Microviscous morphological operators

Fernand Meyer and Jesus Angulo

Centre de Morphologie Mathématique

October 2007

On transitions in levelings

Connected operators: enlarge the existing flat zones and produce new ones. A connected operator transforms an image f into an image g in such a way that $\forall\,(p,q)$ neighbors: $g_p \neq g_q \Rightarrow f_p \neq f_q$

Levelings

Specialising the preceding relation yields levelings :
$\forall\,(p,q)$ neighbors: $g_p > g_q \Rightarrow f_p \geq g_p$ and $g_q \geq f_q$ meaning that any transition in the destination image g is bracketed by a larger variation in the source image.

Such levelings are also characterized by $f \wedge \delta g \leq g \leq f \vee \varepsilon g$

Do levelings level enough ?

Levelings simplify the images but preserve the contours. The following algorithm transforms a marker image g progressively into a leveling of f :

- On $\{g > f\}$, we replace g by $f \vee \varepsilon g$
- On $\{g < f\}$, we replace g by $f \wedge \delta g$

Often, even for strongly simplified marker images, the levelings recontruct amazingly many details : they do not level enough.

A leveling producing larger flat zones would level more !

On transitions in levelings

To $g_p > g_q$ is associated to a transition $f_p > f_q$.

A leveling will level more if only a subset of all transitions in g are linked with transitions in f.

lambda levelings

If only the transitions $g_p > g_q + \lambda$.are associated with a transition in $f : g_p > g_q + \lambda \Rightarrow f_p \geq g_p$ and $g_q \geq f_q$, we get the λ-levelings. This is the first type of viscosity.

Such levelings are also characterized by
$f \wedge [g \vee (\delta g - \lambda)] \leq g \leq f \vee [g \wedge (\varepsilon g + \lambda)]$

ro levelings

Transitions are also less frequent by relaxing the relation $g_p > g_q$ either by lowering the higher term : $\gamma g_p > g_q$ or increasing the lower term $g_p > \varphi g_q$.

We combine the lower leveling $\gamma g_p > g_q \Rightarrow g_q \geq f_q$ and the upper leveling $g_p > \varphi g_q \Rightarrow f_p \geq g_p$ into a ρ-leveling :

$\gamma g_p > g_q$ and $g_p > \varphi g_q \Rightarrow f_p \geq g_p$ and $g_q \geq f_q$

Such levelings are also characterized by $f \wedge \delta\gamma g \leq g \leq f \vee \varepsilon\varphi g$

They give good results, but the computation is relatively heavy.

Bilevelings

Less frequent transitions arise, if one compares the value of g at a pixel with its value at two neighboring pixels :

An image g is a bileveling of the image f iff $\forall\,(p,q,s)$ being the summits of an elementary triangle of the hexagonal grid.
Upper bilevelings : $g_p > g_q$ and $g_p > g_s \Rightarrow f_p \geq g_p$,
Lower bilevelings : $g_p < g_q$ and $g_p < g_s \Rightarrow f_p \leq g_p$.
A leveling being both an upper and a lower leveling
For (p,q,s) and (q,r,t) triangles :
$[(g_p > g_q$ and $g_p > g_s)$ and $(g_t > g_q$ and $g_r > g_q)] \Rightarrow f_p \geq g_p$ and $g_q \geq f_q$

Characterization of upper bilevelings

The criterion for upper bilevelings $[g_p > g_q \text{ and } g_p > g_s \Rightarrow f_p \geq g_p]$ may be interpreted as
$[g_p \leq g_q \text{ or } g_p \leq g_s \text{ or } g_p \leq f_p] \Leftrightarrow [g_p \leq f_p \vee (g_q \vee g_s)]$.

As p and s may be any couple of neighboring pixels of p, we obtain

$$g_p \leq f_p \vee \bigwedge_{(q,s,p)=triangle} (g_q \vee g_s)$$

Combining with $g_p \leq g_p$, we get

$$g_p \leq f_p \vee \left(g_p \wedge \bigwedge_{(q,s,p)=triangle} (g_q \vee g_s) \right)$$

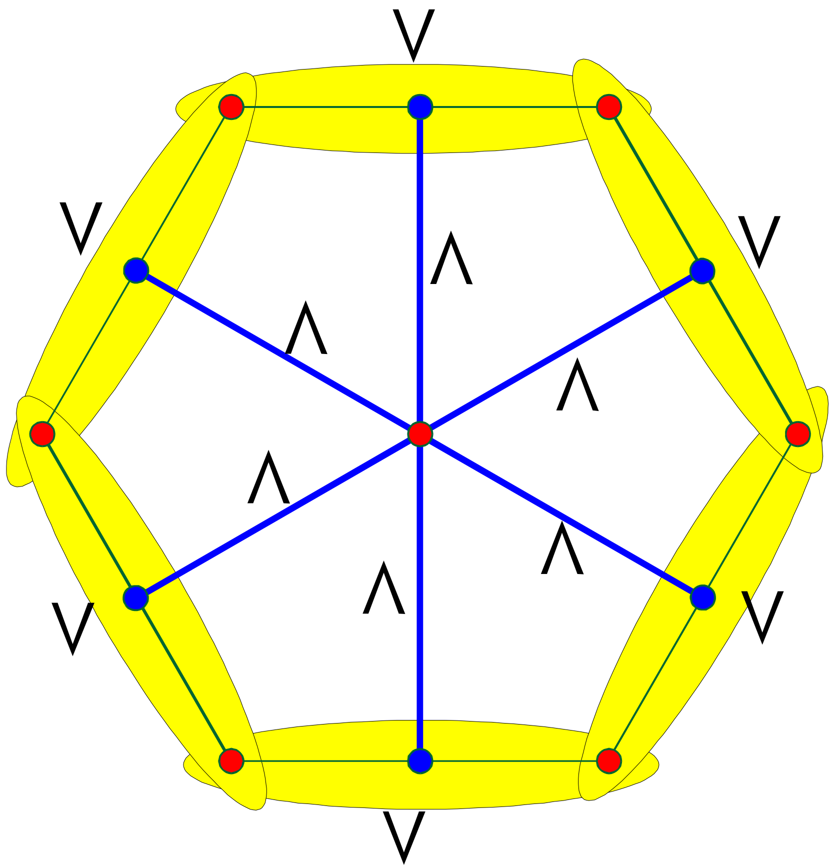


Figure 1: The supremum is taken on each couple of adjacent neighbors, and then the infimum of all these values

Characterization of bilevelings

Bilevelings are boths upper and lower levelings. Both criteria are equivalent :

$$f_p \wedge \left(g_p \vee \bigvee_{(q,s,p)=triangle} (g_q \wedge g_s) \right) \leq g_p \leq$$

$$f_p \vee \left(g_p \wedge \bigwedge_{(q,s,p)=triangle} (g_q \vee g_s) \right)$$

$$f_p \wedge \bigvee_{(q,s,p)=triangle} (g_q \wedge g_s) \leq g_p \leq f_p \vee \bigwedge_{(q,s,p)=triangle} (g_q \vee g_s)$$

Construction of bilevelings

If g is not a bileveling of f, then the relation (6a) does not hold. So we modify g until this relation becomes satisfied:

- On $\{g_p > f_p\}$, we replace g_p by $f_p \vee \left(g_p \wedge \bigwedge_{(q,s,p)=triangle} (g_q \vee g_s) \right)$: this algorithms produces a decreasing series of values bounded by f_p, hence it converges
- On $\{g_p < f_p\}$, we replace g_p by $f_p \wedge \left(g_p \vee \bigvee_{(q,s,p)=triangle} (g_q \wedge g_s) \right)$

Flatzones

Only marked transitions for g correspond to transitions for f : if $(q, s, p) = triangle$, then $g_p > g_q > g_s \Rightarrow f_p \geq g_p$ and $g_s \geq f_s$.

The zones where g departs from f are partly flat : if for the same 3 pixels (q, s, p) forming a triangle we have $f_p > g_p$, $f_s > g_s$ and $f_q > g_q$,then it is not necessarily true that $g_p = g_q = g_s$, but the two lowest values are the same.

Microviscosity reinterpretation of bilevelings

Neighborhood relation of adjacency

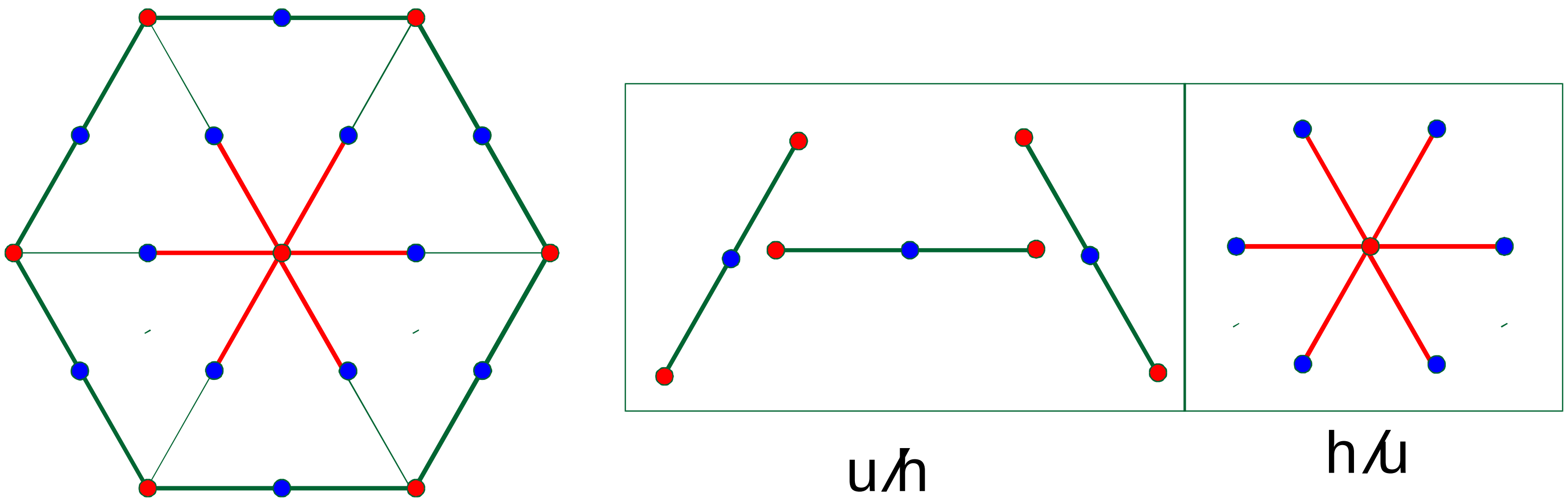
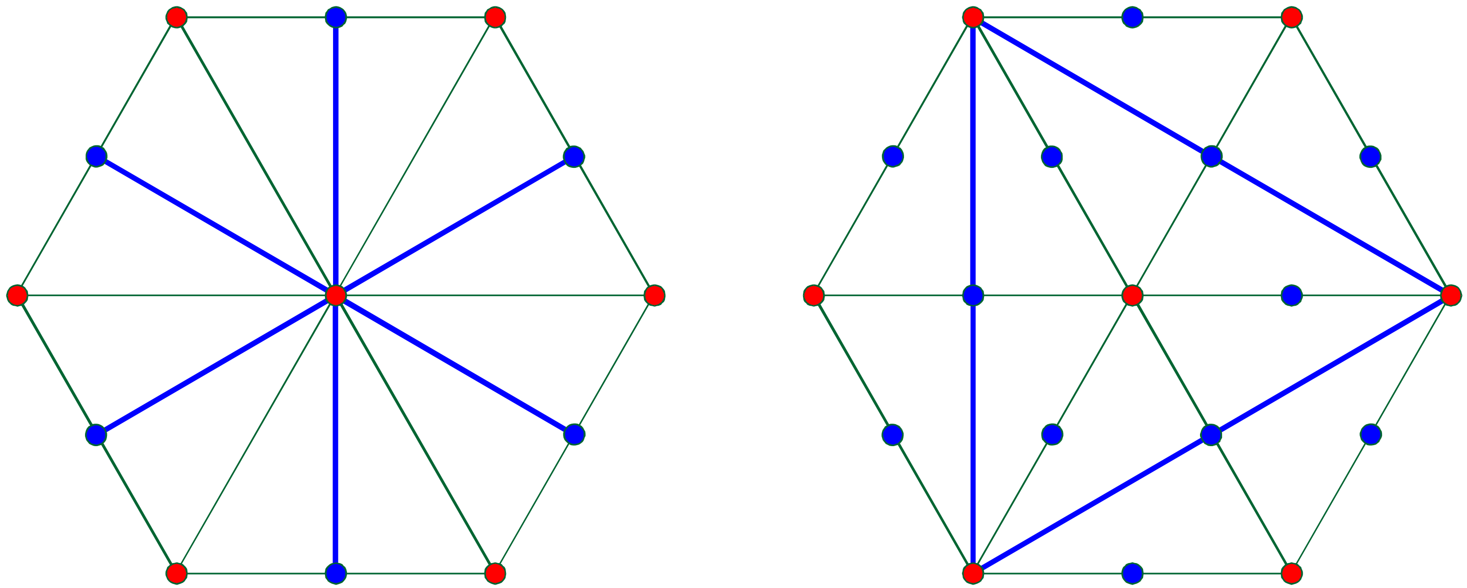


Figure 2: Each pixel ν is extremity of 6 edges; the neighboring edges η of the central pixel appear as small (blue) dots; this neighboring relation is written η/ν, meaning that ν is an extremity of the edge η. Symmetrically, each edge has two extremities; this relation is written ν/η.

Neighborhood relation of opposition



h\u u\h

Figure 3: Each node has 6 opposing edges as neighbors ; this neighborhood relation is written $\eta\backslash\nu$. And each edge η has as neighbors two opposing summits of triangles. This relation is written $\nu\backslash\eta$.

Notations

We consider the elements of the grid itself as operators. The operator ν applied on the function f is the value taken by f on ν : $\nu f = f(\nu)$. Similarly we define $\eta f = f(\eta)$. Let $\overline{\nu}$ be the set of nodes or pixels of the initial grid and $\overline{\eta}$ the set of edges.

Supremum, infimum, complementation of these operators are classically defined as (we illustrate the case for $\overline{\eta}$, the definition for $\overline{\nu}$ being similar):

* $[\eta_1 \vee \eta_2](f) = \eta_1(f) \vee \eta_2(f) = f(\eta_1) \vee f(\eta_2)$

* $[\eta_1 \wedge \vee \eta_2](f) = \eta_1(f) \wedge \eta_2(f) = f(\eta_1) \wedge f(\eta_2)$

* $-\eta_1(f) = \eta_1(-f)$

Adjunction between vertices and adjacent edges

The erosion $\varepsilon_{\eta/\nu} : \mathrm{Fun}(\overline{\nu},\mathcal{T}) \to \mathrm{Fun}(\overline{\eta},\mathcal{T})$ applied to function f is defined by its value at the edge η_i :

$$\eta_i \varepsilon_{\eta/\nu} f = \bigwedge_{\eta_i/\nu_j} f(\nu_j) = \bigwedge_{\eta_i/\nu_j} \nu_j f.$$

Its dual operator, $\delta_{\eta/\nu} : \mathrm{Fun}(\overline{\nu},\mathcal{T}) \to \mathrm{Fun}(\overline{\eta},\mathcal{T})$ is the dilation:

$$\eta_i \delta_{\eta/\nu} = \bigvee_{\eta_i/\nu_j} \nu_j.$$

Its adjunct operator maps $\mathrm{Fun}(\overline{\eta},\mathcal{T})$ into $\mathrm{Fun}(\overline{\nu},\mathcal{T})$ and uses the symmetrical neighborhood relation ν/η:

$$\nu_j \varepsilon_{\nu/\eta} g = \bigvee_{\nu_j/\eta_i} g(\eta_i) = \bigvee_{\nu_j/\eta_i} \eta_i g.$$

Adjunction between vertices and adjacent edges

In the following table each row represents 2 dual operators and each column two adjunct operators:

$\mathrm{Fun}(\overline{\nu},\mathcal{T}) \to \mathrm{Fun}(\overline{\eta},\mathcal{T})$	$\eta_i \varepsilon_{\eta/\nu} = \bigwedge_{\eta_i/\nu_j} \nu_j$	$\eta_i \delta_{\eta/\nu} = \bigvee_{\eta_i/\nu_j} \nu_j$
$\mathrm{Fun}(\overline{\eta},\mathcal{T}) \to \mathrm{Fun}(\overline{\nu},\mathcal{T})$	$\nu_j \delta_{\nu/\eta} = \bigvee_{\nu_j/\eta_i} \eta_i$	$\nu_j \varepsilon_{\nu/\eta} = \bigwedge_{\nu_j/\eta_i} \eta_i$

Introducing the center of the structuring element

By taking into account not only the adjacent edges for computing the value at a node but also the node itself, we define the following erosions and dilations :

$\overline{\eta} \cup \overline{\nu} \rightarrow \overline{\nu}$	$\nu_j \, \overbrace{\delta_{\nu/\eta}} = \nu_j \vee \nu_j \delta_{\nu/\eta}$	$\nu_j \, \overbrace{\varepsilon_{\nu/\eta}} = \nu_j \wedge \nu_j \varepsilon_{\nu/\eta}$
$\overline{\nu} \rightarrow \overline{\eta} \cup \overline{\nu}$	$\nu_j \, \overbrace{\varepsilon_{\eta/\nu}} = \nu_j$ $\eta_i \, \overbrace{\varepsilon_{\eta/\nu}} = \eta_i \varepsilon_{\eta/\nu}$	$\nu_j \, \overbrace{\delta_{\eta/\nu}} = \nu_j$ $\eta_i \, \overbrace{\delta_{\eta/\nu}} = \eta_i \delta_{\eta/\nu}$

Adjunction between vertices and opposing edges

In a similar way, adjunctions may be defined between nodes and opposing edges.

In the following table each row represents 2 dual operators and each column two adjunct operators:

$\mathrm{Fun}(\overline{\nu},\mathcal{T}) \to \mathrm{Fun}(\overline{\eta},\mathcal{T})$	$\eta_i \varepsilon_{\eta\backslash\nu} = \bigwedge_{\eta_i \backslash \nu_j} \nu_j$	$\eta_i \delta_{\eta\backslash\nu} = \bigvee_{\eta_i \backslash \nu_j} \nu_j$
$\mathrm{Fun}(\overline{\eta},\mathcal{T}) \to \mathrm{Fun}(\overline{\nu},\mathcal{T})$	$\nu_j \delta_{\nu\backslash\eta} = \bigvee_{\nu_j \backslash \eta_i} \eta_i$	$\nu_j \varepsilon_{\nu\backslash\eta} = \bigwedge_{\nu_j \backslash \eta_i} \eta_i$

Introducing the center of the structuring element

By taking into account not only the adjacent edges for computing the value at a node but also the node itself, we define the following erosions and dilations :

$\overline{\eta} \cup \overline{\nu} \to \overline{\nu}$	$\nu_j \, \overbrace{\delta_{\nu\backslash\eta}} = \nu_j \vee \nu_j \delta_{\nu\backslash\eta}$	$\nu_j \, \overbrace{\varepsilon_{\nu\backslash\eta}} = \nu_j \wedge \nu_j \varepsilon_{\nu\backslash\eta}$
$\overline{\nu} \to \overline{\eta} \cup \overline{\nu}$	$\nu_j \, \overbrace{\varepsilon_{\eta\backslash\nu}} = \nu_j$ $\eta_i \, \overbrace{\varepsilon_{\eta\backslash\nu}} = \eta_i \varepsilon_{\eta\backslash\nu}$	$\nu_j \, \overbrace{\delta_{\eta\backslash\nu}} = \nu_j$ $\eta_i \, \overbrace{\delta_{\eta\backslash\nu}} = \eta_i \delta_{\eta\backslash\nu}$

Reinterpretation of the bilevelings

The four operators used to characterize and to build the bilevelings can be now reinterpreted in terms of adjunctions between the nodes and the edges of the hexagonal grid:

- $\bigwedge\limits_{(q,s,p)=triangle} (g_q \vee g_s) = \bigwedge\limits_{\nu_p \backslash \eta_i} \eta_i \delta_{\eta/\nu} g = \nu_p \varepsilon_{\nu\backslash\eta} \delta_{\eta/\nu} g$
- $\bigvee\limits_{(q,s,p)=triangle} (g_q \wedge g_s) = \bigvee\limits_{\nu_p \backslash \eta_i} \eta_i \varepsilon_{\eta/\nu} g = \nu_p \delta_{\nu\backslash\eta} \varepsilon_{\eta/\nu} g$
- $g_p \wedge \bigwedge\limits_{(q,s,p)=triangle} (g_q \vee g_s) = g_p \wedge \bigwedge\limits_{\nu_p \backslash \eta_i} \eta_i \delta_{\eta/\nu} g =$
  $\nu_p g \wedge \nu_p \varepsilon_{\nu\backslash\eta} \delta_{\eta/\nu} g = \nu_p \overbrace{\varepsilon_{\nu\backslash\eta}} \delta_{\eta/\nu} g$ (anti-extensive)
- $g_p \vee \bigvee\limits_{(q,s,p)=triangle} (g_q \wedge g_s) = g_p \vee \bigvee\limits_{\nu_p \backslash \eta_i} \eta_i \varepsilon_{\eta/\nu} g =$
  $\nu_p g \vee \nu_p \delta_{\nu\backslash\eta} \varepsilon_{\eta/\nu} g = \nu_p \overbrace{\delta_{\nu\backslash\eta}} \varepsilon_{\eta/\nu} g$ (extensive)

Micro-viscous morphology

Pseudo-inverse operators

To each operator defined above, one may associate its pseudo-inverse operator, obtained by concatenating in reverse order the adjunct operators:

- $\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu} \rightarrow \varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}$
- $\delta_{\nu\backslash\eta}\varepsilon_{\eta/\nu} \rightarrow \delta_{\nu/\eta}\varepsilon_{\eta\backslash\nu}$
- $\overbrace{\varepsilon_{\nu\backslash\eta}}\delta_{\eta/\nu} \rightarrow \varepsilon_{\nu/\eta}\overbrace{\delta_{\eta\backslash\nu}} = \varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}$
- $\overbrace{\delta_{\nu\backslash\eta}}\varepsilon_{\eta/\nu} \rightarrow \delta_{\nu/\eta}\overbrace{\varepsilon_{\eta\backslash\nu}} = \delta_{\nu/\eta}\varepsilon_{\eta\backslash\nu}$

micro-viscous filtering

Concatenating such an operator with its pseudo-inverse produces for instance $\varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu}$: its construction introduces the opening $\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}$ within the closing $\varepsilon_{\nu/\eta}\delta_{\eta/\nu}$.

This operator is increasing, being the product of increasing operators, but it is not a filter as it is not idempotent. However it is an underfilter:

$\varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu}\varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu} \leq \varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu} = \varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu}$ since $\delta_{\eta/\nu}\varepsilon_{\nu/\eta}$ is antiextensive and $\delta_{\eta\backslash\nu}\varepsilon_{\nu\backslash\eta}$ is idempotent.

Similarly $\overbrace{\varepsilon_{\nu\backslash\eta}}\delta_{\eta/\nu}\varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}$ is an underfilter whereas $\delta_{\nu\backslash\eta}\varepsilon_{\eta/\nu}\delta_{\nu/\eta}\varepsilon_{\eta\backslash\nu}$ and $\overbrace{\varepsilon_{\nu\backslash\eta}}\delta_{\eta/\nu}\varepsilon_{\nu/\eta}\delta_{\eta\backslash\nu}$ are overfilters.

micro-viscous filtering 2

For instance, a pseudo-opening can be defined as $\delta_{\nu\backslash\eta}\varepsilon_{\eta/\nu}\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu}$, the corresponding pseudo-closing is obtained as $\varepsilon_{\nu\backslash\eta}\delta_{\eta/\nu}\delta_{\nu\backslash\eta}\varepsilon_{\eta/\nu}$, and mutatis mutandis other pseudo-openings and closings are obtained with the other unitary micro-operations. Then, the product of pseudo-openings and closings leads to more evolved operators such as the pseudo-alternate sequential filters (pseudo-ASF).