

LOW-LEVEL IMAGE PROCESSING BY MAX-MIN FILTERS

P.W. VERBEEK, H.A. VROOMAN and L.J. van VLIET

*Pattern Recognition Group, Faculty of Applied Physics
Delft University of Technology
Lorentzweg 1, 2628 CJ Delft, The Netherlands.*

Abstract. A systematic framework is given that accommodates existing max-min filter methods and suggests new ones. Putting the upper and lower envelopes $UPP = \text{MIN}(\text{MAX})$ and $LOW = \text{MAX}(\text{MIN})$ in the roles that MAX, MIN or original play in existing filters we can distinguish edges in ramp edges and texture (or noise) edges; all methods presented come in three versions: for edges, ramp edges and non-ramp ("texture") edges. The ramp versions of Philips dynamic thresholding and Lee edge detection are considerably less noise sensitive. For images with little noise the texture version of dynamic thresholding brings out fine textures while ignoring ramps. Lee edge-detection can in all versions be extended to a sharp "Laplacian" and an edge enhancer. Starting out from square-full several shapes of the maximum filter are tried out. The round-full filter gives least artefacts; when crescent updating is used it takes size-linear rather than size-quadratic time. The suboptimal round-sparse filter takes size-independent time.

Keywords. Non-linear filtering, edge detection, image enhancement, texture, noise.

1. Introduction

Maximum and minimum filters attribute to each pixel in an image a new value equal to the maximum or minimum value in a neighborhood around that pixel. The neighborhood stands for the shape of the filter. Maximum and minimum filters have been used in contrast enhancement and normalization (Dorst [1]), texture description (Werman & Peleg [2]), edge detection (Lee et al. [3], Van Vliet et al. [4]), and thresholding (Bernsen of Philips Inc.[5]). The filters are grey value analogues of dilation and erosion.

In this study we extensively use upper and lower envelope filters $UPP = \text{MIN}(\text{MAX})$ and $LOW = \text{MAX}(\text{MIN})$, the grey value analogues of closing and opening. Starting from a maximum and minimum filter of one shape (support) other filters are built. Then the same is done for a new shape.

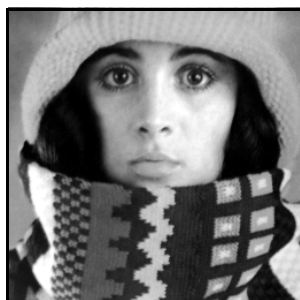
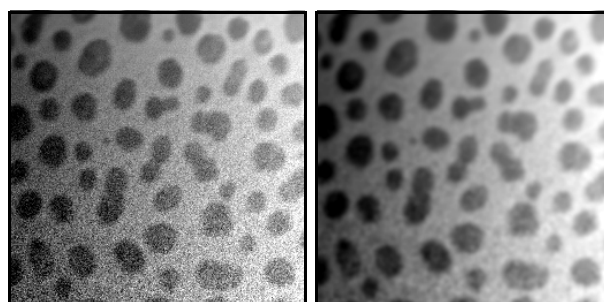


Figure 1. Portrait.



a) Particles; b) presmoothed.

Table 1. Basic operations.

identity operation	ORI
local maximum (grey dilation)	MAX
local minimum (grey erosion)	MIN
upper envelope (grey closing)	$UPP = \text{MIN}(\text{MAX})$
lower envelope (grey opening)	$LOW = \text{MAX}(\text{MIN})$

In the remainder of this article an operation and its result when applied to an original image (figs. 1 and 2) will have the same name. Consequently the identity operation is called "ORI". Where the difference between maximum and minimum filter is irrelevant, the term "max/min filter" will be used.

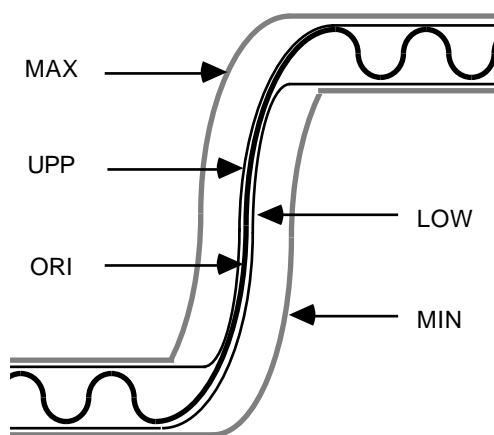


Figure 3. MAX, UPP, ORI, LOW and MIN (see text).

We thus have the basic operations of Table 1 In section 2 we show how these operations can be used to distinguish smooth ("ramp") edges from ripple ("texture") edges. Accordingly, operations relevant to ramp edges have names starting with "RA" , while operations relevant to texture have names starting with "TE". Operations that do not distinguish ramps from texture have names starting with "DY" (from "dynamic"). Names of operations will consist of three characters; the third character characterizes the type of operation. For a compact notation we combine the basic operations into primitives as follows:

$$\begin{aligned} \text{DY+} &= \text{MAX} - \text{ORI} \\ \text{TE+} &= \text{UPP} - \text{ORI} \\ \text{RA+} &= \text{MAX} - \text{UPP} \end{aligned}$$

and

$$\begin{aligned} \text{DY-} &= \text{ORI} - \text{MIN}, \\ \text{TE-} &= \text{ORI} - \text{LOW} \\ \text{RA-} &= \text{LOW} - \text{MIN} \end{aligned}$$

In section 3 we present some extensions of the Lee edge detector. In section 4 the effect of various max/min filters is reported. Results and conclusions are given in section 5.

2. Ramp edges and non-ramp edges

Edge detection by thresholding consists of three steps: subtracting a constant or position dependent value, keeping the sign of the difference as binary value and taking the contour. We concentrate on the first step, with position dependent thresholds. In [5] the dynamic threshold $\text{DYT} = (\text{MAX} + \text{MIN}) / 2$ is proposed. In [1] UPP and LOW are used for contrast stretching.

Our essential observation is that UPP and LOW, and certainly their average, the texture threshold $\text{TET} = (\text{UPP} + \text{LOW}) / 2$ follow a smooth edge, called ramp (cf. fig.3). Hence ramps can be distinguished from

non-ramps such as texture or noise. We have exploited this principle in three ways.

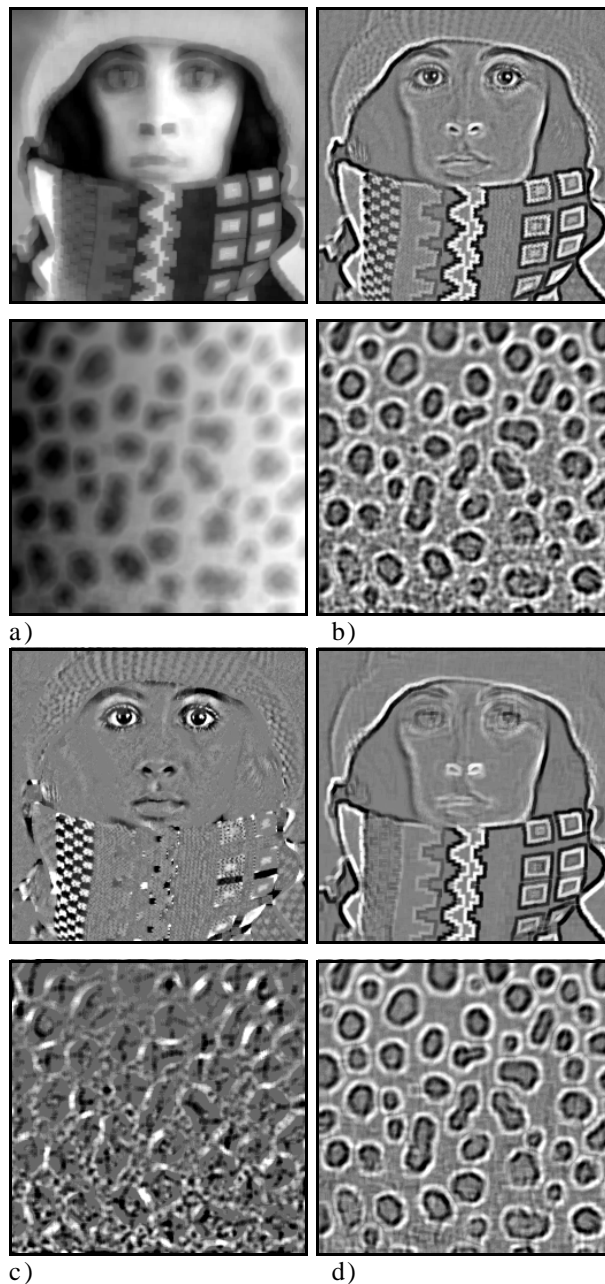


Figure 4. Several levels of detail extracted from the images given in fig. 1 and fig. 2b. (a) Low detail; (b) medium detail (ramps) and high detail (texture or noise); (c) texture (portrait) or noise (particles); (d) ramps.

2.1 Ramp thresholding and texture thresholding

According to [5] subtracting the dynamic threshold DYT from the original (fig. 4a) yields the dynamic gist DYG (fig. 4b). We propose to replace DYT by texture threshold TET to isolate the non-ramp edges

table 2. Low, medium and high detail contributions.

$$\text{ORI} = \text{DYT} + \text{RAG} + \text{TEG} = \text{DYT} + \text{DYG} = \text{RAT} + \text{RAG} = \text{TET} + \text{TEG}$$

$$\text{detail: } \quad l \quad m \quad h \quad l \quad m+h \quad l+h \quad m \quad l+m \quad h$$

such as texture or noise ("texture gist", TEG, fig.4c) instead of all edges (DYG, fig. 4b).

Consequently the difference $\text{RAG} = \text{DYG} - \text{TEG}$ ("ramp gist") must represent the ramps, excluding texture or, more important, noise (fig.4d). We can define a ramp threshold $\text{RAT} = \text{ORI} + \text{DYT} - \text{TET}$ that yields the ramp gist $\text{RAG} = \text{ORI} - \text{RAT}$.

We consider ORI to be split up into the low, medium and high detail contributions DYT, RAG and TEG, non-linear equivalents of frequency bands (Table 2, figs. 4a, 4d, 4c). The size of the max/min filter determines the scaling. The different thresholds and results of thresholding are listed in table 3.

2.2 Texture range

We observe that at a ramp the envelopes UPP and LOW closely follow ORI and differ little there, but that at a non-ramp edge UPP and LOW follow MAX and MIN and differ more (cf. fig.3). Therefore we define the texture range $\text{TER} = \text{UPP} - \text{LOW}$ in analogy to the dynamic range $\text{DYR} = \text{MAX} - \text{MIN}$. We propose to divide any known edge strength by TER to suppress non-ramp edges.

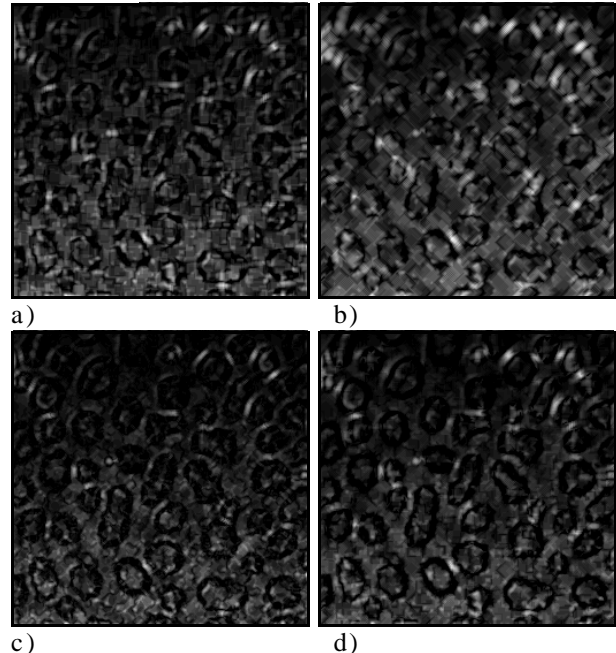


Figure 5. Texture range of the bottom left quadrant of the image given in fig. 2b for different filter shapes. (a) Square; (b) diamond; (c) octagon; (d) minimum of (a) and (b).

Table 3. Ramp an texture versions of dynamic thresholding.

primitives	$\text{DY+} = \text{MAX} - \text{ORI} >0$	$\text{DY-} = \text{ORI} - \text{MIN} >0$
	$\text{TE+} = \text{UPP} - \text{ORI} >0$	$\text{TE-} = \text{ORI} - \text{LOW} >0$
	$\text{RA+} = \text{MAX} - \text{UPP} >0$	$\text{RA-} = \text{LOW} - \text{MIN} >0$
DYT dynamic thresh (Philips PAPS [5])	$\text{DYT} = \frac{1}{2} (\text{MAX} + \text{MIN}) >0$	
TET texture thresh (follows ramps)	$\text{TET} = \frac{1}{2} (\text{UPP} + \text{LOW}) >0$	
DYG dynamic gist	$\text{DYG} = \frac{1}{2} ((\text{DY-}) - (\text{DY+})) = \text{ORI} - \frac{1}{2} (\text{MAX} + \text{MIN}) = \text{ORI} - \text{DYT}$	
TEG texture gist (ramp ignore)	$\text{TEG} = \frac{1}{2} ((\text{TE-}) - (\text{TE+})) = \text{ORI} - \frac{1}{2} (\text{UPP} + \text{LOW}) = \text{ORI} - \text{TET}$	
RAG ramp gist (texture ignore)	$\text{RAG} = \frac{1}{2} ((\text{RA-}) - (\text{RA+})) = \text{DYG} - \text{TEG} = \text{TET} - \text{DYT} = \text{ORI} - \text{RAT}$	
RAT ramp thresh (= by definition)	$\text{RAT} = \text{ORI} + \text{DYT} - \text{TET} >0$	
DYR dynamic range	$\text{DYR} = (\text{DY+}) + (\text{DY-}) = \text{MAX} - \text{MIN} >0$	
TER texture range	$\text{TER} = (\text{TE+}) + (\text{TE-}) = \text{UPP} - \text{LOW} >0$	
RAR ramp range	$\text{RAR} = (\text{RA+}) + (\text{RA-}) = \text{DYR} - \text{TER} >0$	

Using square- (fig.5a) or diamond- (fig.5b) (and even octagon-, fig.5c) shaped max/min filters, one generates small artefacts. Taking the point minimum $\text{pmin}(\text{TER_squ}, \text{TER_dia})$ one avoids them (fig.5d). While TER yields the non-ramp contribution to the dynamic range DYR, the ramp range $\text{RAR} = \text{DYR} - \text{TER}$ represents the ramp contribution.

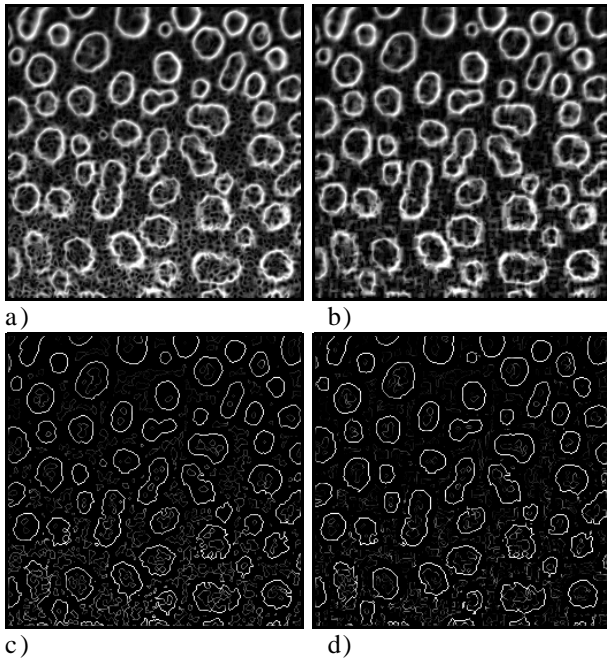


Figure 6. Noise suppression in edge detection. (a) Lee; (b) ramp version of (a); (c) $\text{zerocross's}(\text{DYG}) \text{ AND } \text{DYL}$; (d) ramp version of (c).

2.3 Ramp edge and texture edge detection

As the envelopes UPP and LOW closely follow ORI at a ramp but follow MAX and MIN at a non-ramp edge (cf. fig.3) we propose to replace ORI in the Lee edge-detector by UPP and LOW. The Lee edge detector is defined as a presmoothing (which we adopted for all operations, cf. fig. 2 and section 4.1) followed by the point minimum: $\text{DYL} = \text{pmin}(\text{MAX-ORI}, \text{ORI-MIN})$ ("Lee edge detector", fig. 6a).

Instead we write $\text{RAL} = \text{pmin}(\text{MAX-UPP}, \text{LOW-MIN})$ ("ramp Lee"), in order to suppress non-ramp edges (fig. 6b). The non-ramp edges are selected by:

$\text{TEL} = \text{pmin}(\text{UPP-ORI}, \text{ORI-LOW})$ ("texture Lee"). According to [4] the zero crossing positions of DYG (which are identical to the maximum positions of DYL) are used to mask out the maxima of DYL, the grey contours yielding the contour operator $\text{DYC} = \text{zerocross's}(\text{DYG}) \text{ AND } \text{DYL}$.

In zero areas of DYG the "crossing" is where + and - domain are equally far [4]. Thresholding DYC (fig.6c) eliminates the weak edges to give the binary edge DYE [4] (fig.11a). The ramp versions RAC and RAE are similarly derived from RAL and RAG, again RAC (fig. 6d) and RAE (fig. 11f) give better noise suppression than DYC (fig. 6c) and DYE (fig. 11a). The ramp and texture versions of Lee edge detection are listed in Table 4.

3. More variants on the Lee edge-detector

Expressed in the primitives the Lee edge detector is $\text{DYL} = \text{pmin}(\text{DY+}, \text{DY-})$ (fig.6a); in section 2.3 we defined ramp Lee $\text{RAL} = \text{pmin}(\text{RA+}, \text{RA-})$ and texture Lee $\text{TEL} = \text{pmin}(\text{TE+}, \text{TE-})$.

Table 4. Ramp and texture versions of the Lee edge detector.

primitives	$\text{DY+} = \text{MAX} - \text{ORI} >0$	$\text{DY-} = \text{ORI} - \text{MIN} >0$
	$\text{TE+} = \text{UPP} - \text{ORI} >0$	$\text{TE-} = \text{ORI} - \text{LOW} >0$
	$\text{RA+} = \text{MAX} - \text{UPP} >0$	$\text{RA-} = \text{LOW} - \text{MIN} >0$
DYL dynamic Lee (Lee et al. [3])	$\text{DYL} = \text{pmin}(\text{DY+}, \text{DY-})$	>0 , pmin = point minimum
TEL texture Lee	$\text{TEL} = \text{pmin}(\text{TE+}, \text{TE-})$	>0 , pmin = point minimum
RAL ramp Lee (texture ignoring edge)	$\text{RAL} = \text{pmin}(\text{RA+}, \text{RA-})$	>0 , pmin = point minimum
DYC dynamic contour (Van Vliet [4])	$\text{DYC} = \text{zerocross's}(\text{DYG}) \text{ AND } \text{DYL}$	>0
TEC texture contour	$\text{TEC} = \text{zerocross's}(\text{TEG}) \text{ AND } \text{TEL}$	>0
RAC ramp contour	$\text{RAC} = \text{zerocross's}(\text{RAG}) \text{ AND } \text{RAL}$	>0
DYE dynamic edge	$\text{DYE} = \text{thresholded}(\text{DYC})$	
RAE ramp edge	$\text{RAE} = \text{thresholded}(\text{RAC})$	

Table 5. More variants on the Lee edge detector.

primitives	$DY+ = MAX - ORI >0$	$DY- = ORI - MIN >0$
	$TE+ = UPP - ORI >0$	$TE- = ORI - LOW >0$
	$RA+ = MAX - UPP >0$	$RA- = LOW - MIN >0$
DYL dynamic Lee (Lee edge)	$DYL = pmin(DY+,DY-) >0$, pmin = point minimum	
TEL texture Lee	$TEL = pmin(TE+,TE-) >0$, pmin = point minimum	
RAL ramp Lee (texture ignoring edge)	$RAL = pmin(RA+,RA-) >0$, pmin = point minimum	
DYS dynamic signed Lee	$DYS = smin(DY+,DY-) >0$ if $DY+ < DY-$, smin = signed minimum	
TES texture signed Lee	$TES = smin(TE+,TE-) >0$ if $TE+ < TE-$, smin = signed minimum	
RAS ramp signed Lee	$RAS = smin(RA+,RA-) >0$ if $RA+ < RA-$, smin = signed minimum	
DYF dynamic front	$DYF = ORI + DYS >0$	
TEF texture front	$TEF = ORI + TES >0$	
RAF ramp front	$RAF = ORI + RAS >0$	

The Lee edge detector yields a result similar to the modulus of the result of the linear Laplace operator. Which non-linear, Laplace-like operator would - followed by the modulus operation - generate the Lee edge? We propose the signed minimum $DYS = smin(DY+,DY-)$ which is seen to have a sharp Laplacian-like edge response (figs.7, 8a, 9a). Again the texture and ramp analogues, TES and RAS, (fig. 9c) can be constructed.

Adding high pass to the original gives high emphasis; adding the DYS, TES and RAS results gives steep steps (fronts DYF, TEF, RAF) at edge positions (cf. figs. 7, 8b, 8d, 9b, 9d).

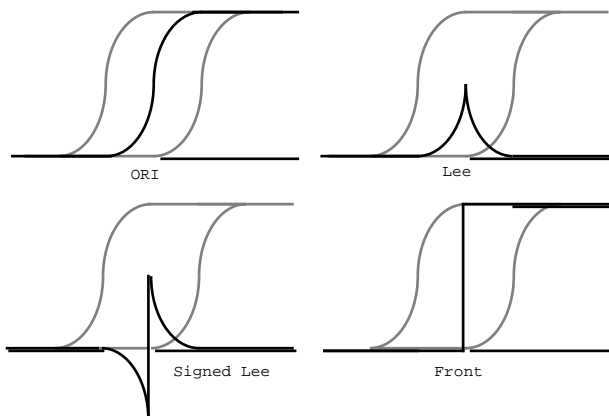


Figure 7. Lee, Signed Lee and Front type edge operators (cf. Table 5).

The result looks similar to that of Kuwahara filtering ([6], fig. 8c). Again the ramp based filter, RAF (fig. 9d) is less noisy than the general one, DYF (fig. 9b). The texture version TEF is only interesting in high SNR images (fig. 8d). The variants on the Lee edge detector are listed in Table 5.

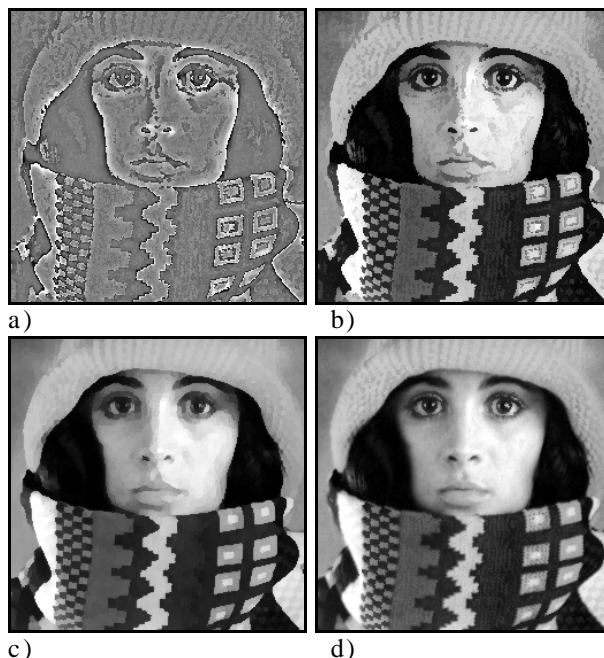


Figure 8. (a) Sharp Laplacian; (b) step shaper; (c) Kuwahara; (d) texture shaper.

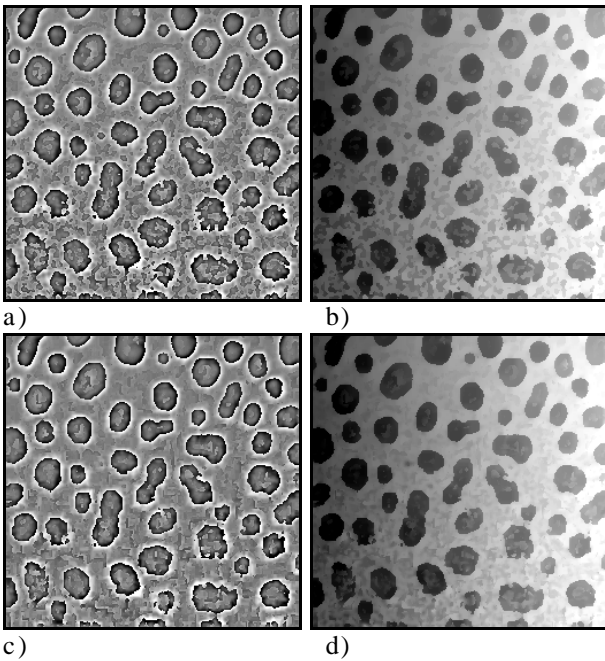


Figure 9. (a) Sharp Laplacian; (b) step shaper; (c), (d) less noise in ramp versions of (a), (b).

4. Different filter-shapes

The filters discussed above were tested for a number of shapes of the underlying max/min filter. To study the impact of isotropy, square, diamond and approximately round (ocatagonal in the 7*7 case, see section 4.2) shapes were used. Also, faster shape approximations like hollow (empty), sampled (net) or a cordon of isolated border points (sparse) were tried (see Table 6).

In particular the influence of filter choice on edge detection was studied on the basis of [4] and [7]. The results of ramp edge detection RAE are shown for all filter shapes (figs.11b-g) together with ground truth obtained from the noise-free image (fig. 11h).

4.1 The images

Two 8-bit images of size 256*256 were used. One is

a standard portrait (fig.1) with little noise and sufficient natural blur (defocusing) to make Lee's presmoothing unnecessary. For this image only the results based on the most simple (square-full) max/min filter are shown.

The other image is constructed from two images:

1. a high SNR electron microscope image of gold particles embedded in glass for which the object contours are precisely known. The mean greyvalue of the objects was 60, that of the background 200.
2. a combined vertical noise wedge and horizontal shading wedge with $\mu + \sigma$ at the four corners given by:

$$\begin{matrix} 448 \pm 0 & 889 \pm 0 \\ 448 \pm 140 & 889 \pm 140. \end{matrix}$$

The unsmoothed original (fig.2a) is given by (microscope image + wedges)/8. The SNR at the top is 1/0 and that at the bottom is 1/1 (shading not included in the signal). This unsmoothed original was presmoothed [3] by threefold linear 1-2-1 filtering in both directions. The result (fig.2b) was used as a (smoothed) original in all filtering operations. The binary contours were compared to those (fig. 11h) known from the noise-free image according to the objective figure-of-merit described by Pratt [7].

In a series of images with different uniform SNR values we found the Pratt figure-of-merit (PFOM) to be roughly linear in s and so the PFOM of the wedge image, being an average of the PFOM for different SNR's, has been taken as a measure for filter performance.

4.2 Shape and computational complexity

All in all, six shapes of the max/min filter were tested (cf. Table 6): a full square, a full diamond, a sampled ("netted") diamond, a discrete approximation of a full circle, ("round", here octagonal) the rim and centre of the same circle ("empty") and eight contour points and the centre of the circle ("sparse").The square-full max/min filter is separable, i.e. it can be obtained by a line filter in

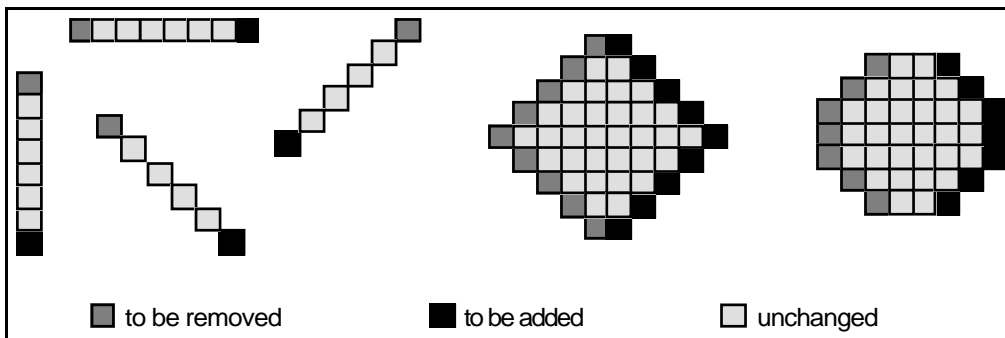


Figure 10. 1-D and 2-D crescent updating according to [8].

the x-direction followed by a line filter in the y-direction. Similarly the netted diamond max/min filter can be found by a line filter in the $(x = y)$ -direction followed by a line filter in the $(x = -y)$ -direction.

Filtering can be implemented in almost size-independent time by updating [8]. The filter window is a line interval. Windows at subsequent positions share most of their pixels. Thus, in going from one position to the next, it suffices to remove one pixel (dark grey in fig. 10) and to add one pixel (black in fig. 10) to update the window. The filter time is thus made independent of window size.

The full diamond and full circle require a proper 2-D window. Again updating can be used. Now the pixels to be removed constitute a crescent (dark grey in fig. 10) and those to be added another crescent (black in fig. 10). The number of pixels in the crescents is proportional to the window filter size d and so is the pixel fetching time. For the empty circle, subsequent windows have little overlap and updating is hardly worth while. All pixels in the window must be fetched at each position, their number is proportional to d . Therefore the empty circle is only 30% faster than the full circle.

The sparse circle rim (eight contour points and centre) has a constant number of 9 pixels to be

fetches, independent of size.

The time indices indicated in Table 6 are measured values, normalized to 10 for the square-full filter. For the filters used the window diameter n is given.

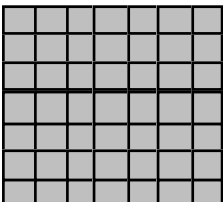
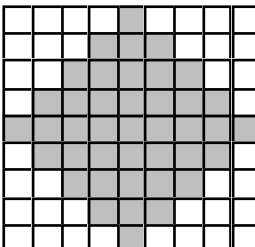
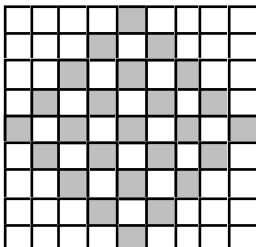
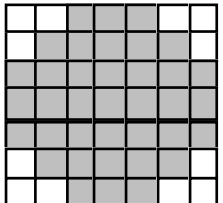
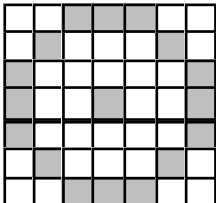
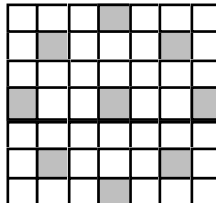
4.3 Evaluation of filter results

All operations have been performed on the electron micrograph for all filter shapes but on the portrait only for the square-full shape. Space prevents us showing all $24 \times (6+1) = 168$ results. In general we have chosen to display the round-sparse results of the electron micrograph. Object contours are most sensitive to filter shape; the RAE results, thresholded contours, most clearly show the artifacts (figs. 11b-g) when compared to the contours of the noiseless original(fig.11h).

Square and diamond filters give contours flattened along the side directions. Round filters, even when approximated by octagons, do better, as expected. Netted diamonds do as well as full diamonds. Round empty is as good as round full. Round sparse is almost as good. In general the round-sparse max/min filter is found to be a fast suboptimal choice especially for images that have been presmoothed according to [3] and [4].

Smoothing reduces noise but at the same time distorts contours. Hence, smoothing must be a

Table 6. Shapes and times of the max/min filters; the time index unit is one tenth of the square-full filter time; d is the filter size expressed in pixel width.

<p>Square full time index = 10 separable $d = 7$</p> 	<p>Diamond full time index = $6 + 5d$ crescent updating $d = 7$</p> 	<p>Diamond net time index = 10 separable $d = 7$</p> 
<p>Round full time index = $6 + 3d$ crescent updating $d = 7$</p> 	<p>Round empty time index = $6 + 2d$ $d = 7$</p> 	<p>Round sparse time index = 13 $d = 7$</p> 

compromise, the outcome of which is reflected in the Pratt figure of merit [4].

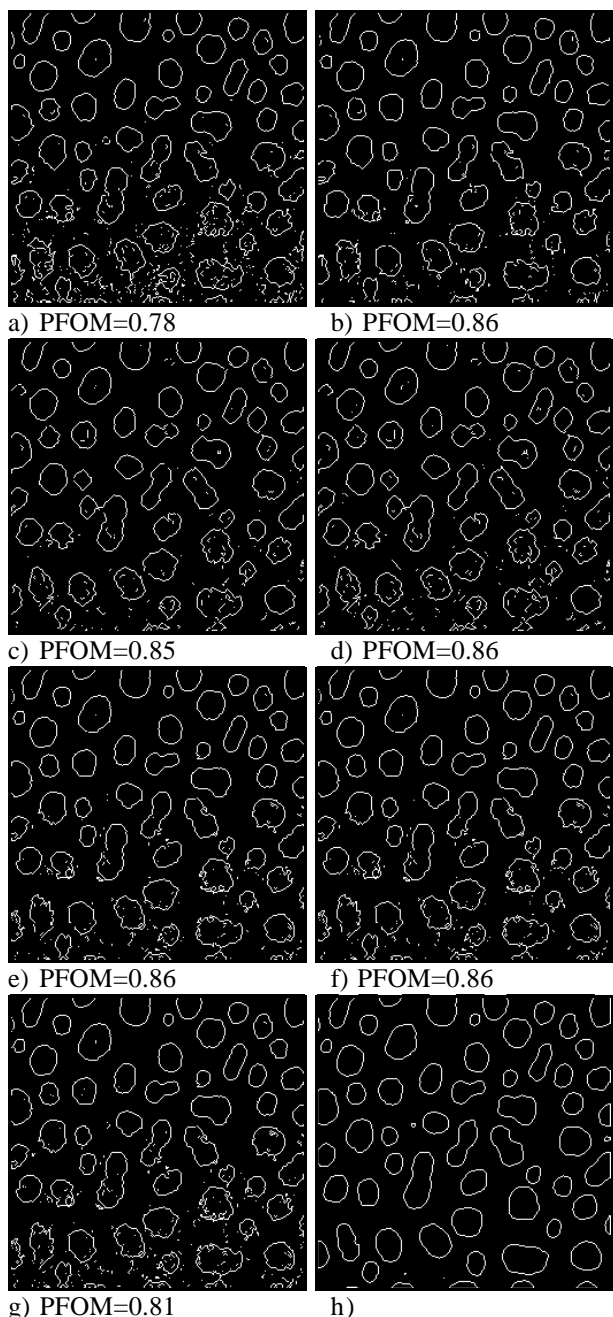


Fig. 11. Impact of filtershape on edge detection. (a) Lee-based, round empty; (b)-(g) ramp versions; (b) square full; (c) diamond full; (d) diamond net; (e) round full; (f) round empty; (g) round sparse; (h) reference contours.

The Pratt figure of merit is a weighted average of distances between a found contour position and the nearest true contour position [7]. No bonus is given for finding closed contours and small erroneous

fragments far from the true contour are heavily penalized. Consequently the round-sparse max/min filter scores less high than the round-empty shape. In practice this may be less serious. Gaps and fragments can be cured by a binary noise filter.

5. Conclusions

5.1. General

- a systematic framework has been given, that accommodates existing max-min filter methods and suggests new ones.
- the envelopes $UPP = \min(\max)$ and $LOW = \max(\min)$ play roles similar to those of MAX and MIN, or ORI, but with different results.
- edges can be distinguished in ramp edges and texture (or noise) edges; all methods presented come in three versions: for edges, ramp edges and non-ramp ("texture") edges.

5.2. New filters

- Philips dynamic thresholding [5] and Lee edge detection [3] can be made considerably less noise sensitive using our ramp equivalents.
- for images with little noise our texture thresholding brings out fine textures while ignoring ramps.
- Lee edge-detection can in all versions be extended to a sharp "Laplacian" and an edge enhancer.

5.3. Variations on the square-full max/min filter

- round max/min filters give fewer artifacts.
- for texture range the artifacts can be removed by taking the point minimum of the square filter and diamond-filter results.
- square-full and diamond-net max/min filters are separable and, like round-sparse filters, take size-independent time.
- diamond-full and round-full max/min filters, using crescent updating, take size-linear rather than size-quadratic time.
- round-empty, especially on pre-smoothed results, gives largely equivalent results to round-full at 30% time saving.
- the round-sparse filter is a good choice in most cases.

Acknowledgement

One of the authors (H.A.V.) contributed to this research as part of the project Quantitative Analysis of Interferograms from Static and Dynamic Experiments Aided by Digital Image Processing no. DTN95.0329 of the Dutch Foundation for Fundamental Research of Matter (FOM).

References

- [1] Dorst, L., Quantitative Analysis of Interferograms Using Image Processing Techniques, ICO-13 Conf. Digest, Sapporo, Japan 1984, pp. 476-477.
- [2] Werman, M. and S. Peleg, Min-Max Filters in Texture Analysis, IEEE PAMI-7, 1986, pp. 730-733.
- [3] Lee, J.S.J., R.M. Haralick and L.G. Shapiro, Morphologic Edge Detection, Proc. 8th ICPR, Paris 1986, pp. 369-373.
- [4] Van Vliet, L.J., I.T. Young and A.L.D. Beckers, A Nonlinear Laplace Operator as Edge Detector in Noisy Images, submitted to CVGIP, 1987.
- [5] Bernsen, J., Dynamic Thresholding of Grey-Level Images, Proc. 8th ICPR, Paris 1986, pp. 1251-1255.
- [6] Kuwahara, M., Hachimura, K., Eiho, S., and Kinoshita, M. (1976), In "Digital Processing of Biomedical Images", (K. Preston and M. Onoe, eds.), pp 187-203, Plenum Press, New York.
- [7] Pratt, W.K., Digital Image Processing, Wiley, New York, 1977.
- [8] Groen, F.C.A., P.P. Jonker and R.P.W. Duin, Hardware versus Software Implementations of Fast Image Processing Algorithms, NATO Advanced Research Workshop on "Real-time Object and Environment Measurement and Classification", Maratea, Italy, August 31-September 3, 1987.