



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2014/06.13.13.01-TDI

RECONHECIMENTO DE OBJETOS EM IMAGENS ORBITAIS COM O USO DE ABORDAGENS DO TIPO DESCRITOR-CLASSIFICADOR

Juliano Elias Cardoso Cruz

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Larmartine Nogueira Frutuoso Guimarães, e Elcio Hideiti Shiguemori, aprovada em 14 de abril de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP5W34M/3GFCTCH>>

INPE
São José dos Campos
2014

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE

Gabinete do Diretor (GB)

Serviço de Informação e Documentação (SID)

Caixa Postal 515 - CEP 12.245-970

São José dos Campos - SP - Brasil

Tel.:(012) 3208-6923/6921

Fax: (012) 3208-6919

E-mail: pubtc@sid.inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE (RE/DIR-204):

Presidente:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Membros:

Dr. Antonio Fernando Bertachini de Almeida Prado - Coordenação Engenharia e Tecnologia Espacial (ETE)

Dr^a Inez Staciarini Batista - Coordenação Ciências Espaciais e Atmosféricas (CEA)

Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)

Dr. Germano de Souza Kienbaum - Centro de Tecnologias Especiais (CTE)

Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)

Dr^a Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação

Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)

Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)

EDITORAÇÃO ELETRÔNICA:

Maria Tereza Smith de Brito - Serviço de Informação e Documentação (SID)

André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÃO
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21b/2014/06.13.13.01-TDI

RECONHECIMENTO DE OBJETOS EM IMAGENS ORBITAIS COM O USO DE ABORDAGENS DO TIPO DESCRITOR-CLASSIFICADOR

Juliano Elias Cardoso Cruz

Dissertação de Mestrado do Curso de Pós-Graduação em Computação Aplicada, orientada pelos Drs. Larmartine Nogueira Frutuoso Guimarães, e Elcio Hideiti Shiguemori, aprovada em 14 de abril de 2014.

URL do documento original:

<<http://urlib.net/8JMKD3MGP5W34M/3GFCTCH>>

INPE
São José dos Campos
2014

Dados Internacionais de Catalogação na Publicação (CIP)

Cruz, Juliano Elias Cardoso.

C889r Reconhecimento de objetos em imagens orbitais com o uso de abordagens do tipo descritor-classificador / Juliano Elias Cardoso Cruz. – São José dos Campos : INPE, 2014.
xviii + 87 p. ; (sid.inpe.br/mtc-m21b/2014/06.13.13.01-TDI)

Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2014.

Orientadores : Drs. Lamartine Nogueira Frutuoso Guimarães, e Elcio Hideiti Shiguemori.

1. Visão computacional. 2. Processamento de imagem. 3. Reconhecimento de padrões. 4. Descritores locais. 5. Detecção de objeto. I.Título.

CDU 681.3.06



Esta obra foi licenciada sob uma Licença [Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada](https://creativecommons.org/licenses/by-nc/3.0/).

This work is licensed under a [Creative Commons Attribution-NonCommercial 3.0 Unported License](https://creativecommons.org/licenses/by-nc/3.0/).

Aprovado (a) pela Banca Examinadora
em cumprimento ao requisito exigido para
obtenção do Título de **Mestre** em
Computação Aplicada

Dr. Haroldo Fraga de Campos Velho



Presidente / INPE / São José dos Campos - SP

Dr. Lamartine Nogueira Frutuoso
Guimarães



Orientador(a) / IEAv/CTA / SJC Campos - SP

Dr. Elcio Hideiti Shiguemori



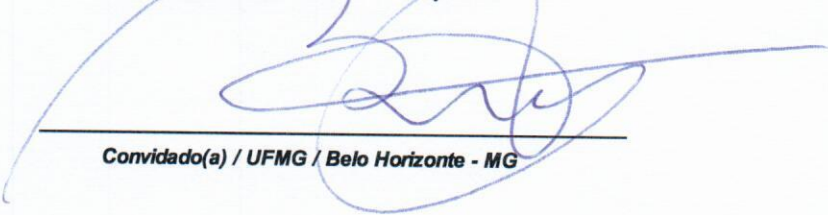
Orientador(a) / IEAv/CTA / São José dos Campos - SP

Dr. Solon Venâncio de Carvalho



Membro da Banca / INPE / SJC Campos - SP

Dr. Antonio de Padua Braga



Convidado(a) / UFMG / Belo Horizonte - MG

Dr. Osamu Saotome



Convidado(a) / ITA / SJC Campos - SP

Este trabalho foi aprovado por:

() maioria simples

unanimidade

Aluno (a): 
Juliano Elias Cardoso Cruz

São José dos Campos, 14 de Abril de 2014

RESUMO

Com a digitalização de informações têm sido apresentadas soluções para problemas que existiam quando as informações eram dispostas de forma analógica, como por exemplo, o fato de alguns sistemas possuírem uma enorme quantidade de dados a serem processados por um operador humano ou sistemas autônomos que necessitam obter informações a fim de tomar decisões sem a ajuda de um operador humano. Portanto, quando se lida com imagens digitais, sendo elas orbitais, aéreas ou obtidas em solo, e deseja-se automatizar um processo que é cansativo e demorado se realizado por um operador humano ou um processo crítico que é responsável pela tomada de decisão em um certo sistema, pode-se recorrer a uma área da computação que é conhecida por Visão Computacional. Assim, esse trabalho de dissertação tem o objetivo de analisar metodologias para o reconhecimento automático de objetos com formato bem definido e com pouca deformação em imagens orbitais. Tais objetos podem ser pessoas, carros, pistas de pouso, campos de futebol, hipódromos, entre outros. São exploradas neste trabalho portanto, abordagens que basicamente utilizam classificadores associados à descritores locais de forma em imagens, mas que tem seu uso em grande maioria aplicadas à detecção e reconhecimento de feições humanas. As abordagens estudadas foram: a associação HOG+SVM, feições Haar-like em cascata e LBP em cascata.

OBJECT RECOGNITION IN SATELLITE IMAGERY WITH DESCRIPTOR-CLASSIFIER APPROACHES

ABSTRACT

Nowadays with the data digitization, it has created solutions to issues that were present particularly at the analog age, e.g. systems that have huge amount of data to be processed by a human operator or an autonomous systems that need to obtain information in order to take decisions without a human operator help. Therefore, when one deals with digital imagery, such as satellite, airborne or ground-level ones, and there is a necessity to automatize a tiring and lengthy process done by a human operator or a critic process responsible for decision-making in a certain system, it is normally employed a computer science known as Computer Vision. Thus, this dissertation work has the objective to present an analysis of methodologies to automatically recognize objects with well-behaved appearance and low deformation level in satellite imagery. Such objects can be human beings, vehicles, runways, soccer fields, hippodromes, among others. It is explored in this work therefore, approaches that basically utilize classifiers associated to image local shape descriptors that have most of their use applied to human features detection and recognition. The studied approaches are: HOG+SVM association, Haar-like feature cascade and LBP cascade.

LISTA DE FIGURAS

	<u>Pág.</u>
2.1 Fase de treinamento.	14
2.2 Fase de classificação.	14
2.3 Etapas da extração do descritor HOG: (a) imagem de entrada, (b) cálculo dos vetores gradientes, (c) agrupamento dos pixels em células, (d) cálculo dos histogramas das células, (e) agrupamento das células em blocos (sobrepósitos) e (f) obtenção do descritor através da cópia dos histogramas de todos os blocos para um vetor.	16
2.4 Imagem orbital de alta resolução de um campo de futebol utilizada como entrada (a), magnitude do vetor gradiente dos pixels (b) e das células (c), e histogramas de orientação das células (d) e dos blocos (e).	17
2.5 Imagem orbital de média resolução de uma pista de pouso utilizada como entrada (a), magnitude do vetor gradiente dos pixels (b) e das células (c), e histogramas de orientação das células (d) e dos blocos (e).	18
2.6 Imagem orbital de alta resolução de uma pista de pouso utilizada como entrada (a), magnitude do vetor gradiente dos pixels (b) e das células (c), e histogramas de orientação das células (d) e dos blocos (e).	19
2.7 Três classificadores possíveis para o problema de duas classes linearmente separáveis. Figura baseada em Theodoridis e Koutroumbas (2008).	21
2.8 Dois hiperplanos e suas margens em um problema de duas classes linearmente separáveis. Figura baseada em Theodoridis e Koutroumbas (2008).	22
2.9 (a) Problema não-linearmente separável com uma fronteira de decisão não-linear, fd_1 , possível, e (b) representação da adição de mais uma dimensão ao problema e a possibilidade de utilizar uma fronteira de decisão linear, fd_2 , para separar o problema. Figura baseada em Theodoridis e Koutroumbas (2008).	24
2.10 Feições Haar-like, onde (a) são feições de borda, (b) feições de linha e (c) feições de ponto. Figura baseada em Lienhart et al. (2003)	26
2.11 Exemplo de aplicação de algumas feições Haar-like, onde (a) é um recorte de uma imagem contendo um campo de futebol, (b) aplicação de feição de borda e (c) aplicação de feição de linha.	26

2.12	Operador LBP: (a) é o fragmento da imagem a ser processado; (b) mostra o nível de cinza de cada pixel; (c) subtrai-se o valor do pixel central dos pixels da periferia; (d) atribui-se 0 para resultados abaixo de zero e 1 para resultados iguais ou maiores a zero; (e) mostra-se os valores que cada pixel representa; e na fase final soma-se os valores de (e) que valem 1 em (d).	27
2.13	(a) Imagem de alta resolução contendo um campo de futebol utilizada como entrada e (b) imagem resultante após a aplicação do operador LBP.	28
2.14	(a) Imagem de média resolução contendo uma pista de pouso utilizada como entrada e (b) imagem resultante após a aplicação do operador LBP.	28
2.15	(a) Imagem de alta resolução contendo uma pista de pouso utilizada como entrada e (b) imagem resultante após a aplicação do operador LBP.	29
2.16	Classificadores em cascata. Onde V representa que um conjunto de atributos passou pelo teste de um determinado nó classificador e F representa a rejeição de um conjunto de atributos.	30
2.17	Algoritmo de treinamento de uma Cascata de Classificadores.	31
2.18	Treinamento de <i>boosting</i> com três classificadores: (a) espaço de atributos com os pesos iniciais, onde o tamanho do círculo representa seu peso, (b) introdução do classificador fraco c_1 , (c) ajuste dos pesos para as amostras mal classificadas, (d) introdução do classificador fraco c_2 , (e) ajuste dos pesos para as amostras mal classificadas, (f) introdução do classificador fraco c_3 , (g) classificador final. Figura baseada em Szeliski (2010)	33
2.19	Algoritmo de treinamento do AdaBoost (FREUND et al., 1999)	34
3.1	Fase de treinamento da abordagem HOG+SVM.	37
3.2	Fase de classificação da abordagem HOG+SVM.	38
3.3	Fase de treinamento das abordagens Haar- <i>like</i> e LBP em cascata.	39
3.4	Fase de classificação das abordagens Haar- <i>like</i> e LBP em cascata.	40
3.5	Amostras do conjunto de treinamento positivo de alta resolução contendo campos de futebol (GOOGLE, 2013).	42
3.6	Amostras do conjunto de treinamento positivo de média resolução contendo pistas de pouso de concreto e asfalto (INPE, 2013). As imagens foram tratadas a fim de permitir uma melhor visualização.	42
3.7	Amostras do conjunto de treinamento positivo de alta resolução contendo pistas de pouso de concreto e asfalto (GOOGLE, 2013).	43
4.1	Representação gráfica dos resultados da abordagem Haar- <i>like</i> em cascata	49
4.2	Representação gráfica dos resultados da abordagem LBP em cascata	51

4.3	Representação gráfica dos resultados da abordagem HOG+SVM	53
4.4	Exemplo de imagens usadas na classificação, onde (a) (b) são verdadeiros positivos e (c), (d) e (e) são falsos positivos. As imagens foram tratadas e recortadas a fim de permitir uma melhor visualização.	54
4.5	Classificadores com melhor equilíbrio entre taxa de acerto e a precisão.	55
4.6	Representação gráfica dos resultados da abordagem Haar- <i>like</i> em cascata	57
4.7	Representação gráfica dos resultados da abordagem LBP em cascata	59
4.8	Representação gráfica dos resultados da abordagem HOG+SVM	61
4.9	Exemplo de imagens usadas na classificação, onde (a), (b) e (c) são verdadeiros positivos e (d) e (e) são falsos positivos.	62
4.10	Classificadores com melhor equilíbrio entre taxa de acerto e a precisão.	63
4.11	Representação gráfica dos resultados da abordagem Haar- <i>like</i> em cascata	66
4.12	Representação gráfica dos resultados da abordagem LBP em cascata	68
4.13	Representação gráfica dos resultados da abordagem HOG+SVM	70
4.14	Exemplo de imagens usadas na classificação, onde (a),(b) e (c) são verdadeiros positivos e (d) e (e) são falsos positivos.	71
4.15	Classificadores com melhor equilíbrio entre taxa de acerto e a precisão.	72

LISTA DE TABELAS

	<u>Pág.</u>
4.1 Reconhecimento de pista de pouso em imagens de média resolução utilizando Haar- <i>like</i> em cascata.	49
4.2 Reconhecimento de pista de pouso em imagens de média resolução utilizando LBP em cascata.	50
4.3 Características do HOG+SVM para esse estudo de caso.	52
4.4 Reconhecimento de pista de pouso em imagens de média resolução utilizando HOG+SVM.	52
4.5 Reconhecimento de pista de pouso em imagens de alta resolução utilizando Haar- <i>like</i> em cascata.	57
4.6 Reconhecimento de pista de pouso em imagens de alta resolução utilizando LBP em cascata.	58
4.7 Características do HOG+SVM para esse estudo de caso.	60
4.8 Reconhecimento de pista de pouso em imagens de alta resolução utilizando HOG+SVM.	61
4.9 Reconhecimento de campo de futebol em imagens de alta resolução utilizando feições Haar- <i>like</i> em cascata.	65
4.10 Reconhecimento de campo de futebol em imagens de alta resolução utilizando LBP em cascata.	67
4.11 Características do HOG+SVM para esse estudo de caso.	69
4.12 Reconhecimento de campo de futebol em imagens de alta resolução utilizando HOG+SVM.	70

LISTA DE ABREVIATURAS E SIGLAS

AdaBoost	–	<i>Adaptative Boosting</i>
FP	–	Falso Positivo
FN	–	Falso Negativo
HOG	–	<i>Histogram of Oriented Gradients</i>
LBP	–	<i>Local Binary Pattern</i>
MB-LBP	–	<i>Multi-Block Local Binary Pattern</i>
RBF	–	<i>Radial Basis Function</i>
SVM	–	<i>Support Vector Machine</i>
TMI	–	Tempo Médio por Imagem
TMpx	–	Tempo Médio por Pixel
VANT	–	Veículo Aéreo Não-Tripulado
VP	–	Verdadeiro Positivo

SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1 Detecção, Reconhecimento ou Identificação?	2
1.2 Processo Humano de Reconhecimento Visual de Objetos	2
1.2.1 Teorias Invariantes ao Ponto de Vista	3
1.2.2 Teorias Dependentes do Ponto de Vista	4
1.2.3 Teorias de Múltiplas Concepções	4
1.3 Reconhecimento Automático de Objetos	4
1.4 Imagens Orbitais	6
1.5 Objetivo	8
2 FUNDAMENTAÇÃO TEÓRICA	11
2.1 Revisão Bibliográfica	11
2.2 Arquitetura Descritor-Classificador	13
2.3 Métodos	14
2.3.1 HOG+SVM	15
2.3.1.1 HOG	15
2.3.1.2 SVM	20
2.3.2 Feições Haar- <i>like</i> e LBP em cascata	25
2.3.2.1 Feições Haar- <i>like</i>	25
2.3.2.2 LBP	26
2.3.2.3 Classificadores em Cascata	29
2.4 Janela Deslizante	35
3 METODOLOGIA	37
3.1 Abordagens de Treinamento e Classificação	37
3.2 Conjunto de Treinamento	39
3.3 Bibliotecas e ferramentas	43
3.3.1 Bibliotecas	43
3.3.1.1 OpenCV	43
3.3.1.2 OpenMP	44
3.3.1.3 Qt	44

3.3.2	Ferramentas	44
3.3.2.1	Extrator de Amostras Semi-Automático	44
3.3.2.2	Treinador e Classificador para Haar- <i>like</i> e LBP em Cascata	45
3.3.2.3	Treinador e Classificador para HOG+SVM	45
4	RESULTADOS	47
4.1	Pista de Pouso em Imagens de Média Resolução	48
4.2	Pista de Pouso em Imagens de Alta Resolução	56
4.3	Campo de Futebol em Imagens Alta Resolução	64
5	CONCLUSÃO	75
	REFERÊNCIAS BIBLIOGRÁFICAS	77

1 INTRODUÇÃO

Com a digitalização de informações surge a possibilidade de criar soluções automatizadas para aplicação em problemas que existiam quando as informações eram dispostas de forma analógica. Neste trabalho apresenta-se o interesse por sistemas automatizados que sejam capazes de lidar com grande quantidade e variações de dados, que geralmente são interpretados por um operador humano para tomada de decisões (AUSTIN, 2010).

Sistemas automatizados ou automáticos são comuns e frequentemente encontrados em diversas situações do cotidiano. Esses sistemas podem ser categorizados de acordo com seu nível de controle que é diretamente relacionado ao grau de intervenção ou interação humana: em sistemas controlados, os operadores possuem controle inteiro ou parcial, como por exemplo em um automóvel comum; sistemas supervisionados realizam uma tarefa instruída pelo operador, pode se dar como exemplo um torno mecânico programável; sistemas automáticos conduzem funções previamente definidas sem a intervenção de um operador, como encontrado em um elevador; e sistemas autônomos, são sistemas adaptativos que aprendem e tomam decisões (THE ROYAL ACADEMY OF ENGINEERING, 2009).

Quando se deseja automatizar processos que fazem o uso de imagens, utiliza-se, normalmente, o que é conhecido por Visão Computacional. Visão Computacional é uma área da ciência da computação que é utilizada quando se lida com imagens digitais sejam elas, orbitais, aéreas ou obtidas em solo, e deseja-se automatizar um processo que é cansativo, perigoso ou demorado se feito por um operador humano ou é um processo crucial responsável pela tomada de decisão em um certo sistema. Esse campo da computação estuda métodos para aquisição, processamento, análise e o entendimento de imagens, sendo em geral dados com alta dimensionalidade obtidos do mundo real para produzir informação numérica ou simbólica (FORSYTH; PONCE, 2011). A visão computacional tem sido empregada em uma grande variedade de problemas, como: reconhecimento óptico de caracteres, inspeção mecânica, fotogrametria, imagens médicas, segurança na condução automotiva, captura de movimentos, vigilância, biometria, entre outros (SZELISKI, 2010).

Esse capítulo de introdução tem por objetivo apresentar ao leitor conceitos chaves para o entendimento desse trabalho de dissertação. Serão abordados temas como: nomenclatura (Seção 1.1), processo humano de reconhecimento (Seção 1.2), processo

de reconhecimento automático (Seção 1.3), imagens orbitais (Seção 1.4) e o objetivo dessa dissertação (Seção 1.5).

1.1 Detecção, Reconhecimento ou Identificação?

Detecção, reconhecimento e identificação são palavras utilizadas no cotidiano em situações em que lhes fazem parecer sinônimas. Elas tratam do mesmo assunto, porém existem diferenças sutis entre elas. A utilização correta desses termos tem ligação direta com o nível de detalhes que um objeto apresenta, isso se tratando de meios visuais, como imagens e vídeos.

Segundo Biberman (1973), a detecção ocorre quando simplesmente um objeto aparece em uma determinada imagem (DONOHUE, 1991), mas esse conceito é vago. Assim a definição feita em Driggers et al. (1997), como sendo a probabilidade de um “borrão” presente na imagem ser o objeto procurado, é mais exata. Para os outros dois termos tanto Biberman, quanto Driggers *et al.* concordam em suas definições. O reconhecimento acontece quando é possível discernir um objeto em uma determinada imagem com clareza suficiente que sua classe possa ser discriminada, por exemplo, automóvel, pessoa, embarcação, etc. Já a identificação, ocorre quando se consegue discernir um objeto presente em uma determinada imagem com clareza suficiente que é possível especificar o tipo do objeto dentro de sua classe, como por exemplo, tanques de guerra T-62, T-72 e M1.

1.2 Processo Humano de Reconhecimento Visual de Objetos

Para um operador humano o processo de reconhecimento de objetos em determinadas imagens pode ser uma tarefa cansativa e suscetível a erro (DONOHUE, 1991). Além de ser entediante, há geralmente em certas aplicações, uma grande quantidade de informação a ser analisada, e ainda em determinados casos, há a necessidade de que o responsável pela tarefa seja especialmente capacitado para o devido fim. Essa dissertação tem como objetivo apresentar diferentes abordagens que automatizam o reconhecimento de objetos. Devido ao fato dos humanos realizarem esse processo, é interessante entender como ele acontece. Esse entendimento traz a possibilidade de se fazer paralelos com os algoritmos computacionais disponíveis atualmente e, também, em um passo mais adiante, criar novos algoritmos tomando como base o próprio processo humano de reconhecimento de objetos.

Apesar de diferentes tipos de objetos necessitarem de diferentes mecanismos para

serem reconhecidos, há etapas em comum, visto que as informações são sempre extraídas do mesmo tipo de dado visual bruto (WARD, 2010). Assim, baseando-se em evidências neuropsicológicas, pode-se dividir o processo de reconhecimento de objetos basicamente em quatro etapas (RAPP, 2001; WARD, 2010):

- processamento de componentes básicos do objeto (cor, profundidade, forma, movimento);
- agrupamento dos componentes básicos de acordo com a similaridade. Essa etapa fornece informações utilizadas na distinção de bordas a fim de montar uma representação visual do objeto e permitir uma segmentação objeto-fundo;
- casamento da representação visual gerada na etapa anterior com descrições estruturais que estão na memória;
- associação de atributos semânticos à representação visual, provendo significado e assim o reconhecimento.

Na maioria das vezes os processos de reconhecimento e identificação de objetos por humanos tem a habilidade de reconhecer um determinado objeto ou classe de objetos independente das condições que este se apresente, como mudança de orientação, tamanho, cor, iluminação ou qualquer outra mudança na apresentação (HUMPHREYS; QUINLAN, 1987). Portanto, o processo necessita ser invariante as mudanças de apresentação dos objetos. Isso é obtido através da extração de atributos comuns existentes na descrição de objetos por diferentes pontos de vista (HUMPHREYS; QUINLAN, 1987; TARR; BÜLTHOFF, 1995). Algumas teorias tentam explicar como os processos de reconhecimento e identificação obtêm invariância às mudanças de apresentação do objeto, dentre as principais pode-se citar as teorias invariantes ao ponto de vista, as teorias dependentes do ponto de vista e as teorias de múltiplas concepções.

1.2.1 Teorias Invariantes ao Ponto de Vista

As teorias invariantes ao ponto de vista defendem que o reconhecimento de objetos é baseado em informações estruturais, como partes individuais, permitindo assim que o reconhecimento aconteça independente do ponto de vista. Existem duas mais conhecidas na literatura: representação por modelo 3D e o reconhecimento por componentes (TARR; BÜLTHOFF, 1995).

A representação por modelo 3D foi proposto em Marr e Nishihara (1978), onde se afirma que o reconhecimento de objetos é alcançado com o casamento da representação do modelo 3D obtido do objeto no campo de visão com representações de modelos 3D armazenados na memória. No casamento dos modelos rotaciona-se mentalmente a representação em 3D do objeto e compara-se com o modelo em memória. Isso se deve ao fato de somente a descrição canônica do objeto ser armazenado em memória.

Já o reconhecimento por componentes é uma extensão do modelo anterior e foi proposto em Biederman (1987). Esse modelo afirma que qualquer objeto pode ser decomposto em componentes geométricos básicos chamados de *geons* (*geometric ions*). Os *geons* são em um objeto o que os fonemas são em uma palavra. Assim, no processo de casamento de padrões procura-se na memória pela representação do objeto mais similar que houver.

1.2.2 Teorias Dependentes do Ponto de Vista

As teorias dependentes do ponto de vista são baseadas em uma abordagem mais geral do que por partes. Assim, afirma-se que a imagem do objeto é armazenada na memória em diferentes ângulos e pontos de vista. Nesse tipo de reconhecimento, há uma grande utilização da memória, pois cada ponto de vista precisa ser armazenado. A precisão do reconhecimento depende de quanto o ponto de vista observado do objeto é familiar (LOGOTHETIS et al., 1994; BIEDERMAN; GERHARDSTEIN, 1995).

1.2.3 Teorias de Múltiplas Concepções

Essas teorias misturam as duas abordagens anteriores (Seção 1.2.1 e 1.2.2), onde as teorias invariantes ao ponto de vista são responsáveis pela categorização do objeto e as teorias dependentes do ponto de vista pela discriminação do objeto dentro de sua classe, utilizando as definições apresentadas pela Seção 1.1 pode-se dizer que a primeira teoria é responsável pelo reconhecimento e a segunda pela identificação de um dado objeto (TARR; BÜLTHOFF, 1995).

1.3 Reconhecimento Automático de Objetos

Com o avanço da tecnologia, computadores e câmeras digitais estão cada vez mais acessíveis, com poder computacional e qualidade de imagem cada vez melhores. Por esse motivo, a automatização de aplicações que empregam o reconhecimento de ob-

jetos tem se tornado popular. O principal exemplo é identificação e reconhecimento facial (VIOLA; JONES, 2001; XIAO et al., 2004; ZHANG et al., 2007a). É comum que aplicações de visão computacional e sensoriamento remoto utilizem a classificação pixel a pixel (FORSYTH; PONCE, 2011; GONZALEZ; WOODS, 2007). Porém, em alguns casos essa abordagem não apresenta um bom desempenho, como por exemplo no reconhecimento de objetos. Um objeto, de formato bem definido e baixa deformação, pode aparecer na imagem em diversas poses e normalmente apresenta variação intraclasse. Assim, não é possível afirmar que um determinado pixel pertence a um objeto somente pelos valores obtidos no espectro visível. O que define a aparência ou o formato de um objeto é um conjunto de pixels que organizados de uma maneira específica formam feições ou características. Portanto, um pixel isolado não traz muita informação se ele pertence ou não à um objeto. Entretanto, quando se analisa um conjunto de pixels, onde estes fazem vizinhança entre si, é possível extrair características que permitem reconhecer, ou não, um objeto, ou parte deste. A arquitetura atualmente mais utilizada para realizar o reconhecimento automático de objetos é a combinação descritor-classificador (FORSYTH; PONCE, 2011; SZELISKI, 2010; SHAPIRO; STOCKMAN, 2001).

Descritores são métodos que combinam determinados atributos de imagens para descrevê-las de uma forma específica em um determinado contexto. Para a aplicação de reconhecimento de objetos utiliza-se geralmente descritores locais, onde são responsáveis por descreverem objetos levando em conta a necessidade de possuírem invariância à mudança de iluminação e a ruídos. Descritores locais utilizam atributos locais para descrever uma região da imagem. Pode-se dar como exemplo de atributos: cor, gradiente, valor médio de nível de cinza etc. (ROTH; WINTE, 2008; FORSYTH; PONCE, 2011).

Classificadores são métodos utilizados para organizar dados em aglomerados ou classes, podendo-se utilizar ou não um conhecimento prévio. Existem dois tipos principais de classificação: supervisionada e não-supervisionada. Onde a primeira visa rotular dados a serem classificados através de um mecanismo que foi treinado com um conjunto de dados específicos e rotulados. Já no segundo tipo, deseja-se encontrar grupos ou aglomerados dentre um conjunto de dados a partir de características particulares (WEBB, 2002). Na maioria das aplicações de reconhecimento de objetos que adotam a abordagem descritor-classificador é utilizado classificação supervisionada em conjunto com descritores locais (FORSYTH; PONCE, 2011). A principal

explicação de se utilizar classificação supervisionada com maior frequência nesse tipo de aplicação é que já sabe o que se deseja encontrar, portanto, é necessário somente rotular os dados apresentados. Nesse tipo de classificação existem duas etapas: treinamento e classificação. Na fase de treinamento utiliza-se um conjunto de treinamento, o qual possui amostras positivas (imagens contendo o alvo desejado) e negativas (imagens que não contém o alvo em questão), para se treinar o método até atingir níveis toleráveis de acerto e falsos positivos na classificação. Já na fase de classificação a tarefa do classificador é dizer a qual classe (positiva ou negativa) aquela amostra apresentada pertence. O descritor local é utilizado em ambas etapas. O que os algoritmos de treinamento e classificação utilizam como atributo, não são os níveis brutos de cinza ou de cor de cada pixel e sim o descritor, ou descritores, extraídos de cada amostra.

Com pesquisas na literatura científica e mesmo fora da comunidade acadêmica, é possível encontrar facilmente abordagens que fazem uso da arquitetura descritor-classificador especificamente para reconhecimento automático de objetos. Dentre as mais utilizadas pode-se citar: HOG+SVM (DALAL; TRIGGS, 2005), Feições Haar-like em cascata (VIOLA; JONES, 2001) e LBP em cascata (MA et al., 2008). O que essas abordagens têm em comum é o fato dos descritores utilizarem textura e o formato dos objetos, e os trabalhos que as empregam relataram bom desempenho, robustez e até execução em tempo real (XIA et al., 2010; VIOLA; JONES, 2004; DALAL; TRIGGS, 2005; LIENHART et al., 2003). No Capítulo 2 serão apresentados maiores detalhes a respeito da arquitetura descritor-classificador e das abordagens citadas.

Em um paralelo com a seção anterior, a abordagem descritor-classificador se assemelha com as teorias invariantes ao ponto de vista. Apesar de as teorias neuropsicológicas se basearem em modelos 3D, a abordagem desse trabalho utiliza um modelo no espaço de atributos, no qual o classificador aprende a classificar a amostra em objeto e não-objeto. Durante processo de classificação, não é feito nenhum tipo de casamento visual entre a amostra apresentada ao classificador e o conjunto de treinamento no qual é composto por imagens de objetos em diferentes poses.

1.4 Imagens Orbitais

Sensoriamento remoto é uma área da ciência que visa obter informações sobre objetos e fenômenos sem que se entre diretamente em contato com os alvos. Assim, utiliza-se sensores a bordo satélites, aeronaves, balões, entre outros, para alcançar tal

objetivo (EHLERS et al., 2002). Sensores são dispositivos que realizam a conversão de energia proveniente de objetos em registros (imagens ou gráficos) permitindo assim, a associação da distribuição da radiância, emitância, ou retroespalhamento com as propriedades físicas, químicas, biológicas ou geométricas do alvo (NOVO, 2008).

Em relação a sua fonte de energia, os sensores podem ser rotulados como ativos ou passivos. Os sensores ativos se caracterizam por possuir sua própria radiação, como por exemplo radares e *lasers*. Já os sensores passivos captam a radiação refletida ou a emitida pelos objetos da superfície, ou seja, dependem de uma fonte de radiação externa. Um exemplo desse tipo de sensor, são os sensores ópticos, pois captam a radiação do Sol refletida ou emitida pela Terra. Os sensores podem ser classificados também como imageadores e não-imageadores. Um sensor imageador é aquele que possui a capacidade de produzir uma imagem bidimensional da radiância, emitância ou retroespalhamento do alvo. Outras características dos sensores podem ser agrupadas em três grupos: características geométricas, espectrais e radiométricas. Características geométricas definem a qualidade geométrica da imagem adquirida em termos de posição e forma dos objetos imageados em relação à sua posição e forma do terreno. A capacidade do sensor em diferenciar os níveis de energia captado é representado pelas características radiométricas. Características espectrais informam quais regiões do espectro eletromagnético o sensor opera e com que detalhe ele consegue mostrar as propriedades dos alvos captados (EHLERS et al., 2002; EHLERS, 2007; NOVO, 2008).

A resolução espacial do sensor informa qual é a menor feição passível de ser detectada, ou seja, é através dessa característica que é possível saber qual é o nível de detalhe encontrado em uma imagem (NOVO, 2008). A resolução espacial no solo pode ser dividido, de acordo com Ehlers et al. (2002), em: muito baixa (maior que 250 m); baixa (entre 250 e 50 m); média (entre 10 e 50 m); alta (entre 4 e 10 m); muito alta (entre 1 e 4 m); e ultra alta (menor que 1 m) (EHLERS, 2007). Há, porém, a expressão “sensores de alta resolução” que se refere a sensores que capturam imagens da superfície terrestre com resolução espacial menor que 5 metros (NOVO, 2008).

Neste trabalho são realizados estudos de caso com imagens de alta e média resolução espacial no espectro visível. As imagens de média resolução foram obtidas utilizando a banda pancromática (resolução espacial de 15 metros) do satélite Landsat 7 (USGS, 2014). As imagens desse satélite para a América do Sul podem ser obtidas gratuitamente em (INPE, 2013). Para esse tipo de imagem o objetivo é de reconhecer somente

pistas de pouso de asfalto ou concreto. Já as imagens de alta resolução foram obtidas através do aplicativo Google Maps (GOOGLE, 2013). Nesse tipo de imagem o objetivo é reconhecer campos de futebol e pistas de pouso de asfalto ou concreto. Maiores detalhes a respeito dos tipos de alvos escolhidos é apresentado na Seção 3.2.

1.5 Objetivo

Em diversas aplicações há a necessidade da utilização de reconhecimento automático de objetos presentes em imagens digitais, onde geralmente é necessário processar grandes quantidades de imagens de maneira rápida ou simplesmente automatizar a tarefa de um operador humano. Plataformas que utilizam sistemas autônomos ou híbridos são onde as abordagens que propõem o reconhecimento automático de objetos são normalmente empregadas. Aplicações possíveis para o reconhecimento automático de objetos em plataformas aéreas são: utilização em procedimentos de auto-localização (RODRIGUES et al., 2009), detecção de carros, pessoas ou outro tipo de objeto para um posterior rastreamento ou acompanhamento (BRECKON et al., 2009; GASZCZAKA et al., 2011), localização de alvos estratégicos etc.

Nesse estudo, aborda-se uma metodologia para o reconhecimento automático de objetos em imagens orbitais compostas por bandas do espectro visível. Tal metodologia pode ser utilizada para objetos, como carros, pistas de pouso, campos de futebol, hipódromos, entre outros, ou seja, objetos que possuam forma bem definida e baixa deformação. No entanto, existem diversos fatores que dificultam o processo de reconhecimento. Os fatores presentes na captura de imagens são, por exemplo, o uso de diferentes sensores, condições meteorológicas, resolução espacial em solo, entre outros. Já os fatores inerentes ao objeto são: variação de pose, variação intraclasse, transformações geométricas, entre outros. Deste modo, as abordagens utilizadas devem possuir invariância total ou parcial a esses fatores complicadores.

Algumas abordagens gerais, mas utilizadas popularmente para o reconhecimento de feições humanas, como faces e perfil humano, lidam bem com problemas relacionados aos fatores complicadores citados e normalmente possuem um baixo custo computacional (DALAL; TRIGGS, 2005; VIOLA; JONES, 2001; MA et al., 2008). Assim, a principal contribuição deste trabalho é o estudo da aplicabilidade no campo do sensoriamento remoto de abordagens de visão computacional empregadas na maioria das vezes em feições humanas. Esse estudo é feito levando em consideração o tipo de alvo, os tipos de sensores empregados e a variação de parâmetros das abordagens.

Os procedimentos são realizados de forma a permitir, também, a comparação entre as abordagens empregadas.

Esta dissertação de mestrado está organizada da seguinte maneira: este capítulo é destinado a introduzir alguns conceitos, como a definição de detecção, reconhecimento e identificação (Seção 1.1), o processo humano de reconhecimento visual de objetos (Seção 1.2), reconhecimento automático de objetos (Seção 1.3), imagens orbitais (Seção 1.4) e apresenta o objetivo deste trabalho (Seção 1.5); o Capítulo 2 apresenta a revisão bibliográfica (Seção 2.1), o modelo da arquitetura utilizada (Seção 2.2), descreve os métodos utilizados nesse trabalho (Seção 2.3) e o conceito de janela deslizante (Seção 2.4); o Capítulo 3 detalha a metodologia utilizada no treinamento e classificação (Seção 3.1), detalha a criação dos conjuntos de treinamento (Seção 3.2) e apresenta as ferramentas e bibliotecas utilizadas (Seção 3.3); no Capítulo 4 são apresentados os resultados dos estudos de caso; e a conclusão é apresentada no Capítulo 5.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Revisão Bibliográfica

Com pesquisas na literatura a respeito das abordagens mais populares na área de reconhecimento de objetos em imagens digitais, chegou-se entre outras abordagens a HOG+SVM (DALAL; TRIGGS, 2005), feições Haar-*like* em cascata (VIOLA; JONES, 2001) e LBP em cascata (MA et al., 2008). A escolha dessas três abordagens em detrimento das demais se deve ao fato de que os trabalhos científicos que as empregam normalmente relatam bom desempenho, robustez e, em certos casos, execução em tempo real (XIA et al., 2010; VIOLA; JONES, 2004; DALAL; TRIGGS, 2005; LIENHART et al., 2003). Outro fato é que fora da comunidade acadêmica o uso das três abordagens é grande (OPENCV, 2014). Isso pode ser constatado com o número de tutoriais e discussões em fóruns que podem ser encontrados na internet. Como será apresentado a seguir, a grande maioria dos trabalhos fazem uso das abordagens para o reconhecimento de feições humanas apesar delas serem de uso geral. O que todos esses trabalhos têm em comum é o reconhecimento de alvos bem definidos e com baixa deformação.

A combinação de HOG e SVM foi primeiramente empregada por Dalal e Triggs (2005), utilizada para o reconhecimento do perfil humano. Utilizado HOG+SVM pode se dar como exemplo, Gritti et al. (2008) que fazem um estudo a respeito do reconhecimento facial; Zhang et al. (2007b) propõem modificações para melhorar o desempenho para uso geral; Oreifej et al. (2010) realizam o reconhecimento do perfil humano em imagens aéreas de baixa qualidade; e Felzenszwalb et al. (2010) propõem um sistema que faz o reconhecimento de objetos através da detecção de partes deste. Vale ressaltar que todos os trabalhos utilizaram o SVM (BOSER et al., 1992) somente no modo linear.

Em relação a Haar-*like* em cascata, um dos primeiros trabalhos a utilizar essa técnica foi o trabalho Viola e Jones (2001), onde era proposto o reconhecimento facial. Subsequentemente em Viola e Jones (2004), mostrou-se a robustez dessa abordagem para objetos em geral. Em sua maioria os trabalhos que empregam Haar-*like* em cascata são para aplicações que envolvem o reconhecimento de faces em imagens, como pode-se ver nos trabalhos a seguir. Em Xiao et al. (2004) é proposto um sistema de três estágios onde deseja-se aumentar a robustez e o desempenho na aplicação de reconhecimento facial. A fim de reduzir o número de falsos positivos, Wilson e

Fernandez (2006) faz um levantamento estatístico de onde certas feições mais aparecem antes de classificar a imagem. Em Paliy (2008), propõe-se um sistema com duas fases para o reconhecimento de faces, onde a primeira faz o emprego do Haar-like em cascata. No intuito de melhorar o desempenho do reconhecimento facial, Jiang et al. (2010) propõem algumas modificações nas feições utilizadas. Existem alguns artigos que trabalham com outros tipos de alvos, como em Chen et al. (2007) onde é proposto um sistema de dois estágios para o reconhecimento de gestos manuais, onde Haar-like em cascata é empregada no primeiro estágio. Utilizando VANTs, pode-se citar: Breckon et al. (2009) que fazem o reconhecimento de veículos utilizando imagens do espectro visível e Gaszczaka et al. (2011) que fazem o reconhecimento de veículos e pessoas utilizando imagens do espectro visível e termal.

Uma outra técnica amplamente utilizada, o LBP com classificadores em cascata, foi empregado primeiramente no reconhecimento de objetos em Ahonen et al. (2004), onde foi proposto uma abordagem para o reconhecimento de faces. A maioria dos trabalhos que empregam o LBP em cascata, como Ma et al. (2008), Gritti et al. (2008) e Wang et al. (2011), realizam o reconhecimento de faces. Também propondo o reconhecimento de face, Zhang et al. (2007a) utilizam o LBP na abordagem multi-bloco. Em Fang e Wang (2008) estende-se a ideia do trabalho anterior e implementa-se o reconhecimento de gênero para faces. Existem também trabalhos que fazem o reconhecimento de pedestres, como Xia et al. (2010) que utilizam imagem no espectro visível e termal, Ma et al. (2011) que além do reconhecimento faz o rastreamento, e Corvee e Bremond (2011) que além de detectar pedestres, fazem o reconhecimento de cabeça e face. Há trabalhos que visam realizar identificação biométrica, como por exemplo em Wang et al. (2006) que propõem a identificação da palma da mão. Fazendo um teste mais amplo, o trabalho Trefny e Matas (2010) faz pequenas modificações no operador LBP e o testa em um conjunto de dados de face, carros e de reconhecimento de gênero.

Um fato interessante é que não foram encontrados trabalhos que façam a comparação entre as três abordagens para fins gerais ou específicos. Exceto pelos trabalhos que utilizam imagens de VANTs (GASZCZAKA et al., 2011; BRECKON et al., 2009), não foi encontrado nenhum outro trabalho que faça o uso de imagens aéreas ou orbitais. Alguns dos trabalhos apresentados fazem certas modificações nas abordagens originais. Na maioria das vezes, essas modificações se traduzem em aumento do desempenho, porém esse fato geralmente ocorre somente pra um determinado tipo de

problema, alvo ou situação (JIANG et al., 2010; TREFNÝ; MATAS, 2010; ZHANG et al., 2007b).

2.2 Arquitetura Descritor-Classificador

A arquitetura descritor-classificador é um conceito abstrato amplamente utilizado na literatura (FORSYTH; PONCE, 2011; SZELISKI, 2010; SHAPIRO; STOCKMAN, 2001). Pelo fato de ser um conceito abstrato, pode ser construída com diferentes descritores e algoritmos de classificação. Os métodos descritores geralmente são compostos por somente um estágio de operação, enquanto os métodos de classificação são compostos por um ou dois estágios. Na arquitetura que se utiliza neste trabalho somente classificadores supervisionados são empregados. Esse tipo de classificador possui dois estágios de operação: treinamento e classificação.

Na fase de treinamento, Figura 2.1, apresenta-se dois conjuntos de treinamentos rotulados: positivo, amostras contendo o alvo desejado, e o negativo, onde as amostras não contém o alvo em questão. Antes de serem apresentadas ao algoritmo de treinamento, ou treinador, é necessário extrair os atributos de cada amostra. Essa etapa é importante pois extrai informações relevantes dos pixels da amostra. Nesse caso, quem é responsável por extrair e organizar os atributos de modo a ajudarem no treinamento e na classificação é o descritor local, apresentado na Seção 1.3. A função do treinador é encontrar um hiperplano, ou hiperplanos, que consiga separar de maneira satisfatória as duas classes, positiva e negativa, no espaço de atributos. Essa tarefa é computacionalmente custosa podendo levar de horas até dias a ser concluída (WEBB, 2002). Tal período de tempo ocorre devido a busca por um hiperplano, ou hiperplanos, que seja o mais generalista possível, ou seja, que consiga classificar de forma satisfatória amostras que não foram utilizadas no treinamento. Ao final desta fase, obtêm-se a melhor configuração para o classificador de acordo com os parâmetros, conjunto de treinamento e heurística utilizados.

Na fase de classificação, Figura 2.2, apresenta-se uma amostra, não rotulada, para análise. A amostra nada mais é que uma imagem ou um recorte. Em seguida, extrai-se o descritor e o apresenta para o classificador, que por sua vez, já está configurado com os parâmetros obtidos na fase de treinamento. O que o classificador retorna é se a amostra apresentada é positiva ou negativa, ou seja, se contém ou não o objeto que se deseja encontrar.

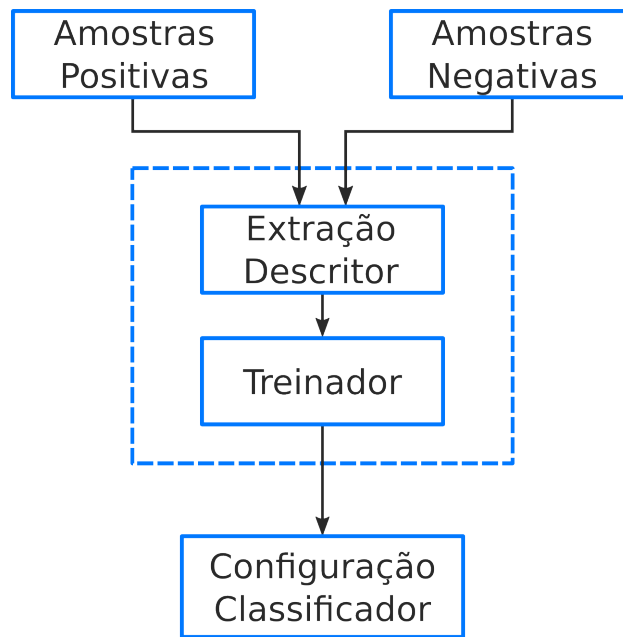


Figura 2.1 - Fase de treinamento.

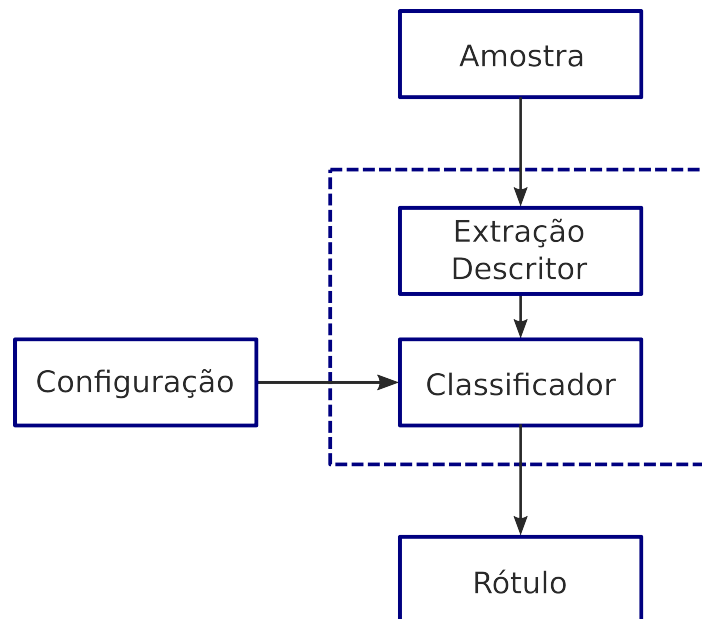


Figura 2.2 - Fase de classificação.

2.3 Métodos

Como mencionado na Seção 1.3, três das abordagens mais utilizadas nos últimos anos para o reconhecimento de objetos são: HOG+SVM (DALAL; TRIGGS, 2005), feições

Haar-*like* em cascata (VIOLA; JONES, 2001) e LBP em cascata (MA et al., 2008). O que as três abordagens possuem em comum é o fato de utilizarem, mantendo suas devidas particularidades, textura e formato dos objetos para realizarem o processo de reconhecimento.

Feições humanas e marcos em solo (exemplo, campo de futebol, hipódromo, entre outros) possuem semelhança no fato de serem objetos que possuem forma bem definida e baixa deformação. Olhando indivíduos de uma mesma classe de objetos, pode-se notar variações intraclasse. Tais variações podem ser pequenas alterações no formato, cor ou textura. As abordagens utilizadas neste trabalho possuem certa invariância a esse tipo de problema.

2.3.1 HOG+SVM

A associação de HOG com SVM, funciona da seguinte maneira: o primeiro é um descritor que utiliza histogramas de orientação dos vetores gradientes, provendo assim, informações sobre textura e bordas; o segundo é um classificador muito popular na literatura e utiliza os dados do primeiro método para executar o treinamento e classificação (DALAL; TRIGGS, 2005).

2.3.1.1 HOG

O Histograma de Gradientes Orientados (HOG, do inglês *Histogram of Oriented Gradients*) foi primeiramente descrito em (DALAL; TRIGGS, 2005) e é uma técnica que utiliza as orientações dos gradientes de uma imagem para obter seus descritores.

A ideia principal deste descritor é que a aparência e forma de objetos em uma imagem podem ser descritos através da distribuição dos gradientes de intensidade dos pixels ou pelas direções das bordas (GRITTI et al., 2008). O processo para gerar o descritor pode ser dividido em quatro etapas (Figura 2.3): cálculo do gradiente em cada pixel, agrupamento dos pixels em células, agrupamento das células em blocos e obtenção do descritor.

Na primeira etapa, calcula-se o gradiente de cada pixel da imagem em relação a sua vizinhança, como pode-se ver na Figura 2.3(b) e nos exemplos, Figura 2.4(b), Figura 2.5(b) e Figura 2.6(b). No método original usa-se uma máscara unidimensional de derivada discreta pontual tanto no eixo vertical como horizontal (Eq. 2.1) (DALAL; TRIGGS, 2005).

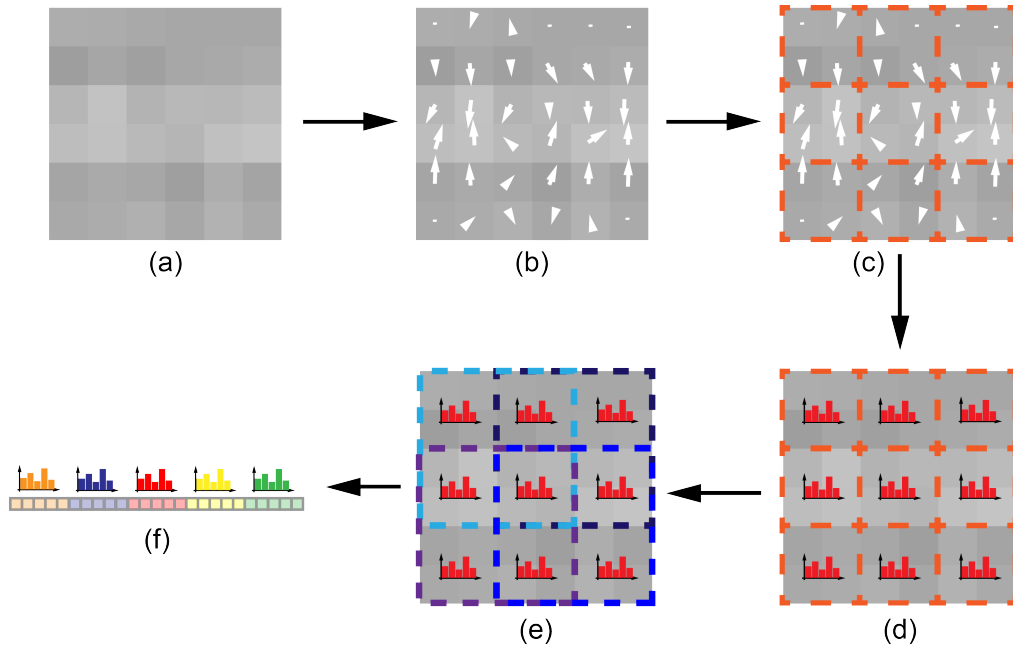


Figura 2.3 - Etapas da extração do descritor HOG: (a) imagem de entrada, (b) cálculo dos vetores gradientes, (c) agrupamento dos pixels em células, (d) cálculo dos histogramas das células, (e) agrupamento das células em blocos (sobrepostos) e (f) obtenção do descritor através da cópia dos histogramas de todos os blocos para um vetor.

$$[-1, 0, 1] \text{ e } [-1, 0, 1]^T \quad (2.1)$$

Outros tipos de filtros que também fazem o cálculo de gradiente já foram testados (GRITTI et al., 2008) nessa etapa mas não apresentaram um melhor custo-benefício.

O passo seguinte é responsável por agrupar os pixels de uma determinada região, criando-se o que se chama de célula, como pode-se ver na Figura 2.3(c) e nos exemplos, Figura 2.4(c), Figura 2.5(c) e Figura 2.6(c). Todas as células criadas na imagem possuem mesmo formato e tamanho. Cria-se um histograma com orientação do vetor gradiente dos pixels que compõe essa célula, Figura 2.3(d), onde são computados os valores de magnitude de acordo com o ângulo do vetor. O histograma possui uma quantidade finita de divisões. Por apresentar melhor desempenho (GRITTI et al., 2008), utiliza-se o histograma com nove divisões.

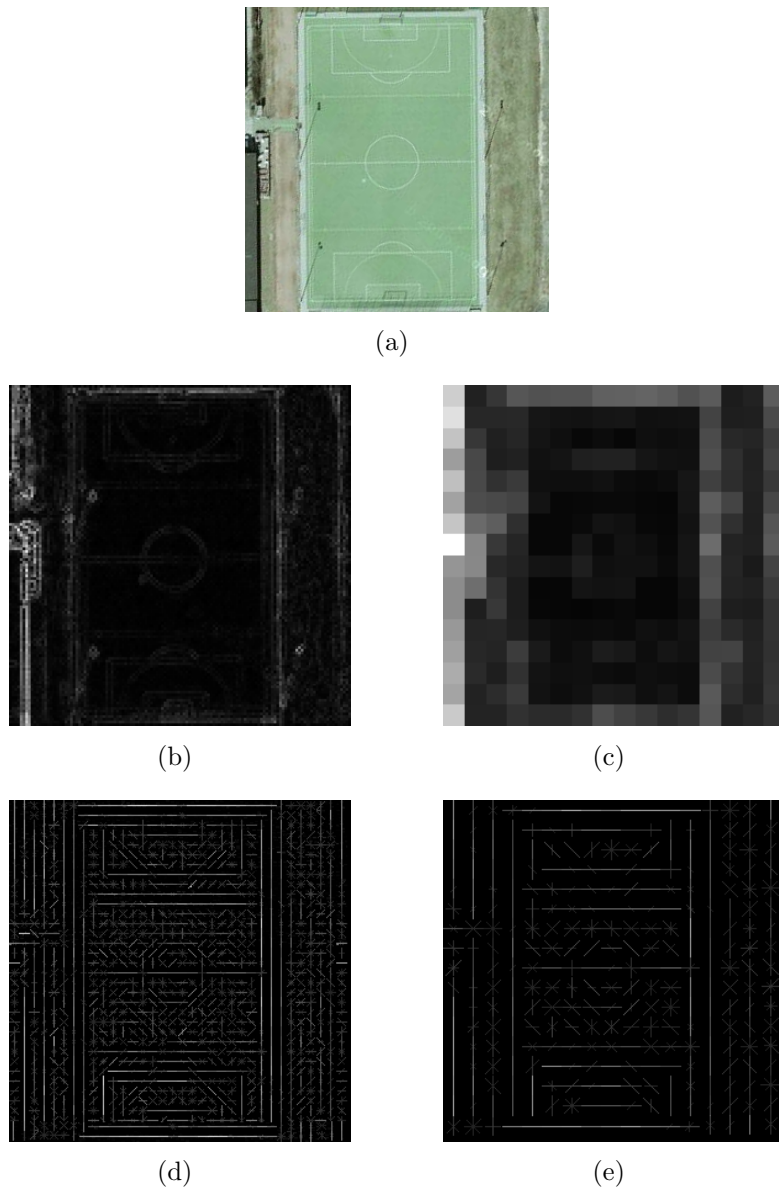


Figura 2.4 - Imagem orbital de alta resolução de um campo de futebol utilizada como entrada (a), magnitude do vetor gradiente dos pixels (b) e das células (c), e histogramas de orientação das células (d) e dos blocos (e).

Após a segunda etapa, os blocos são criados com o uso do agrupamento de células de uma certa região, como pode-se ver na Figura 2.3(e) e nos exemplos, Figura 2.4(e), Figura 2.5(e) e Figura 2.6(e). Assim como as células, os blocos também sempre possuem o mesmo formato e tamanho em toda a imagem por motivos de padronização. Existem áreas dos blocos em que há uma sobreposição proposital com o bloco vizinho, Figura 2.3(e), o que torna o método mais eficiente em relação a uma abordagem

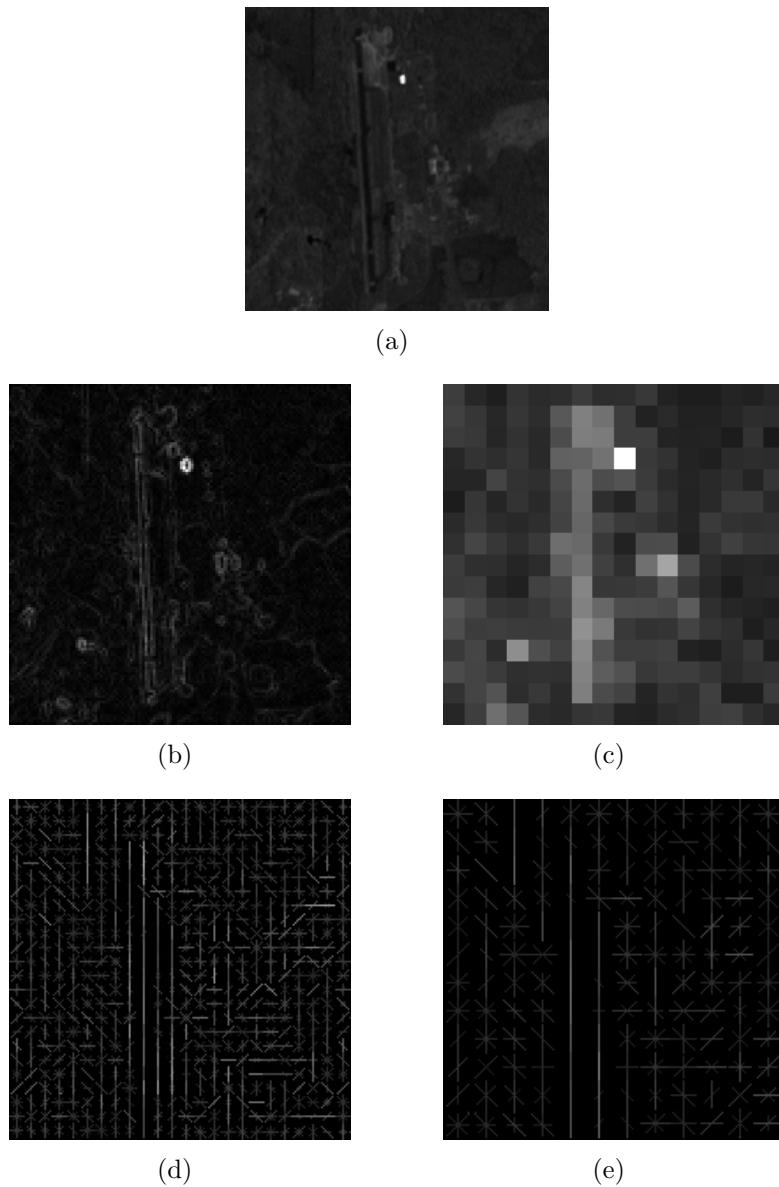


Figura 2.5 - Imagem orbital de média resolução de uma pista de pouso utilizada como entrada (a), magnitude do vetor gradiente dos pixels (b) e das células (c), e histogramas de orientação das células (d) e dos blocos (e).

sem essas sobreposições (DALAL; TRIGGS, 2005).

Na etapa final, cria-se o descritor, Figura 2.3(f), que é uma lista dos histogramas de todos os blocos.

Em qualquer tipo de imagem existe o problema de variações locais de iluminação, seja por causa de sombras ou fontes secundárias de iluminação. Esse tipo de problema

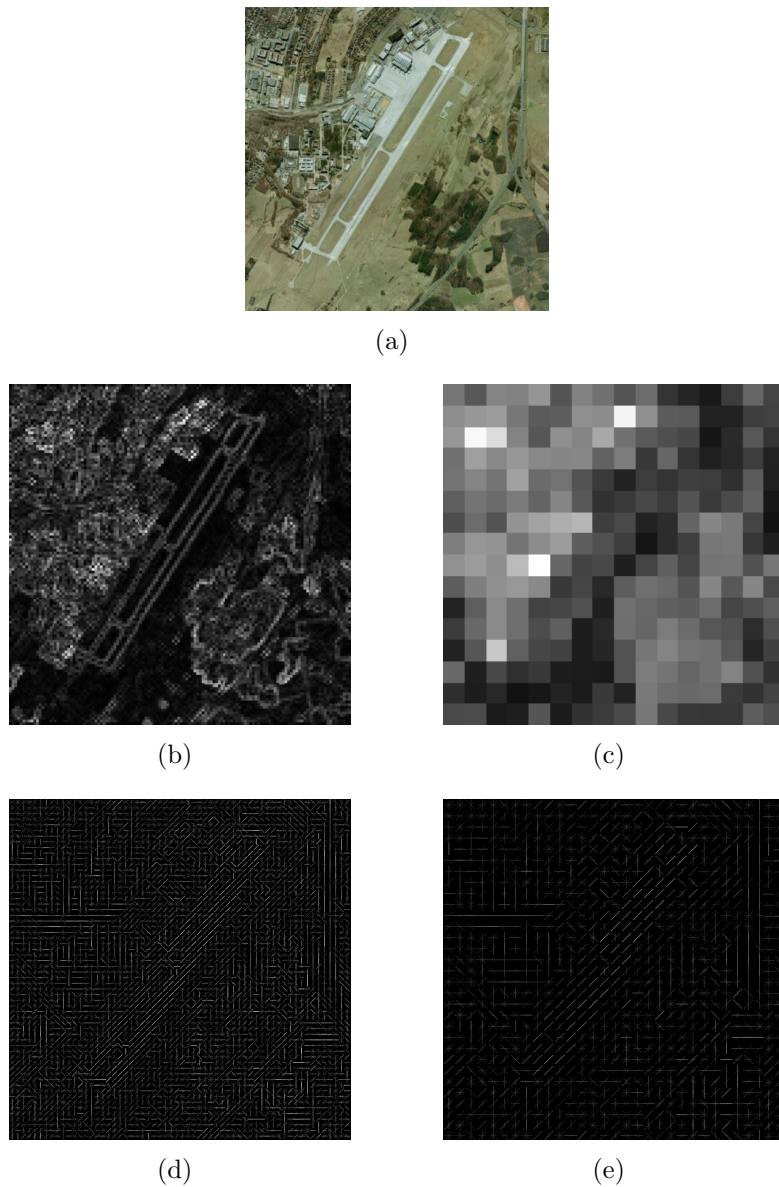


Figura 2.6 - Imagem orbital de alta resolução de uma pista de pouso utilizada como entrada (a), magnitude do vetor gradiente dos pixels (b) e das células (c), e histogramas de orientação das células (d) e dos blocos (e).

pode afetar o desempenho do classificador. A atenuação desse problema, assim como o problema de alto contraste entre o primeiro plano e o plano de fundo, acontece pela normalização dos histogramas dentro de um mesmo bloco (DALAL; TRIGGS, 2005). No método canônico, o método de normalização do vetor utilizado é o $L2-hys$. $L2-hys$ consiste em aplicar $L2-norm$ e limitar os resultados em um teto, em seguida calcula-se $L2-norm$ novamente. A $L2-norm$ é descrita pela Eq. 2.2.

$$L2-norm = \frac{v}{\sqrt{\|v\|_2^2 + e^2}}, \quad (2.2)$$

onde v é o vetor descritor, $\|v\|_k$ a sua k -norma para $k = 1, 2$ e e uma constante muito pequena. Outros métodos de normalização já foram testados em [Gritti et al. \(2008\)](#), mas *L2-hys* apresenta os melhores resultados.

Mudanças não-lineares de iluminação podem ocorrer devido à saturação causada pela câmera ou devido a mudanças de iluminação em superfícies tridimensionais vindo de diferentes ângulos e com diferentes intensidades ([LOWE, 2004](#)). Esses tipos de mudanças podem afetar diretamente na magnitude relativa de alguns gradientes. Assim, reduz-se a influência de grandes magnitudes de gradientes estipulando um valor máximo para a magnitude. O valor de teto de 0,2 foi encontrado em [Lowe \(2004\)](#) depois da execução de testes com imagens utilizando diferentes iluminações para os mesmos objetos tridimensionais.

2.3.1.2 SVM

Máquina de Vetores de Suporte (SVM, do inglês *Support Vector Machine*) foi primeiramente descrita em ([BOSER et al., 1992](#)) e é um método de aprendizado supervisionado, que analisa dados e reconhece padrões, usado para classificação e análise de regressão. O SVM é um classificador linear binário, mas existem abordagens que o tornam capaz de lidar com problemas não-linearmente separáveis ou mesmo com mais de duas classes. Por ser um classificador com poder de generalização e conseguir lidar com dados não-linearmente separáveis, esse método tem sido muito utilizado na literatura ([THEODORIDIS; KOUTROUMBAS, 2008](#)).

Um conjunto de treinamento pode ser descrito pela equação abaixo

$$C = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in R^d, y_i \in \{-1, 1\}\}_{i=1}^n, \quad (2.3)$$

onde \mathbf{x} são vetores contendo d atributos das n amostras e y são os respectivos rótulos. Assumindo que essas amostras pertencem somente a duas classes e que são

linearmente separáveis ou próximo disso, o objetivo é então encontrar um hiperplano que classifique as amostras de treinamento de forma satisfatória. Esse hiperplano é descrito através da equação abaixo

$$\mathbf{w}^T \mathbf{x} + b = 0, \quad (2.4)$$

onde \mathbf{w} é o vetor de pesos, b é o *bias*.

Como pode-se ver na Figura 2.7, tal hiperplano não é único. Neste caso, nota-se a importância do classificador possuir uma capacidade considerável de generalização, ou seja, conseguir operar satisfatoriamente com dados não presentes no conjunto de treinamento. Para alcançar tal capacidade, o SVM escolhe, durante o treinamento, o hiperplano que possuir a maior margem. Pode-se ter como exemplo a Figura 2.8, em que o h_2 será o hiperplano escolhido pelo SVM por possuir maior margem em relação a h_1 e h_3 .

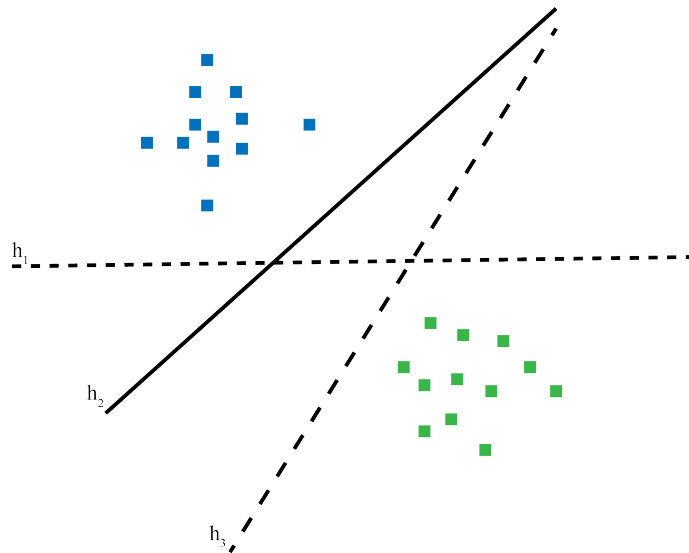


Figura 2.7 - Três classificadores possíveis para o problema de duas classes linearmente separáveis. Figura baseada em Theodoridis e Koutroumbas (2008).

Aplicações reais dificilmente possuem conjuntos de dados linearmente separáveis, devido a presença de ruído, amostras discrepantes ou à própria natureza do problema

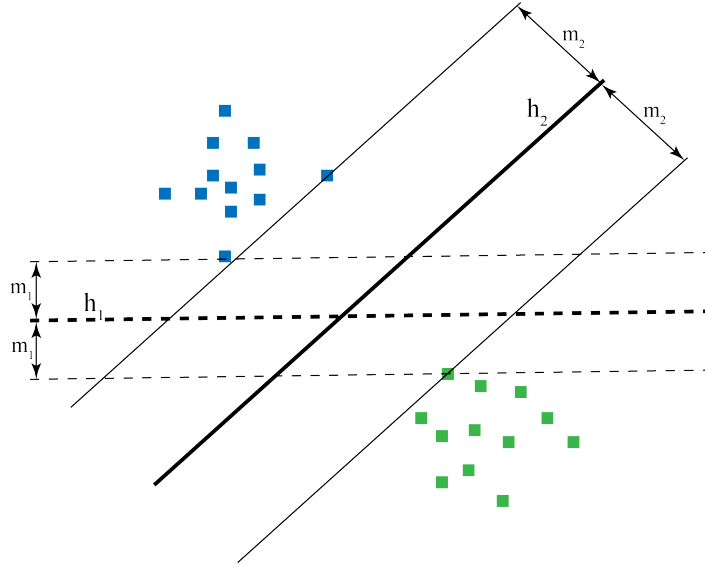


Figura 2.8 - Dois hiperplanos e suas margens em um problema de duas classes linearmente separáveis. Figura baseada em Theodoridis e Koutroumbas (2008).

abordado (LORENA; CARVALHO, 2003). Uma das abordagens que o SVM utiliza para lidar com classes que possuem fronteiras de separação próximas ao linear é a chamada Margem Suave (CORTES; VAPNIK, 1995), onde permite-se a presença de algumas amostras entre as margens e a ocorrência de alguns erros de classificação, mas é atribuída uma certa penalidade a cada amostra que se encaixa em um desses dois casos. Assim, o algoritmo de treinamento escolhe o hiperplano que apresenta melhor harmonia de duas características: baixa penalização, ou seja, separação das classes de forma mais limpa possível, e alta generalização, ou seja, a maior distância entre as margens (HASTIE et al., 2001). O treinamento é descrito pela equação abaixo.

$$\min_{\mathbf{w}, \xi, b} \max_{\alpha, \beta} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i [y_i (K(\mathbf{w}, \mathbf{x}_i) - b) - 1 + \xi_i] - \sum_{i=1}^n \beta_i \xi_i \right\}, \quad (2.5)$$

com $\alpha_i, \beta_i \geq 0$ e onde \mathbf{w} é o vetor de pesos, C e ξ são parâmetros da Margem Suave, onde o primeiro é o coeficiente de penalidade e o segundo é o quanto uma determinada amostra violou as margens, α e β são multiplicadores de Lagrange, y_i é o rótulo associado a amostra \mathbf{x}_i , $K()$ é a função *kernel* e b é o *bias*.

Na fase de treinamento, Eq. 2.5, tem-se como objetivo encontrar um hiperplano que

melhor separe as classes de um determinado problema. Um hiperplano, Eq. 2.4, é definido pelos parâmetros \mathbf{w} e b . Portanto, estes parâmetros são imprescindíveis para o funcionamento do classificador, em que a equação que o representa está descrita abaixo

$$g(\mathbf{x}) = \text{sin}al(K(\mathbf{w}^T \mathbf{x}) + b), \quad (2.6)$$

onde $\text{sin}al()$ é a função que retorna se o valor é positivo ou negativo, $K()$ é a função *kernel*, \mathbf{w} é o vetor de pesos, \mathbf{x} a amostra a ser analisada e b o *bias*. O que o classificador faz é verificar a posição da amostra não rotulada em relação ao hiperplano, assim é possível rotulá-la como sendo da classe positiva ou negativa.

Para situações onde só a margem suave não consegue separar de forma satisfatória as classes, devido provavelmente à natureza do problema, utilizam-se as funções *kernel* não-lineares. O uso desse tipo de função é motivado pelo teorema de Cover (COVER, 1965), onde afirma-se que é possível tornar um problema de separação não-linear, Fig. 2.9(a), em linear através do aumento da dimensionalidade do espaço de atributos, Fig. 2.9(b). Assim, as funções *kernel* não-lineares são responsáveis por realizar o mapeamento do espaço de atributos atual para um espaço de atributos com mais dimensões. Portanto, além do linear,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j \quad (2.7)$$

pode-se encontrar na literatura *kernels* do tipo RBF (*Radial Basis Function*) gaussiano,

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)}, \gamma > 0 \quad (2.8)$$

sigmoidal,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i^T \mathbf{x}_j + r) \quad (2.9)$$

entre outros. Onde γ e r são parâmetros do *kernel* (THEODORIDIS; KOUTROUMBAS, 2008; SHAWE-TAYLOR; CRISTIANINI, 2004).

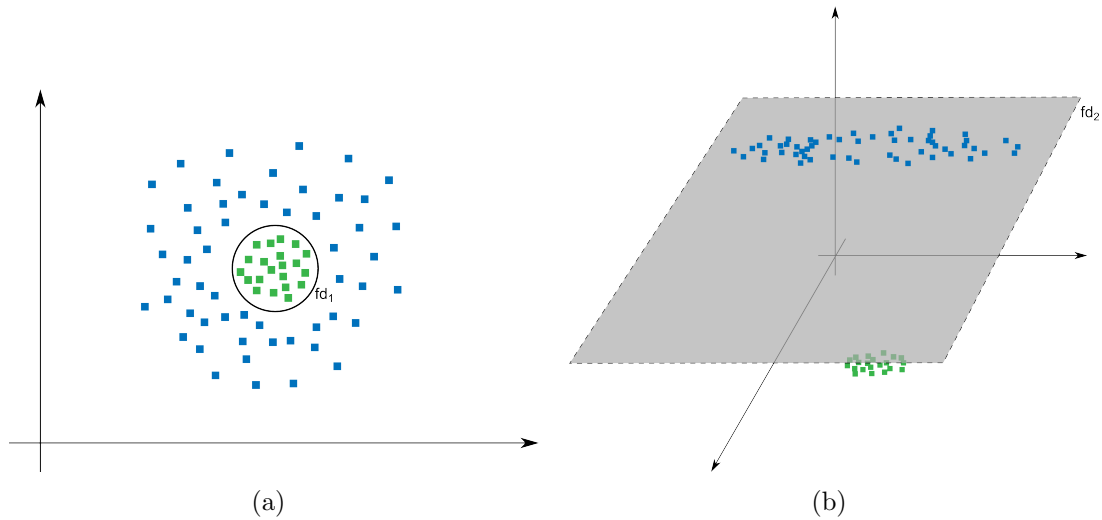


Figura 2.9 - (a) Problema não-linearmente separável com uma fronteira de decisão não-linear, fd_1 , possível, e (b) representação da adição de mais uma dimensão ao problema e a possibilidade de utilizar uma fronteira de decisão linear, fd_2 , para separar o problema. Figura baseada em Theodoridis e Koutroumbas (2008).

A precisão da classificação de um dado SVM é diretamente influenciado pelos parâmetros utilizados na fase de treinamento. Os parâmetros que devem ser ajustados são: parâmetros específicos do *kernel*, Equação 2.8 e Equação 2.9, e o coeficiente de penalização da Margem Suave. Por isso existem abordagens que visam encontrar, de forma aproximada, os melhores parâmetros para um determinado problema. Uma das abordagens mais utilizadas (CHANG; LIN, 2011) é a utilização de uma lista pré-estabelecida de parâmetros, método conhecido como *grid search*, junto com a validação cruzada (KOHAVI et al., 1995). Essa abordagem consiste em dividir o conjunto de treinamento em subconjuntos e treinar cada subconjunto com uma combinação diferente dos parâmetros da lista. Assim, utiliza-se para o treinamento do conjunto de treinamento total os parâmetros do conjunto que obtiver a melhor precisão na

classificação. Quando se trabalha com um conjunto de treinamento desbalanceado, é possível replicar amostras do conjunto que possui uma menor população para que os subconjuntos, ao se utilizar validação cruzada, possuam a mesma quantidade de indivíduos dos subconjuntos da outra classe.

2.3.2 Feições Haar-*like* e LBP em cascata

A associação de Feições Haar-*like* e LBP com o conceito de Classificadores em Cascata funciona da seguinte maneira: os dois primeiros são descritores de textura e forma, e o segundo é um conceito que dispõe classificadores de modo sequencial a fim de obter um classificador com baixa taxa de erro e com baixo custo computacional.

2.3.2.1 Feições Haar-*like*

Feições Haar-*like* são atributos extraídos de imagens e possuem esse nome devido a similaridade com *wavelets* Haar (HAAR, 1910; VIOLA; JONES, 2004). Em (PAPAGEORGIOU et al., 1998) foi proposta a utilização desse tipo de feição para o reconhecimento de objetos em imagens digitais (VIOLA; JONES, 2001).

Cada feição, vista na Figura 2.10, é um filtro em que se deve subtrair as regiões positivas (brancas) das negativas (pretas) para se obter um valor que é posteriormente utilizado na categorização das sub-regiões de uma determinada imagem. Em uma imagem pode-se encontrar entre dezenas e centenas de milhares de feições Haar-*like*. Cada uma das feições encontradas pode ser considerada um classificador fraco, ou seja, onde a taxa de acerto de classificação é geralmente um pouco melhor do que uma escolha aleatória (FORSYTH; PONCE, 2011), pois sozinho não se espera que se consiga classificar uma imagem de forma satisfatória. O modelo matemático que rege esses classificadores pode ser visto na equação abaixo.

$$h_i(x) = \begin{cases} 1, & \text{se } p_i f_i(x) < p_i \theta_i \\ 0, & \text{caso contrário} \end{cases} \quad (2.10)$$

onde f é a feição (Figura 2.10), p a polaridade, que indica a direção da inequação, e θ o limiar de classificação (VIOLA; JONES, 2001).

Alguns exemplos da aplicação das feições Haar-*like* podem ser vistos na Figura 2.11.

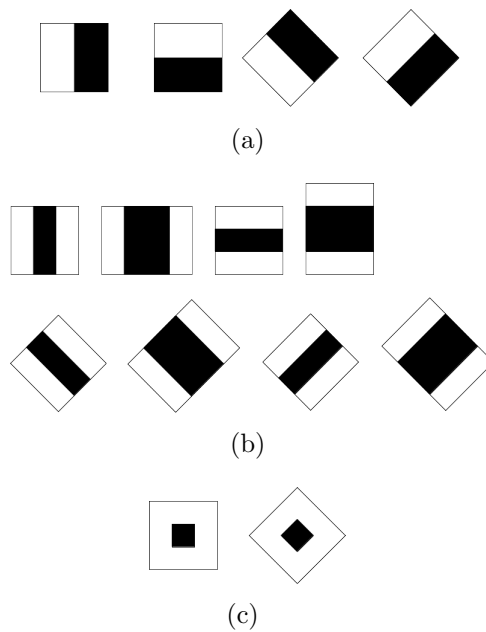


Figura 2.10 - Feições Haar-like, onde (a) são feições de borda, (b) feições de linha e (c) feições de ponto. Figura baseada em Lienhart et al. (2003)

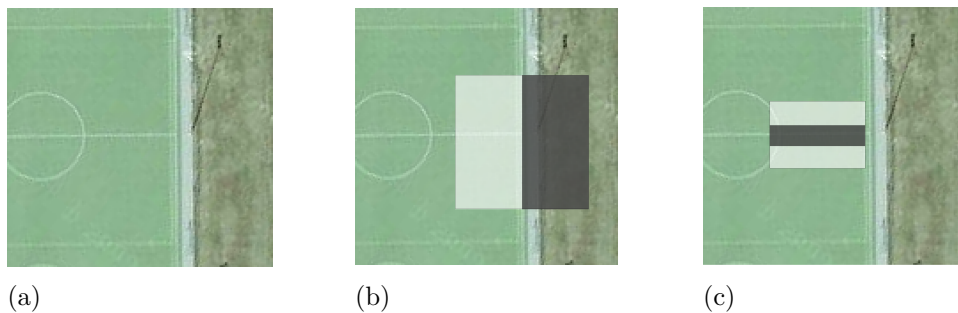


Figura 2.11 - Exemplo de aplicação de algumas feições Haar-like, onde (a) é um recorte de uma imagem contendo um campo de futebol, (b) aplicação de feição de borda e (c) aplicação de feição de linha.

2.3.2.2 LBP

Textura é definida como função de variação espacial na intensidade dos pixels de uma imagem, sendo utilizada em uma grande variedade de aplicações (CHEN, 2009). O operador Padrão Binário Local (LBP, do inglês, *Local Binary Pattern*), foi introduzido primeiramente em (OJALA et al., 1994) através da adaptação da proposta de (WANG; HE, 1990) e tem mostrado ser um poderoso descritor de texturas. A ideia por trás desse operador é que feições comuns, como bordas, linhas, ponto entre outros,

podem ser representadas através de um valor de uma determinada escala numérica. Portanto, utilizando um conjunto de valores extraídos *a priori* é possível fazer o reconhecimento de um determinado objeto em uma imagem.

Na Figura 2.12, ilustra-se o processo de extração do operador LBP.

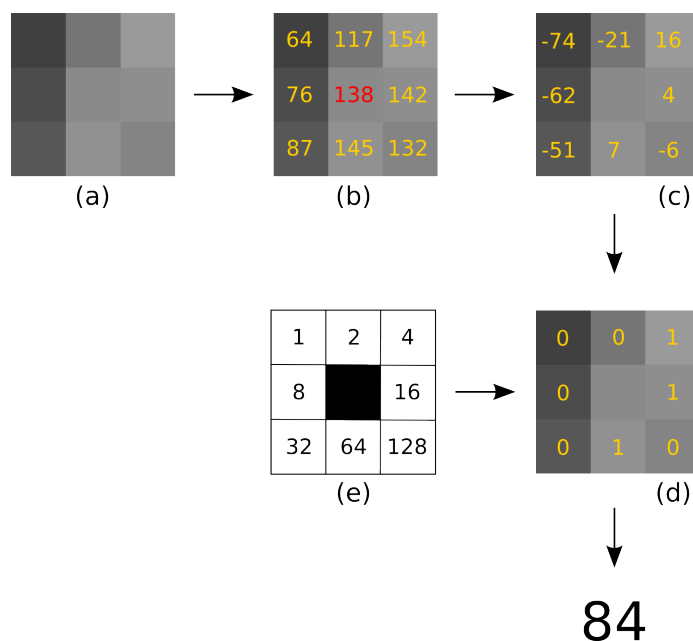


Figura 2.12 - Operador LBP: (a) é o fragmento da imagem a ser processado; (b) mostra o nível de cinza de cada pixel; (c) subtrai-se o valor do pixel central dos pixels da periferia; (d) atribui-se 0 para resultados abaixo de zero e 1 para resultados iguais ou maiores a zero; (e) mostra-se os valores que cada pixel representa; e na fase final soma-se os valores de (e) que valem 1 em (d).

O LBP original rotula os pixels com o uso da limiarização da vizinhança 3x3 em relação ao valor do pixel central, mas existem trabalhos mais recentes que testam outros tipos de vizinhanças (OJALA et al., 2002; TREFNÝ; MATAS, 2010). O “bit” obtido para cada pixel vizinho é utilizado através de uma ordem pré-definida para compor um valor final, que utilizando 8 pixels vizinhos tem seu valor entre 0 e 255, como é ilustrado abaixo:

$$LBP(x_p, y_p) = \sum_{n=0}^7 s(i_n - i_p) 2^n, \quad (2.11)$$

onde (x_p, y_p) é um pixel da imagem que se deseja avaliar, n representa os pixels

vizinhos, i_n e i_p os níveis de cinza do pixel vizinho e do pixel central respectivamente, e $s(x)$ pode ser descrito pela equação:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (2.12)$$

A Figura 2.13, Figura 2.14 e Figura 2.15 ilustram a aplicação do LBP.

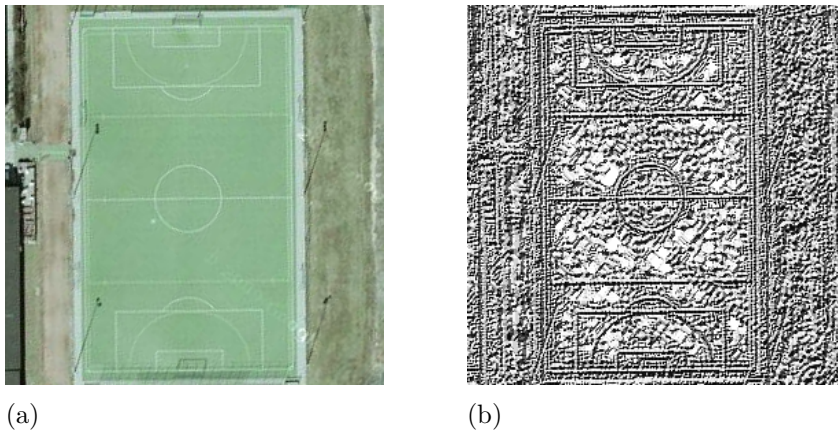


Figura 2.13 - (a) Imagem de alta resolução contendo um campo de futebol utilizada como entrada e (b) imagem resultante após a aplicação do operador LBP.

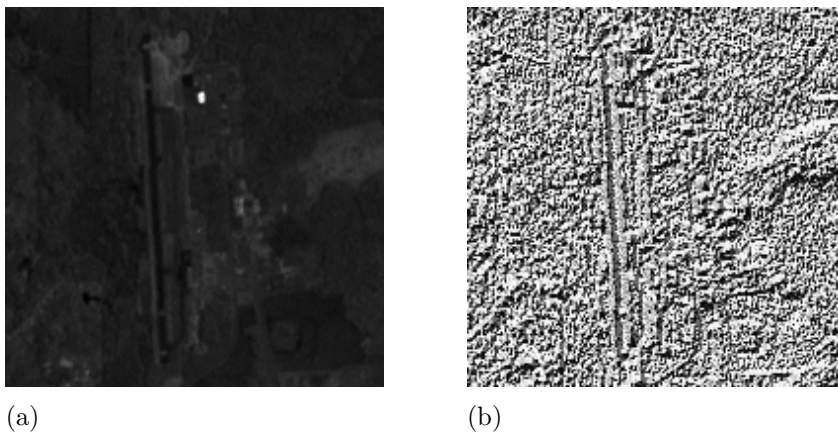


Figura 2.14 - (a) Imagem de média resolução contendo uma pista de pouso utilizada como entrada e (b) imagem resultante após a aplicação do operador LBP.

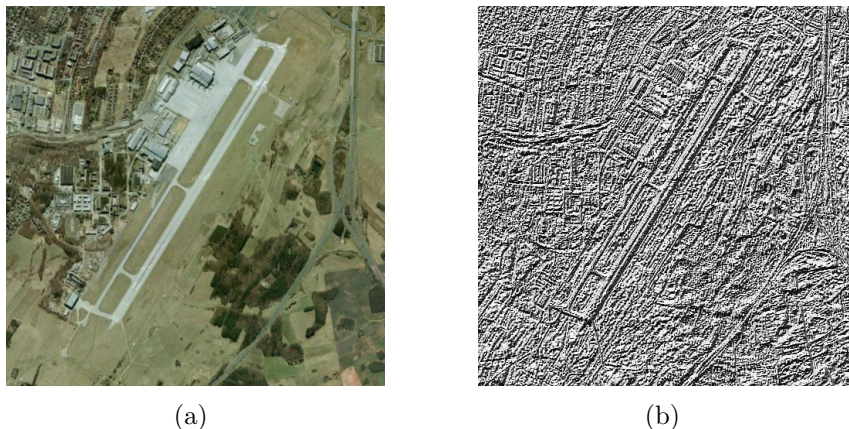


Figura 2.15 - (a) Imagem de alta resolução contendo uma pista de pouso utilizada como entrada e (b) imagem resultante após a aplicação do operador LBP.

O descritor LBP é portanto, um histograma com valores obtidos pelo operador LBP após os pixels de uma determinada região serem processados (GRITTI et al., 2008). Devido a arquitetura interna, esse método possui invariância a mudança local e global da iluminação na imagem.

Uma extensão do LBP é o que se chama de LBP de múltiplos blocos (ZHANG et al., 2007a), ou simplesmente MB-LBP (do inglês, *Multi-Block Local Binary Pattern*). Ao invés de se utilizar pixels, o MB-LBP aplica o operador LBP à blocos de pixels. Todos os blocos devem ter o mesmo tamanho e devem respeitar a formação 3x3, como mostrado na Fig. 2.12. O que o operador LBP utiliza para seu cálculo é o valor médio de cinza do bloco. Assim como a versão original do LBP, os padrões binários obtidos pelo MB-LBP permitem detectar diversas estruturas em imagens em diferentes escalas e localização, como: bordas, linhas, pontos, cantos e superfícies homogêneas. Porém a vantagem do MB-LBP é permitir que o classificador seja mais rápido e preciso (ZHANG et al., 2007a).

2.3.2.3 Classificadores em Cascata

Classificadores em Cascata é uma abordagem que combina classificadores em sequência, onde cada classificador é um nó binário de decisão, como mostrado na Fig. 2.16 em que V é o caminho que uma determinada amostra percorre se ela passar pelo teste de um dos classificadores fracos e F é o caminho que ela percorre se for rejeitada. O emprego de classificadores mais complexos em tarefas de reconhecimento de

objetos, normalmente resulta em um processamento mais lento da imagem (ZHANG; MA, 2012). Assim, a estrutura dos Classificadores em Cascata foi projetada visando obter um classificador com alta velocidade de processamento e com baixas taxas de erro.

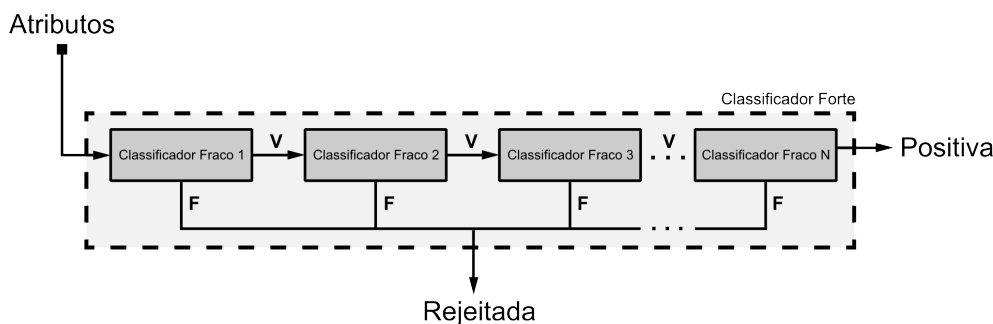


Figura 2.16 - Classificadores em cascata. Onde V representa que um conjunto de atributos passou pelo teste de um determinado nó classificador e F representa a rejeição de um conjunto de atributos.

Essa abordagem se aproveita do fato que em uma imagem real, não sintética, existem normalmente muito mais regiões negativas do que positivas, ou seja, o objeto só aparece em determinadas regiões da imagem. Esse classificador possui teoricamente uma alta velocidade de detecção, pois se for retirado aleatoriamente um recorte, com tamanho próximo ao do objeto, da imagem, a probabilidade que essa seção da imagem passe por todos os nós do classificador é baixa, onde provavelmente será rejeitada nos nós iniciais (ZHANG; MA, 2012). Essa velocidade de detecção se dá portanto, pela suposta raridade dos objetos alvos em uma imagem.

Em um classificador, quanto mais feições forem utilizadas, maiores são as possibilidades de se obter altas taxas de detecção e baixas taxas de falsos positivos, porém um número maior de feições implica em um maior tempo de processamento. Como o método de Classificadores em Cascata tem por objetivo ter um bom desempenho de classificação com um baixo custo computacional, deve-se então realizar uma otimização em que se leva em consideração o número de estágios da cascata, o número de feições e o limiar utilizados em cada estágio, visando minimizar o número total de feições utilizadas no classificador para um dado conjunto de treinamento. Encontrar a combinação ótima é uma tarefa árdua (VIOLA; JONES, 2004).

A fase de treinamento de uma Cascata se dá como mostrado na Fig. 2.17, onde

obtem-se o classificador H dispondo os classificadores h_1, \dots, h_N de forma sequencial. A cada iteração, o algoritmo cria um nó que atenda os objetivos de treinamento f_{meta} e v_{meta} , e que utilize o mínimo de feições possíveis. A construção de cada nó se dá normalmente com o uso de algoritmos de *boosting*, sendo AdaBoost um dos mais utilizados (VIOLA; JONES, 2004; MA et al., 2008; LIENHART et al., 2003).

Dado:

P conjunto de amostras positivas
N conjunto de amostras negativas
V conjunto de validação.

Meta de aprendizado:

f_{meta} é a taxa de falsos positivos
 v_{meta} é a taxa de verdadeiros positivos

Para $i = 1$ até N :

faça

Enquanto $f_i > f_{meta}$:

faça

(a) Treine nó h_i com **P** e **N** utilizando n feições

Enquanto $v_i < v_{meta}$:

faça

(b) Reduza os limiares utilizados classificador h_i .

(c) Classifique **V** com h_i para obter f_i e v_i .

fim

(d) Incremente o número de feições n

fim

(e) Classifique **N** com H e retire os exemplos corretamente classificados.

fim

Figura 2.17 - Algoritmo de treinamento de uma Cascata de Classificadores.

Boosting é uma abordagem geral que visa melhorar o desempenho de um dado classificador impreciso através da combinação de várias instâncias deste seguindo determinadas regras (FREUND et al., 1999). O conceito por trás desse método, remonta a questão apresentada por Valiant e Kearns (VALIANT, 1984; KEARNS; VALIANT, 1994) onde se questionava se um algoritmo de aprendizagem fraco, ou seja, onde a taxa de acerto de classificação é geralmente um pouco melhor do que uma escolha aleatória, poderia ser melhorado se empregado em grupo de modo a se tornar um algoritmo forte (THEODORIDIS; KOUTROUMBAS, 2008).

AdaBoost, abreviação em inglês para *Adaptive Boosting*, foi proposto em (FREUND; SCHAPIRE, 1996) e é um dos algoritmos de *boosting* mais conhecidos e estudados, onde tem aplicação geral em reconhecimento de padrões. Este método se baseia portanto, na utilização de classificadores fracos para sua construção que são, por sua vez, criados iterativamente utilizando uma ponderação diferente para cada amostra do conjunto de treinamento. Os pesos, que a cada iteração são atualizados, têm a função de enfatizar as amostras incorretamente classificadas, como visto na Figura 2.18. O classificador final obtido comporta-se como uma média ponderada dos classificadores fracos projetados. Utilizando-se um número suficiente de iterações, é possível chegar a uma baixa taxa de erro do classificador final medido sobre o conjunto de treinamento (SCHAPIRE et al., 1998; THEODORIDIS; KOUTROUMBAS, 2008).

O algoritmo, mostrado na Figura 2.19, tem como entrada um conjunto de treinamento $(x_1, y_1), \dots, (x_m, y_m)$ onde x_i corresponde a um ponto no espaço de atributos \mathbf{X} e y_i é o rótulo correspondente da amostra. O AdaBoost treina um tipo classificador fraco repetidamente para a série de ciclos $t = 1, \dots, T$, mas a configuração obtida para cada classificador tende a ser diferente. Uma das principais ideias desse algoritmo é possuir um peso para cada amostra. O peso do exemplo de treinamento i no ciclo t é simbolizado por $D_t(i)$. Inicialmente, todos os pesos são iguais, mas a cada ciclo, os pesos dos exemplos mal classificados são incrementados forçando o classificador fraco “focar” nesses exemplos do conjunto de treinamento.

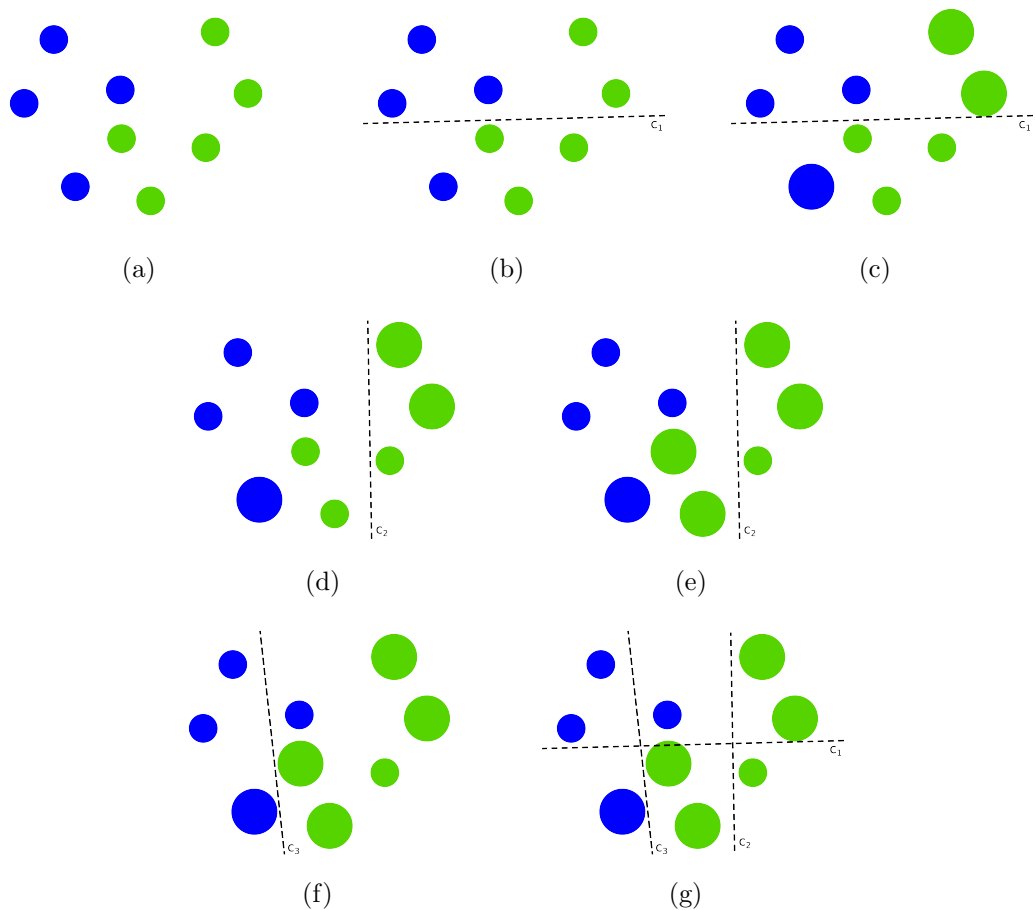


Figura 2.18 - Treinamento de *boosting* com três classificadores: (a) espaço de atributos com os pesos iniciais, onde o tamanho do círculo representa seu peso, (b) introdução do classificador fraco c_1 , (c) ajuste dos pesos para as amostras mal classificadas, (d) introdução do classificador fraco c_2 , (e) ajuste dos pesos para as amostras mal classificadas, (f) introdução do classificador fraco c_3 , (g) classificador final. Figura baseada em Szeliski (2010)

Dado: $(x_1, y_1), \dots, (x_m, y_m)$ onde $x_i \in \mathbf{X}, y_i \in \mathbf{Y} = \{-1, +1\}$

Inicialize $D_1(i) = \frac{1}{m}$

Para $t = 1, \dots, T$:

faça

(a) Treine o classificador fraco usando a distribuição D_t .

(b) Obtenha a hipótese $h_t : \mathbf{X} \rightarrow \{-1, +1\}$ com o erro $\epsilon_t = Pr_{i \sim D_t}[h_t(x_i) \neq y_i]$.

(c) Calcule: $\alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$.

(d) Atualize: $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{-\alpha_t}, & \text{se } h_t(x_i) = y_i \\ e^{\alpha_t}, & \text{se } h_t(x_i) \neq y_i \end{cases}$

onde Z_t é o fator de normalização

fim

Hipótese final: $H(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t h_t(x) \right)$

Figura 2.19 - Algoritmo de treinamento do AdaBoost (FREUND et al., 1999)

A tarefa do classificador fraco é encontrar a hipótese $h_t : \mathbf{X} \rightarrow \{-1, +1\}$ mais apropriada para o conjunto de pesos D_t . A precisão da hipótese do classificador fraco pode ser medida através de seu erro:

$$\epsilon_t = Pr_{i \sim D_t}[h_t(x_i) \neq y_i] = \sum_{i: h_t(x_i) \neq y_i} D_t(i), \quad (2.13)$$

que é medido de acordo com D_t com o qual o classificador fraco i foi treinado. Se o classificador fraco utilizado não permitir o uso de pesos D_t nas amostras de treinamento, utiliza-se uma segunda abordagem que consiste na amostragem, de acordo com D_t , de um subconjunto a partir do conjunto de treinamento e esses exemplos amostrados são usados para treinar o classificador fraco do próximo estágio.

Uma vez que a hipótese h_t é obtida, o AdaBoost calcula o parâmetro α_t que é o peso associado à cada classificador fraco h_t (FREUND et al., 1999). A etapa seguinte, como mostrada na Figura 2.19, é responsável por atualizar o conjunto de pesos D_{t+1} . O efeito dessa regra é aumentar o peso dos exemplos mal classificados por h_t e diminuir o peso dos exemplos corretamente classificados. Consequentemente, os pesos tendem a se concentrar nos exemplos que são normalmente classificados de forma incorreta. A hipótese final H é o voto ponderado da maioria das T hipóteses fracas onde α_t é

o peso associado a h_t . (FREUND et al., 1999)

Além da versão original do Adaboost, conhecido como *Discrete* AdaBoost, pois retorna um rótulo discreto binário (THEODORIDIS; KOUTROUMBAS, 2008), existem variantes que implementam algumas mudanças no algoritmo visando melhorias no desempenho final. Algumas das variantes mais conhecidas na literatura são o Real AdaBoost (SCHAPIRE; SINGER, 1999), onde o rótulo é um número real, e o Gentle AdaBoost (FRIEDMAN et al., 2000), onde o classificador final é normalmente mais estável em relação às outras variantes.

O Adaboost não foi projetado especificamente para atingir altas taxa de detecção com altas taxas de falsos positivos, ele simplesmente tenta minimizar os erros (VIOLA; JONES, 2004). Assim, para permitir que o AdaBoost se comporte dessa maneira e possa ser utilizado no treinamento de uma Cascata, modifica-se os limiares de seus classificadores fracos. Portanto, limiares altos tendem a produzir classificadores com baixa taxa de detecção e de falsos positivos, assim como limiares baixos tendem a produzir classificadores com alta taxa de detecção e de falsos positivos (VIOLA; JONES, 2004).

2.4 Janela Deslizante

Em qualquer abordagem de reconhecimento automático existe o problema em que o objeto pode aparecer em diversas regiões da imagem e com variações de escala. Janela deslizante é um procedimento utilizado para auxiliar abordagens de detecção, reconhecimento ou identificação de objetos com aparência bem definida e de baixa deformação visual, e tem o intuito de mitigar o problema de localização e escala de objetos alvos (FORSYTH; PONCE, 2011). O procedimento constitui na construção de um conjunto de dados com imagens rotuladas de tamanho fixo, $n \times m$, que são utilizadas no treinamento do classificador. As amostras ditas positivas, contêm instâncias de tamanho considerável do objeto a ser encontrado e amostras negativas há a ausência deste objeto. Na fase de classificação, varre-se a imagem com uma janela de tamanho $n \times m$ e espera-se que o classificador rotule corretamente, em positivas ou negativas, as janelas sucessivamente apresentadas a ele.

Existem dois fatores que se devem levar em consideração quando se utiliza janelas deslizantes. O primeiro é que nem todas as instâncias de um objeto terão o mesmo tamanho na imagem. Portanto, é necessário realizar um busca também em escala.

Um das abordagens normalmente empregadas para solucionar esse problema é obter uma Pirâmide Gaussiana (WITKIN, 1984) da imagem e realizar as buscas com uma janela de tamanho $n \times m$ em cada uma das camadas da pirâmide. Uma Pirâmide Gaussiana de Imagem é uma coleção de sucessivas representações, onde o nome vem da analogia visual, pois cada camada é obtida através da sub-amostragem da camada anterior após esta ter sido suavizada com um *kernel* gaussiano simétrico. A utilização de uma janela $n \times m$ em uma imagem onde seu tamanho foi redimensionado por um fator s é preferível do que realizar a busca na imagem original com uma janela $(sn) \times (sm)$. A diferença está na resolução; na facilidade do treinamento, pois as amostras de treinamento estão na mesma escala; no custo computacional, pois o número de atributos não aumenta no decorrer do processo de classificação; e no fato de sistemas de classificação geralmente serem configurados a receber uma janela de tamanho fixo para análise (FORSYTH; PONCE, 2011). O segundo fator, é que as janelas vizinhas, seja na mesma escala ou em outras, possuem uma certa sobreposição e cada uma pode conter se não o todo, uma fração substancial do objeto. Assim, há a possibilidade que cada janela seja rotulada como positiva pelo classificador, o que significa que um mesmo objeto será contado diversas vezes. Normalmente a estratégia utilizada para mitigar esse problema é a supressão não-máxima, onde janelas com um máximo local da resposta do classificador suprimem janelas positivas próximas.

A busca de objetos utilizando a abordagem de janela deslizante é um conceito genérico e se comporta muito bem na prática, mas a distância do objeto para a borda da imagem utilizada no conjunto de treinamento é uma variável que deve ser ajustada pois influencia o desempenho final do classificador (FORSYTH; PONCE, 2011).

3 METODOLOGIA

3.1 Abordagens de Treinamento e Classificação

As três abordagens empregadas neste trabalho: HOG+SVM (Seção 2.3.1), LBP em cascata (Seção 2.3.2) e Haar-like em cascata (Seção 2.3.2), utilizam a mesma arquitetura (Seção 2.2). Como apresentadas a seguir, as fases de treinamento e classificação são semelhantes.

No treinamento da abordagem HOG+SVM, Figura 3.1, extrai-se o descritor das amostras positivas e negativas do conjunto de treinamento e apresenta-se ao treinador SVM. O treinador então, encontra o hiperplano que melhor separa essas amostras no espaço de atributo. O melhor hiperplano é aquele que consegue melhor equilíbrio entre a maximização da margem e a minimização da penalidade total. A margem é a distância entre o hiperplano e as amostras localizadas nas fronteiras das classes e a penalização acontece quando uma amostra viola a margem do hiperplano. Tal penalidade é proporcional à distância da amostra em relação à margem. Ao fim do processo de treinamento obtém-se a configuração do melhor hiperplano encontrado.

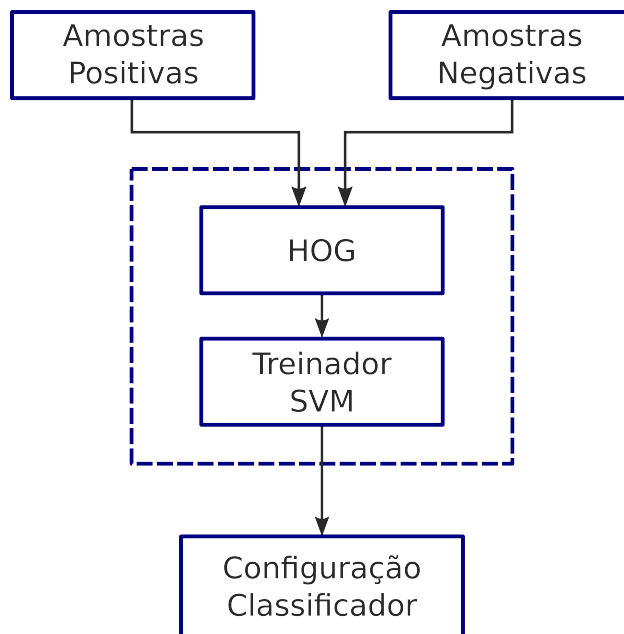


Figura 3.1 - Fase de treinamento da abordagem HOG+SVM.

Na fase de classificação, Figura 3.2, novamente extrai-se o descritor da amostra e

apresenta ao SVM. O SVM, já com a configuração obtida durante a fase de treinamento, calcula a distância entre a amostra e o hiperplano. Assim, as amostras que estão na face positiva do hiperplano são rotuladas como positivas e consequentemente, as amostras que estão na face negativa são rotuladas como negativas. Cada amostra é extraída da imagem utilizando o conceito de janela deslizante (Seção 2.4), onde uma janela de tamanho fixo varre a imagem. Devido busca também ser realizada em escala, há a necessidade se usar outro conceito, a supressão não-máxima (Seção 2.4), onde amostras com respostas positivas com valores mais altos suprimem as amostras ao redor da mesma ou de outras escalas.

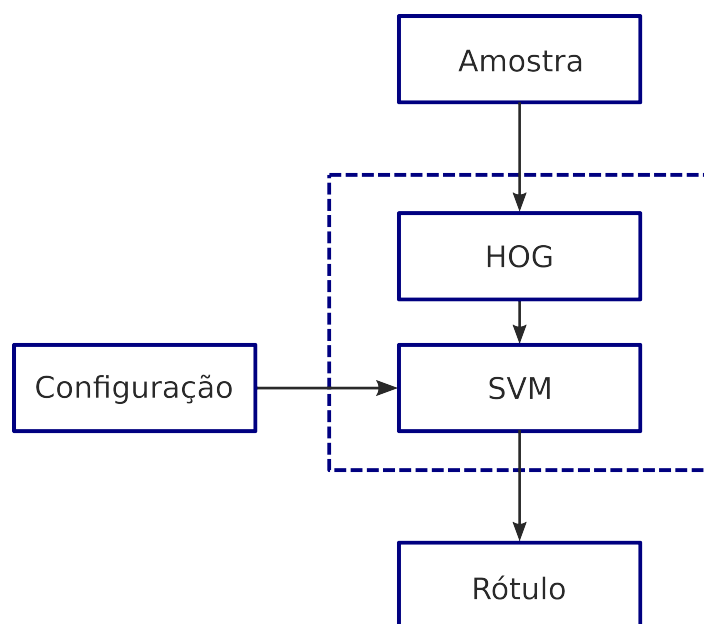


Figura 3.2 - Fase de classificação da abordagem HOG+SVM.

No treinamento da abordagem Haar-like e LBP em cascata, Figura 3.3, extrai-se os descritores das amostras positivas e negativas do conjunto de treinamento e as apresenta ao treinador. Cria-se então, nó a nó, a cascata de classificadores. O treinador deve respeitar a meta de falsos positivos e verdadeiros positivos, e utilizar o mínimo de feições possíveis em cada nó. Como resultado da fase de treinamento, obtém-se a configuração do classificador.

Na classificação, Figura 3.4, extrai-se os descritores das amostras e apresenta-se ao classificador, onde este foi previamente configurado com os parâmetros obtidos

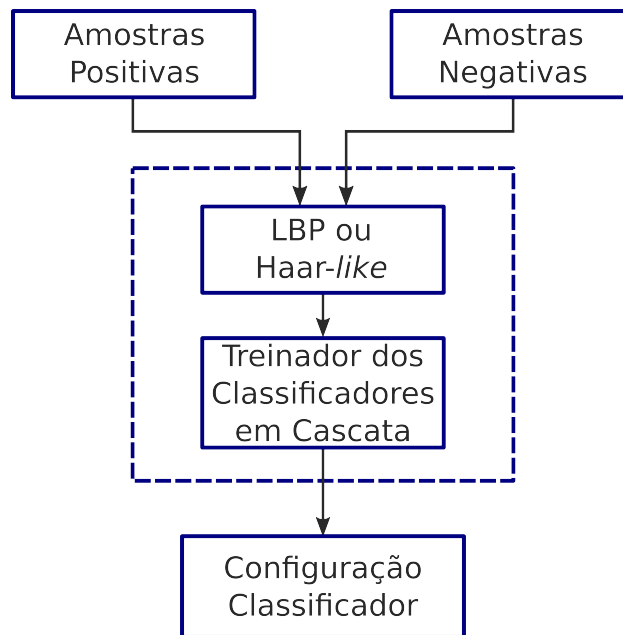


Figura 3.3 - Fase de treinamento das abordagens Haar-like e LBP em cascata.

no treinamento. A cada nível da cascata procura-se por determinadas feições na amostra. Se esse determinado conjunto de feições é encontrado, passa-se para o próximo nó, se não, descarta-se a possibilidade de ser uma amostra positiva e ela é como rotulada como negativa. Caso a amostra consiga passar por todos os nós, ela é rotulada como positiva. Assim como a abordagem HOG+SVM, os conceitos de janela deslizante e supressão não-máxima são empregados na classificação. Uma diferença dessa abordagem em relação à HOG+SVM é o fato de não se utilizar todos os atributos fornecidos pelo descritor e sim só os mais cruciais.

3.2 Conjunto de Treinamento

Um conjunto de treinamento é composto por amostras que descrevem no espaço de atributos as classes envolvidas no problema abordado. Sua função é apresentar ao classificador a aparência espacial da classe positiva e negativa, em um problema de separação de duas classes, ou da classe A, B, C etc., em um problema de separação de múltiplas classes. O classificador por sua vez tem o papel de encontrar, de modo satisfatório, uma forma de separar as determinadas classes representadas pelas amostras. O conjunto de treinamento é portanto, um ponto chave no desempenho final do classificador. Geralmente, quanto mais representativos são os dados presentes no processo de treinamento, melhor é o classificador obtido. No entanto,

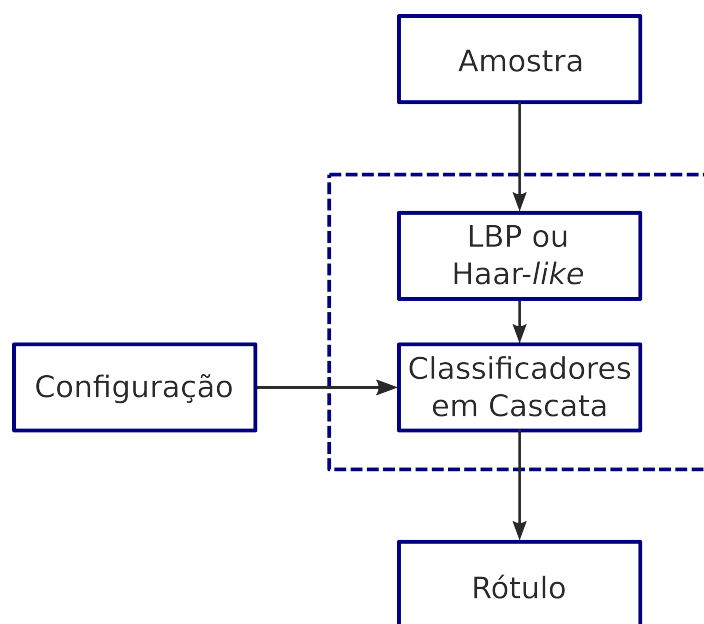


Figura 3.4 - Fase de classificação das abordagens *Haar-like* e LBP em cascata.

treinar classificadores com um grande conjunto de dados pode ser uma tarefa difícil, pois em certos casos a obtenção de dados suficientes pode ser algo muito trabalhoso. Normalmente, somente uma pequena quantidade de exemplos são realmente importantes para se determinar o comportamento de um classificador, isto porque esses casos afetam a posição da fronteira de decisão de forma considerável. Os exemplos realmente importantes tendem assim, a serem casos raros que são bem difíceis de discriminar (FORSYTH; PONCE, 2011). É necessário portanto, um conjunto de dados representativos para se garantir que estes casos estejam presentes.

Neste trabalho, deseja-se reconhecer objetos com formato bem definido e com pouca deformação em imagens orbitais e aéreas. Exemplo de tais objetos são: pessoas, carros, pistas de pouso, campos de futebol, hipódromos, entre outros. Montar um conjunto de treinamento de imagens por si só já é algo trabalhoso, assim muitos trabalhos utilizam conjuntos de treinamento já prontos, como SUN Database (MIT, 2013; XIAO et al., 2010; LIU et al., 2011), *datasets* do grupo de visão computacional da CALTECH (CALTECH, 2013; PERONA et al., 2004; FEL-FEI et al., 2006; GRIFFIN et al., 2007), conjuntos de treinamento do LEAR (LEAR, 2013; DALAL; TRIGGS, 2005; MARSZATEK; SCHMID, 2007; WEIJER et al., 2007), entre outros. Porém as bases de dados citadas, não possuem nenhum tipo de conjunto de dados de imagens aéreas ou orbitais. Elas possuem somente conjuntos de dados de imagens em solo, exemplo

desses objetos são: cadeiras, bicicletas, carros, motos, entre outros.

As abordagens apresentadas na Seção 2.3 requerem entre milhares e dezenas de milhares de amostras para o treinamento, onde o tipo de alvo é um dos fatores que define o número de amostras necessárias. Após pesquisas na literatura, não foi encontrado nenhum conjunto de dados de imagens aéreas ou orbitais já pronto contendo os tipos de objetos que se deseja encontrar. Assim, foi necessário criar os conjuntos de treinamento. As imagens utilizadas foram obtidas através do aplicativo Google Maps (GOOGLE, 2013) e do catálogo *online* da Divisão de Geração de Imagens do INPE (INPE, 2013).

Os objetos escolhidos para testar as abordagens listadas anteriormente foram: campos de futebol e pistas de pouso de asfalto ou concreto. Campo de futebol foi escolhido por ser um alvo que pode ser encontrado facilmente e em grande quantidade em imagens orbitais e aéreas de alta resolução espacial devido a popularidade do esporte pelo mundo. O segundo tipo de alvo, pistas de pouso de asfalto ou concreto, não são tão comuns quanto o primeiro, mas normalmente há no mínimo uma pista deste tipo em cidades médias e grandes ao redor do mundo. Pode-se também utilizar imagens de média resolução espacial, onde esse tipo de alvo é bem visível devido ao seu tamanho.

Os alvos escolhidos para serem utilizados neste trabalho podem aparecer em diversas orientações nas imagens, onde são sempre consideradas na visada nadir. As abordagens que foram utilizadas não apresentam invariância a grandes mudanças de rotação, por isso os alvos que compõem o conjunto de treinamento tiveram suas amostras rotacionadas previamente. Cada amostra inicial foi então rotacionada 40 vezes, onde em cada iteração rotacionava-se 9° . Tais valores foram encontrados de forma empírica. Assim, o conjunto de treinamento de campos de futebol em imagens de alta resolução, Figura 3.5, é composto por 3419 amostras positivas e 4000 amostras negativas. O conjunto de treinamento de pistas de pouso em imagens de média resolução, Figura 3.6, é composto por 1200 amostras positivas e 1500 amostras negativas. E o conjunto de treinamento de pistas de pouso em imagens de alta resolução, Figura 3.7, é composto por 2021 amostras positivas e 2500 amostras negativas. O número de amostras geralmente reflete a frequência de ocorrência de um determinado alvo em um conjunto de imagens, ou seja, quanto maior a ocorrência de um determinado objeto, maior pode ser seu conjunto de treinamento.

Durante o processo de treinamento, todas as abordagens redimensionam as amostras positivas para o tamanho do descritor, enquanto as amostras negativas são obtidas através da extração de sucessivas janelas de imagens contendo somente regiões negativas.

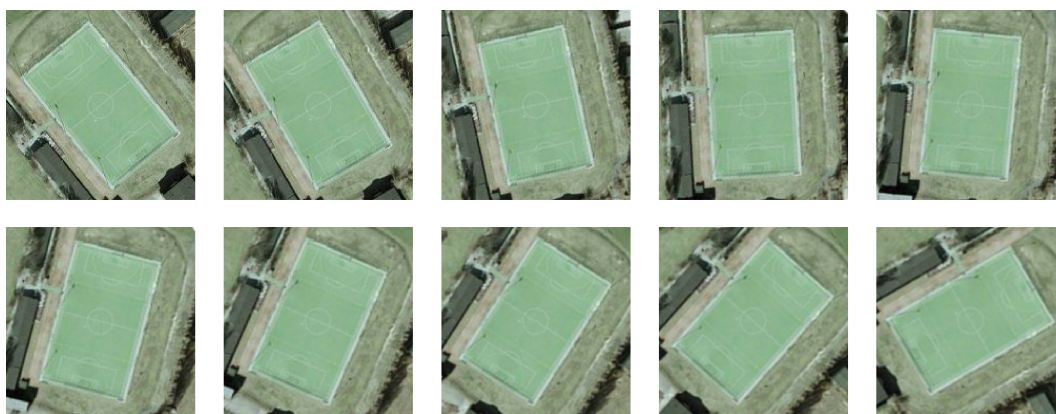


Figura 3.5 - Amostras do conjunto de treinamento positivo de alta resolução contendo campos de futebol (GOOGLE, 2013).

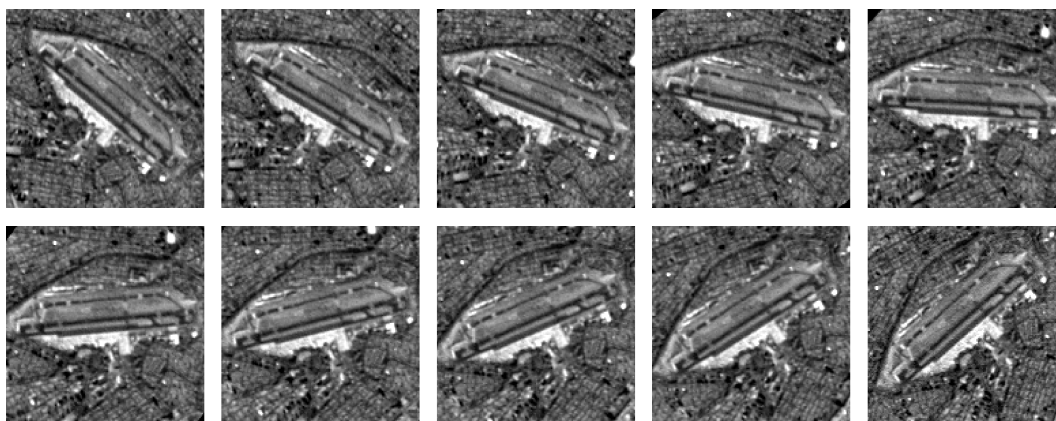


Figura 3.6 - Amostras do conjunto de treinamento positivo de média resolução contendo pistas de pouso de concreto e asfalto (INPE, 2013). As imagens foram tratadas a fim de permitir uma melhor visualização.

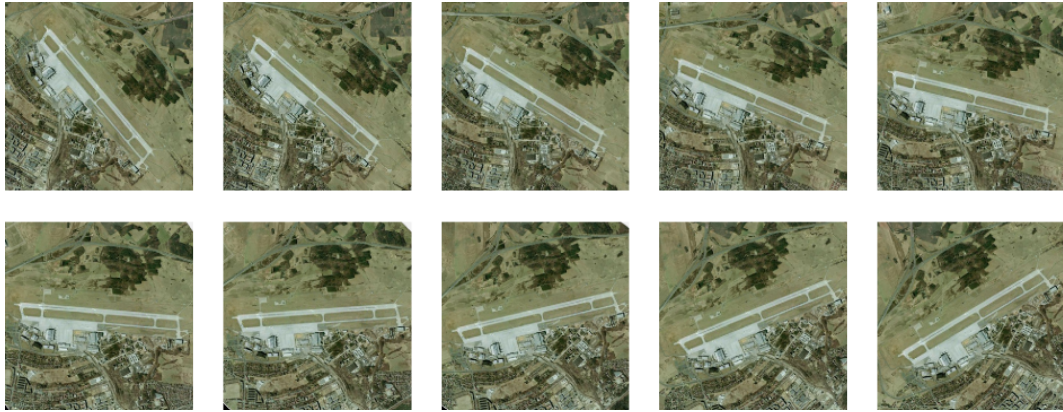


Figura 3.7 - Amostras do conjunto de treinamento positivo de alta resolução contendo pistas de pouso de concreto e asfalto (GOOGLE, 2013).

3.3 Bibliotecas e ferramentas

Para implementar os testes visando verificar a aplicabilidade dos métodos HOG+SVM, Haar-*like* em cascata e LBP em cascata na área de sensoriamento remoto, foi necessário utilizar bibliotecas para utilizar, criar e modificar ferramentas. As bibliotecas e ferramentas utilizadas serão descritas a seguir.

3.3.1 Bibliotecas

Foram utilizadas três bibliotecas para desenvolver esse trabalho de dissertação: OpenCV (OPENCV, 2014), OpenMP (OPENMP, 2014) e Qt (QT, 2014). A primeira teve papel essencial, enquanto as outras duas um papel complementar. Na Seção 3.3.2 é possível entender o porquê disso.

3.3.1.1 OpenCV

OpenCV (Visão Computacional de Código Aberto, do inglês *Open Source Computer Vision*) é uma biblioteca de visão computacional criada inicialmente em 2000 pela Intel (Intel Research) a fim de permitir avanços em aplicações que utilizavam CPU. Hoje em dia ela é mantida por uma fundação sem fins lucrativos e é uma das bibliotecas de visão computacional mais utilizadas no mundo, com uma comunidade com mais de 47 mil usuários e com mais de 7 milhões de pacotes de instalação baixados desde sua criação (dados obtidos em janeiro de 2014 (OPENCV, 2014)). Além de muito utilizada na pesquisa acadêmica, ela tem seu uso bem difundido na

indústria, em que os principais exemplos são: Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota, entre outros. Exemplos de seu uso são: criação de mosaico, detecção de intrusão em vídeos de vigilância, monitoramento de equipamento de mineração, navegação autônoma, auxílio visual para braços robóticos, arte visual interativa, monitoramento de entulho em estradas, inspeção de rótulos em produtos, reconhecimento facial, entre outros (OPENCV, 2014). O OpenCV possui interfaces para C/C++, Python e Java e pode ser utilizada em Windows, Linux, Mac OS, iOS e Android.

3.3.1.2 OpenMP

OpenMP (Multi-Processamento Aberto, do inglês *Open Multi-Processing*) é utilizada na programação multi-processo em memória compartilhada em C/C++ e Fortran e funciona na maioria das arquiteturas de processadores e sistemas operacionais. O que faz o OpenMP atraente é que ele utiliza um modelo escalável e portátil, permitindo assim, oferecer ao programador uma interface simples e flexível para o desenvolvimento de aplicações paralelas para plataformas que vão de computadores pessoais até super computadores (OPENMP, 2014).

3.3.1.3 Qt

Qt é um arcabouço para aplicações multiplataformas que permite a criação de interfaces gráficas. Para se utilizar o Qt a aplicação deve utilizar a linguagem C++ e utilizar plataformas como o Windows, Linux, OS X, Android, iOS, entre outros (QT, 2014).

3.3.2 Ferramentas

Ao longo do período em que essa dissertação foi realizada, ferramentas foram desenvolvidas e aperfeiçoadas e outras simplesmente utilizadas sem a necessidade de qualquer modificação. Todas as ferramentas utilizadas foram desenvolvidas em C/C++ e utilizadas em Linux.

3.3.2.1 Extrator de Amostras Semi-Automático

A construção de conjuntos de treinamento de maneira manual é algo extremamente trabalhoso quando se precisa do número de amostras na ordem de milhares a dezenas de milhares de unidades. No intuito de agilizar esse processo, foi desenvolvido uma

ferramenta utilizando OpenCV e Qt. Essa ferramenta consiste de um aplicativo com interface visual que permite acelerar o processo de obtenção de amostras de forma semi-automática, uma vez que é necessário que o usuário sinalize onde é o centro de rotação e os limites da amostra para que assim, o programa rotacione a imagem no grau estipulado e recorte a amostra. A sequência de rotacionar e recortar é realizada ininterruptamente de acordo com o número de vezes pedido pelo usuário. As amostras obtidas são parecidas às que foram mostradas nas Figuras 3.5, 3.6 e 3.7. O uso dessa ferramenta permite realizar o processo de construção de conjunto de treinamento de 40 a 60 vezes mais rápido em relação ao processo manual em um programa de manipulação de imagens.

3.3.2.2 Treinador e Classificador para Haar-*like* e LBP em Cascata

O treinador utilizado tanto para o Haar-*like* quanto para o LBP em cascata foi o treinador disponível na biblioteca OpenCV, não foi realizada nenhuma modificação. Vale ressaltar que o OpenCV utiliza o MB-LBP, uma extensão do LBP, apresentado na Seção 2.3.2.2.

Foi implementado um código que permite fazer a classificação em lote, ou seja, recebe de entrada um arquivo texto contendo as imagens a serem processadas e retorna as imagens classificadas e dados relevantes do processo de classificação. Porém o código destinado a classificadores em cascata contido no OpenCV não foi modificado.

3.3.2.3 Treinador e Classificador para HOG+SVM

Apesar dos trabalhos encontrados com HOG+SVM só utilizarem a versão linear do SVM, optou-se por também empregar *kernels* não-lineares no intuito de testar o desempenho do classificador com essa configuração para o problema de reconhecimento de objetos.

O treinador e classificador HOG+SVM que existe no OpenCV só realiza o treinamento e classificação com *kernel* linear. Portanto, para poder fazer o teste com *kernels* não-lineares foi necessário criar um novo treinador. No treinador, utilizou-se a abordagem de *grid search* para conseguir se chegar o mais perto possível de um conjunto ideal de parâmetros. Assim como na versão do LBP e Haar-*like* em cascata, esse classificador pode executar a classificação em lote e disponibiliza ao final estatística do processo. Para o classificador foi necessário modificar o código já existente a fim de permitir a utilização de *kernels* não-lineares. Devido a essa abordagem ter

um tempo de processamento um pouco elevado, foram paralelizados, com OpenMP, os laços de repetições onde as iterações não dependiam das iterações anteriores e podiam assim, ser quebradas em *threads*.

4 RESULTADOS

As abordagens LBP em cascata, Haar-*like* em cascata e HOG+SVM são de uso geral, apesar de terem seu uso mais difundido para problemas envolvendo feições humanas. A fim de testar a aplicabilidade dessas abordagens no campo do sensoriamento remoto, utilizou-se três estudos de caso: pista de pouso em imagens de média resolução, pista de pouso em imagens de alta resolução e campos de futebol em imagens de alta resolução. Os três alvos possuem forma bem definida e de baixa deformação. Tais objetos foram escolhidos devido a facilidade de serem encontrados em imagens de satélite em relação a outros tipos de alvos. Essa facilidade impacta diretamente no tamanho e na qualidade do conjunto de treinamento.

Para permitir a comparação entre as abordagens, seja ela no mesmo estudo de caso ou entre os estudos de casos, utilizou-se duas métricas para medir o desempenho da classificação e duas métricas para medir a velocidade de processamento. Há a possibilidade de comparar com outros trabalhos, desde que sejam utilizadas as mesmas métricas ou que estes forneçam dados suficientes para que as métricas sejam calculadas.

Para medir o desempenho da classificação das abordagens, utilizou-se as métricas: Taxa de Acerto (**TA**) e **Precisão** (FAWCETT, 2006).

$$\mathbf{TA} = \frac{\mathbf{VP}}{\mathbf{VP} + \mathbf{FN}} \cdot 100 \quad (4.1)$$

$$\mathbf{Precisão} = \frac{\mathbf{VP}}{\mathbf{VP} + \mathbf{FP}} \cdot 100 \quad (4.2)$$

onde Verdadeiros Positivos (**VP**) são objetos corretamente reconhecidos, Falso Positivos (**FP**) são regiões da imagem que foram classificadas erroneamente como sendo o objeto, e Falso Negativo (**FN**) que são objetos não detectados.

Para medir a velocidade no processamento do conjunto de imagens de avaliação, foram utilizadas duas métricas: Tempo Médio por Imagem (**TMI**) e Tempo Médio por Pixel (**TMpx**).

$$\mathbf{TMI} = \frac{\text{Tempo Total}}{\text{Número de Imagens}} \quad (4.3)$$

$$\mathbf{TM}_{\text{px}} = \frac{\text{Tempo Total}}{\text{Número Total de Pixels}} \quad (4.4)$$

4.1 Pista de Pouso em Imagens de Média Resolução

Para o teste de reconhecimento de pistas de pouso de concreto ou asfalto em imagens orbitais de média resolução espacial, foram utilizadas imagens da banda pancromática do LANDSAT-7 (INPE, 2013). Esse tipo de alvo foi utilizado como estudo de caso, pois são comuns em bases aéreas e em cidades médias e grandes. Devido ao seu tamanho, podem ser reconhecidas em imagens de média resolução.

No conjunto de treinamento foram utilizados 1200 amostras positivas e 1500 amostras negativas. Esses números foram alcançados, pois é necessário ter um conjunto com amostras representativas, porém havia a limitação da quantidade de imagens disponíveis.

No teste de classificação foram utilizadas 30 imagens, onde elas não possuíam nenhuma região dentro o conjunto de treinamento positivo ou negativo e possuíam um tamanho médio de 1497 x 1497 pixels. Cada imagem teste possuía um aeroporto.

Foram testadas diversas configurações das abordagens. Nas abordagens em que se utilizou o conceito de classificadores em cascata, variou-se, na fase de treinamento, os números de estágios da cascata mas manteve-se os parâmetros de taxa mínima de acerto (99,5%) e taxa máxima de falsos positivos (50%).

Os resultados obtidos com Haar-*like* em cascata podem ser vistos na Tabela 4.1 e a representação gráfica dos resultados na Figura 4.1. Pode-se notar que a taxa de acerto e precisão ficaram em torno de 27% para todos os classificadores. Ocorreu variação nos tempos de processamento, que em média era menor para os classificadores com mais estágios.

Tabela 4.1 - Reconhecimento de pista de pouso em imagens de média resolução utilizando Haar-like em cascata.

Tamanho	Níveis	VP	TA	FP	FN	Precisão	TMI	TMpx
32px	20	8	27%	22	22	27%	77597 ms	$346,28 \cdot 10^{-4}$ ms
	25	8	27%	22	22	27%	63374 ms	$282,81 \cdot 10^{-4}$ ms
	30	8	27%	22	22	27%	55329 ms	$246,91 \cdot 10^{-4}$ ms
	35	8	27%	22	22	27%	36761 ms	$164,05 \cdot 10^{-4}$ ms
	40	8	27%	22	22	27%	33639 ms	$150,11 \cdot 10^{-4}$ ms
50px	20	7	23%	23	23	23%	191578 ms	$854,92 \cdot 10^{-4}$ ms
	25	8	27%	22	22	27%	506858 ms	$2261,9 \cdot 10^{-4}$ ms
	30	8	27%	22	22	27%	355871 ms	$1588,1 \cdot 10^{-4}$ ms
	35	8	27%	22	22	27%	361155 ms	$1611,7 \cdot 10^{-4}$ ms
	40	8	27%	22	22	27%	311019 ms	$1387,9 \cdot 10^{-4}$ ms

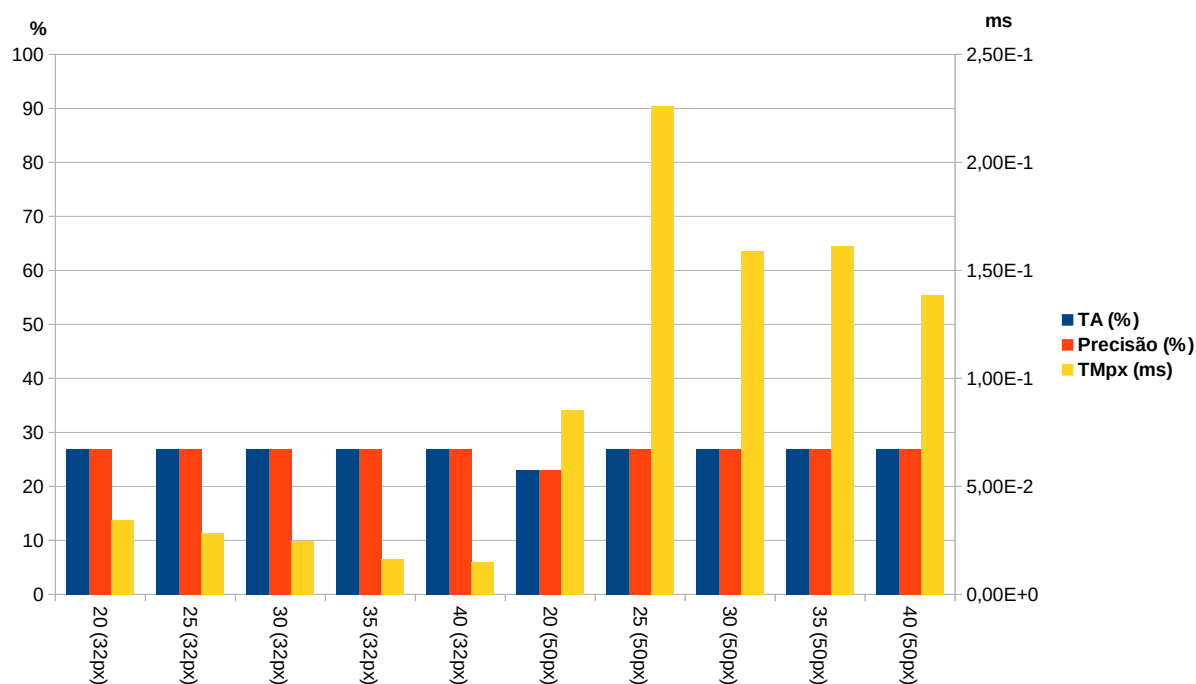


Figura 4.1 - Representação gráfica dos resultados da abordagem Haar-like em cascata

Os resultados obtidos com LBP em cascata podem ser vistos na Tabela 4.2 e a representação gráfica dos resultados na Figura 4.2. Pode-se notar que a taxa de acerto e precisão também ficaram em torno de 27% para todos os classificadores, exceto pelo classificador de janela de detecção de 32 pixels e com 20 níveis, que teve uma taxa de acerto de 100% e precisão de 2,1%. Uma explicação para tal fato é o que o classificador tenha se generalizado muito.

Tabela 4.2 - Reconhecimento de pista de pouso em imagens de média resolução utilizando LBP em cascata.

Tamanho	Níveis	VP	TA	FP	FN	Precisão	TMI	TMpx
32px	20	30	100%	1357	0	2,1%	35,74 ms	$0,159 \cdot 10^{-4}$ ms
	25	7	23%	23	23	23%	47502 ms	$211,98 \cdot 10^{-4}$ ms
	30	8	27%	22	22	27%	29877 ms	$133,32 \cdot 10^{-4}$ ms
	35	8	27%	22	22	27%	24639 ms	$109,95 \cdot 10^{-4}$ ms
	40	8	27%	22	22	27%	15709 ms	$70,1 \cdot 10^{-4}$ ms
50px	20	8	27%	22	22	27%	220230 ms	$982,78 \cdot 10^{-4}$ ms
	25	8	27%	22	22	27%	156728 ms	$699,41 \cdot 10^{-4}$ ms
	30	8	27%	22	22	27%	134298 ms	$599,31 \cdot 10^{-4}$ ms
	35	8	27%	22	22	27%	54905 ms	$245,02 \cdot 10^{-4}$ ms
	40	8	27%	22	22	27%	41133 ms	$183,56 \cdot 10^{-4}$ ms

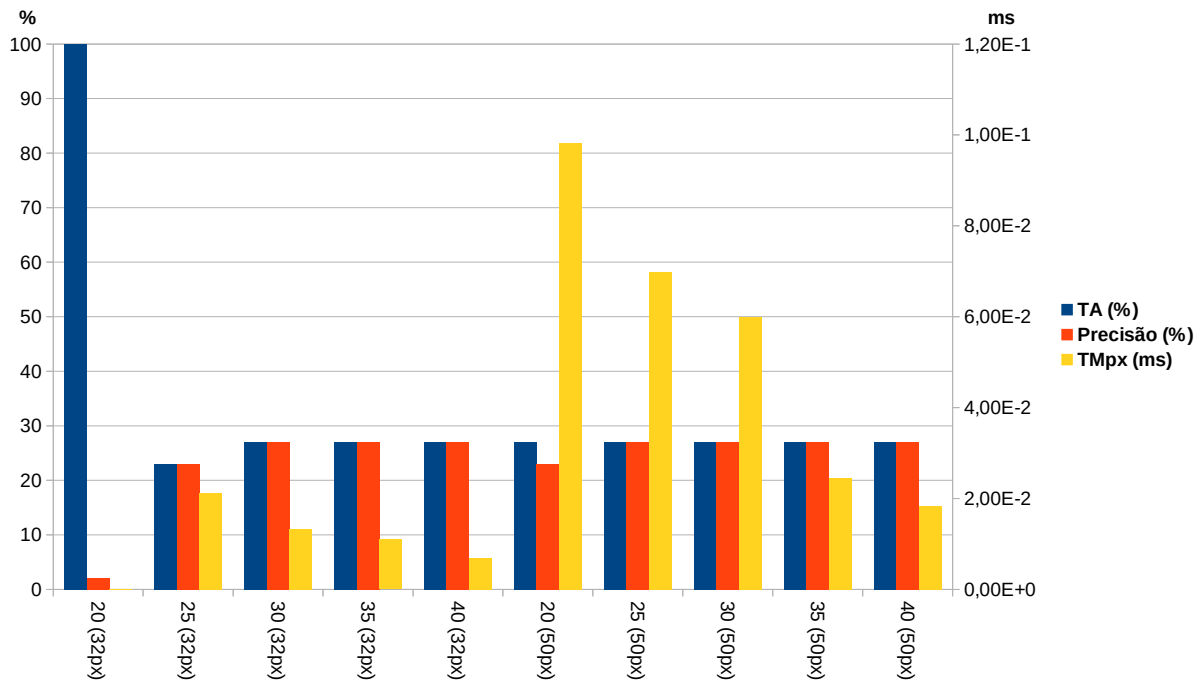


Figura 4.2 - Representação gráfica dos resultados da abordagem LBP em cascata

Na fase de classificação da abordagem HOG+SVM utilizou-se os parâmetros de *kernel* encontradas no processo de treinamento. A Tabela 4.3 mostra as configurações utilizadas nesse estudo de caso, a Tabela 4.4 mostra os resultados obtidos e a Figura 4.3 ilustra graficamente os resultados.

Na Tabela 4.3 são mostrados todos os parâmetros obtidos na fase de treinamento, que variam de *kernel* para *kernel*. Foram obtidos ao todo, nove classificadores, ou seja, cada janela (32, 64 e 128 pixels) foi treinada com os três tipos de *kernels* utilizados nesse trabalho, Linear, RBF e Sigmoidal. A maneira que as funções *kernel* são calculadas foi mostrado na Seção 2.3.1.2. A tabela também traz outras informações, como o número de atributos que aquele tamanho da janela de detecção gera, assim como o número de vetores de suporte encontrados no treinamento.

O desempenho dos classificadores HOG+SVM ficou muito abaixo das abordagens que utilizam classificadores em cascata. Em relação ao tempo de processamento, pode-se notar que o *kernel* linear, em média, é o mais rápido, seguido pelo RBF.

Tabela 4.3 - Características do HOG+SVM para esse estudo de caso.

Janela	Dimensões	Kernel	γ	r	C	Vetores de Suporte
32 px	324	Linear	-	-	0,1	1
		RBF	$0,50625$	-	12,5	4957
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	2400
64 px	1764	Linear	-	-	0,5	1594
		RBF	$3,375 \cdot 10^{-2}$	-	62,5	2336
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	2400
128 px	8100	Linear	-	-	0,1	1014
		RBF	$2,25 \cdot 10^{-3}$	-	62,5	1124
		Sigmoidal	$3,375 \cdot 10^{-2}$	0,1	62,5	2400

Tabela 4.4 - Reconhecimento de pista de pouso em imagens de média resolução utilizando HOG+SVM.

Janela	Kernel	VP	TA	FP	FN	Precisão	TMI	TMpx
32 px	Linear	30	100%	481	0	5,87%	218815,4 ms	$976,5 \cdot 10^{-4}$ ms
	RBF	29	96,67%	665	1	4,18%	292011,3 ms	$1303,1 \cdot 10^{-4}$ ms
	Sigmoidal	30	100%	696	0	4,13%	302685,4 ms	$1350,7 \cdot 10^{-4}$ ms
64 px	Linear	3	10%	37	37	7,5%	440637,4 ms	$1966,4 \cdot 10^{-4}$ ms
	RBF	2	6,67%	64	28	3,03%	794476,2 ms	$3545,4 \cdot 10^{-4}$ ms
	Sigmoidal	4	13,33%	34	26	10,53%	1016158,4 ms	$4534,7 \cdot 10^{-4}$ ms
128 px	Linear	4	13,33%	118	26	3,28%	333833,3 ms	$1489,7 \cdot 10^{-4}$ ms
	RBF	5	16,67%	207	25	2,36%	1528649,5 ms	$6821,7 \cdot 10^{-4}$ ms
	Sigmoidal	5	16,13%	142	26	3,4%	2576707,4 ms	$11498,7 \cdot 10^{-4}$ ms

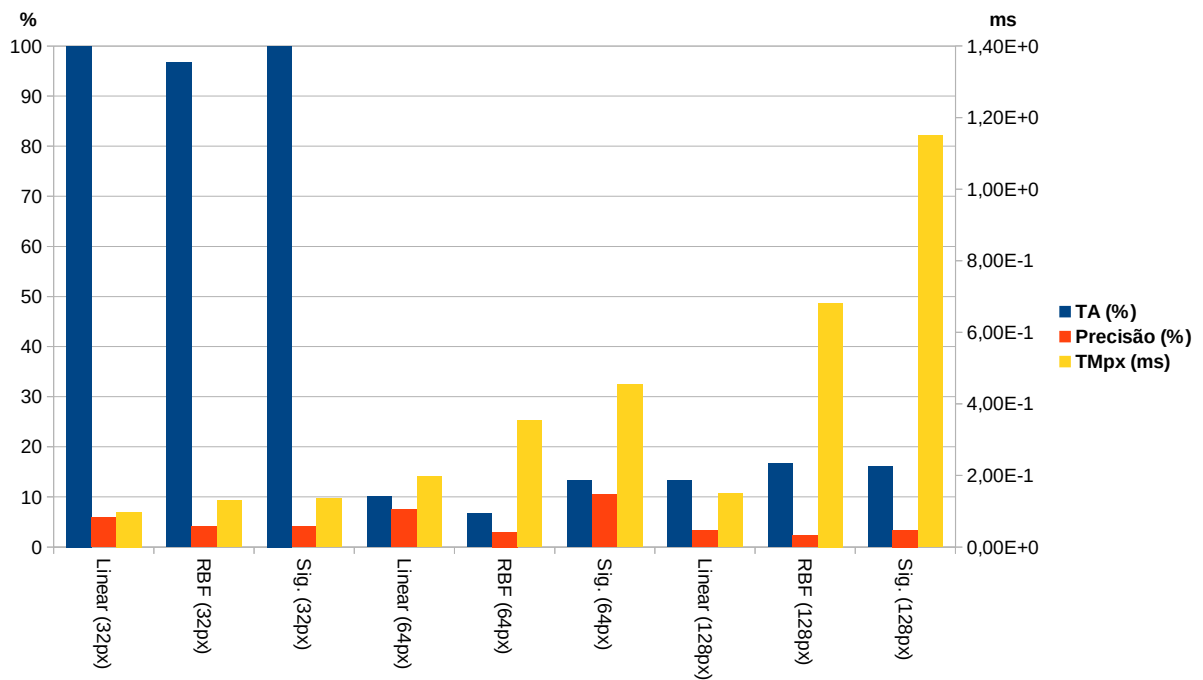


Figura 4.3 - Representação gráfica dos resultados da abordagem HOG+SVM

Na Figura 4.4 são mostrados exemplos de verdadeiros positivos e falsos positivos obtidos nesse estudo de caso. Não foi possível notar nenhum tipo de padrão na ocorrência dos falsos positivos.

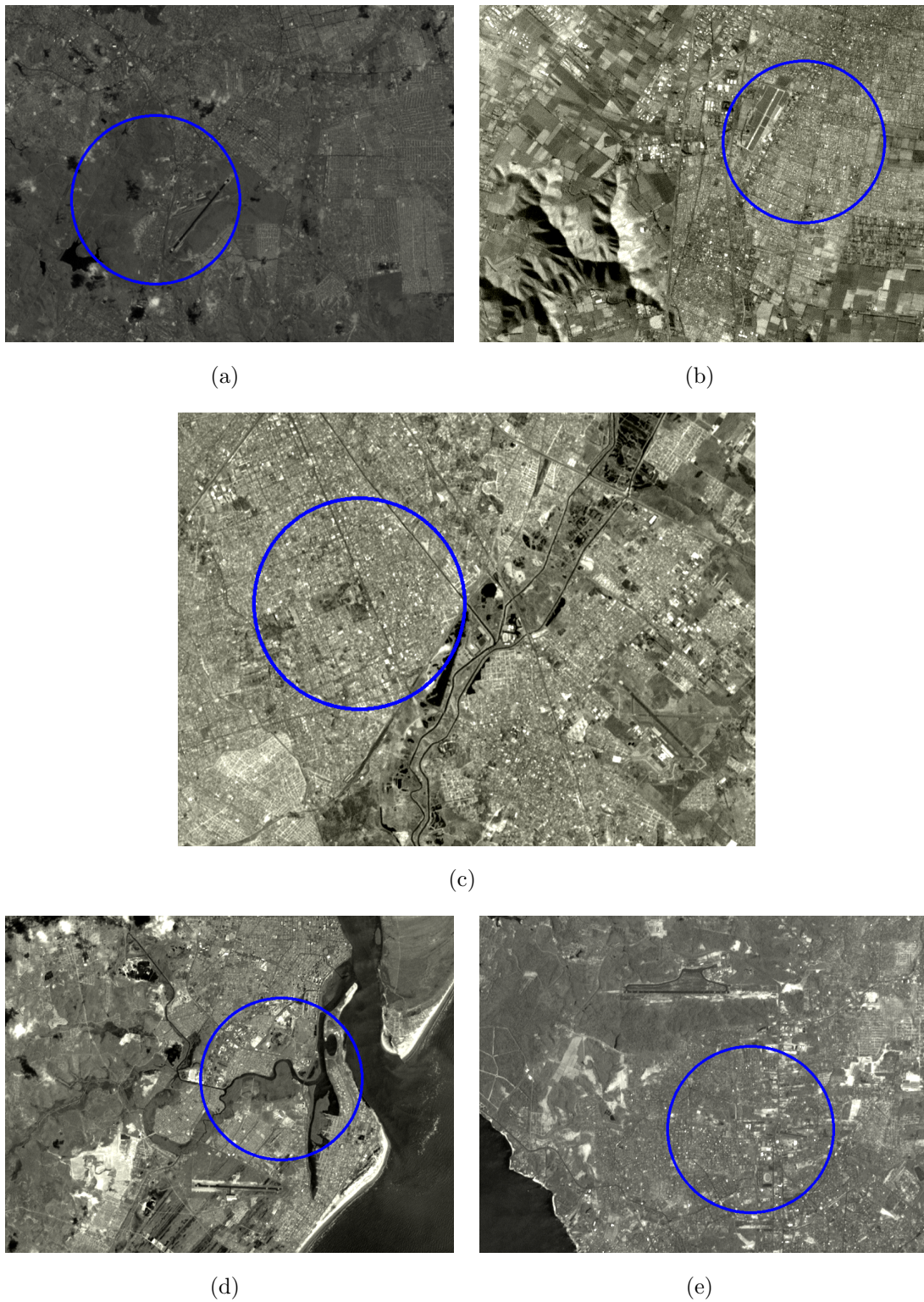


Figura 4.4 - Exemplo de imagens usadas na classificação, onde (a) (b) são verdadeiros positivos e (c), (d) e (e) são falsos positivos. As imagens foram tratadas e recortadas a fim de permitir uma melhor visualização.

No intuito de permitir uma melhor análise em relação ao desempenho das abordagens para o estudo de caso de reconhecimento de pistas de pouso em imagens de média resolução, criou-se um gráfico, Figura 4.5, com os classificadores de cada abordagem que apresentaram melhor equilíbrio entre taxa de acerto e a precisão.

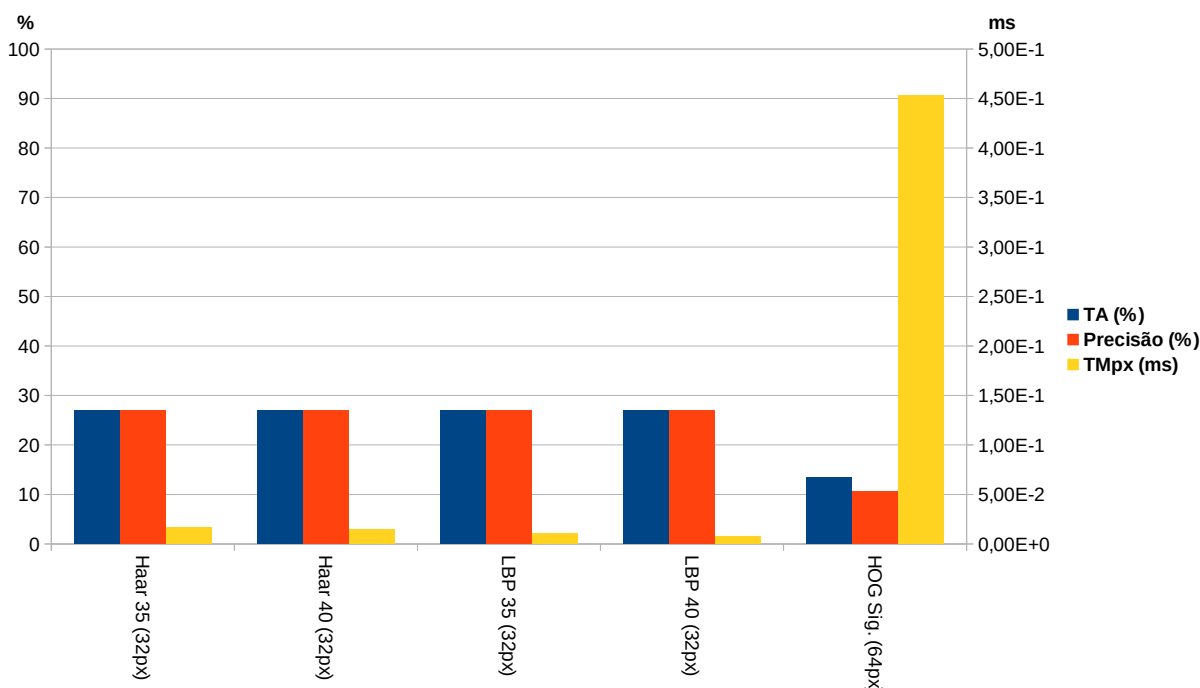


Figura 4.5 - Classificadores com melhor equilíbrio entre taxa de acerto e a precisão.

É possível visualizar no gráfico apresentado na Figura 4.5 que o desempenho de todas as abordagens foi relativamente ruim, pois a precisão e a taxa de acerto ficaram abaixo de 30%. Com a análise dos resultados de cada abordagem, Tabela 4.1, 4.2 e 4.4, encontram-se classificadores com taxa de acerto acima de 90%, mas no entanto, a precisão é baixa, abaixo de 6%. Nestes casos acerta-se o alvo, porém encontra-se um número muito elevado de falsos positivos. Pode-se notar também que em média os métodos que utilizam classificadores em cascata são mais rápidos do que HOG+SVM e obtêm melhores resultados.

Uma possível explicação para o baixo desempenho de modo geral, é que talvez as imagens de média resolução da banda pancromática não tenham a quantidade de

detalhes suficiente.

O artigo publicado com resultados preliminares desse estudo de caso, foi em [Cruz et al. \(2013c\)](#). Nesse trabalho é feito um estudo do emprego do Haar-*like* em cascata para imagens orbitais de média e alta resolução.

4.2 Pista de Pouso em Imagens de Alta Resolução

Para o teste de reconhecimento de pistas de pouso de concreto ou asfalto em imagens orbitais de alta resolução espacial, foram utilizadas imagens do aplicativo Google Maps ([GOOGLE, 2013](#)). Esse tipo de objeto pode ser encontrado com facilidade em imagens de satélite, pois normalmente está presente em bases aéreas e em cidades de grande e médio porte ao redor do mundo.

No conjunto de treinamento foram utilizados 2021 amostras positivas e 2500 amostras negativas. Tal tamanho foi obtido na tentativa de formar um conjunto com dados que representassem o melhor possível a classe de objetos em questão. No teste de classificação foram utilizadas 50 imagens, onde elas não possuíam nenhuma região dentro o conjunto positivo ou negativo e possuíam uma tamanho médio de 858 por 858 pixels. Cada imagem teste possuía somente um aeroporto.

Foram testadas diversas configurações das abordagens. Assim como no estudo de caso anterior, variou-se, nas abordagens de classificadores em cascata, os números de estágios da cascata mas manteve-se os parâmetros de taxa mínima de acerto (99,5%) e taxa máxima de falsos positivos (50%) durante o treinamento.

Os resultados obtidos com Haar-*like* em cascata podem ser vistos na Tabela 4.5 e a representação gráfica dos resultados na Figura 4.6. A taxa de acerto ficou por volta de 80% enquanto a precisão ficou na faixa de 50% a 80%. Em relação a velocidade de processamento pode-se notar que janelas de detecção menores são mais rápidas.

Tabela 4.5 - Reconhecimento de pista de pouso em imagens de alta resolução utilizando Haar-like em cascata.

Tamanho	Níveis	VP	TA	FP	FN	Precisão	TMI	TMpx
32px	20	40	80%	47	9	46%	302,5 ms	$4,11 \cdot 10^{-4}$ ms
	25	40	80%	36	9	52,6%	354,1 ms	$4,81 \cdot 10^{-4}$ ms
	30	40	80%	37	9	51,9%	276,87 ms	$3,76 \cdot 10^{-4}$ ms
	35	41	82%	39	8	51,3%	265,31 ms	$3,61 \cdot 10^{-4}$ ms
	40	43	86%	60	7	43%	178,81 ms	$2,43 \cdot 10^{-4}$ ms
50px	20	39	78%	13	11	75%	2788,8 ms	$37,90 \cdot 10^{-4}$ ms
	25	40	80%	10	10	80%	3536 ms	$48,10 \cdot 10^{-4}$ ms
	30	39	78%	12	11	76,5%	3580,1 ms	$48,66 \cdot 10^{-4}$ ms
	35	39	78%	11	11	78%	2660,6 ms	$36,16 \cdot 10^{-4}$ ms
	40	40	80%	11	10	78,4%	2477,5 ms	$33,67 \cdot 10^{-4}$ ms

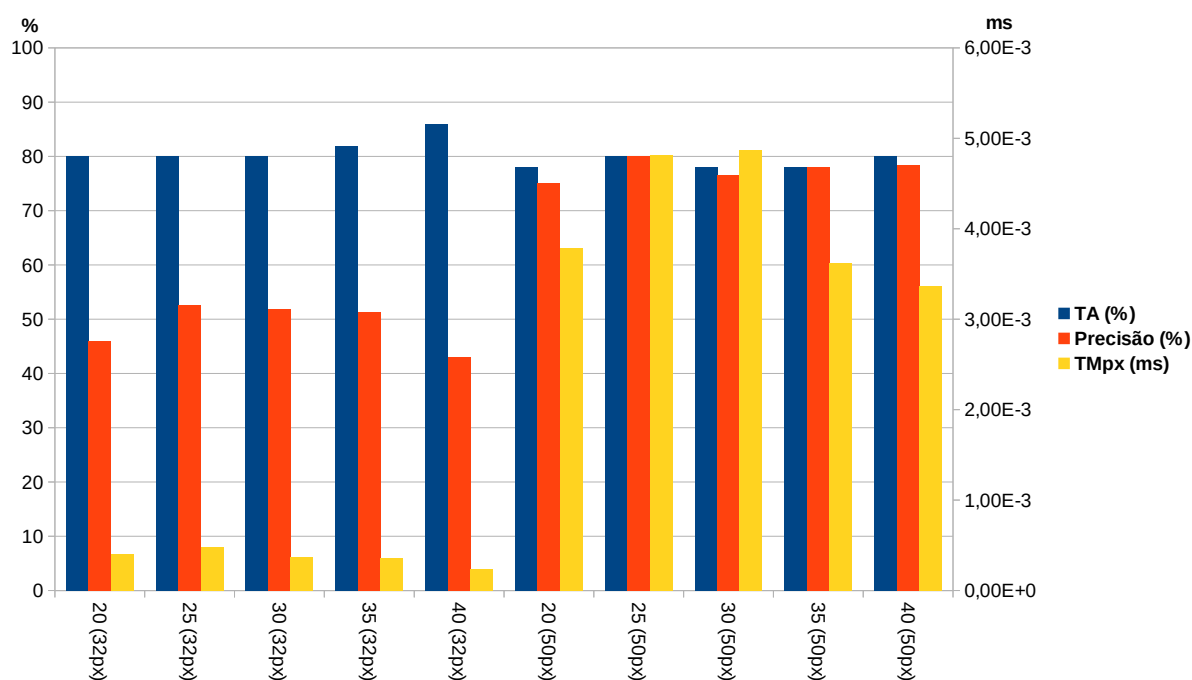


Figura 4.6 - Representação gráfica dos resultados da abordagem Haar-like em cascata

Os resultados obtidos com LBP em cascata podem ser vistos na Tabela 4.6. Na maioria dos casos a taxa de acerto e de precisão ficaram por volta de 80%. Na representação gráfica dos resultados, Figura 4.7, pode-se ver claramente o que, às vezes, acontece com classificadores em cascata: quanto mais estágios a cascata tem, mais rápida ela é; e quanto menor a janela de detecção, mais rápido o classificador fica.

Tabela 4.6 - Reconhecimento de pista de pouso em imagens de alta resolução utilizando LBP em cascata.

Tamanho	Níveis	VP	TA	FP	FN	Precisão	TMI	TMpx
32px	20	39	78%	11	11	78%	367,66 ms	$5 \cdot 10^{-4}$ ms
	25	39	78%	11	11	78%	209,65 ms	$2,85 \cdot 10^{-4}$ ms
	30	39	78%	11	11	78%	124,57 ms	$1,69 \cdot 10^{-4}$ ms
	35	40	80%	11	10	78,4%	81,75 ms	$1,11 \cdot 10^{-4}$ ms
	40	38	76%	24	12	61,3%	57,15 ms	$0,78 \cdot 10^{-4}$ ms
50px	20	39	78%	11	11	78%	3002 ms	$40,8 \cdot 10^{-4}$ ms
	25	39	78%	11	11	78%	1945,7 ms	$26,4 \cdot 10^{-4}$ ms
	30	40	80%	10	10	80%	1124,4 ms	$15,3 \cdot 10^{-4}$ ms
	35	40	80%	10	10	80%	649,24 ms	$8,82 \cdot 10^{-4}$ ms
	40	40	80%	10	10	80%	393,55 ms	$5,35 \cdot 10^{-4}$ ms

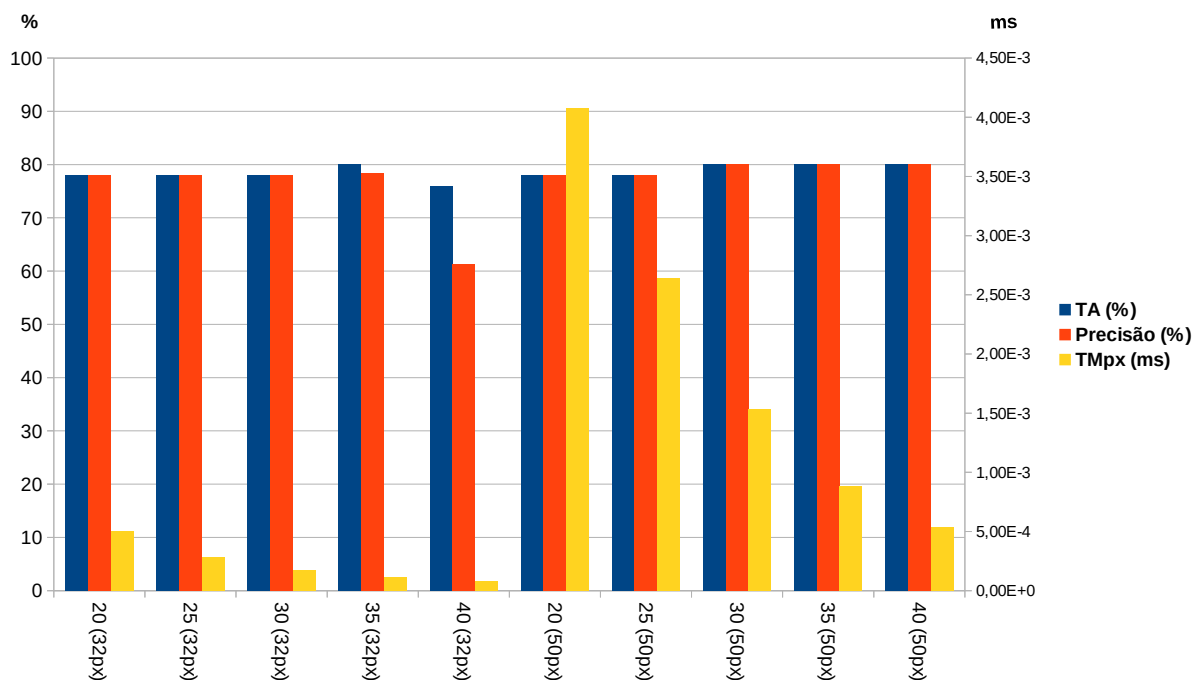


Figura 4.7 - Representação gráfica dos resultados da abordagem LBP em cascata

Na fase de classificação da abordagem HOG+SVM utilizou-se os parâmetros de *kernel* encontradas no processo de treinamento. A Tabela 4.7 mostra as configurações utilizadas nesse estudo de caso, a Tabela 4.8 mostra os resultados obtidos e a Figura 4.8 ilustra graficamente os resultados.

Na Tabela 4.7 são mostrados todos os parâmetros obtidos na fase de treinamento, que variam de *kernel* para *kernel*. Assim como no estudo de caso anterior, foram obtidos ao todo, nove classificadores, ou seja, cada janela (32, 64 e 128 pixels) foi treinada com os três tipos de *kernels* utilizados nesse trabalho, Linear, RBF e Sig-moidal. A tabela também informa o número de atributos que aquele tamanho da janela de detecção gera, assim como o número de vetores de suporte encontrados no treinamento.

Tabela 4.7 - Características do HOG+SVM para esse estudo de caso.

Janela	Dimensões	Kernel	γ	\mathbf{r}	\mathbf{C}	Vetores de Suporte
32 px	324	Linear	-	-	0,1	1
		RBF	$0,50625$	-	62,5	5880
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	4042
64 px	1764	Linear	-	-	0,5	3072
		RBF	$3,375 \cdot 10^{-2}$	-	12,5	3315
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	4042
128 px	8100	Linear	-	-	0,1	1823
		RBF	$3,375 \cdot 10^{-2}$	-	2,5	3356
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	3752

Analisando a Tabela 4.8, pode-se destacar que os classificadores com janela de detecção de 32 pixels tiveram precisão abaixo de 5%. Estes classificadores se generalizaram mais que o esperado. Uma possível explicação para esse fato, é que esse tamanho de janela talvez não consiga fornecer a quantidade suficiente de feições para que se possa realizar o processo de reconhecimento de forma adequada utilizando a abordagem HOG+SVM.

Pode-se também notar que quanto maior a janela de detecção, maior é o tempo de processamento. Isso se deve principalmente à necessidade de se realizar mais cálculos quando se tem uma quantidade maior de atributos. Outro ponto a se destacar, é o fato que os classificadores que utilizaram *kernel* linear, foram mais rápidos, isso analisando os classificadores com o mesmo tamanho de janela de detecção. O segundo mais rápido foram aqueles que utilizaram a função *kernel* Sigmoidal.

Tabela 4.8 - Reconhecimento de pista de pouso em imagens de alta resolução utilizando HOG+SVM.

Janela	Kernel	VP	TA	FP	FN	Precisão	TMI	TMpx
32 px	Linear	47	94%	961	3	4,66%	20124,7 ms	$273,5 \cdot 10^{-4}$ ms
	RBF	45	90%	992	5	4,34%	62182,6 ms	$845,1 \cdot 10^{-4}$ ms
	Sigmoidal	46	92%	923	4	4,75%	48644,9 ms	$661,1 \cdot 10^{-4}$ ms
64 px	Linear	23	46%	27	27	46%	46024,1 ms	$625,5 \cdot 10^{-4}$ ms
	RBF	12	24%	41	38	22,64%	254805,2 ms	$3463 \cdot 10^{-4}$ ms
	Sigmoidal	23	46%	27	27	46%	226746 ms	$3081,6 \cdot 10^{-4}$ ms
128 px	Linear	27	54%	26	23	50,94%	58312,8 ms	$792,5 \cdot 10^{-4}$ ms
	RBF	24	48%	95	26	20,17%	1142245,2 ms	$15523,6 \cdot 10^{-4}$ ms
	Sigmoidal	23	46%	27	27	46%	803039,5 ms	$10913,7 \cdot 10^{-4}$ ms

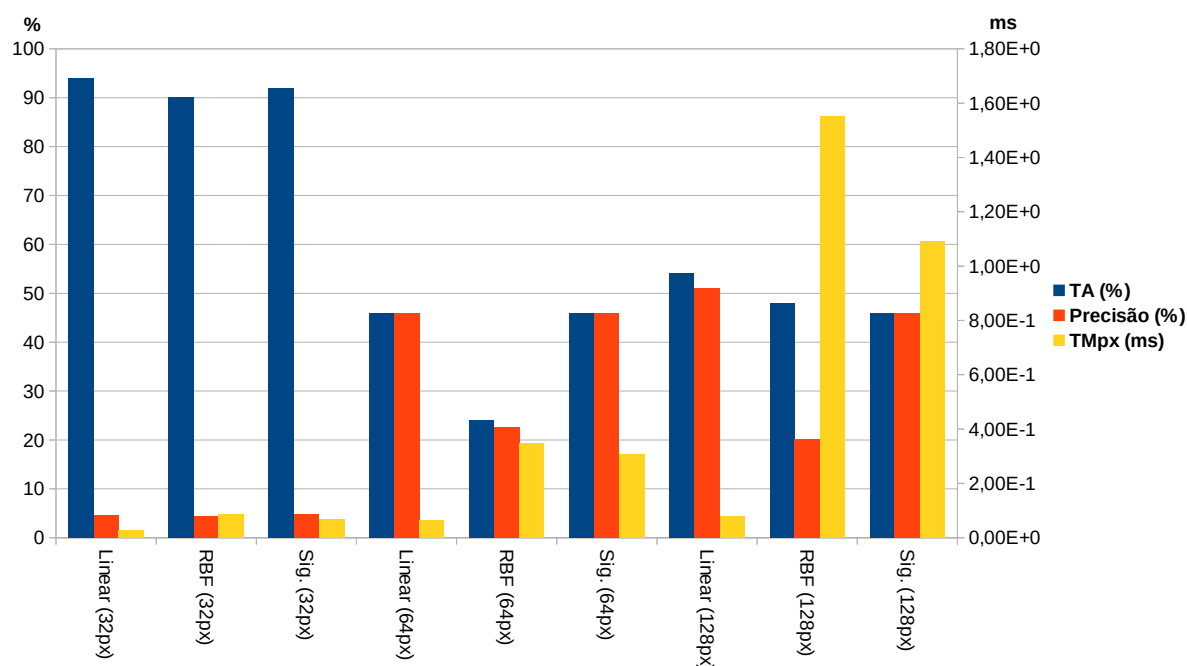


Figura 4.8 - Representação gráfica dos resultados da abordagem HOG+SVM

A Figura 4.9 mostra exemplos de falsos positivos e de verdadeiros positivos. Assim

como o estudo de caso anterior, não foi possível identificar nenhum tipo de padrão na ocorrência dos falsos positivos.

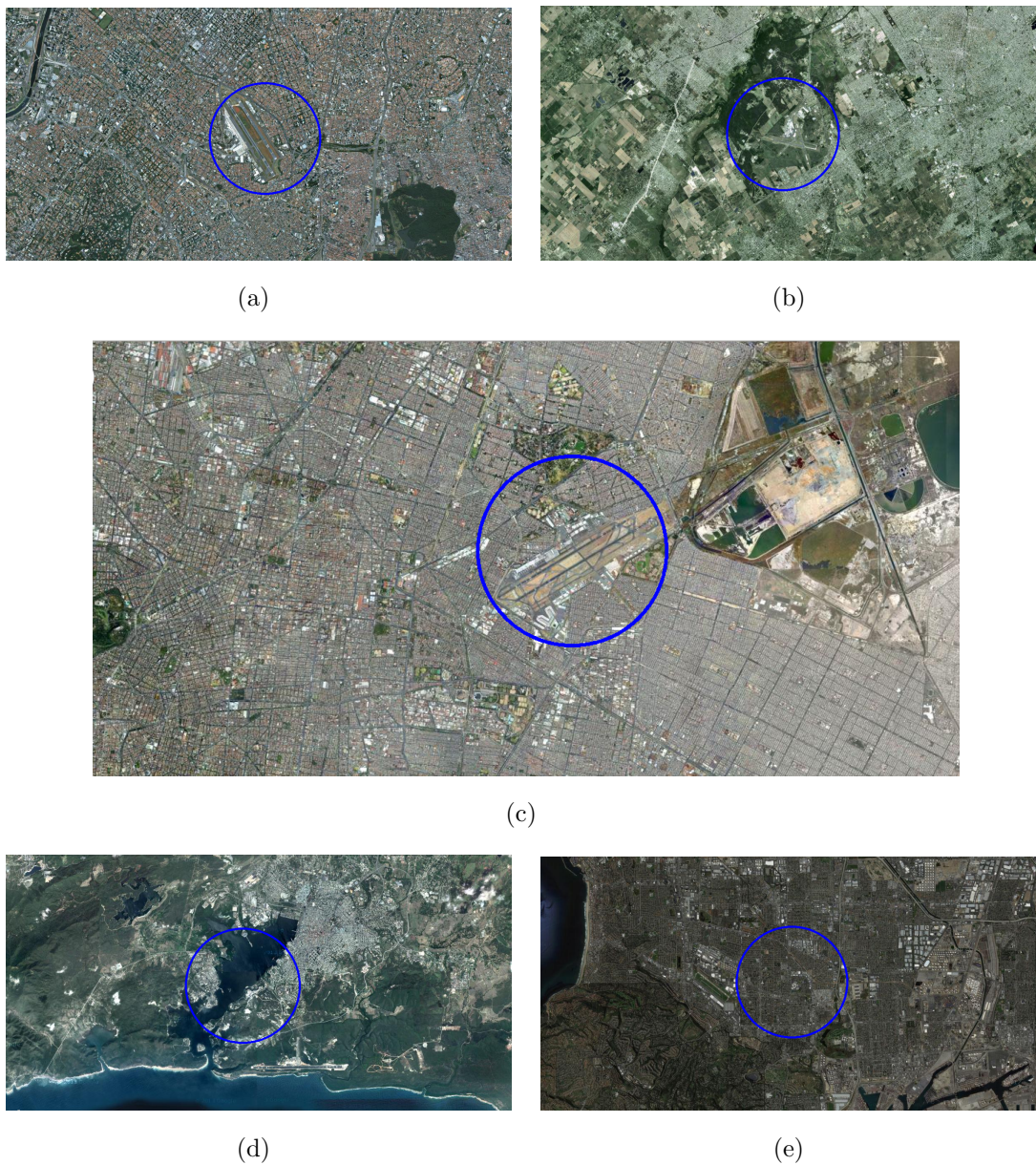


Figura 4.9 - Exemplo de imagens usadas na classificação, onde (a), (b) e (c) são verdadeiros positivos e (d) e (e) são falsos positivos.

A fim de permitir uma melhor visualização dos resultados desse estudos de caso,

foi criado um gráfico, Figura 4.10, com os classificadores que apresentaram melhor equilíbrio entre taxa de acerto e precisão.

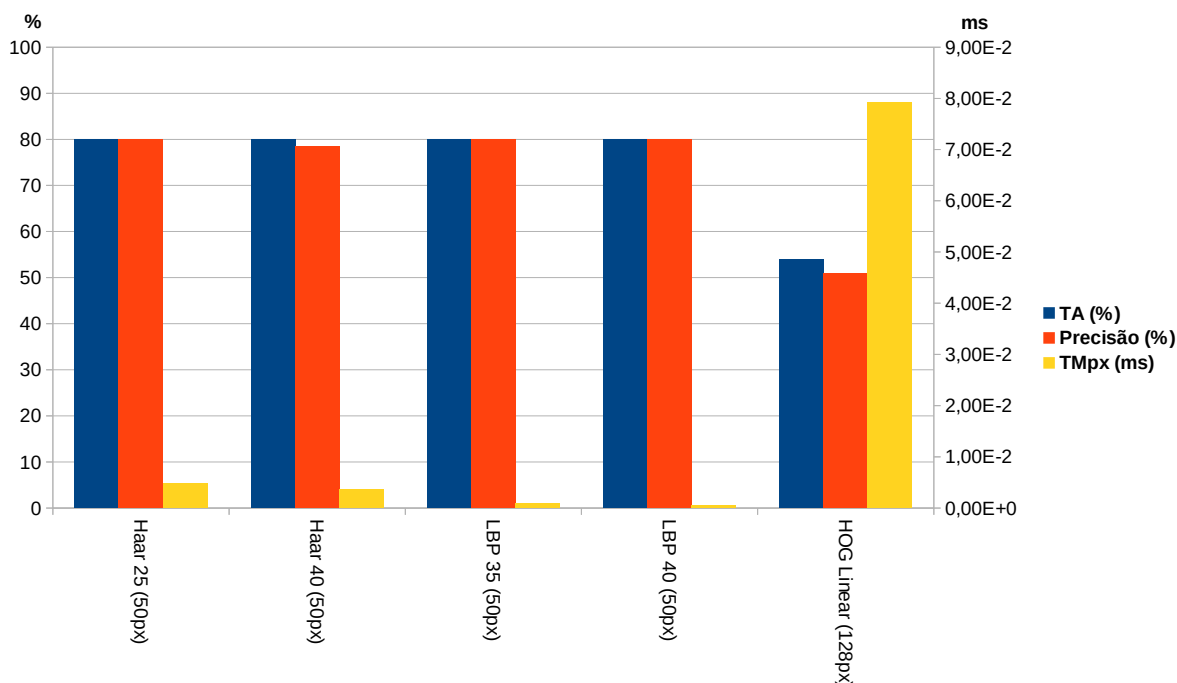


Figura 4.10 - Classificadores com melhor equilíbrio entre taxa de acerto e a precisão.

Os resultados desse estudo de caso foram satisfatórios, devido principalmente às abordagens com classificadores em cascata, onde os melhores classificadores obtiveram por volta de 80% de taxa de acerto e precisão, e tiveram tempo de processamento relativamente baixo, o melhor caso da Figura 4.10 processa uma imagem de 1000x1000 pixels em 535 ms.

Apesar de utilizar janelas de detecção maiores que nas outras abordagens, HOG+SVM teve desempenho pior, mas razoável, taxa de acerto de 54% e precisão por volta de 51% para seu melhor classificador. Outro ponto é o tempo de processamento, que em geral foi muito acima do que obtido nas outras abordagens. Na Tabela 4.8, pode-se encontrar classificadores com taxa de acerto de 90% ou mais, no entanto a precisão é muito baixa, menor que 5%, o que resulta em uma alta taxa de falsos positivos.

Com resultados preliminares desse estudo de caso, foi possível fazer algumas publicações. Em Cruz et al. (2013b), emprega-se LBP em cascata em imagens orbitais. Já com Haar-*like* em cascata, foram publicados os trabalhos Cruz et al. (2013d) e Cruz et al. (2013c). O primeiro faz o estudo da aplicação da abordagem somente em imagens de alta resolução, enquanto o segundo faz uma comparação entre o uso de imagens de alta e média resolução para o reconhecimento do alvo.

4.3 Campo de Futebol em Imagens Alta Resolução

Para o teste de reconhecimento de campos de futebol em imagens orbitais de alta resolução espacial, foram utilizadas imagens do aplicativo Google Maps (GOOGLE, 2013). Esse tipo de alvo pode ser encontrado facilmente e em grande quantidade em imagens orbitais de alta resolução espacial devido a popularidade do esporte pelo mundo.

A fim de criar um conjunto de treinamento com representatividade dos indivíduos desse tipo classe, foram utilizadas 3419 amostras positivas e 4000 amostras negativas. No teste de classificação foram utilizadas 69 imagens, onde elas não possuíam nenhuma região dentro o conjunto positivo ou negativo. Cada imagem teste possuía no mínimo um campo de futebol e no total são 113 campos. O tamanho médio dessas imagens era de 856 por 856 pixels.

As abordagens foram testadas com diferentes configurações. Naquelas em que o conceito de classificadores em cascata é utilizado, variou-se, no estágio de treinamento, os números de estágios da cascata mas manteve-se os parâmetros de taxa mínima de acerto (99,5%) e taxa máxima de falsos positivos (50%).

Os resultados obtidos com Haar-*like* em cascata podem ser vistos na Tabela 4.9 e a representação gráfica dos resultados na Figura 4.11. Inicialmente o treinamento foi feito com 15, 20, 25, 30 e 35 estágios na cascata. Observou-se então que o classificador com 15 estágios apresentava uma taxa de acerto maior que o de 20 estágios, porém apresentava uma precisão bem menor. Desejando encontrar um classificador com melhor balanceamento entre taxa de acerto e precisão, treinou-se classificadores com 16, 17, 18 e 19 estágios. Exceto por um ou outro classificador, a precisão aumentou quando se aumentou o número de estágios. O inverso aconteceu com a taxa de acerto, que diminuiu quando se aumentou o número de estágios. Uma possível explicação para isso, é que nesse caso, quanto mais estágios a cascata possuía, mas especializada

ela ficou.

Tabela 4.9 - Reconhecimento de campo de futebol em imagens de alta resolução utilizando feições Haar-like em cascata.

Tamanho	Níveis	VP	TA	FP	FN	Precisão	TMI	TMpx
32px	15	89	78,76%	261	24	25,43%	20,05 ms	$27,37 \cdot 10^{-4}$ ms
	16	79	69,91%	138	34	36,41%	19,74 ms	$26,95 \cdot 10^{-4}$ ms
	17	74	65,49%	94	39	44,05%	19 ms	$25,94 \cdot 10^{-4}$ ms
	18	67	59,29%	79	49	45,89%	20,77 ms	$28,35 \cdot 10^{-4}$ ms
	19	52	46,02%	29	61	64,20%	20,46 ms	$27,94 \cdot 10^{-4}$ ms
	20	49	43,36%	24	64	67,12%	19,39 ms	$26,47 \cdot 10^{-4}$ ms
	25	33	29,20%	4	80	89,19%	20,82 ms	$28,42 \cdot 10^{-4}$ ms
	30	20	17,70%	0	93	100%	20,70 ms	$28,26 \cdot 10^{-4}$ ms
50px	35	18	15,93%	0	95	100%	20,47 ms	$27,94 \cdot 10^{-4}$ ms
	15	93	82,30%	190	20	32,86%	26,70 ms	$36,45 \cdot 10^{-4}$ ms
	16	77	68,14%	60	36	56,20%	25,01 ms	$34,14 \cdot 10^{-4}$ ms
	17	80	70,80%	68	33	54,05%	25,80 ms	$35,23 \cdot 10^{-4}$ ms
	18	69	61,06%	36	44	65,71%	24,31 ms	$33,19 \cdot 10^{-4}$ ms
	19	69	61,06%	46	44	60%	30,98 ms	$42,29 \cdot 10^{-4}$ ms
	20	63	55,75%	32	52	66,32%	29,04 ms	$39,64 \cdot 10^{-4}$ ms
25	36	31,86%	1	77	97,30%	26,88 ms	$36,70 \cdot 10^{-4}$ ms	

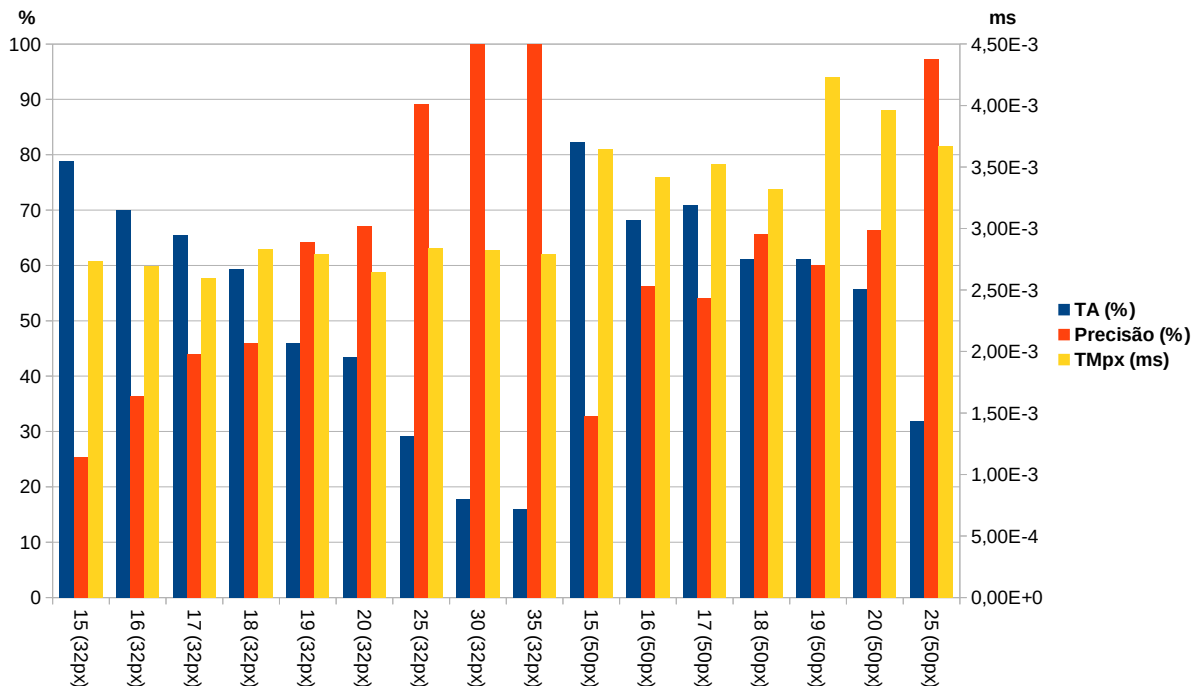


Figura 4.11 - Representação gráfica dos resultados da abordagem Haar-like em cascata

Os resultados obtidos com LBP em cascata podem ser vistos na Tabela 4.10 e a representação gráfica dos resultados na Figura 4.12. Assim como no caso do Haar-like, treinou-se classificadores com 20, 25, 30, 35 e 40 estágios na cascata. Nos classificadores de 30, 35 e 40 estágios o desempenho foi ruim, com taxa de acerto menor que 5%. Investigou-se então, classificadores com cascatas entre 20 e 25, já que esse classificadores possuíam os melhores desempenhos dentre os demais. No geral, todos os classificadores obtidos tiveram precisão abaixo de 30% e somente três classificadores tiveram taxa de acerto acima de 50%. Uma possível explicação para tal comportamento, é a especialização extrema para os classificadores com 30, 35 e 40 estágios e por outro lado os classificadores de 20 estágios ficaram muito generalistas.

Tabela 4.10 - Reconhecimento de campo de futebol em imagens de alta resolução utilizando LBP em cascata.

Tamanho	Níveis	VP	TA	FP	FN	Precisão	TMI	TMpx
32px	20	106	93,81%	1031	7	9,32%	27,53 ms	$37,58 \cdot 10^{-4}$ ms
	21	63	55,75%	204	52	23,60%	24,11 ms	$32,92 \cdot 10^{-4}$ ms
	22	47	41,59%	140	66	25,13%	24,26 ms	$33,11 \cdot 10^{-4}$ ms
	23	38	33,63%	97	75	28,15%	23,89 ms	$32,61 \cdot 10^{-4}$ ms
	24	34	30,09%	128	79	20,99%	25,92 ms	$35,39 \cdot 10^{-4}$ ms
	25	25	22,12%	67	88	27,17%	27,44 ms	$37,47 \cdot 10^{-4}$ ms
	30	1	0,88%	8	112	11,11%	27,08 ms	$36,97 \cdot 10^{-4}$ ms
	35	0	0%	2	113	0%	26,63 ms	$36,36 \cdot 10^{-4}$ ms
	40	0	0%	2	113	0%	26,41 ms	$36,05 \cdot 10^{-4}$ ms
50px	20	96	84,96%	476	17	16,78%	49,69 ms	$67,83 \cdot 10^{-4}$ ms
	21	44	38,94%	103	69	29,93%	40,72 ms	$55,59 \cdot 10^{-4}$ ms
	22	32	28,32%	50	81	39,02%	40,54 ms	$55,34 \cdot 10^{-4}$ ms
	23	31	27,43%	64	82	32,63%	42,85 ms	$58,50 \cdot 10^{-4}$ ms
	24	18	15,93%	41	95	30,51%	41,72 ms	$56,95 \cdot 10^{-4}$ ms
	25	12	10,62%	27	101	30,77%	48,25 ms	$65,86 \cdot 10^{-4}$ ms
	30	3	2,65%	3	110	50%	47,01 ms	$64,17 \cdot 10^{-4}$ ms
	35	3	2,65%	3	110	50%	48,22 ms	$65,82 \cdot 10^{-4}$ ms
	40	0	0%	2	113	0%	49,28 ms	$67,27 \cdot 10^{-4}$ ms

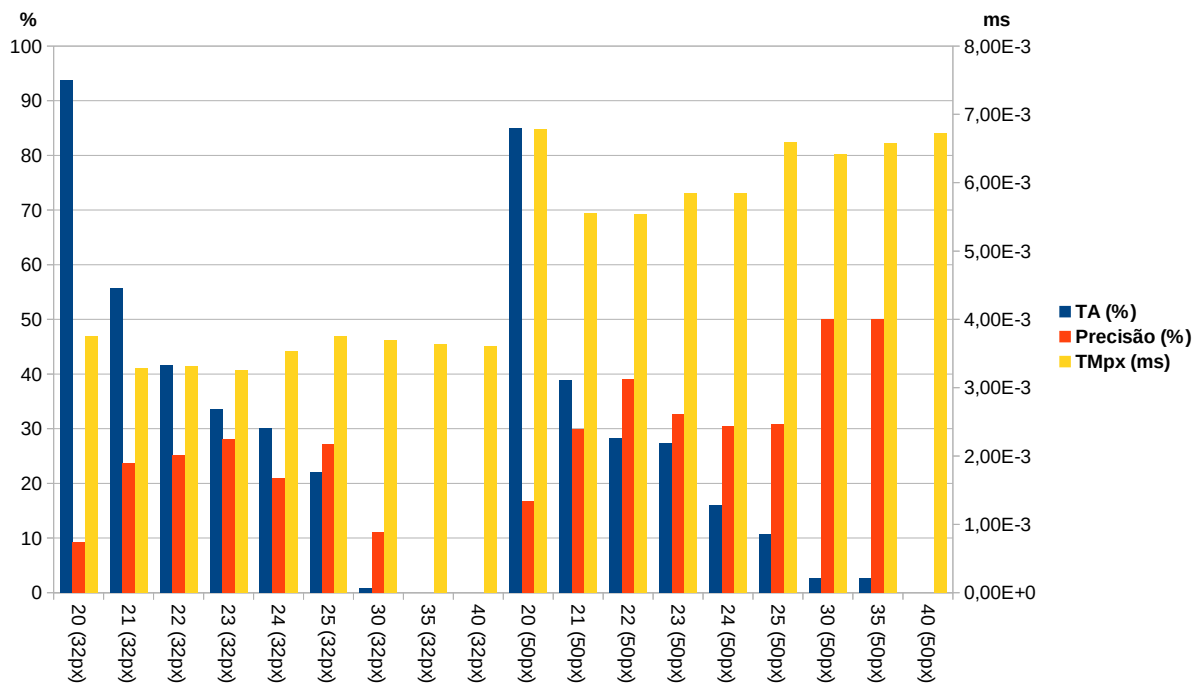


Figura 4.12 - Representação gráfica dos resultados da abordagem LBP em cascata

A Tabela 4.11 mostra os parâmetros utilizados na fase de classificação da abordagem HOG+SVM. Assim como nos estudos de caso anteriores, foram obtidos ao todo, nove classificadores, ou seja, cada janela (32, 64 e 128 pixels) foi treinada com os três tipos de *kernels* utilizados nesse trabalho, Linear, RBF e Sigmoidal. A tabela também informa o número de atributos que aquele tamanho da janela de detecção gera, assim como o número de vetores de suporte encontrados no treinamento.

Tabela 4.11 - Características do HOG+SVM para esse estudo de caso.

Janela	Dimensões	Kernel	γ	\mathbf{r}	\mathbf{C}	Vetores de Suporte
32 px	324	Linear	-	-	2,5	1
		RBF	$0,50625$	-	12,5	2275
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	2156
64 px	1764	Linear	-	-	0,1	1
		RBF	$3,375 \cdot 10^{-2}$	-	12,5	1036
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	6838
128 px	8100	Linear	-	-	0,1	1
		RBF	$2,25 \cdot 10^{-3}$	-	12,5	831
		Sigmoidal	$1 \cdot 10^{-5}$	0,1	0,1	6838

A Tabela 4.12 mostra os resultados obtidos e a Figura 4.13 ilustra graficamente os resultados obtidos. Assim como no estudo de caso de pista de pouso em imagens de alta resolução, os classificadores com janela de detecção de 32 pixels apresentaram precisão baixa, abaixo de 8%. Devido talvez ao número insuficiente de feições encontrados em uma janela de 32 por 32 pixels, os classificadores tenham se generalizado mais que o desejado.

Em relação ao tempo de processamento, pode-se notar que quanto maior a janela de detecção maior é o tempo de processamento. Isso se deve principalmente à necessidade de se realizar mais cálculos por causa do número maior de atributos. Outro ponto a se destacar é que olhando dentre os classificadores com a mesma janela de detecção, aqueles que utilizaram *kernel* linear, foram mais rápidos, seguidos pelo RBF.

Tabela 4.12 - Reconhecimento de campo de futebol em imagens de alta resolução utilizando HOG+SVM.

Janela	Kernel	VP	TA	FP	FN	Precisão	TMI	TMpx
32 px	Linear	103	91,15%	1188	10	7,98%	25073,2 ms	$342,3 \cdot 10^{-4}$ ms
	RBF	100	88,50%	1203	13	7,67%	43374,5 ms	$592,1 \cdot 10^{-4}$ ms
	Sigmoidal	101	89,38%	1233	12	7,57%	70384,5 ms	$960,8 \cdot 10^{-4}$ ms
64 px	Linear	25	22,12%	44	88	36,23%	56886,4 ms	$776,6 \cdot 10^{-4}$ ms
	RBF	23	20,35%	46	90	33,33%	120654,8 ms	$1647,1 \cdot 10^{-4}$ ms
	Sigmoidal	34	30,36%	42	79	44,74%	328079,6 ms	$4478,7 \cdot 10^{-4}$ ms
128 px	Linear	33	28,95%	38	80	46,48%	111023,4 ms	$1515,7 \cdot 10^{-4}$ ms
	RBF	36	32,14%	37	77	49,32%	402882,5 ms	$5499,8 \cdot 10^{-4}$ ms
	Sigmoidal	44	38,94%	30	69	59,46%	1722938,9 ms	$23520,2 \cdot 10^{-4}$ ms

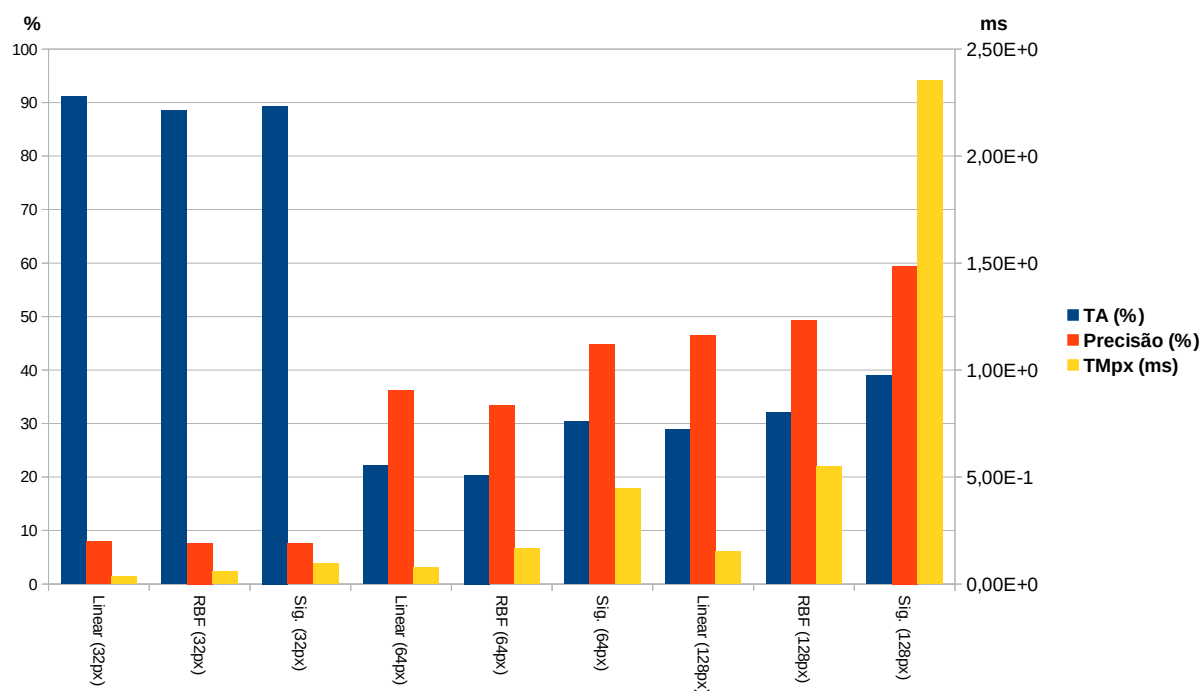


Figura 4.13 - Representação gráfica dos resultados da abordagem HOG+SVM



Figura 4.14 - Exemplo de imagens usadas na classificação, onde (a),(b) e (c) são verdadeiros positivos e (d) e (e) são falsos positivos.

Com intenção de levar a uma melhor análise em relação ao desempenho das abordagens para o estudo de caso de reconhecimento de campos de futebol em imagens de alta resolução, criou-se um gráfico, Figura 4.15, com os classificadores de cada abordagem que apresentaram melhor equilíbrio entre taxa de acerto e precisão.

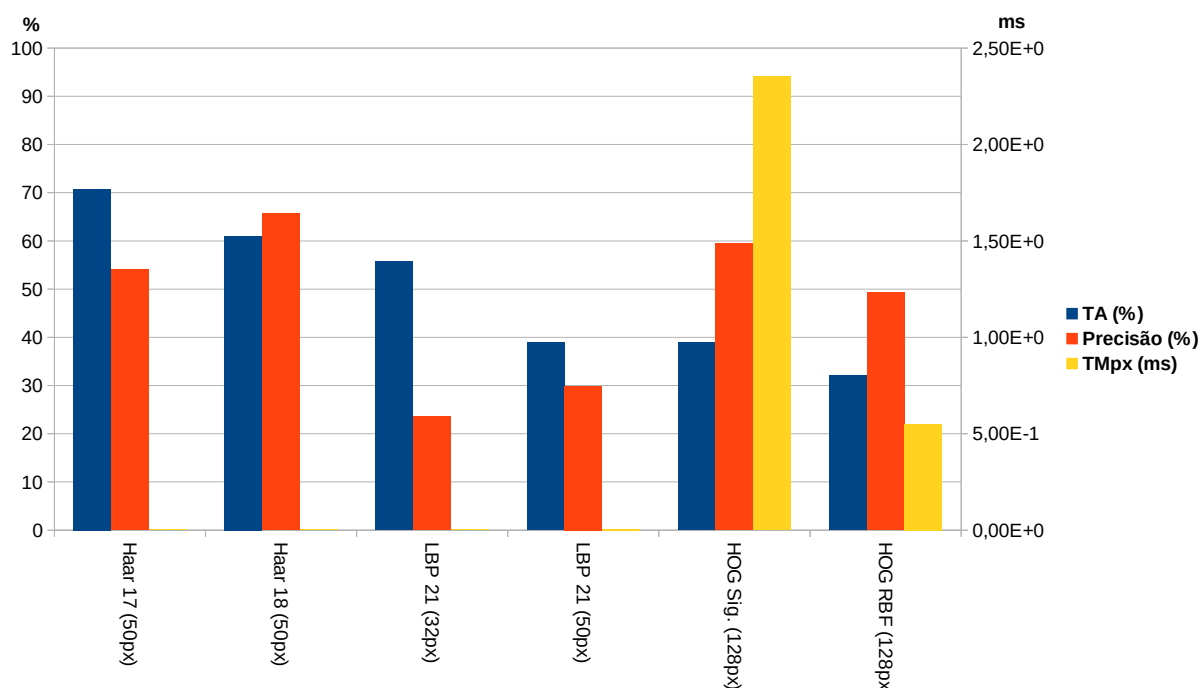


Figura 4.15 - Classificadores com melhor equilíbrio entre taxa de acerto e a precisão.

Nesse estudo de caso, os melhores classificadores foram da abordagem Haar-like em cascata, onde obtiveram aproximadamente 71% e 61% de taxa de acerto e respectivamente, 54% e 66% de precisão. Eles tiveram tempo médio de processamento razoável, em que uma imagem de 1000x1000 pixels levaria em torno de 3,4 segundos para ser processada.

Novamente, HOG+SVM utiliza janelas de detecção maiores que nas outras abordagens e ainda assim, não obteve os melhores resultados. O alto tempo médio de processamento também é algo que se repete em relação aos estudos de caso anteriores. Na Tabela 4.8, nota-se que existem classificadores com taxa de acerto acima de 88%, porém a precisão é muito baixa, abaixo de 8%, o que se traduz em uma alta taxa de falsos positivos.

Foram realizadas publicações com resultados preliminares desse estudo de caso. O artigo Cruz et al. (2012a) fez um estudo da utilização de Haar-like em cascata. Em Cruz et al. (2012b), utilizou-se a abordagem HOG+SVM em imagens aéreas e orbitais. A comparação desse dois métodos em imagens orbitais foi publicado em

Cruz et al. (2013a).

5 CONCLUSÃO

Este trabalho de dissertação teve como propósito abordar metodologias para o reconhecimento automático de objetos em imagens orbitais compostas por bandas do espectro visível. Ao longo desse estudo, mostrou-se a necessidade das abordagens de reconhecimento automático terem uma certa invariância à fatores que dificultam o processo de reconhecimento, como por exemplo, variação intraclasse, variação de pose, de escala etc. Após realizar pesquisas na literatura e mesmo em fóruns de discussão não científicos, chegou-se as abordagens: LBP em cascata, Haar-*like* em cascata e HOG+SVM. A fim de testar a aplicabilidade dessas abordagens em problemas de sensoriamento remoto, foram feitos três estudos de caso: pista de pouso em imagens de média resolução, pista de pouso em imagens de alta resolução e campos de futebol em imagens de alta resolução.

Como já mencionado, as abordagens têm sido utilizadas amplamente no reconhecimento de feições humanas, apesar de terem sido concebidas como abordagens de uso geral. De fato elas são, porém isso não é garantia que elas funcionem satisfatoriamente com qualquer tipo de objeto e sensor. Prova disso são os resultados ruins (taxa de acerto abaixo de 30% no casos em que a precisão ficou maior que 10%) para imagens da banda pancromática de média resolução, resultados razoáveis para o reconhecimento de campos de futebol em imagens de alta resolução e resultados satisfatórios para o reconhecimento de pista de pouso, também, em imagens de alta resolução. A maior disponibilidade de imagens de alta resolução atualmente e os resultados obtidos são motivadores para utilização desse tipo de imagem com as abordagens apresentadas. Apesar de os estudos de caso só terem abordado dois tipos de alvos, as metodologias funcionam com outros tipos de objetos, desde que estes tenham forma bem definida e baixa deformação. Pode-se, também, aplicar essas abordagens em imagens aéreas desde que estas sejam de alta resolução e do espectro visível.

Nesse estudo, foi possível comparar as abordagens entre si. Notou-se uma grande vantagem dos classificadores em cascata em relação ao HOG+SVM. Analisando os resultados pode-se verificar que na média, Haar-*like* e LBP em cascata são mais rápidos e possuem melhor desempenho na classificação do que o HOG+SVM. Alguns motivos para isso são: enquanto os classificadores em cascata procuram por características mais relevantes de um determinado objeto, o SVM utiliza todas as características fornecidas; quanto mais vetores de suporte o SVM tem, mais custoso

fica o processo; o emprego de *kernel* não linear aumenta o custo computacional do SVM. Porém, há uma desvantagem do uso de classificadores em cascata em relação ao SVM: o tempo de treinamento, onde o primeiro é de 4 a 5 vezes mais demorado que o segundo.

Uma das motivações dessa dissertação é o emprego de reconhecimento de objetos em sistemas autônomos ou simplesmente para auxiliar o operador humano. Em sistemas autônomos ou híbridos de *software* embarcado, deseja-se que o tempo de resposta da abordagem seja baixo, pois o resultado influencia diretamente a tomada de decisão. Já em sistemas de *software* não-embarcado, pode-se utilizar abordagens com um custo computacional mais alto pois podem ser utilizados em computador com alto poder de processamento, desde que o emprego de um determinado método se traduza em melhor desempenho. Haar-*like* e LBP em cascata são abordagens que podem ser perfeitamente utilizadas em sistemas embarcados autônomos ou híbridos. Já o emprego de HOG+SVM em sistemas embarcados que necessitem de resposta rápida fica comprometido devido ao tempo médio de processamento ser relativamente alto.

Existem muitas variações na metodologia apresentada neste trabalho que podem ser testadas a fim de melhorar o desempenho no processo de reconhecimento automático de objetos. Levando em consideração o fato de algumas abordagens terem melhor desempenho com determinados tipos de alvos e imagens, pode-se testar outras combinações de descritores locais e classificadores. Pode-se também fazer um estudo profundo em relação ao conjunto de amostra, onde procura-se saber como os ruídos e o ângulo de rotação presente nas amostras, assim como a quantidade de indivíduos do conjunto de treinamento interferem no desempenho do classificador. Outros estudos que podem estender essa dissertação são: a aplicação dessa metodologia, seja com as mesmas ou outras abordagens, em outros tipos de alvos ou imagens; desenvolvimento de reconhecimento automático de objetos para sistemas embarcados autônomos e de tempo real.

REFERÊNCIAS BIBLIOGRÁFICAS

AHONEN, T.; HADID, A.; PIETIKÄINEN, M. Face recognition with local binary patterns. In: PAJDLA, T.; MATAS, J. (Ed.). **Computer vision**. Springer Berlin Heidelberg, 2004, (Lecture Notes in Computer Science, v. 3021). p. 469–481. ISBN 978-3-540-21984-2. Disponível em:

<http://dx.doi.org/10.1007/978-3-540-24670-1_36>. 12

AUSTIN, R. **Unmanned aircraft systems: UAVs design, development and deployment**. [S.l.]: Wiley, 2010. (Aerospace Series). ISBN 9780470664803. 1

BIBERMAN, L. **Perception of displayed information**. Plenum Press, 1973. (Optical physics and engineering). ISBN 9780306307249. Disponível em:

<<http://books.google.com.br/books?id=JiRTAAAAMAAJ>>. 2

BIEDERMAN, I. Recognition-by-components: A theory of human image understanding. **Psychological Review**, v. 94, p. 115–147, 1987. 4

BIEDERMAN, I.; GERHARDSTEIN, P. Viewpoint-dependent mechanisms in visual object recognition: reply to Tarr and Bülthoff. **Journal of Experimental Psychology: Human Perception and Performance**, American Psychological Association, v. 21, n. 6, p. 1506 – 1514, 1995. 4

BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: ANNUAL WORKSHOP ON COMPUTATIONAL LEARNING THEORY. **Proceedings...** New York, NY, USA: ACM, 1992. p. 144–152. ISBN 0-89791-497-X. 11, 20

BRECKON, T.; BARNES, S.; EICHNER, M.; WAHREN, K. Autonomous real-time vehicle detection from a medium-level UAV. In: INTERNATIONAL CONFERENCE ON UNMANNED AIR VEHICLE SYSTEMS. **Proceedings...** [S.l.], 2009. 8, 12

CALTECH. **CALTECH computational vision group archive**. abr. 2013. Disponível em: <http://www.vision.caltech.edu/Image_Datasets/>. 40

CHANG, C. C.; LIN, C. J. LIBSVM: A library for support vector machines. **ACM Transactions on Intelligent Systems and Technology**, ACM, New York, NY, USA, v. 2, n. 3, p. 27:1–27:27, maio 2011. ISSN 2157-6904. Disponível em: <<http://doi.acm.org/10.1145/1961189.1961199>>. 24

CHEN, C. **Handbook of pattern recognition and computer vision**. World Scientific Publishing Company, Incorporated, 2009. ISBN 9789814273398.

Disponível em: <<http://books.google.com.br/books?id=TaevS0ibLqwC>>. 26

CHEN, Q.; GEORGANAS, N. D.; PETRIU, E. M. Real-time vision-based hand gesture recognition using haar-like features. In: INSTRUMENTATION AND MEASUREMENT TECHNOLOGY CONFERENCE. **Proceedings...** Warsaw, Poland, 2007. p. 1–6. 12

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, Kluwer Academic Publishers, v. 20, n. 3, p. 273–297, 1995. ISSN 0885-6125. Disponível em: <<http://dx.doi.org/10.1007/BF00994018>>. 22

CORVEE, E.; BREMOND, F. Haar like and LBP based features for face, head and people detection in video sequences. In: INTERNATIONAL WORKSHOP ON BEHAVIOUR ANALYSIS AND VIDEO UNDERSTANDING. **Proceedings...** Sophia Antipolis, France, 2011. p. 10. Disponível em: <<http://hal.inria.fr/inria-00624360>>. 12

COVER, T. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. **IEEE Transactions on Electronic Computers**, EC-14, n. 3, p. 326–334, 1965. ISSN 0367-7508. 23

CRUZ, J. E. C.; SHIGUEMORI, E. H.; GUIMARÃES, L. N. F. An approach to automatic object recognition in satellite and airborne images using a decision tree. In: SYMPOSIUM SELPER. **Proceedings...** Cayenne, 2012. 72

_____. Um estudo da detecção automática de campos de futebol de imagens aéreas e orbitais utilizando SVM e descritores HOG. In: WORKSHOP DE COMPUTAÇÃO APLICADA. **Anais...** São José dos Campos, 2012. 72

_____. Comparação entre HOG+SVM e Haar-like em cascata para a detecção de campos de futebol em imagens aéreas e orbitais. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO. **Anais...** Foz do Iguaçu, 2013. 73

_____. Concrete and asphalt runway detection in high resolution images using lbp cascade classifier. In: BRICS COUNTRIES CONGRESS ON COMPUTATIONAL INTELLIGENCE. **Proceedings...** Recife, 2013. 64

_____. Detecção de pista de pouso de concreto ou asfalto em imagens de média e alta resolução utilizando feições haar-like em cascata. In: SIMPÓSIO DE CIÊNCIA E TECNOLOGIA DO IEAV. **Anais...** São José dos Campos, 2013. 56, 64

_____. Detecção de pista de pouso em imagens orbitais de alta resolução utilizando feições Haar-like em cascata. In: WORKSHOP DE COMPUTAÇÃO APLICADA. **Proceedings...** São José dos Campos, 2013. 64

DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** Washington, DC, USA: IEEE, 2005. (CVPR '05, v. 1), p. 886–893. ISBN 0-7695-2372-2. ISSN 1063-6919. 6, 8, 11, 14, 15, 18, 19, 40

DONOHUE, J. **Introductory review of target discrimination criteria.** Hanscom Air Force Base, Massachusetts: [s.n.], 1991. 2

DRIGGERS, R. G.; COX, P.; KELLEY, M. National imagery interpretation rating system and the probabilities of detection, recognition, and identification. **Optical Engineering**, v. 36, n. 7, p. 1952–1959, 1997. Disponível em: <<http://dx.doi.org/10.1117/1.601381>>. 2

EHLERS, M. Sensoriamento remoto para usuários de SIG - sistemas sensores e métodos: entre as exigências do usuário e a realidade. In: BLASCHE, T.; KUX, H. (Ed.). **Sensoriamento Remoto e SIG Avançados.** 2. ed. São Paulo: Oficina de Textos, 2007. p. 19 – 38. ISBN 978-85-86238-57-4. 7

EHLERS, M.; JANOWSKY, R.; GAEHLER, M. New remote sensing concepts for environmental monitoring. In: EHLERS, M. (Ed.). **Remote Sensing for Environmental Monitoring, GIS Applications and Geology.** Toulouse: SPIE, 2002. v. 4545, p. 1 – 12. ISBN 978-3-540-33832-1. Disponível em: <<http://dx.doi.org/10.1117/12.453657>>. 7

FANG, Y.; WANG, Z. Improving LBP features for gender classification. In: INTERNATIONAL CONFERENCE ON WAVELET ANALYSIS AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2008. v. 1, p. 373–377. 12

FAWCETT, T. An introduction to ROC analysis. In: **Pattern Recognition Letters.** New York, NY, USA: Elsevier Science Inc., 2006. v. 27, n. 8, p. 861–874. 47

FEI-FEI, L.; FERGUS, R.; PERONA, P. One-shot learning of object categories. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 28, n. 4, p. 594–611, 2006. ISSN 0162-8828. 40

FELZENSZWALB, P. F.; GIRSHICK, R. B.; MCALLESTER, D.; RAMANAN, D. Object detection with discriminatively trained part-based models. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, Los Alamitos, CA, USA, v. 32, n. 9, p. 1627–1645, set. 2010. ISSN 0162-8828. 11

FORSYTH, D.; PONCE, J. **Computer vision**. Prentice Hall, 2011. ISBN 9780136085928. Disponível em:
<<http://books.google.com.br/books?id=gM63QQAACAAJ>>. 1, 5, 13, 25, 35, 36, 40

FREUND, Y.; SCHAPIRE, R. Experiments with a new boosting algorithm. In: INTERNATIONAL CONFERENCE OF MACHINE LEARNING. **Proceedings...** San Francisco, 1996. p. 148 –156. 32

FREUND, Y.; SCHAPIRE, R.; ABE, N. A short introduction to boosting. **Journal of the Japanese Society For Artificial Intelligence**, Japanese Society For Artificial Intelligence, v. 14, n. 771-780, p. 1612, 1999. x, 31, 34, 35

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. Additive logistic regression: a statistical view of boosting. **The Annals of Statistics**, Institute of Mathematical Statistics, v. 28, n. 2, p. 337–374, 2000. ISSN 00905364. Disponível em:
<<http://dx.doi.org/10.2307/2674028>>. 35

GASZCZAKA, A.; BRECKON, T.; HANA, J. Real-time people and vehicle detection from UAV imagery. In: SPIE CONFERENCE INTELLIGENT ROBOTS AND COMPUTER VISION. **Proceedings...** [S.l.], 2011. 8, 12

GONZALEZ, R. C.; WOODS, R. E. **Digital image processing**. 3. ed. [S.l.]: Prentice Hall, 2007. Hardcover. ISBN 013168728X. 5

GOOGLE. **Google Maps**. jun. 2013. Disponível em:
<<http://maps.google.com>>. x, 8, 41, 42, 43, 56, 64

GRIFFIN, G.; HOLUB, A.; PERONA, P. **Caltech-256 object category dataset**. Pasadena. California Institute of Technology: [s.n.], 2007. 40

GRITTI, T.; SHAN, C.; JEANNE, V.; BRASPENNING, R. Local features based facial expression recognition with face registration errors. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATIC FACE AND GESTURE RECOGNITION. **Proceedings...** [S.l.], 2008. 11, 12, 15, 16, 20, 29

HAAR, A. Zur theorie der orthogonalen funktionensysteme. **Mathematische Annalen**, Springer-Verlag, v. 69, p. 331–371, 1910. ISSN 0025-5831. Disponível em: <<http://dx.doi.org/10.1007/BF01456326>>. 25

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The elements of statistical learning**. New York: Springer New York, 2001. 22

HUMPHREYS, G.; QUINLAN, P. Normal and pathological processes in visual object constancy. **Visual object processing: a cognitive neuropsychological approach**, Lawrence Erlbaum Associates, p. 43 – 105, 1987. 3

INPE. **Divisão de Geração de Imagens - Arquivos e Documentos**. jun. 2013. Disponível em: <<http://www.dgi.inpe.br>>. x, 8, 41, 42, 48

JIANG, N.; LU, Y.; TANG, S.; GOTO, S. Rapid face detection using a multi-mode cascade and separate haar feature. In: INTERNATIONAL SYMPOSIUM ON INTELLIGENT SIGNAL PROCESSING AND COMMUNICATION SYSTEMS. **Proceedings...** [S.l.], 2010. p. 1 –4. 12, 13

KEARNS, M.; VALIANT, L. Cryptographic limitations on learning boolean formulae and finite automata. **Journal of the ACM**, ACM, New York, NY, USA, v. 41, n. 1, p. 67–95, jan. 1994. ISSN 0004-5411. Disponível em: <<http://doi.acm.org/10.1145/174644.174647>>. 31

KOHAVI, R. et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE. **Proceedings...** [S.l.], 1995. v. 14, p. 1137–1145. 24

LEAR. **LEAR image datasets**. abr. 2013. Disponível em: <<http://lear.inrialpes.fr/data>>. 40

LIENHART, R.; KURANOV, A.; PISAREVSKY, V. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. **Pattern Recognition**, p. 297–304, 2003. ix, 6, 11, 26, 31

- LIU, C.; YUEN, J.; TORRALBA, A. Nonparametric scene parsing via label transfer. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 33, n. 12, p. 2368–2382, 2011. ISSN 0162-8828. 40
- LOGOTHETIS, N.; PAULS, J.; BÜLTHOFF, H.; POGGIO, T. et al. View-dependent object recognition by monkeys. **Current Biology**, London, UK: Current Biology Ltd., c1991-, v. 4, n. 5, p. 401–414, 1994. 4
- LORENA, A. C.; CARVALHO, A. C. P. L. F. **Uma introdução à support vector machines**. São Carlos. ICMC-USP: [s.n.], 2003. 22
- LOWE, D. G. Distinctive image features from scale-invariant keypoints. **International Journal of Computer Vision**, Springer Netherlands, Hingham, MA, USA, v. 60, n. 2, p. 91–110, nov. 2004. ISSN 0920-5691. 20
- MA, C.; TAN, T.; YANG, Q. Cascade boosting LBP feature based classifiers for face recognition. In: INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEM AND KNOWLEDGE ENGINEERING. **Proceedings...** [S.l.], 2008. v. 1, p. 1100 –1104. 6, 8, 11, 12, 15, 31
- MA, Y.; CHEN, X.; CHEN, G. Pedestrian detection and tracking using HOG and oriented-LBP features. In: ALTMAN, E.; SHI, W. (Ed.). **Network and Parallel Computing**. Springer Berlin Heidelberg, 2011, (Lecture Notes in Computer Science, v. 6985). p. 176–184. ISBN 978-3-642-24402-5. Disponível em: <http://dx.doi.org/10.1007/978-3-642-24403-2_15>. 12
- MARR, D.; NISHIHARA, H. K. Representation and recognition of the spatial organization of three-dimensional shapes. **Royal Society of London**, v. 200, n. 1140, p. 269–294, fev. 1978. ISSN 0080-4649. Disponível em: <<http://dx.doi.org/10.1098/rspb.1978.0020>>. 4
- MARSZATEK, M.; SCHMID, C. Accurate object localization with shape masks. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2007. p. 1–8. ISSN 1063-6919. 40
- MIT. **SUN Database Project**. abr. 2013. Disponível em: <<http://groups.csail.mit.edu/vision/SUN/>>. 40
- NOVO, E. M. L. M. **Sensoriamento remoto: princípios e aplicações**. 3. ed. São Paulo: Editora Blucher, 2008. ISBN 978-85-212-0441-1. 7

- OJALA, T.; PIETIKAINEN, M.; HARWOOD, D. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In: INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION. **Proceedings...** [S.l.], 1994. v. 1, p. 582–585 vol.1. 26
- OJALA, T.; PIETIKAINEN, M.; MAENPAA, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 24, n. 7, p. 971–987, jul 2002. ISSN 0162-8828. 27
- OPENCV. **The OpenCV library**. fev. 2014. Disponível em: <<http://opencv.org/>>. 11, 43, 44
- OPENMP. **The OpenMP library**. fev. 2014. Disponível em: <<http://openmp.org/>>. 43, 44
- OREIFEJ, O.; MEHRAN, R.; SHAH, M. Human identity recognition in aerial images. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2010. p. 709–716. ISSN 1063-6919. 11
- PALIY, I. Face detection using haar-like features cascade and convolutional neural network. In: INTERNATIONAL CONFERENCE ON MODERN PROBLEMS OF RADIO ENGINEERING, TELECOMMUNICATIONS AND COMPUTER SCIENCE. **Proceedings...** Lviv-Slavsko, Ukraine, 2008. 12
- PAPAGEORGIOU, C.; OREN, M.; POGGIO, T. A general framework for object detection. In: INTERNATIONAL CONFERENCE ON COMPUTER VISION. **Proceedings...** [S.l.], 1998. 25
- PERONA, P.; FERGUS, R.; FEI-FEI, L. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In: CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2004. p. 178–178. 40
- QT. **Qt library**. fev. 2014. Disponível em: <<http://qt-project.org/>>. 43, 44
- RAPP, B. **The handbook of cognitive neuropsychology: what deficits reveal about the human mind**. Psychology Press, 2001. ISBN 9781841690445. Disponível em: <<http://books.google.ca/books?id=JJKoJWz0F4C>>. 3

RODRIGUES, R.; SHIGUEMORI, H.; FORSTER, C.; PELLEGRINO, S. color and texture features for landmarks recognition on UAV navigation. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO. **Anai...** [S.l.], 2009. 8

ROTH, P.; WINTE, M. **Survey of appearance-based methods for object recognition**. Graz. Technische Universität Graz: [s.n.], 2008. 5

SCHAPIRE, R. E.; FREUND, Y.; BARTLETT, P.; LEE, W. S. Boosting the margin: a new explanation for the effectiveness of voting methods. **The Annals of Statistics**, Institute of Mathematical Statistics, v. 26, n. 5, p. 1651–1686, 1998. ISSN 00905364. Disponível em: <<http://dx.doi.org/10.2307/120016>>. 32

SCHAPIRE, R. E.; SINGER, Y. Improved boosting algorithms using confidence-rated predictions. **Machine Learning**, Kluwer Academic Publishers, Hingham, MA, USA, v. 37, n. 3, p. 297–336, dez. 1999. ISSN 0885-6125. Disponível em: <<http://dx.doi.org/10.1023/A:1007614523901>>. 35

SHAPIRO, L.; STOCKMAN, G. **Computer vision**. Prentice Hall, 2001. ISBN 9780130307965. Disponível em: <<http://books.google.com.br/books?id=FftDAQAAIAAJ>>. 5, 13

SHAWE-TAYLOR, J.; CRISTIANINI, N. **Kernel methods for pattern analysis**. Cambridge University Press, 2004. ISBN 9780521813976. Disponível em: <<http://books.google.com.ar/books?id=9i0vg12lti4C>>. 24

SZELISKI, R. **Computer vision: algorithms and applications**. Londres: Springer, 2010. x, 1, 5, 13, 33

TARR, M. J.; BÜLTHOFF, H. H. Is human object recognition better described by geon structural descriptions or by multiple views? Comment on Biederman and Gerhardstein. **Journal of experimental psychology: Human perception and performance**, Department of Psychology, Yale University, USA., v. 21, n. 6, p. 1494–1505, dez. 1995. ISSN 0096-1523. Disponível em: <<http://view.ncbi.nlm.nih.gov/pubmed/7490590>>. 3, 4

THE ROYAL ACADEMY OF ENGINEERING. **Autonomous systems: social, legal and ethical issues**. London: [s.n.], 2009. Disponível em: <http://www.raeng.org.uk/societygov/engineeringethics/pdf/Autonomous_Systems_Report_09.pdf>. 1

THEODORIDIS, S.; KOUTROUMBAS, K. **Pattern Recognition**. 4. ed. Elsevier Science, 2008. ISBN 9780080949123. Disponível em:

<<http://books.google.com.br/books?id=QgD-3Tcj8DkC>>. ix, 20, 21, 22, 24, 31, 32, 35

TREFNÝ, J.; MATAS, J. Extended set of local binary patterns for rapid object detection. In: COMPUTER VISION WINTER WORKSHOP. **Proceedings...** [S.l.], 2010. v. 10, p. 37–43. 12, 13, 27

USGS. **Landsat missions**. jan. 2014. Disponível em:

<<http://landsat.usgs.gov/>>. 8

VALIANT, L. G. A theory of the learnable. **Communications of the ACM**, ACM, New York, NY, USA, v. 27, n. 11, p. 1134–1142, nov. 1984. ISSN 0001-0782. Disponível em: <<http://doi.acm.org/10.1145/1968.1972>>. 31

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. In: IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** Los Alamitos, CA, USA, 2001. v. 1, p. 511–518. ISBN 0-7695-1272-0. ISSN 1063-6919. 5, 6, 8, 11, 15, 25

_____. Robust real-time face detection. **International Journal of Computer Vision**, 2004. 6, 11, 25, 30, 31, 35

WANG, L.; HE, D. Texture classification using texture spectrum. **Pattern Recognition**, Elsevier, v. 23, n. 8, p. 905–910, 1990. 26

WANG, W.; CHEN, W.; XU, D. Pyramid-based multi-scale LBP features for face recognition. In: INTERNATIONAL CONFERENCE ON MULTIMEDIA AND SIGNAL PROCESSING. **Proceedings...** [S.l.], 2011. v. 1, p. 151–155. 12

WANG, X.; GONG, H.; ZHANG, H.; LI, B.; ZHUANG, Z. Palmprint identification using boosting local binary pattern. In: INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION. **Proceedings...** [S.l.], 2006. v. 3, p. 503–506. ISSN 1051-4651. 12

WARD, J. **The student's guide to cognitive neuroscience**. Psychology Press, 2010. ISBN 9781848720022. Disponível em:

<<http://books.google.com.br/books?id=429FAQAAIAAJ>>. 3

- WEBB, A. **Statistical pattern recognition**. Londres: Wiley, 2002. 5, 13
- WEIJER, J. van de; SCHMID, C.; VERBEEK, J. Learning color names from real-world images. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2007. p. 1–8. ISSN 1063-6919. 40
- WILSON, P. I.; FERNANDEZ, J. Facial feature detection using haar classifiers. **Journal of Computing in Small Colleges**, Consortium for Computing Sciences in Colleges, USA, v. 21, n. 4, p. 127–133, 2006. ISSN 1937-4771. 12
- WITKIN, A. Scale-space filtering: A new approach to multi-scale description. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING. **Proceedings...** [S.l.], 1984. v. 9, p. 150 – 153. 36
- XIA, D.; SUN, H.; SHEN, Z. Real-time infrared pedestrian detection based on multi-block LBP. In: INTERNATIONAL CONFERENCE ON COMPUTER APPLICATION AND SYSTEM MODELING. **Proceedings...** [S.l.], 2010. v. 12, p. 139 – 142. 6, 11, 12
- XIAO, J.; HAYS, J.; EHINGER, K.; OLIVA, A.; TORRALBA, A. Sun database: large-scale scene recognition from abbey to zoo. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2010. p. 3485–3492. ISSN 1063-6919. 40
- XIAO, R.; LI, M. J.; ZHANG, H. J. Robust multipose face detection in images. **IEEE Transactions on Circuits and Systems for Video Technology**, v. 14, n. 1, p. 31 – 41, jan. 2004. ISSN 1051-8215. 5, 11
- ZHANG, C.; MA, Y. **Ensemble machine learning: methods and applications**. Springer, 2012. ISBN 9781441993250. Disponível em: <<http://books.google.com.br/books?id=CjAs4stLXhAC>>. 30
- ZHANG, L.; CHU, R.; XIANG, S.; LIAO, S.; LI, S. Face detection based on multi-block LBP representation. In: LEE, S.-W.; LI, S. (Ed.). **Advances in Biometrics**. Springer Berlin Heidelberg, 2007, (Lecture Notes in Computer Science, v. 4642). p. 11–18. ISBN 978-3-540-74548-8. Disponível em: <http://dx.doi.org/10.1007/978-3-540-74549-5_2>. 5, 12, 29

ZHANG, L.; WU, B.; NEVATIA, R. Pedestrian detection in infrared images based on local shape features. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.], 2007. p. 1 –8. ISSN 1063-6919. 11, 13