ONE-DIMENSIONAL SCALE-SPACE PRESERVING FILTERS.

Richard Harvey, Alison Bosson and J. Andrew Bangham
School of Information Systems, University of East Anglia
Norwich, NR4 7TJ, UK
Tel: +44 1603 593257; fax: +44 1603 593345
E-mail: {rwh,bosson,ab}@sys.uea.ac.uk

ABSTRACT
We show how graph-morphology processors may be specialized to one dimension and how in this case they amount to parsers of extrema. We compare these scale-space processors on the basis of their performance in additive and replacement noise. Of the filters studied we find that M- and N-filters behave similarly to the recursive median filter and hence inherit their robustness whereas multiscale openings and closings produce much less stable representations.

1 INTRODUCTION

Scale-spaces are an emerging idea of increasing popularity in image processing (see [1] for a collection of relevant papers). The idea is to progressively simplify an image in a way that does not introduce any more detail – the scale-space causality property. Since details are characterised by local maxima or minima this is usually enunciated as the "no new extrema principle." In one dimension the unique solution to this problem, if one insists on a linear filter, is to filter with a Gaussian kernel. These Gaussian filters may be extended into two-dimensions but, as has been noted by Pizer [2], they do not obey this principle. Proponents of linear filtering avoid this difficulty by redefining causality to be the "non-enhancement of existing regional extrema."

Scale-space preserving morphological filters may also be constructed. Examples include multiscale openings, closings [3, 4], and combined open/close or close/open operations [5] over a graph. Such filters preserve causality, in its original sense, over an n-dimensional image. Unfortunately openings and closings are known to be sensitive to noise. This paper applies these new graph algorithms to the one dimensional case and examines their robustness to noise and other distortions via a series of experiments.

2 SCALE-SPACE ALGORITHMS

We adopt the notation of an n-dimensional image as a graph [5, 6], \( G = (V,E) \), in which the vertices, \( V \), are pixel labels and the edges, \( E \), define the adjacencies. In this notation \( C_s(G) \) is the set of all connected subsets of \( s \) elements and \( C_s(G,x) = \{ \xi \in C_s(G) | x \in \xi \} \) is the set of all \( C_s(G) \)'s that contain pixel \( x \). For one-dimensional signals, \( C_s(G) \), becomes \( C_s \) the set of all intervals of length \( s \)

\[ C_s = \{ [x, x+s-1] | x \in \mathbb{Z} \}. \]  (1)

This definition allows a compact definition for an opening and closing as

\[ \gamma_s f(x) = \min_{\xi \in C_s,G^0 \supseteq \xi} \max_{u \in \xi} f(u) \]  \hspace{1cm} (2)

\[ \psi_s f(x) = \max_{\xi \in C_s,G^0 \supseteq \xi} \min_{u \in \xi} f(u) \]  \hspace{1cm} (3)

From (2) and (3) compound operators

\[ M = \psi_s \gamma_s \text{ and } N = \gamma_s \psi_s \]  \hspace{1cm} (4)

may be defined and are known as \( M \)- and \( N \)-filters. Openings correspond to identifying local maxima and reducing their amplitude to that of the largest adjacent pixel. Closings operate similarly on local minima. Thus the \( M \)- and \( N \)-filters differ only in the order in which they process maxima and minima. The application of these filters in cascade forms a sieve [7] in which the output at scale \( s \) is given by

\[ M_s f(x) = M_s M_{s-1} f(x) \]  \hspace{1cm} (5)

where \( M_s \) is the Gaussian filter, [8], in which the input signal is filtered with a discretised version of the Gaussian kernel

\[ f_s(x) = \sum_{n=-\infty}^{\infty} T_s(n) f(x+n) \]  \hspace{1cm} (6)

and \( T_s(n) \) is the modified Bessel function of the first kind. Examples of the simplification produced by the Gaussian filter and the \( m \)-sieve are shown in Figure 1 in which an example signal consisting of two pulses and an impulse is progressively simplified. Note how the Gaussian system filters
by blurring the input signal. As scale increases it becomes increasingly difficult to distinguish between the three main pulses. The m-sieve scale-space is very different, leaving sharp discontinuities where pulses are completely removed at the appropriate scale.

To provide scale-selection one plots, for the Gaussian filter, scale-space curvature \[ d = (s f_{xx})^2 \] or, for the sieve, the granules \[ d_s = |P_s f(x) - P_{s+1} f(x)| \] where \( P \) is one of \( m, M, N, O \) or \( C \). The scale selection surfaces are shown in Figure 2 in which there are local maxima at the positions corresponding to features in the original signal.

Such simplifications in themselves are interesting, but they do not form an easy basis for comparison. Which of the plots in Figures 1 or 2 is more useful? Since a primary purpose of these systems is to localise objects in images and objects in images are characterised by sharp edges we examine the performance of these systems for localising sharp-edged pulses in noise.

3 ROBUSTNESS TO NOISE

Two types of synthetic noise were generated. Additive zero-mean Gaussian noise with a standard deviation of 16, and replacement impulsive noise in which samples were replaced with a probability of 0.2 with a value uniformly distributed in the range (0-255). These were used to corrupt a 100 sample signal containing a pulse of width 20 samples. Figure 3 shows two typical test signals. The amplitude of the pulse is 32, on a background intensity of 112.

For an uncorrupted, noiseless signal, filtering to increasing scales reveals the actual scale of the pulse. This appears as a single maximum in the scale-selection surface of both fil-
The two systems perform similarly in the presence of Gaussian noise; the diffusion processor is slightly more accurate in position estimation and the sieve more accurate in scale. For impulsive noise the systems behave very differently. The sieve is hardly affected. The noise is localised to one scale and is therefore easily removed. The scale-selection surface is almost completely noise free, since the search range does not include small-scale features. The slight variation in the sieve’s estimates occur when the noise falls on or next to the edge of the pulse, thereby increasing or decreasing the pulse length. The performance is summarised in Table 1. The opening sieve, \((O-)\), which generates granularities, performs very badly here since it is biased towards positive-going pulses, the results of the closing sieve are not included since it is bias towards negative-going pulses and the positive-going target is never located. The Gaussian filter also has poor performance – the requirement that the filter preserve scale-space causality has meant that the filter is matched to only Gaussian signals. The results of the normalised cross-correlation system are shown for comparison with the other processors since cross-correlation is known to be optimal for position estimation in Gaussian noise.

As expected, the recursive median \((m-)\), \(M-\) and \(N-\) systems have the best performance in impulsive noise since in this case the structure of the linear scale-space is destroyed and the target is lost.

Of course Table 1 shows results at only one noise level. To investigate this as a parameter, the experiments were repeated for different noise levels. The standard deviation of the Gaussian noise was varied \((2 \leq \sigma \leq 26)\), and the impulsive noise density (probability that a pixel is corrupted) in the range \((0 – 0.1)\). The results of these tests are shown in Figure 5; the Gaussian noise test results are in the top panel and the impulsive noise tests in the bottom panel. In Gaussian noise the sieve proves to be better at scale estimation than the Gaussian filter for all but the highest noise levels. However in position estimation the Gaussian filter consistently performs better than the sieve. This is because in Gaussian noise there are always corruptions at the end of the pulse that affect the sieve’s localisation. In many image processing applications Gaussian noise is not significant and the corruptions are dominated by occlusion for which our impulsive noise model is a better match – here the sieve performs better than the Gaussian processor in both position and scale estimates.

### 4 DISCUSSION

In one dimension \(m-, M\) and \(N\) type systems perform well, and better than Gaussian filters, openings or closings. These results augment and strengthen those obtained in two-dimensions [10] where it was found that sieves and Gaussian filters had comparable performance in Gaussian noise and improved performance in impulsive noise. This robustness has led to the development of real computer vision systems based on these one-dimensional systems [11]. Furthermore a scale-space analysis by computing granularity is quick since the removal of extrema simplifies the signal as the algorithm proceeds – the algorithm is of order complexity \(n\) where \(n\) is the number of pixels.

### References


Figure 5: Top panel: Effect of varying $\sigma$ of the Gaussian noise, left are the position estimates and right is the scale. Bottom panel: Effect of varying the noise density of the impulsive noise, left are the position estimates and right is the scale. Vertical axis corresponds to the standard deviation of the estimates. The sieve results are those of the $m$-sieve.

Graphical models and image processing, 57:1–12, 1995.


