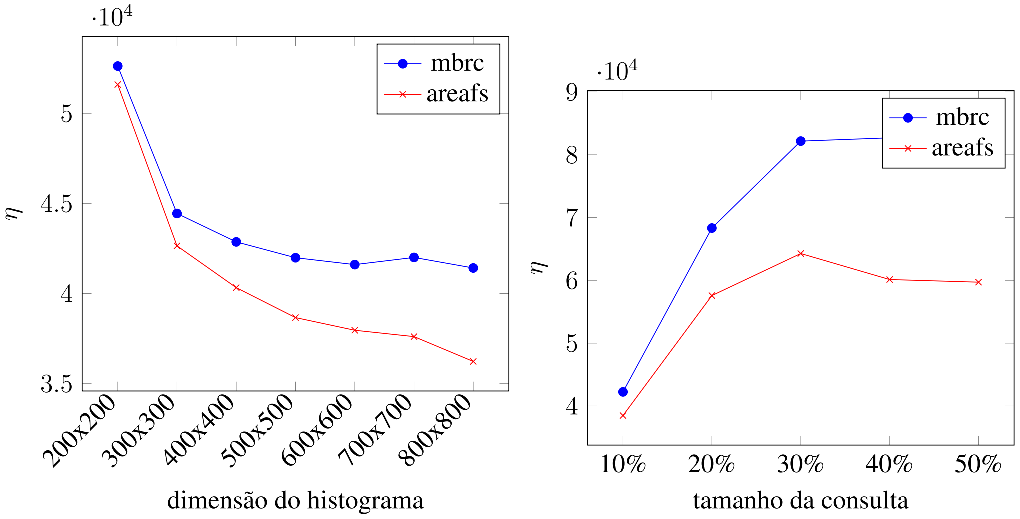
5. Avaliação

Para realizar uma avaliação inicial do método proposto usou-se o *dataset* `ca_roads`, que contém $1.128.694$ objetos espaciais do tipo linha, representando as ruas, avenidas e rodovias do estado da Califórnia - EUA. Este *dataset* é frequentemente empregado em experimentos com estruturas de dados espaciais.

O experimento executado consistiu em construir histogramas espaciais usando dois métodos de enquadramento de objetos: o método proposto em [Mamoulis and Papadias 2001], que usa o centro do MBR para enquadrar os objetos (mbrc), e a proposta descrita na Seção 4 (areafs). Para cada um dos histogramas construídos foi estimada[1] a cardinalidade do conjunto de resultados (c^e) para 500 consultas de janela distribuídas aleatoriamente e uniformemente pelo espaço geográfico do *dataset*. O resultado da estimativa (c^e) para cada consulta foi comparado com a cardinalidade real (c^r), obtida através da execução da consulta no *dataset*. Dado o conjunto de consultas $\mathcal{Q}$, o erro de estimativa η foi obtido somando o erro absoluto para cada consulta, conforme a Equação 1. Para avaliar a eficácia do método proposto, variou-se também as dimensões do histograma criado e o tamanho das consultas de janela, conforme indicado a seguir.

$$\eta = \sum_{q \in \mathcal{Q}} |c_q^r - c_q^e| \quad (1)$$

[1] A estimativa da cardinalidade do conjunto resultante de cada consulta foi obtida usando o método de estimativa sobre histogramas de grade proposto por [Mamoulis and Papadias 2001].



(a) (b)

Figura 3. Comparação de η para os dois tipos de enquadramentos, variando as dimensões do histograma (a), e variando o tamanho da consulta (b).

O gráfico da Figura $5a$ apresenta o resultado do experimento com a variação da dimensão do histograma. Foram construídos histogramas de dimensões $200x200$, $300x300$, $\ldots$, $800x800$, usando os dois métodos de enquadramento. Observa-se que o erro diminui a medida que a dimensão do histograma aumenta. A proporção da redução também é maior nas maiores dimensões. No método (mbrc) o erro praticamente se estabiliza a partir do histograma com dimensão $500x500$. No método proposto (areafs), o erro continua diminuindo, mesmo para os histogramas com maior dimensão. A maior diferença de erro foi de 30.16%, para o histograma de dimensão $800x800$.

Avaliou-se também o efeito do tamanho das consultas na estimativa. Foram executadas consultas de tamanhos 10%, 20%, $\ldots$, 50%, calculados sobre o tamanho de cada dimensão do *dataset*, formando retângulos (janelas), posicionadas de forma aleatória e uniforme na área geográfica do *dataset*. Com as consultas de tamanho maior há uma tendência a aumentar o erro (valor de η), devido as mesmas retornarem mais objetos. O gráfico da Figura $5b$ apresenta os resultados obtidos. O método proposto (areafs) foi mais preciso que o método (mbrc) para todos os tamanhos de consultas testados.

6. Conclusão

A estimativa de seletividade é uma importante métrica para escolha de planos de execução eficientes em banco de dados espaciais. Neste artigo propusemos um aprimoramento no método de construção dos histogramas multidimensionais de grade, que resultou numa redução do erro de estimativa de até 30.16%, quando estimando a cardinalidade do conjunto resultante de consultas espaciais de janela, comparado com a técnica original de histograma de grade para dados espaciais.

O método proposto e experimentado procura exaustivamente a divisão ótima do MBR do objeto espacial, a qual retorna a menor área morta. Devido sua complexidade, pode não ser recomendado na prática para um banco de dados espacial. Na continuação deste projeto, pretendemos investigar algoritmos mais eficientes para encontrar a divisão ótima ou uma boa divisão do MBR através de métodos heurísticos, bem como testar outras técnicas de aproximação dos objetos espaciais, como as propostas por [Lee et al. 1996]. Pretendemos ainda expandir o conjunto de experimentos, incluindo mais *datasets* de objetos espaciais complexos para confirmar a aplicabilidade do método.

Referências

Acharya, S., Poosala, V., and Ramaswamy, S. (1999). Selectivity estimation in spatial databases. *SIGMOD Rec.*, 28(2):13–24.

An, N., Yang, Z.-Y., and Sivasubramaniam, A. (2001). Selectivity estimation for spatial joins. In *Proceedings 17th International Conference on Data Engineering*, pages 368–375. IEEE.

Brinkhoff, T. and Kriegel, H.-P. (1994). *Approximations for a Multi-step Processing of Spatial Joins*, pages 25–34. Springer Berlin Heidelberg, Berlin, Heidelberg.

Brinkhoff, T., Kriegel, H.-P., Schneider, R., and Seeger, B. (1994). Multi-step processing of spatial joins. *SIGMOD Rec.*, 23(2):197–208.

Campbell, J. E. and Shin, M. (2012). *Geographic Information System Basics*.

de Oliveira, T. B., Costa, F. M., and Rodrigues, V. J. d. S. (2015). Definição de Planos de Execução Distribuídos para Consultas de Junção Espacial usando Histogramas Multidimensionais. In *Proceedings of the Brazilian Symposium on Databases*, pages 89–100, Petrópolis, RJ, Brazil.

de Oliveira, T. B., Costa, F. M., and Rodrigues, V. J. d. S. (2016). Distributed execution plans for multiway spatial join queries using multidimensional histograms. 7(1):To appear.

Gatti, S. D. (2000). Fatores que afetam o desempenho de métodos de junções espaciais: um estudo baseado em dados reais. Mestrado, UNICAMP, Campinas.

Ioannidis, Y. E. and Poosala, V. (1999). Histogram-based approximation of set-valued query-answers. In *Proceedings of the 25th International Conference on Very Large Data Bases*, VLDB '99, pages 174–185, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Lee, Y.-J., Lee, D.-M., Ryu, S.-J., and Chung, C.-W. (1996). *Controlled decomposition strategy for complex spatial objects*, pages 207–223. Springer Berlin Heidelberg, Berlin, Heidelberg.

Mamoulis, N. and Papadias, D. (2001). Multiway Spatial Joins. *ACM Transactions on Database Systems*, 26(4):424–475.

Rigaux, P., Scholl, M., and Voisard, A. (2002). *Spatial Databases: With Application to GIS*. Series in Data Management Systems. Morgan Kaufmann Publishers.

Teotônio, F. A. B. (2008). Comparação do desempenho dos ídices r-tree, grades fixas, e curvas de hilbert para consultas espaciais em bancos de dados geográficos. Mestrado do curso de pós-graduação em computação aplicada, Instituto Nacional de Pesquisas Espaciais - INPE, São José dos Campos.

Aplicação em Banco de Dados Espaciais Para o Rastreio de Células Convectivas

Marcos Lima Rodrigues[1], Stephan Stephany[2], Lúbia Vinhas[3], Karine Reis Ferreira[3], Gilberto Ribeiro de Queiroz[3], João Victor Cal Garcia[4], Carlos Frederico de Angelis[4]

[1]Programa de Pós-graduação Computação Aplicada (CAP)
Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 515 - 12.245-97 – São José dos Campos – SP – Brasil

[2]Laboratório Associado de Computação e Matemática Aplicada (LAC)
Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 515 - 12.245-97 – São José dos Campos – SP – Brasil

[3]Divisão de Processamento de Imagens (DPI)
Instituto Nacional de Pesquisas Espaciais (INPE)
Caixa Postal 515 - 12.245-97 – São José dos Campos – SP – Brasil

[4]Centro Nacional de Monitoramento e Alertas de Desastres Naturais (CEMADEN)
Rodovia Presidente Dutra, Km 40 SP-RJ - 12630-000 – Cachoeira Paulista – SP – Brasil

{marcos.rodrigues,stephan.stephany, lubia.vinhas}@inpe.br

{karine.ferreira, gilberto.queiroz}@inpe.br

{joao.garcia, carlos.angelis}@cemaden.gov.br

Abstract. *In this work, we propose the development of a system for thunderstorms cell tracking, based on a spatial database. We also introduce a that performs spatio-temporal clustering operations of lightning data using the DBSCAN algorithm. The spatial database query language were used to define the time window, in order to characterize cells associated with storms. In the analyzed period it was identified and tracked 13 precipitation systems in the radar coverage area located in Jaraguari-MS. Identified cells exhibited good consistency with the intensity core of the thunderstorms.*

Resumo. *Neste trabalho, propomos o desenvolvimento de um sistema para o rastreio de células de tempestades, com o suporte de banco de dados espaciais (SBDE). O método empregado realiza operações de agrupamento espaço-temporal de descargas atmosféricas, através do algoritmo DBSCAN. O SBDE foi usado para dar suporte a consultas com a definição das janelas de tempo do monitoramento. A fim de caracterizar células associadas a tempestades, no período analisado foram identificados e rastreados 13 sistemas precipitantes na área de alcance do radar de Jaraguari-MS. As células identificadas exibiram boa coerência com a parte mais intensa das tempestades.*

1. Introdução

O Brasil, assim como outros países têm experimentado, principalmente na última década um incremento no número de ocorrências de desastres naturais, vide Figura 1. Estudos apontam que existe uma conexão com os fenômenos das mudanças climáticas, bem como uma relação com a ocupação desordenada das cidades [Xavier et al. 2014]. Como forma de mitigar os problemas causados por essas ocorrências, é muito importante que se estude a construção de aplicações que processem dados coletados em tempo real com o intuito de auxiliar no monitoramento e alerta de eventos meteorológicos extremos. As tempestades são eventos extremos que podem causar desastres naturais com perdas de vidas, como deslizamentos de terra e inundações.

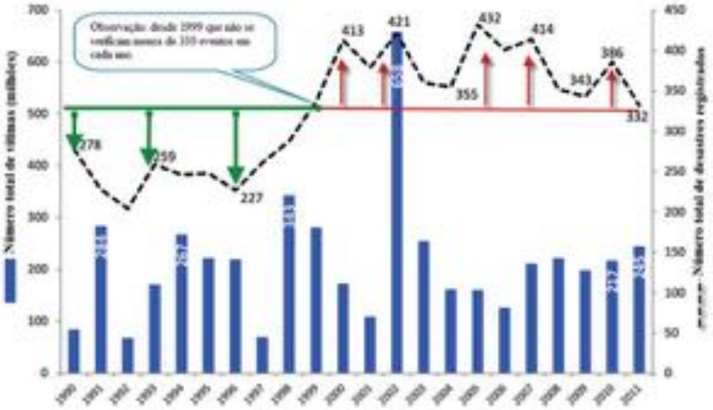


Figura 1. Incremento no número de desastres naturais no Brasil. Adaptada de [MATA-LIMA et al. 2013]

Existem diversos métodos para a realização do rastreio (*tracking*) de células de tempestades, normalmente definidos para a identificação de áreas com precipitação intensa. É comum a utilização de métodos computacionais e estatísticos para determinação do centro de gravidade de áreas definidas a partir de um limiar relacionado à característica da tempestade. Outro método utilizado é a extrapolação de um campo que pode ser de refletividade ou de precipitação, registrado por um satélite ou radar meteorológico. Pode-se ainda definir células de tempestades a partir da definição de agrupamentos de descargas atmosféricas baseados em função da densidade [Steinacker et al. 2000].

Outros trabalhos recomendam a utilização, de forma complementar, da combinação dos dados de descargas atmosféricas, radares meteorológicos e imagens obtidas por satélites meteorológicos, a fim de aumentar a eficiência do processo de monitoramento e previsão de curto alcance de tempestades ([Steinacker et al. 2000] e [Bonelli and Marcacci 2008]). Isso permite analisar com maior acurácia todo o ciclo de vida das tempestades, pois o processo de eletrificação está relacionado ao início do desenvolvimento do processo de convecção e dura até depois da maturação do sistema em média entre 10 e 20 minutos após a primeira descarga ocorre então a precipitação [Steinacker et al. 2000].

Segundo Betz et al. [2008], o registro de descargas elétricas totais, ou seja, de ambos os tipos, intra-nuvem (*IC*) e nuvem-solo (*CG*), servem como um indicativo e precursor de condições de tempestades extremas, especialmente quando a taxa de *IC* aumenta de forma abrupta num curto espaço de tempo. O trabalho apresentado por [Liu and Heckman 2011] utiliza ambos os tipos de descarga atmosférica para a emissão de alertas de tempestades, através do rastreio de células de precipitação convectiva. O uso de funções de densidade e limiares sobre elas permite a delimitação das células de interesse.

Este trabalho tem como objetivo caracterizar e rastrear tempestades a partir de dados de descargas atmosféricas, explorando as capacidades de um Sistema Gerenciador de Banco de Dados (SGBD) com extensão espacial. A abordagem aqui apresentada faz uso de operações espaço-temporais, providas pelo SBDE, para identificar as células de tempestades a partir de agrupamento de descargas atmosféricas. Os sensores possibilitam o registro de descargas atmosféricas, até mesmo em pequenas quantidades, de forma localizada ao longo do tempo, tornando possível a identificação de células associadas a tempestades mais fracas em diferentes instantes de tempo, favorecendo assim o seu agrupamento e rastreio [Betz et al. 2008].

Uma contribuição importante do trabalho é aumentar o conjunto de ferramentas disponíveis em Meteorologia, através da integração de ferramentas de Geoinformática, na solução de problemas desse domínio. Para isso, descrevemos como foram utilizadas ferramentas baseadas no PostgreSQL e sua extensão espacial PostGIS. Esta solução se mostra uma alternativa robusta e sem custos, com potencial de atender a demandas operacionais de um centro para o monitoramento e alerta de desastres.

2. Região do Estudo e Dados

Nesse trabalho foram utilizados registros de descargas atmosféricas da rede Earth Networks em parceria com a BrasilDAT[1], e as imagens do radar meteorológico instalado no município de Jaraguari-MS, ambos fornecidos pelo Centro Nacional de Monitoramento e Alerta de Desastres Naturais (Cemaden). O Cemaden foi criado pelo governo federal em resposta às catástrofes ocorridas em 2011 em todo o país, especialmente na região serrana do Rio de Janeiro. Sua missão é "realizar o monitoramento das ameaças naturais em áreas de riscos em municípios brasileiros suscetíveis à ocorrência de desastres naturais, além de realizar pesquisas e inovações tecnológicas que possam contribuir para a melhoria de seu sistema de alerta antecipado, com o objetivo final de reduzir o número de vítimas fatais e prejuízos materiais em todo o país" [CEMADEN 2016].

A rede BrasilDAT possui 56 sensores instalados no Brasil, e emprega tecnologia que possibilita detecção, localização e identificação do tipo de descarga ocorrida. A Figura 2 apresenta os locais de instalação dos sensores. Segundo [Naccarato et al. 2016], a rede apresenta eficiência de detecção entre 70 e 85%, precisão na localização entre 400 e 700 metros e capacidade de caracterização do tipo de descarga entre 60 e 80%.

O radar instalado no município de Jaraguari-MS é do tipo banda S e fornece uma varredura volumétrica a cada 10 minutos com 13 elevações de *Plan Position Indicator* (PPI). Os PPIs são então reprojetados para formar uma imagem de refletividade em altitude constante denominada *Constant Altitude Plan Position Indicator* (CAPPI), como demonstra a Figura 3. Esse tipo de imagem é comumente utilizada em Meteorologia para observação e estimativa de precipitação. Neste trabalho, servirá para a validação dos resultados alcançados, pela aplicação desenvolvida.

[1] Rede BrasilDAT - `https://www.earthnetworks.com/networks/brazil/`

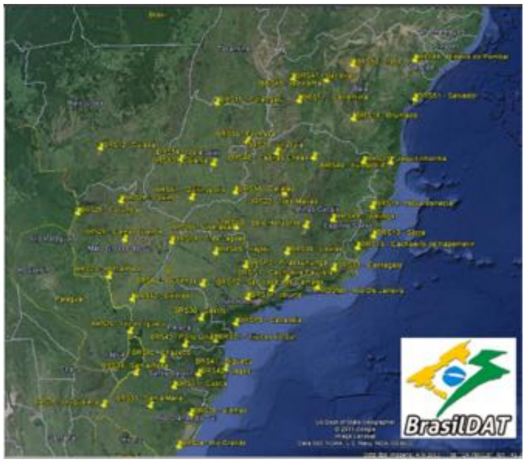
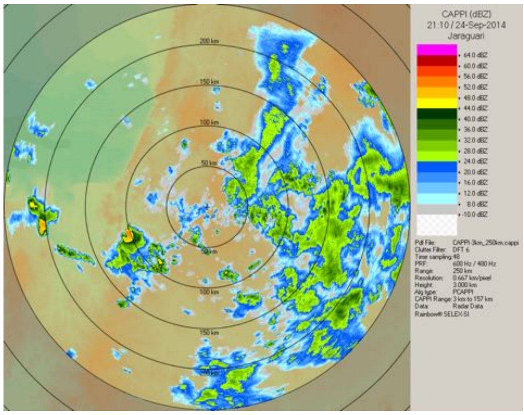


Figura 2. Localização dos sensores da rede BrasilDat. Adaptada de [Naccarato et al. 2016]



Pdf File: CAPPI-3km_250km.cappi
Clutter Filter: DFT 6
Time sampling: 48
PRF: 600 Hz / 480 Hz
Range: 250 km
Resolution: 0.667 km/pixel
Height: 3.000 km
Alg type: PCAPPI
CAPPI Range: 3 km to 157 km
Data: Radar Data
Rainbow® SELEX-SI

Figura 3. Imagem do CAPPI para o radar de Jaraguari do dia 24/09/2014 às 21:10 UTC.
Fonte: Screenshot do software Rainbow 5.

3. Medotologia

Foi criado um banco de dados com os atributos e localização espacial dos registros de descargas atmosféricas de ambos os tipos *IC* e *CG*. Por se tratar de um estudo exploratório, que visa comprovar a possibilidade de se usar um banco de dados geográficos para o acompanhamento de tempestades, usou-se apenas um subconjunto dos dados de descargas elétricas disponíveis. Os dados analisados são referentes ao dia 24/09/2014, quando foram registrados eventos meteorológicos extremos em toda região Centro-Oeste do Brasil, especialmente na região de alcance do radar. Os dados são disponibilizados no formato *Universal ASCII Lightning Format* (UALF); estes são arquivos de texto formatados em colunas de atributos, onde cada linha é referente a uma detecção registrada com os seus atributos como: tipo, data e horário, localização, pico de corrente e altitude (no caso de descargas IC), conforme Tabela 1.

Foi desenvolvido um *Shell Script* em linguagem *Bourn Again Shell* (BASH), que

Tabela 1. Extrato das informações contidas no arquivo de dados de descargas atmosféricas.

tipo	datahora	latitude	longitude	pico_corrente	altura_ic
IC	2014-09-24 00:00:00.807	-26.8998734	-57.8639393	-8801	16175
IC	2014-09-24 00:00:01.362	-28.4522589	-54.8191756	7511	13838
CG	2014-09-24 00:00:01.592	-7.0376185	-54.8472766	-15698	0
IC	2014-09-24 00:00:01.644	-28.3011662	-54.7069739	-12911	10561

serviu de interface para a inserção dos registros no banco. Essa rotina teve como finalidade facilitar o carregamento dos dados, uma vez que em um dia foram registrados mais de 120 mil ocorrências de descargas para toda a América do Sul. Como resultado desse pré-processamento, foi gerado um arquivo com comandos SQL para inserção no banco de dados.

O uso de funções de cálculo de densidade permite transformar as observações discretas das descargas em uma superfície de densidade; e a aplicação de limiares sobre elas permitem a delimitação das células. No presente trabalho, a densidade dos pontos de descargas são determinadas com o auxílio do algoritmo de agrupamento (*clustering*) denominado *Density Based Spatial Clustering of Application with Noise* (DBSCAN) [Ester et al. 1996], que atualmente faz parte do core da extensão espacial PostGIS. O algoritmo DBSCAN é baseado na conectividade entre os pontos através da densidade de pontos de vizinhança, a abordagem é caracterizada pela definição de pontos de centro e de borda do agrupamento a partir da definição de 2 parâmetros: densidade de vizinhança (**Eps**), ou seja, distância máxima entre os pontos e a quantidade mínima de pontos (**MinPts**) de vizinhança para que ele pertença ao agrupamento encontrado. Esses parâmetros foram determinados empiricamente a partir de vários testes e da análise dos resultados obtidos. Ele apresenta como principais vantagens a descoberta automática do numero de agrupamentos, o que facilita o processo para agrupamentos não conhecidos. Além da capacidade de identificar agrupamentos com formatos arbitrários e eliminação de *outliers* [Cassiano 2014].

Além da tabela para o armazenamento dos dados descargas atmosféricas, o banco possui as seguintes tabelas: *cells_table* e *cells_table_tmp*. Essa última serve para armazenar as células identificadas na última janela de tempo processada e possibilitar a determinação de continuidade das tempestades, através da intersecção entre as células de janelas de tempo distintas; todo o registro nessa tabela é mapeado para a tabela *cells_table* através de uma função gatilho ou *trigger*. A detecção e o rastreio das células de tempestades são baseados na sobreposição espacial e temporal dos dados de descargas atmosféricas [Li et al. 2008], para isso foi criada uma função em PL/pgSQL, a qual recebe como parâmetros a data e hora inicial e final para o processamento, além dos valores de Eps e $MinPts$ para o algoritmo de *clustering*. A partir desse intervalo, são definidas janelas de tempo de 10 minutos para o agrupamento e os seguintes parâmetros para o algoritmo de *clustering*: $Eps \approx 8km$ e $MinPts = 2$. Para cada agrupamento foi então definido um círculo envolvente para delimitar as células de descargas associadas a tempestades, essa configuração gerou agrupamentos bem ajustados aos eventos de interesse como demonstra Figura 4. A partir da sobreposição espaço-temporal dessas células, tornou-se possível a identificação, e rastreio das tempestades pela conexão entre as centroides das células "pais" e suas descendentes células "filhas" ([Liu and Heckman 2011] e [Meyer et al. 2013]); essa é uma abordagem simples que possibilita a investigação e extração de informações sobre o ciclo de vida dos sistemas atuantes [Betz et al. 2008].

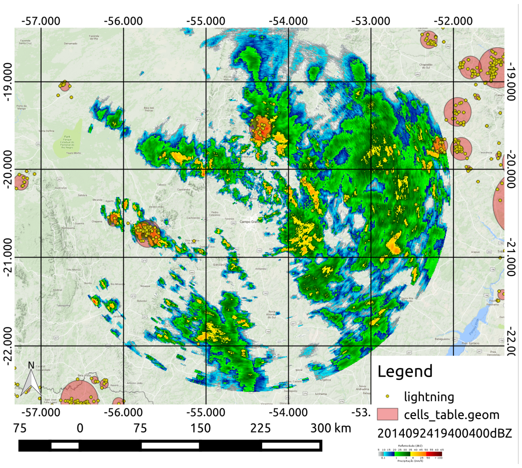


Figura 4. Mapa que evidência a criação do círculo envolvente dos agrupamentos determinados a partir das descargas atmosféricas na janela de 19:40 UTC à 19:50 UTC.

A Figura 5 exibe um fluxograma que ilustra a aplicação da metodologia descrita. Identificando a entrada de dados, processos envolvidos e os resultados alcançados, com a identificação e registros das células de tempestades na tabela *cells_table* na estrutura do banco de dados, bem como a geração de mapas para a visualização através de um Sistema Informações Geográficas (SIG).

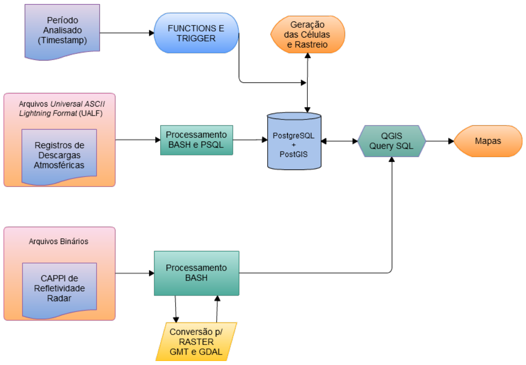

Período Analisado (Timestamp)
FUNCTIONS E TRIGGER
Geração das Células e Rastreio
Arquivos *Universal ASCII Lightning Format* (UALF)
Registros de Descargas Atmosféricas
Processamento BASH e PSQL
PostgreSQL + PostGIS
QGIS Query SQL
Mapas
Arquivos Binários
CAPPI de Refletividade Radar
Processamento BASH
Conversão p/ RASTER GMT e GDAL

Figura 5. Fluxograma dos principais passos descritos na metodologia, para a identificação e rastreio das células de tempestades a partir do processo de clustering dos dados de descargas atmosféricas.

4. Resultados

A data escolhida para esse estudo apresentou um grande número de descargas em todo o país, especialmente concentradas nas regiões Centro-Oeste e Sul, devido ao avanço de uma frente fria, ver Figura 6.

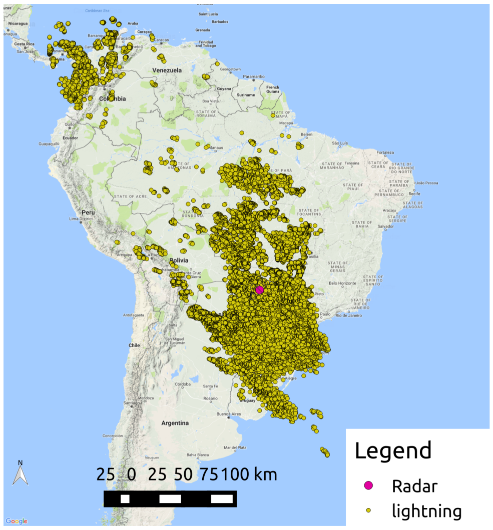


Figura 6. Mapa com a distribuição total de descargas atmosférica para o dia 24/09/2014, foram registrados 128663 eventos entre IC e CG.

A partir do total de registros de ocorrência de descargas atmosféricas, considerou-se o seguinte intervalo de tempo: das 19:30 UTC às 22:50 UTC para análise, e a resolução temporal da janela definida. Foram geradas 1114 células, das quais 116 foram registradas dentro da área de cobertura do radar, conforme demonstra a Figura 7.

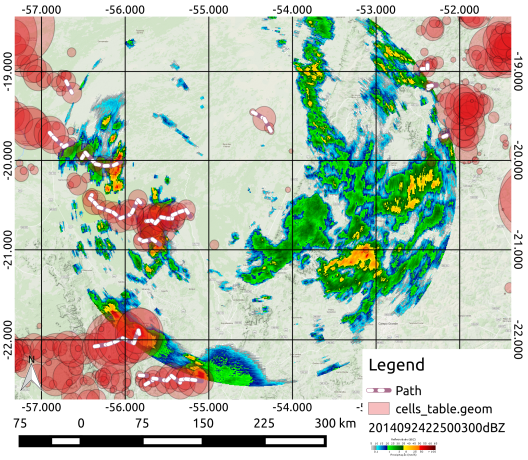


Figura 7. Mapa com o total de agrupamentos criados próximos a área de cobertura do radar, cada célula está associada a um núcleo de tempestade no tempo.

A partir da intersecção do caminho seguido por essas células com a área de do alcance do radar, foram identificados ao longo do período o acompanhamento de 13 sistemas convectivos com deslocamentos territoriais importantes, conforme destaca a Figura

8. Cada cluster que foi gerado foi associado a um sistema convectivo identificado pelo radar por meio de comparação visual.

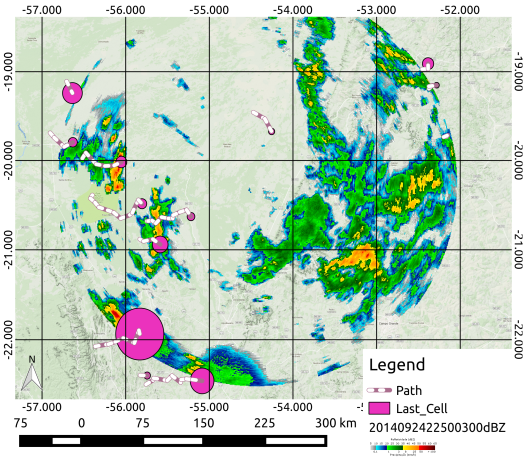


Figura 8. Rastreio dos sistemas precipitantes que que interagiram com o radar de Jaraguari durante o período de analise.

Além de identificar e rastrear a células associadas a tempestades, o sistema permite a coleta de informações sobre o ciclo de vida das mesmas. Essa informação pode ser determinante para orgãos como a defesa civil, fornecendo informações que ajudem a definir a severidade de um evento, favorecendo assim a tomada de decisão por parte de seus agentes. A Figura 9 apresenta, em detalhes, o rastreio e a posição da última célula que demostra o fim do ciclo de vida desses sistemas.

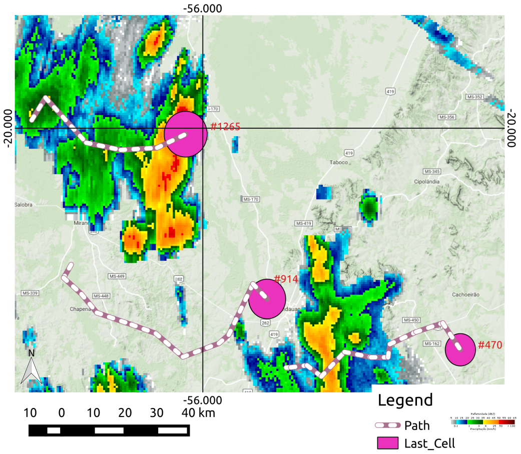


Figura 9. Destaque de sistemas rastreados na área de alcance do radar entre o intervalo de 19:30 UTC à 22:50 UTC.

A partir da informação de identificação das células rastreadas, foram então exportadas do banco de dados algumas das características desses sistemas. A Figura 10 apresenta uma visão geral dessas informações, para tempestades registradas na data de 24 de setembro de 2014. Essas informações permitem acompanhar a iniciação, desenvolvimento e extinção dos sistemas convectivos atuantes. O processo ilustrado pela Figura 5 permite perceber que é possível a operacionalização desse método, e a exibição de forma automática das células de tempestades identificadas ao longo do tempo, bem como o caminho feito por elas. Esses resultados podem ser exibidos através de uma interface web baseada em um servidor de mapas, para que possa auxiliar o trabalho de equipes responsáveis pelo monitoramento e alerta de desastres naturais ligados a eventos extremos de precipitação.

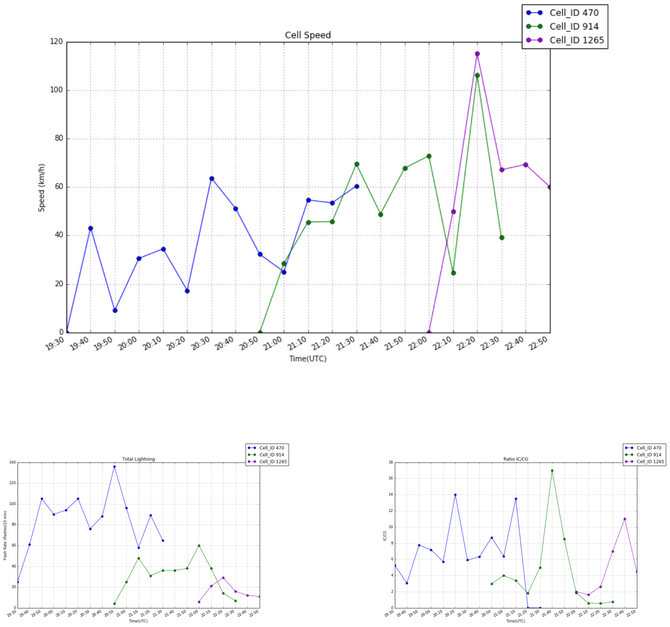


Figura 10. Informações de velocidade, total de descargas e relação entre as descargas do tipo IC/CG para as células de tempestades com ID 470, 914, 1265.

5. Conclusão

O processo de identificação e rastreio de tempestades é uma etapa importante para o desenvolvimento de previsões de curto alcance (*nowcasting*), geralmente realizadas a partir da extrapolação da informação em instantes anteriores [Betz et al. 2008]. O método aqui desenvolvido, com o uso de operações espacias e do algoritmo de *clustering* DBSCAN, apresentou boa coerência com os núcleos de tempestades visualizados na imagem do radar. Como trabalho futuro recomenda-se à adoção de um algoritmo de *clustering* que não dependa da ordem dos dados e que seja flexível na definição da densidade local dos elementos agrupados, como por exemplo o apresentado por [Birant and Kut 2007], favorecendo assim a delimitação de *clusters* adjacentes. As operações providas pelo SBDE facilitaram o processo de geração da informação da continuidade dos sistemas precipitantes devido à sobreposição espacial das células de tempestades.

Por se tratar de um trabalho em andamento, a estrutura desenvolvida para esse estudo foi baseada num aporte inicial de dados e imagens para validação dos resultados, porém para sua utilização de forma operacional, bastaria algumas pequenas adaptações como:

1. Modificar o script para adquirir os dados de descarga de forma automática e em tempo quase real a partir do *webservice* do Cemaden;
2. Utilizar um agendador de tarefas como o crontab[2], para a execução da função PL/pgSQL responsável pelo processo de identificação e rastreio de tempestades;
3. Integrar a consulta de resultados do processamento registrado no banco, a um sistema servidor de mapas como o GeoServer[3], para exibição das células de tempestades e da sua informação de rastreio.

A partir da integração desse trabalho com um sistema servidor de mapas, é possível exibir em *layers* as células de tempestades identificadas e sua informação de rastreio, constituindo assim um ferramental importante para os orgãos que necessitam de informações sobre tempestades no menor tempo possível. A escalabilidade da abordagem proposta será analisada num trabalho futuro, mas foram gerados agrupamentos para uma região extensa, que corresponde a quase toda a América do Sul, apenas a validação desses agrupamentos ficou restrita ao radar de Jaguari.

Referências

[Betz et al. 2008] Betz, H. D., Schmidt, K., Oettinger, W. P., and Montag, B. (2008). Cell-tracking with lightning data from LINET. *Advances in Geosciences*, 17:55–61.

[Birant and Kut 2007] Birant, D. and Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial-temporal data. *Data and Knowledge Engineering*, 60(1):208–221.

[Bonelli and Marcacci 2008] Bonelli, P. and Marcacci, P. (2008). Thunderstorm nowcasting by means of lightning and radar data: Algorithms and applications in northern Italy. *Natural Hazards and Earth System Sciences*, 8(5):1187–1198.

[Cassiano 2014] Cassiano, K. M. (2014). Análise de Séries Temporais Usando Análise Espectral Singular (SSA) e Clusterização de Suas Componentes Baseada em Densidade. *din.uem.br*, page 172.

[CEMADEN 2016] CEMADEN (2016). Missão.

[Ester et al. 1996] Ester, M., Kriegel, H.-p., S, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, pages 226–231.

[Li et al. 2008] Li, X., Ramachandran, R., Movva, S., Graves, S., Plale, B., and Vijayakumar, N. (2008). Storm Clustering for Data driven Wather Forecasting. In *24th Conference on IIPS*, pages 1–5.

[Liu and Heckman 2011] Liu, C. and Heckman, S. (2011). The application of total lightning detection and cell tracking for severe weather prediction. *WMO Technical Conference on Instruments and ...*.

[MATA-LIMA et al. 2013] MATA-LIMA, H., ALVINO-BORBA, A., PINHEIRO, A., MATA-LIMA, A., and ALMEIDA, J. A. (2013). Impactos dos desastres naturais nos sistemas ambiental e socioeconômico: o que faz a diferença. *Ambient. soc., São Paulo*, 16(3).

[2] Manual do crontab agendador de tarefas do Linux, disponível em: `http://man7.org/linux/man-pages/man5/crontab.5.html`, consultado em set. 2016.

[3] GeoServer - servidor de mapas de código aberto, disponível em: `http://geoserver.org/`, consultado em set. 2016.

[Meyer et al. 2013] Meyer, V. K., Höller, H., and Betz, H. D. (2013). Automated thunderstorm tracking: Utilization of three-dimensional lightning and radar data. *Atmospheric Chemistry and Physics*, 13(10):5137–5150.

[Naccarato et al. 2016] Naccarato, K. P., Santos, W. A., Carretero, M. A., Moura, C., and Tikami, A. (2016). Total Lightning Flash Detection from Space A CubeSat Approach. *Proceedings of the ILDC*.

[Steinacker et al. 2000] Steinacker, R., Dorninger, M., Wolfelmaier, F., and Krennert, T. (2000). Automatic Tracking of Convective Cells and Cell Complexes from Lightning and Radar Data. *Meteorology and Atmospheric Physics*, 72:101–110.

[Xavier et al. 2014] Xavier, D. R., Barcellos, C., Barros, H. d. S., Magalhães, M. d. A. F. M., de Matos, V. P., and Pedroso, M. d. M. (2014). Organização, disponibilização e possibilidades de análise de dados sobre desastres de origem climática e seus impactos sobre a saúde no Brasil. *Ciência & Saúde Coletiva*, 19(9):3657–3668.

Index of authors