Experiments With Scale-Space Vision Systems

Alison Bosson

A thesis submitted for the Degree of
Doctor of Philosophy
in the
School of Information Systems,
University of East Anglia, Norwich

January 21, 2000

Abstract

This thesis studies the implementation and filter performance of four different approaches to scale-space analysis. These approaches are the linear diffusion system, the anisotropic diffusion system and two mathematical morphology based filters; one based on erosions and dilations with fixed-shape paraboloids and the other on openings and closings over flexible connected-sets, also called *sieve*-operators.

Investigations into the performances of these systems for different conditions of input signal are carried out in both one and two dimensions. Parametric estimation problems are then used to study system reliability under a range of different noise conditions, the task of the systems being to estimate the scale and position of simple target features from synthetic images. The effect of image occlusion is also studied using both synthetic and real image occlusion models.

This thesis finds that the anisotropic diffusion based and morphological, erosion-dilation paraboloid-based approaches to scale-space analysis are impractical for the estimation of scale and position of image features. The linear diffusion system can produce these estimates, but, there are many problems associated with obtaining these results. The sieve-based scale-space is found to provide a scale-based image decomposition that is as robust to common levels of Gaussian noise as the linear diffusion filter and that is extremely robust in the presence of impulsive noise and occlusion.

Contents

List of Figures

List of Tables

Publications

These are the publications which have been produced by, or in conjunction with, the author, during her Ph.D candidacy.

1. Alison Bosson and Richard Harvey, Using occlusion models to evaluate scale-space processors. In *Proceedings of the IEEE International Conference on Image Processing*, October 1998.

2. Richard Harvey, Alison Bosson, and J. Andrew Bangham, One dimensional scale-space preserving filters. In *Signal Processing IX: Theories and Applications*, Volume II, pages 821-824, Rhodes, 1998.

3. Richard Harvey, Alison Bosson and J. Andrew Bangham, The robustness of some scale-spaces. In *Proceedings of the British Machine Vision Conference*, Volume 1, pages 11-20, 1997.

4. Richard Harvey, J. Andrew Bangham and Alison Bosson. Scale-space filters and their robustness. In *Scale-Space Theory in Computer Vision, Lecture Notes in Computer Science*, 1252, Springer, pages 341-344, 1997.

5. Alison Bosson, Richard W. Harvey and J. Andrew Bangham, Scale-space pattern processors: are they robust to noise and occlusion? In *Proceedings of the IEE Colloquium on Pattern Recognition*, Ref number 1997/018, pages 8/1-8/6, Savoy Place, London, March 1997.

6. Richard W. Harvey, Alison Bosson and J. Andrew Bangham, A comparison of linear and non-linear scale-space filters in noise. In *Signal Processing VIII: Theories and Applications*, Volume III, pages 1777-1780, Trieste, September 1996.

Acknowledgements

I would like to thank my supervisor Dr. Richard Harvey for his support and encouragement during this period of study. I would also like to thank Prof. Andrew Bangham for his ideas and comments.

I am also grateful to the faculty, support staff and especially to the postgraduate students, past and present, of the Image and Signal Processing Labs in the School of Information Systems.

This work was funded by the the Engineering and Physical Sciences Research Council.

Chapter 1

Introduction

Scale-space systems are an increasingly popular means of representing images and use filters that decompose an image with respect to some *scale* parameter. The structure of the decomposition is such that, as scale increases, the details in the input image are removed or suppressed. A low-level scale-based analysis such as this can be useful to higher-level systems since the large or *coarse*-scale representations retain what Koenderink calls the *deep-structure* of the image, that is the basic underlying image structure, and this is separate from the often confusing small-scale detail.

These scale-based systems are being prototyped for several practical applications, including

- Feature detection [11, 12], for example fingerprint analysis [105], audio-visual speech recognition [80] and stereo vision [74, 84, 88].
- Edge [31, 91] and corner [105] detection.
- Three-dimensional object recognition [57].

These examples indicate a variety of different approaches to scale-space analysis. Early representations were described in terms of Gaussian convolution filtering and linear diffusion [51, 63, 118] however, alternative non-linear approaches have arisen in the form of non-linear diffusion [23, 91] and mathematical morphology based techniques [57, 111]. Indeed, one of the interests of this thesis is in the evaluation of a new scale-space processor called a *sieve* [14, 15]. Since a useful method of analysis in the evaluation of any system involves comparisons with other systems, an integral part of this thesis therefore looks at the implementation of some of the more popular scale-space systems. These different approaches are rarely discussed together or compared under similar conditions. The real images that are used to compare the systems (Chapter 3) are helpful as they allow us to make subjective comparisons on the performances of these filters and foresee possible problems in the analysis of these images. However, if the behaviour of filters is to be fully understood and explored then real images are too complicated. For this reason, synthetic images with fully controllable signal parameters are often used.

The experimental work of this thesis concentrates on using increasingly complicated synthetic images to evaluate the performances of these different scale-space filters to different conditions of signal and noise. We describe the use of simple one- and two-dimensional synthetic signals that are corrupted with different distributions of noise. The deterioration in system performance is then studied for increasing levels of noise. Occlusion experiments are also described since image occlusion is considered here to be a special case of noise corruption. Both synthetic and real occlusion test images are created and are used to further study system performance.

1.1 Structure of this thesis

In Chapter 2 the concept of a scale-based analysis of signals and images is introduced. This includes examples of multi-resolution and multi-scale methods and a description of scale-space theory. Finally, the general axioms or properties that are required from a scale-space system are discussed.

Chapter 3 gives a description of four different scale-space systems showing their implementation and example filtered images. These systems include the linear diffusion filter of [71], the non-linear diffusion filters of [23, 91] and the mathematical morphology-based systems of [60, 111], using fixed-shape poweroid structuring elements and lastly the sieve, [14, 17] which uses a connected-set based area decomposition.

One of the recommended properties of any practical scale-space system, mentioned in Chapter 2, is access to a *scale-selection surface*. This provides a means of labelling features within a signal as being of a particular scale, and if possible, calibrating this scale in some way to the true pixel-dimensions of the feature. The final section of

Chapter 3 looks at the methods used to obtain this scale-selection surface - if one is available.

In Chapter 4 a series of tests are described. These involve using simple one dimensional synthetic pulse-signals that are corrupted with different types of noise. It is the task of each system to return an estimate of the position and size of the pulse for different distributions of noise. Chapter 5 extends this work, presenting the results of some two-dimensional robustness experiments, again using synthetic signals of noise-corrupted discs, the synthetic nature of these images giving full control over the signal parameters. Chapter 5 also presents the results of some increasingly complicated cluttered images containing occluding and occluded discs. These tests include simple synthetic images, more complex anti-aliased and noise corrupted synthetic images and finally real images. Here the systems must return position and size estimates of all image features.

Chapter 6 gives a summary and evaluates the performances of the systems. In particular, the problems of implementation and obtaining results from the linear diffusion scale-space processor are discussed.

1.2 Statement of originality

Unless otherwise noted or referenced in the text, the work described in this thesis is that of the author. One section in particular is highlighted,

- Chapter 3 shows the implementation of four different approaches to scale-space analysis. Although these systems are not original to this thesis, they are rarely considered together and evaluated under similar signal conditions.

Chapter 2

Multi-Scale Systems

This chapter studies the basic concept of *scale* in relation to digitised images and computer vision. Some scale-based image processing algorithms are described that use both multi-scale and multi-resolution techniques. The history of scale-space theory is then introduced and the axioms of a scale-space representation are presented.

2.1 What is scale?

In terms of human vision, scale may be described as the measurement that is used to describe the size of an object. However, this is rarely considered as an absolute length, but as relative size compared to other features. This scale-labelling is somewhat arbitrary since objects may generally be viewed at a series of scales ranging from the atomic level to some external perimeter level and the *correct* scale, if there is such a thing, is very dependent on the distance at which that object is viewed. A useful analogy for this situation is described in [64, 71] in terms of cartography. There are many scales at which one may view a map; ranging from world maps that show the boundaries of countries and continents and that may possibly show the locations of large cities, to smaller-scale country maps that show the detail of towns and roads, to even smaller-scale city-guide maps that may focus on buildings. To mix the scale of detail shown at any scale-representation (for example to place the location of buildings onto a world map) or to have one map showing all detail at the true scale of the earth would be inconsistent with the needs of the map reader, the features on any particular map should be of a scale that is relevant to the scale of the map. It is the task of the cartographer to *suppress* all irrelevant detail whilst retaining an accurate description of the remaining relevant information. This multi-scale nature allows us to view areas at a range of different levels, where large-scale representations show the deep-structure and small-scale representations show the detail.

Presenting the visual information that surrounds us in a scale-based manner not only makes intuitive sense, especially for the cartography example described above, but is also a practical way to reduce the data rate. There is evidence to suggest that the human vision system also follows this scale-based approach. In [29] the results of psychophysical experiments showed that humans see periodic patterns or gratings by means of parallel visual filters that are each tuned to a band of spatial frequency and specific orientation.

There are many examples of literature that discuss the nature of scale and in particular the nature of scale in relation to computer vision [63, 71, 77]. When the notion of scale is transfered to digitised images the problem is simplified greatly by the limits that are immediately imposed. Whereas in nature the range of scales at which objects may be viewed is infinite, in computer vision this range is reduced by the constraints imposed by the imaging process. The minimum scale of any image is the smallest resolution unit which corresponds roughly to the *grains* of a photograph or the *pixels* of a digitised image. Conversely, the largest scale of an image corresponds to the largest size object that may be represented which is the total area of the image; these limits are sometimes referred to as the *inner* and *outer* scales [63].

As with the real-world cartography example, analysing an image as a function of some increasing-size scale parameter will allow the image to be seen at a variety of levels; a scale spectrum. Ideally, small-scale features are visible at the lower levels of this spectrum and conversely, large-scale features will be more apparent at large scales. A general multi-scale image analysis is simply a means of decomposing an image according to some notion of scale.

Why is there a need for this multi-scale analysis? Digitised images of the real world contain large amounts of information that may be unnecessary and confusing to the majority of higher-level processes. If there is any prior knowledge of the size or *scale* of a system's region of interest then it makes sense to suppress all unwanted smaller and larger-scale detail from an image at the low-level process stage, before the higher-level analysis takes place. Even if there is no such prior knowledge, it is still advantageous to have as an input, an image that is viewed at multiple scales, especially if this multi-scale representation provides information on the basic structure of the image and on the *scale-lifetime* (the longevity of objects over scales), sometimes linked to the *importance* of image features.

Analysing an image with respect to a scale parameter is not a new concept. This chapter starts by providing a basic introduction into some of the more popular scale-based representations and describing some practical applications. The final example that is discussed is that of a *scale-space* analysis and it is this approach that is followed throughout this thesis. The scale-space approach is not new [51], but, it has only recently gained popularity as a useful image-description tool for decomposing an image purely by scale.

2.2 Examples of multi-scale analysis

An examination of several multi-scale representations reveals that the image-based notion of scale is vague. Indeed, the main multi-scale methods that are implemented in Chapter 3 all have a different definition of scale. In general, this is defined by the parameters of the filtering operation, which may indicate that scale is related to the size of the filtering mask, the number of filtering iterations or to the dimensions of features within the image. Later we will argue for a definition of scale that is related to the size of image features and not to the analysis method, but first we review a number of multi-scale methods.

2.2.1 Quad-trees

The quad-tree representation was developed in the 1970's [62] and was one of the earliest types of multi-scale structures. Images are divided into increasingly small block regions, decomposing the image into regions of similar intensity. This similarity is measured by some homogeneity criterion determined by the user or the intended application. For a square image, I, of size $2^k \times 2^k$ pixels, the algorithm first treats the image as a single block region (g) and takes some measure of the variation in pixel intensity, such as the region's intensity range, $|g_{max} - g_{min}|$, or the variance of the regions pixels

(2.1).

$$\sigma^2 = \frac{\sum_{i=1}^{N} (g_{(x,y)} - \mu)^2}{N} \tag{2.1}$$

If this measure is greater than some threshold, T, the image is divided into n similar-area sub-regions. For simplicity, n is usually set to 4 and the image is divided into quarters. The intensity variations of all new sub-regions are then calculated and the segmentation continues iteratively until all regions are of relatively uniform intensity, as determined by T.

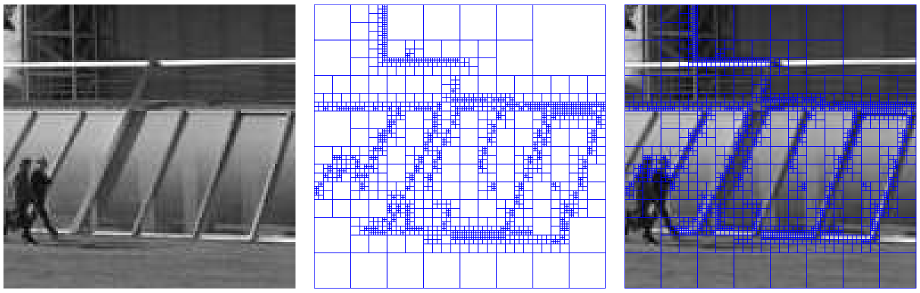


Figure 2.1: Example quad-tree decomposition of an image. Left: original image, middle: quad-tree decomposition, right: quad-tree superimposed onto original image.

Figure 2.1 shows a quad tree decomposition of a simple test image. The original image (left) is 128×128 pixels and contains intensities in the range $[0, 255]$. Throughout this thesis the images will, unless indicated otherwise, be 8-bit greyscale with no headroom so 0 will represent black and 255 white. The number of sub-regions in the quad-tree, n, at each division is 4. The variation over each region is calculated using the maximum intensity difference over the region and the threshold, T, is set to 15.

The large regions of the original image that have low mean contrast, such as the top right hand corner, are segmented into large regions in the quad-tree decomposition. High contrast regions such as the white hand rail stretching across the image, the edges of the glass panels and the couple are segmented into small regions.

The algorithm is efficient and results in large data reductions if the image contains a small number of low contrast regions. For images with multiple high contrast areas, with a small threshold, T, the analysis is not so efficient and in a worst case situation all of the pixels in the image may be identified as sub-regions of the quad-tree.

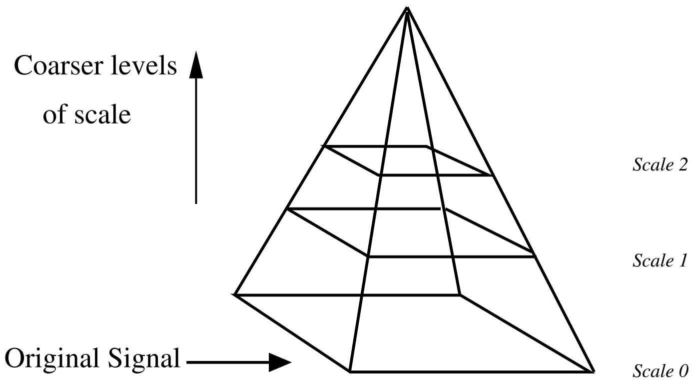


Figure 2.2: Structure of a pyramid representation.

The quad-tree splitting algorithm is often used in conjunction with a *merging* algorithm. The final partition of a quad-tree analysis is likely to contain adjacent regions with identical properties, and so, within the quad-tree algorithm it is useful to merge any adjacent regions that obey the similarity condition.

2.2.2 Pyramids

Pyramid representations are both multi-scale and multi-resolution. The idea of this data-structure is to successively reduce the amount of data, by reducing the size of the image for higher-levels of the pyramid, hence 'multi-resolution', see Figure 2.2. The subsampling method used is crude; simply, every other pixel is removed from the image to create the next level. This approach would be meaningless, due to the anti-aliasing effects of the subsampling if there were no multi-scale representation combined into the pyramid generation. This is achieved through pre-filtering. Each level is filtered with a smoothing signal before subsampling occurs. It is the choice of filtering signal that determines the type of pyramid implementation. Whilst the simplest approach is to average non-overlapping blocks of 2×2 pixels to create corresponding pixels in subsequent levels, the more popular alternatives are *low-pass* pyramids generated by, for example, Gaussian smoothing, *wavelet* pyramids generated from wavelet filtering and *Laplacian* pyramids, otherwise known as bandpass or difference of low-pass pyramids, (DOLP), where each pyramid level is defined as the difference between two adjacent levels in a low-pass pyramid [116].

Figure 2.3 shows an example of a Gaussian pyramid decomposition. The original image lies at the base of the pyramid (level zero). It is 221×331 pixels. The level-one resolution image is created by filtering the original image with a sampled Gaussian with a mean of zero and a standard deviation of two. This smoothed image is then subsampled and the process continues to create the lower resolution images. The results are shown in the first to fourth (top) pyramid layers. Further reductions in image size are impractical for display purposes, however for base images of size $2^k \times 2^k$, the topmost layer of the pyramid reduces to a single pixel in the $k + 1$ level.

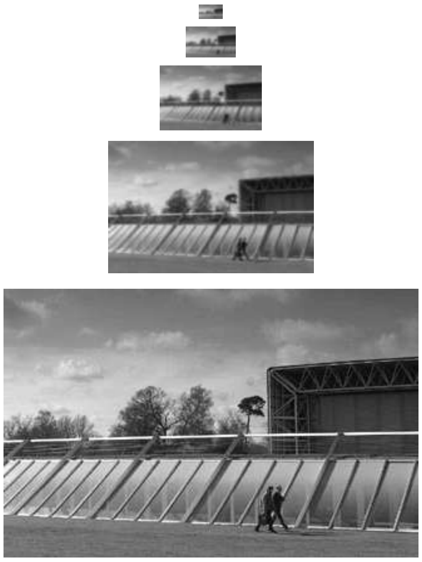


Figure 2.3: Example Gaussian pyramid decomposition of an image. Original image (level 0) is 221×331 pixels, successive layers are created by filtering with a Gaussian kernel ($\mu = 0, \sigma = 2$) followed by subsampling. level 1 image: 111×166 pixels, level 2 image: 56×83 pixels, level 3 image: 28×42 pixels, level 4 image: 14×21 pixels.

The main advantage in creating a pyramid data-structure is in the reduction in spatial sampling that is needed for decreasing resolution representations. The rapidly decreasing image size leads to fast processing and small memory requirements, making the pyramid structure very efficient. The intermediate pre-filtering operations help to reduce any anti-aliasing effects caused by the subsampling. However, there are problems with this method of presentation. There is an implied fixed sampling rate in resolution since the image sampling is fixed to remove every other pixel at every stage of the pyramid.[1] This makes matching and tracking features through the different resolutions difficult. The definition of scale in the pyramidal structure is slightly ambiguous. There are two different types of processing that must occur to form each pyramid layer. If each layer is to be referred to as a particular scale representation then the definition of scale is dependent on both the pre-filtering and the subsampling operations. This ambiguity makes it difficult to find an analytical description of the movement of features through the pyramid. The subsampling destroys the mathematical model of movement caused by the smoothing filter and it is this subsampling that must first be removed to implement a scale-space structure.

2.2.3 Scale-space

A scale-space analysis is similar to the multi-scale analysis of the pyramid representation in that images are filtered to increasing scales. However, the major difference is in the lack of multi-resolution. A scale-space system provides a means of analysing a signal at multiple scales with no change in the signal resolution; the scale parameter is the *only* free parameter of the filtering function. The absence of the pyramids subsampling means that (with careful choice of filtering operator) an analytical model of the behaviour of the signal may be found – this was one of the main reasons behind the development of scale-space [118].

For an unknown signal, there is rarely any *a priori* knowledge of a suitable scale at which to analyse the contents of that signal. In fact, as stated by Witkin [118], for many tasks no one scale description is categorically correct. A scale-space representation provides a purely scale-based analysis of an image at a range of scales. Fine-scale detail is present only in the small-scale levels and, as scale increases this fine detail is removed until, at large scales a minimum of detail remains.

Figure 2.4 shows the structure of a scale-space and how it differs from the pyramid structure. The base or *scale zero* level is simply the original signal. All subsequent levels are produced though filtering with increasing-scale functions, the idea being to separate objects with respect to scale. Coarse scale representations provide useful simplifi-

[1] note: p38 in Lindeberg [71]. The harsh scale-sampling can be avoided to a certain extent through the introduction of "oversampled pyramids", here, not every smoothing operation is followed by subsampling. Several representations are generated by filtering the same resolution image to increasing scales. This creates a denser sampling of the scale parameter.

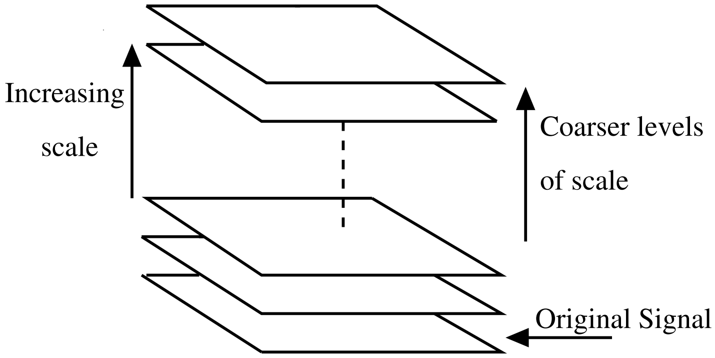


Figure 2.4: Structure of a scale-space representation

cations of the original signal whilst the detail is still retained at small scales. Figure 2.5 shows a simple one-dimensional example scale-space, generated through convolution with an analogue of a Gaussian kernel (described in Section 3.2). The original signal is at the base of the scale-space and as the width of the kernel increases (increasing scale) detail is gradually removed.

Although the notion of the scale parameter is now fully determined by the filtering operation there is still a degree of ambiguity in the definition of scale with respect to the filtered signal. In a general sense, scale can refer to the *size* of objects in the filtered signal, however, different approaches to scale-space filtering produce scale-images that are determined by, for example, the dimensions of the filtering operator, the number of filtering operations or the dimensions of objects within the signal. This makes it difficult to compare the different systems.

The fundamental nature of the scale-space representation is that at *any* scale, all image-features may be linked back to corresponding features at a smaller scale, and finally back to the original, scale-zero image. There has, therefore, been a great deal of discussion over the choice of a suitable filtering operator. The problem has been to choose a filter that allows features at a coarse scale to be traced back to corresponding fine-scale features. A scaling parameter is easily introduced by filtering with any mask of slowly increasing size. The behaviour of the filtered signal may be easily modelled for increasing scale, but if the mask function is not carefully chosen the filtering process will more than likely introduce new features into the scale-space. This creates discontinuities between consecutive levels of scale which, in turn causes problems when tracing features from coarse to fine scales since one can never be certain that points at some in-

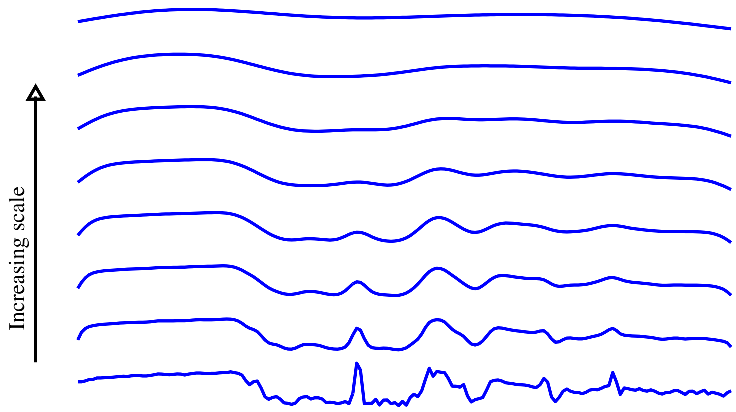


Figure 2.5: Example one-dimensional scale-space representation

termediate scale correspond to lower-scale features or to the weights of the filtering operator. That scale-space filters do *not* introduce artificial features into the scale-space is a fundamental requirement and is described in more detail in Section 2.3.

Early formulations in scale-space analysis [51, 117, 118] were developed along the lines of linear diffusion. In [77], Marr recognises that attributes carrying valuable information in a scene may emerge at any range of scales. To make allowances for this when searching for objects (as intensity changes) in an image, Marr filters with the operator $\nabla^2 G$ where ∇^2 is the Laplacian operator $(\partial^2/\partial x^2 + \partial^2/\partial y^2)$ and G is the two-dimensional Gaussian distribution. This operator provides a scale-based approach to searching for intensity changes in an image. The Gaussian filter, having the desirable properties of smoothness and localisation in both space and frequency, blurs the image for increasing standard deviation kernels, whilst the Laplacian operator is used to locate the intensity changes as zero-crossings in the second derivative (see Figure 2.6).

Marr's Laplacian of Gaussian approach has strong connections to linear scale-space theory. In [77] he notes that one of the useful properties of the Gaussian blurring function is that it is "least likely to introduce any changes that were not present in the original image", however it was in [118] that Witkin coined the phrase *scale-space filtering* as a filtering method that relates one scale-description directly to another. He introduces the idea of a Gaussian-based, one-dimensional scale-space which is formed by blurring the original signal with a family of increasing scale Gaussian functions; the larger the function's scale, the more intense the blurring effect and the greater the reduction in

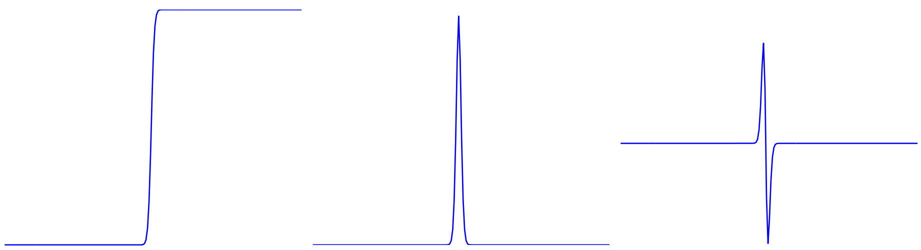


Figure 2.6: Detection of intensity changes in a signal. On the left is the original signal, x, middle panel: the first derivative (d/dx) (peak at highest gradient point of x), right-hand panel: the second derivative of x, (d^2/dx^2), (zero crossing locates the highest gradient point of x.)

image detail.

The filtering operator is a zero-mean Gaussian kernel,

$$g_s(n) = \frac{1}{\sqrt{2\pi s}} \exp^{\left(\frac{-n^2}{2s}\right)} \tag{2.2}$$

with a scale parameter, s, that is equal to the variance of the kernel, $s = \sigma^2$. Using this approach it is straightforward to generate a 'family' of increasing scale kernels that are convolved with a sample signal

$$I_s(x) = \int_{n=-\infty}^{\infty} g_s(n) I(x-n) dn \tag{2.3}$$

to produce different scale representations. The advantage of such a representation lies partly in the easily obtained mathematical model of the behaviour of the filtered signal and also in the fact that the number of extrema (the detail of the signal) reduces for increasing scale.

In [8], Babaud proves that the Gaussian function is the only linear filter with the property that in one dimension, new extrema points or *zero-crossings* of the first derivative are *never* created for increasing scales. This property is very important because it allows for the coarse to fine scale tracking mentioned earlier – a necessary property for any multi-scale blurring system. Once features are located at a coarse scale there is a need to track extrema points back to the fine-scale image in order to localise features. If new extrema are created by the filter then the tracking algorithms become confused.

In [120] a proof that the number of zero-crossings decreases for increasing scales is presented for *two-dimensional* Gaussian filtering (and > 2 dimensions). In particular, Yuille and Poggio favour the Laplacian of Gaussians suggested by [78], mentioned earlier. At a similar time, Koenderink [63], described a two-dimensional scale-space filtering that is based on the assumptions of *homogeneity and isotropy*, treating all spatial positions across the image in a similar manner. He demands that there be no creation of new extrema for increasing scales – a property that he names *causality*, and he states that the only linear solution to the scale-space filter problem, given these constraints, is the solution of the isotropic diffusion equation,

$$\frac{\partial I_s}{\partial s} = c\frac{\partial^2 I_s}{\partial^2 x} \tag{2.4}$$

where I_s is the image as a function of space, x, and scale, s. Koenderink also notes that the Green's function of the diffusion equation is the Gaussian kernel; and so if the initial condition of the diffusion equation is the original image $I(x)$, then the scaled image I_s is given by the convolution of $I(x)$ with the Gaussian kernel of equation (2.2). Conversely, the convolution of $I(x)$ with this kernel satisfies the diffusion equation of (2.4).

In [117] there is a review of some of these different approaches to generating a continuous Gaussian scale-space and the properties that each of the different approaches holds. Table 2.1 (from [117]) gives a summary of the findings. The basic definitions of these general properties follow, but, for more detailed descriptions the reader is referred to [117] and to the individual papers referenced in Table 2.1.

- *Convolution kernel:* The scale-space may be generated via convolution with a family of increasing scale kernels (T_s), where s denotes scale and I is the original image.

$$I_s = \int_{\mathbf{R}^N} T_s(n)I(x-n)dn$$

- *Semigroup property:* Ensures that the scale-space filter may be implemented as a cascaded function.

$$T_{s_1+s_2} = T_{s_2}(T_{s_1})$$

	I1	I2	O	K	Y	B	L1	F1	A	N	L2	F2
Convolution kernel	•	•	•		•	•	•	•	•		•	•
Semigroup property	•	•					•	•	•	•	•	•
Locality									•			
Regularity					•	•	•	•	•		•	•
Causality				•	•	•	•				•	
Nonnegativity	•	•	•						•			•
Tikhonov reg.										•		
Av. greylevel invar.			•			•	•		•			
Flat kern. for $s \to \infty$					•			•				
Isometry invariance		•	•		•	•	•	•	•	•	•	•
Homogen. and isotrop.				•								
Separability			•					•				
Scale invariance	•	•			•	•		•		•		•
Valid for dimension	1	2	2	1,2	1,2	1	1	>1	N	N	N	N

Table 2.1: Overview of continuous Gaussian axiomatics. I1 = Iijima [50, 51], I2 = Iijima [52, 53], O = Otsu [90], K = Koenderink [63], Y = Yuille & Poggio [120], B = Babaud [8], L1 = Lindeberg [68], F1 = Florack [37], A = Alvarez [5], N = Nielsen [87], L2 = Lindeberg [105], F2 = Florack [36]

- *Locality:* Limited action at a distance. For small scales, the value of the filtered image I_s at any point x, is determined by its vicinity:

$$\lim_{s \to 0^+} (T_s * I_1 - T_s * I_2)(x) = o(x)$$

  for all images $I_1, I_2 \in C^\infty$ whose derivatives of order ≥ 0 are identical.

- *Regularity:* A precise definition of the smoothness requirements for a scale-space operator varies from author to author, however, from [37]: Since the original image creates the scale-space, it is natural to assume that it is continuously embedded, i.e., $\lim_{s \to 0^+} I_s = I$. In the linear convolution case, this means that the scale kernel T_s tends to Dirac's delta distribution and its Fourier transform becomes unity everywhere.

- *Causality:* The scale-space evolution should not create any artificial extrema for increasing scale. See Section 2.3.1 for more detail.

- *Non-negativity:* If the non-negativity of the convolution kernel,

$$T_s(x) \geq 0 \quad \forall x, \ \forall s \geq 0,$$

  is violated, new level-crossings may occur for $t > 0$, such that the causality property does not hold.

- *Tikhonov regularization:* In the one-dimensional case, u is called a Tikhonov regularization of the image I, if it minimises the energy functional

$$E_I[u] = \int_{\mathbf{R}^N} \left[(I-u)^2 + \sum_{i=1}^{\infty} \lambda_i \left(\frac{d^i u}{dx^i} \right)^2 \right] dx$$

  This concept (and an N-dimensional generalisation of it) has been used by Nielsen [86]. The first term under the integral ensures that u remains close to the original image, while the second term determines the smoothness of u. The positive coefficients of λ_i are weights that guarantee the smoothness of the i^{th} derivative.

- *Average greylevel invariance:* The average greylevel invariance

$$\int_{\mathbf{R}^N} I_s(x)dx = \int_{\mathbf{R}^N} I(x)dx \quad \forall s > 0$$

  can be achieved by means of the conservation principle; that the blurring transformation does not change the total light energy within the image, in connection with reflecting or periodic boundary conditions.

- *Flat kernel for $s \to \infty$:* For $s \to \infty$, the kernel may be expected to spread the information uniformly over the image. Therefore if the integral over the kernel should remain finite, it follows that the kernel has to become entirely flat: $\lim_{s\to\infty} T_s(x) = 0$.

- *Isometry invariance:* Filtering the mirrored image I_m, with the kernel T_s produces the same result as filtering the original image I with the mirrored kernel T_{m_s} and mirroring the output. In one dimension, for the case of linear convolution, this is only true for kernels that obey the symmetry condition,

$$T_s(x) = T_s(-x) \tag{2.5}$$

- *Homogeneity & isotropy:* All spatial points should be filtered in the same manner and the kernel should be symmetrical, treating all points equally around the point of symmetry. For more detail see Section 2.3.2.

- *Separability:* The $N-$dimensional convolution kernel T_s may be separated into N one-dimensional convolution kernels, each of which act along one coordinate axis.

- *Scale invariance:* Filtering an image that is stretched by a scalar λ, by a kernel T_s should produce the same result as filtering the original image with a kernel T_s that has been stretched by the same scalar λ. To achieve this for N dimensions, the convolution kernel T_s has the structure

$$T_s(x) = \frac{1}{\Psi^N(s)}\Phi\left(\frac{x}{\Psi(s)}\right) \tag{2.6}$$

  All kernels may therefore be derived by stretching a parent kernel such that its area remains constant.

2.3 Essential properties of a scale-space representation

The properties described in the previous table are simply the *general* properties of some selected approaches to scale-space and are not the *required* properties for scale-space analysis. So far, all the approaches have been linear, but this is not demanded. Since the introduction of scale-space theory, alternative *non-linear* solutions have become increasingly popular. The following chapter follows four different approaches ranging from the more classical linear diffusion filter to different mathematical morphology based filters. The one common feature to these systems is that they all observe the *causality* criterion that was introduced by Koenderink. Causality is *the* essential property of scale-space, however there are other attractive axioms that are adopted by different representations. A collection of these properties are described below:

2.3.1 Causality

If the scale-space structure is to show the progressive *simplification* of a signal with respect to a scale parameter, then there can be no artificial structures introduced by the filtering operation, i.e. the process must be *causal*.

In one dimensional discrete space, extrema may be defined as any length of flat-signal that is bounded on *both* sides by, either lower amplitude points (maxima), or higher amplitude points (minima). This is easily translated into two dimensions as a flat-zone area or region within an image that is bounded by pixels that are either, *all* of lower intensity than that of the flat-zone (maxima), or are *all* of higher intensity (minima).

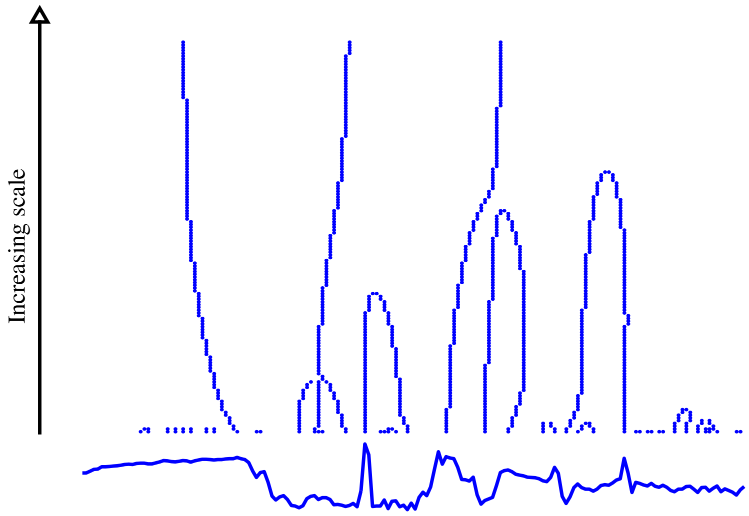


Figure 2.7: Zero-crossings plot for an example one-dimensional signal.

To comply with the causality constraint, a processor must not introduce any new extrema to the filtered signal and any existing extrema must not increase in magnitude. The same constraint, in some publications, is described in terms of the number of zero-crossings in the first derivative of the signal. The paths of these zero-crossings form loops (see Figure 2.7 for an example) that are closed from above but never from below. Zero-crossings at any scale must be traced back to zero-crossings at scale-zero, to obey causality, the number of zero-crossings must not increase as the signal is filtered to coarser scales.

2.3.2 Homogeneity and isotropy

Along with the necessity of causality, Koenderink [63] also chose to apply the constraints of homogeneity and isotropy. These are not necessary, but do simplify the analysis. *Homogeneity* causes the filtering operation to be identical over the entire signal, and *isotropy* causes the filtering function to act identically in each direction (i.e. a symmetrical filtering element). Together, these properties ensure that all spatial points and all scale levels are treated in a similar manner.

If these axioms are adhered to, when combined with the causality restraint, the diffusion equation is referred to as the unique solution to creating a linear scale-space representation [8, 63].

2.3.3 Immediate localisation

If the constraints of homogeneity and isotropy are relaxed, other equally-valid, causal solutions appear and these solutions have their own useful attributes. The anisotropic diffusion based scale-space (to be discussed in Chapter 3) was first introduced in [91]. The algorithm allows the conductivity, c, of the diffusion equation (2.4), to vary as a function of the gradient of the image at any location. The main advantage of this process is that it overcomes the problematic blurring that is inherent with linear diffusion systems.

Immediate localisation requires that "at each resolution, the region boundaries should be sharp and coincide with meaningful boundaries at that resolution" [91]. This axiom strengthens the case of the non-linear alternative to linear diffusion; anisotropic diffusion [23, 91]. In low-contrast regions the image detail blurs to a constant greylevel whilst in high-contrast regions the diffusion is reduced, leaving sharp edges.

The advantages of such a property are clear. An image filtered to a coarse scale contains details that are easily referenced and linked back to objects in the original image. Extrema do not move as a result of filtering and so tracing objects through the scale-space is a much simpler task.

2.3.4 Piecewise smoothing

Advocates for anisotropic diffusion [91, 107] also require the property of piecewise smoothing; that intra-region smoothing should occur preferentially over inter-region smoothing. A popular description of this property involves an image of treetops against a sky background. *Piecewise smoothing* requires that "the leaf regions should be collapsed to a treetop before being merged with the sky background" [91]. In the case of linear diffusion, the leaves blur into the sky background before the treetop emerges as an object.

2.3.5 Scale-selection

We might also add that a system should provide a *scale-selection* surface. If a processor is to produce meaningful results, and its aim is to separate out signal-features according to scale, then it would be useful to generate a scale-selection surface. This surface should provide a means of obtaining scale and position estimates of individual features and should be robust to common image corruptions.

Chapter 3

Some Practical Scale-Spaces

3.1 Introduction

This chapter studies four different types of scale-space processor, some of which have several sub varieties. The four systems are the conventional linear diffusion processor [38, 71] that continues from the work of [8, 51, 63, 117, 118], the nonlinear or anisotropic diffusion system [22, 23, 91, 105, 107], the sieve, which is a connected-set based morphological system, [13–17] developed by Bangham, and lastly the fixed-shape morphological system of Boomgaard [111] and Jackway [57] which is based on poweroids or quadratic structuring elements.

Figure 3.1 shows a selection of images from the scale-spaces of the above-mentioned systems. The original image (on the left) is a synthetic image of 100×100 pixels in the range [0,255], containing discs and rectangles that are corrupted with additive Gaussian noise of distribution ($\mu = 0, \sigma = 20$). All of the filters are classified as scale-space processors since they obey the causality criterion, however, they "simplify" the input signal in different ways, ranging from the complete blurring of the linear diffusion (top row), to the controlled blurring of the anisotropic diffusion (second row) to the non-blurring, area-decomposition of the sieve (third row) and finally the mottled or bubble-effect of the images morphologically filtered by poweroid structuring elements (bottom row). The values of scale for these representations are also very different. This stems from the alternative descriptions of the scale parameter for each filter. The linear diffusion filter and the erosion-dilation scale-space have a scale parameter that is determined by

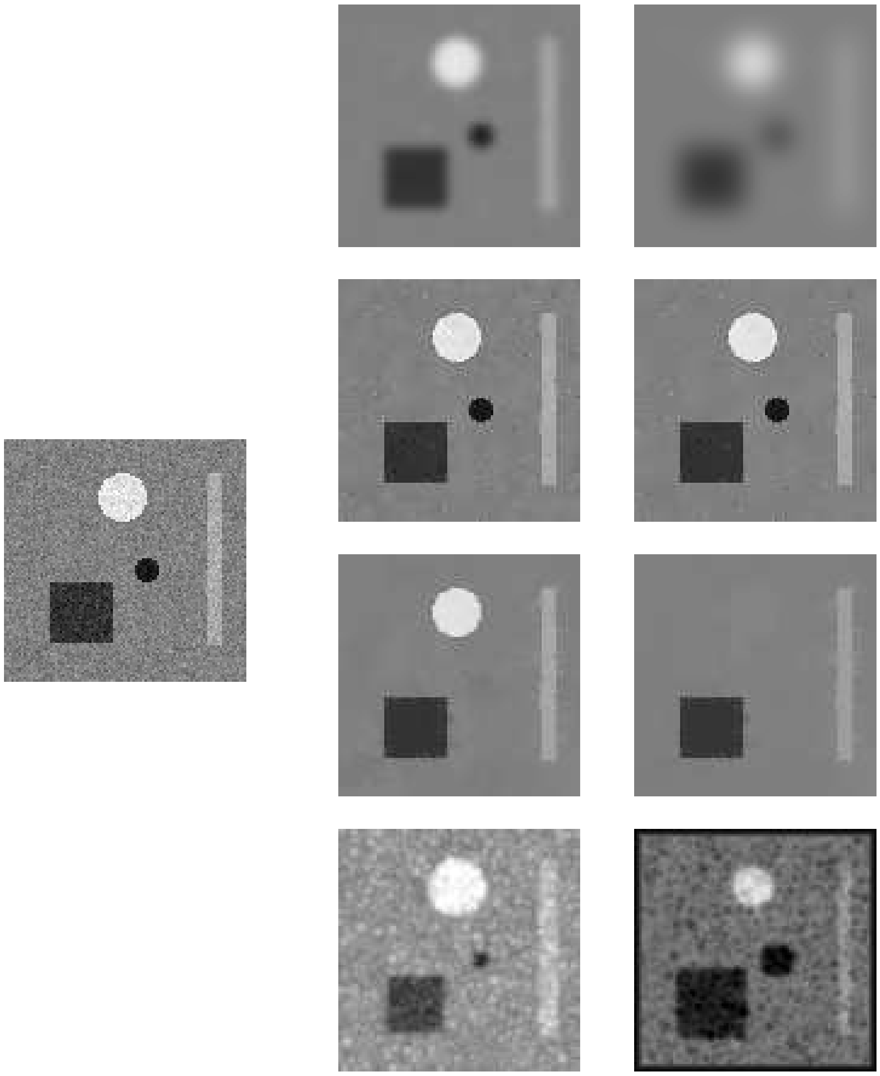


Figure 3.1: Example images from the scale-spaces of four different processors. The original input image (left) is a synthetic image corrupted with additive Gaussian noise. The top row shows the image filtered to scales 5 and 20 by the linear diffusion system, the second row shows the result of filtering using a type of anisotropic diffusion (scales 10 (left), and 20 (right)), the third row shows the results of one of the sieve-based filters at scales 100 and 200 and the fourth row shows the result of morphological dilation (left) and erosion (right) by a paraboloid structuring element of scale 0.01.

the filtering function. The anisotropic diffusion system's scale parameter is determined by the number of filtering iterations and the sieve's scale refers to pixel area in the input image. These differences are explained in more detail later. These figures are simply instructive in that they allow us to make a visual comparison of the systems and they also allow us an insight into possible problems that must be overcome if the desirable properties of scale-space processors are to be preserved. The following sections show some of the theory of these different approaches to scale-space filtering along with example images that further highlight the filter differences.

3.2 Linear diffusion

In [71] Lindeberg popularised the linear diffusion approach to scale-space filtering, continuing the theory originally proposed by Iijima, Witkin, Koenderink and Florack etc. The main contribution of [71] was to bring together many of the previous publications on linear scale-space theory and to present a *new* scale-space convolution kernel that is a discrete analogue to the Gaussian kernel. This new kernel is proved to be causal and is shown to have advantages over the discretised Gaussian-based scale-space.

This alternative filtering kernel is based on Bessel functions. The reasoning behind the new approach is explained through the problems that occur when translating continuous scale-space theory into discrete scale-space. Here, the properties of causality, homogeneity and isotropy are taken as the basis for linear multi-scale analysis.

The scale-spaces described in Chapter 2, section 2.2.3 refer to a linear scale-space representation where I is a continuous signal. From [8, 63, 118], to create a causal scale-space, the input signal must be convolved with a continuous Gaussian kernel, g_s.

$$I_s(x) = \int_{n=-\infty}^{n} g_s(n) I(x-n) dn \tag{3.1}$$

where the subscript, s, denotes the level of scale, equal to the variance, $(s = \sigma^2)$, of the Gaussian kernel

$$g_s(n) = \frac{1}{\sqrt{2\pi s}} \exp^{\left(\frac{-n^2}{2s}\right)} \tag{3.2}$$

This solution is said to be unique because the Gaussian is the *only* causal filter in these circumstances. All other techniques introduce extrema into the higher-scale representations.

However, in general, the area of computer vision deals with discrete rather than continuous image data. This limits us to two kinds of practicable scale-spaces; those with a discrete space and discrete scale parameter, and those with a discrete space and continuous scale parameter.

Since the continuous Gaussian is the unique kernel solution for filtering a continuous signal, a discrete Gaussian kernel would seem to be an sensible choice for filtering a discrete signal and creating a discrete scale-space. If the causality constraint is the only requirement of this discrete scale-space kernel, then convolution with a discretised Gaussian will provide a valid scale-space. Evaluating a signal, I, filtered to scale, s, using

$$y_s(x) = \sum_{k=0}^{\infty} T_s(k) I(x-k), \tag{3.3}$$

where T_s is a sampled version of (3.2), produces a causally-filtered signal. However, in [71], it is observed that the discretised Gaussian does not obey the *semi-group* property,

$$T_{s_2}(\cdot) = T_{s_2-s_1}(\cdot) * T_{s_1}(\cdot) \tag{3.4}$$

where T_s is the discrete kernel and '$*$' denotes a convolution operation. This demands that filtering a signal that has been pre-filtered to a scale s_1, to a scale $s_2 - s_1$, *must* produce an output signal that is equivalent to the original signal filtered to a scale of s_2.

Although imposing the semi-group property is not strictly necessary, it does simplify the analysis of the linear scale-space. Lindeberg shows that in the case of a discretised Gaussian, the causality property does not hold between two arbitrary filtering levels (i.e. the semi-group property does not hold). If s_1 and s_2 are two levels of scale, $(s_2 > s_1 > 0)$, and the corresponding filter outputs from (3.3) are I_{s_1} and I_{s_2}, then it can be shown [71] that the causality property between the two filter outputs holds if and only if the ratio s_2/s_1 is an odd integer or if s_1 is zero.

It is impractical to create a scale-space using a kernel that operates in this manner. Figure 3.2 shows how such a representation must be formed; increasing-scale output signals must be generated *only* from the preceding output signal. If intermediary-scale outputs are then required, for example a *scale* 10 output where *scale* $10 <$ *scale* m, the relationship between this new level and all consecutive filtered images are no longer

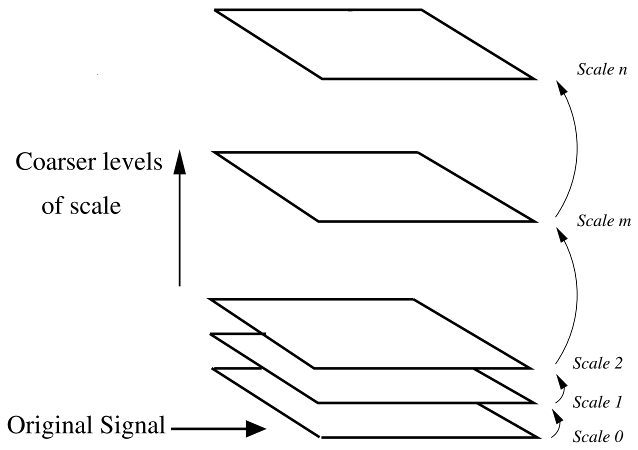


Figure 3.2: Scale-space structure from a *sampled Gaussian* kernel

causal and the scale-space fails. Hence the scale-space properties of the continuous Gaussian kernel are not fully preserved after discretisation.

The semi-group property is not a necessary constraint for scale-space processors. Indeed, some of the systems discussed later do not observe the property. However, it is necessary that no new extrema be created though the scale-space. For the causality constraint to be observed between two arbitrary levels of scale, s_1 and s_2, using the discretised Gaussian described above, then the signal at scale s_2 must be generated from the s_1 output signal. The main disadvantage of such a system is that some prior knowledge of the required scale-sampling rate is needed.

Investigations by Lindeberg [68–71, 107] lead to the definition of an alternative kernel; the T_s of (3.3) is taken to be

$$T_s(n) = e^{-\alpha s} I_n(\alpha s), \tag{3.5}$$

where I_n are the modified Bessel functions of integer order and α is a parameter that affects the scaling of the scale parameter, s. For simplicity α is set to 1.

This new filter is an approximation to the Gaussian kernel, but unlike the sampled Gaussian, the semi-group property is observed and so causality holds between two arbitrary levels of scale. The kernel also satisfies the requirements for both the discrete scale and continuous scale systems. A scale-space generated in this way is equivalent to the solution of a semi-discretised version of the diffusion equation (3.6) – discretised

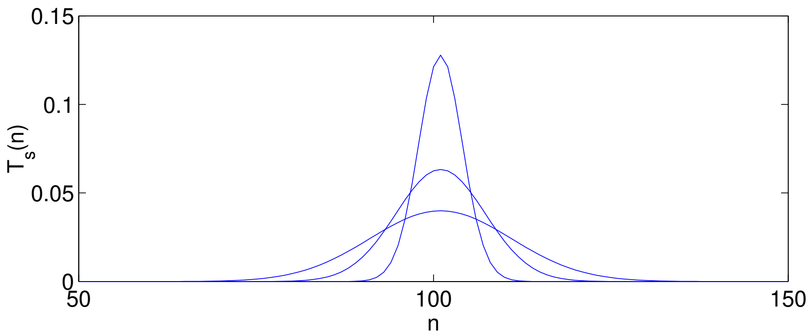


Figure 3.3: One-dimensional diffusion kernels: scales 10, 40 and 100 where s^2 = standard deviation

in space, but solved analytically in time, with the original signal, $I(x, y, 0)$, as the initial condition and a constant diffusion coefficient $c(x, y, s)$.

$$\frac{\partial I(x, y, s)}{\partial s} = \text{div}(c(x, y, s)\nabla I) \tag{3.6}$$

Examples of some increasing-scale kernels generated from (3.5) are shown in Figure 3.3. The kernels are normalised such that, for all s,

$$\sum_{n=-\infty}^{\infty} T_s(n) = 1. \tag{3.7}$$

Figure 3.4 shows the result of applying increasing scale diffusion kernels (3.5) to a synthetic one-dimensional signal. The signal is filtered between scales 1 and 120. The small-scale detail of the jagged-topped right hand pulse is lost after only a small amount of blurring. As the degree of blurring increases the three significant features start to merge until, at large scales (not shown), they become indistinguishable.

3.2.1 Moving from one to two dimensions

To apply the diffusion filter in two dimensions it is possible to create a two-dimensional approximation to the diffusion kernel such as those shown in Figure 3.5. Scales $s = 10$, $s = 50$ and $s = 100$ are shown, where s is the square of the standard deviation of the kernels. However, it should be noted that convolving an image with two-dimensional kernels is inefficient. The kernel defined in equation (3.5) is separable and a two-

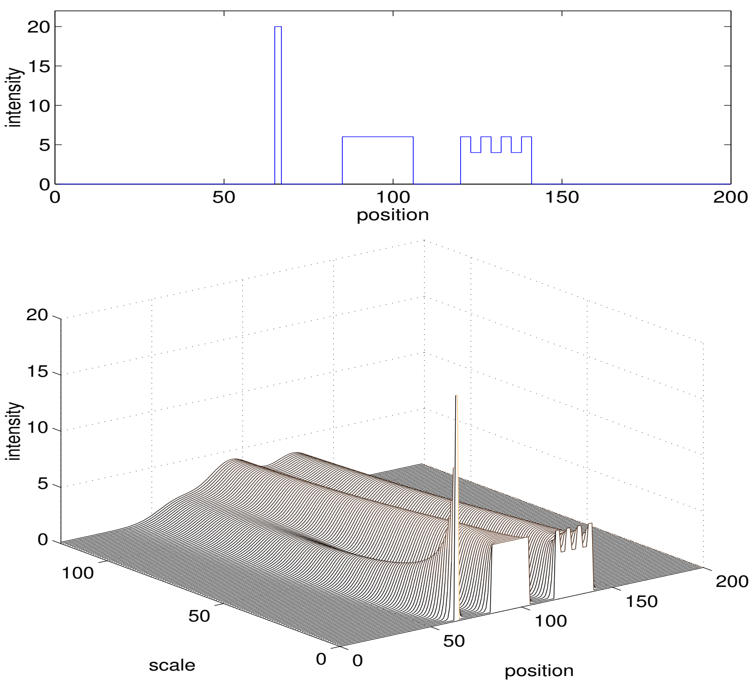


Figure 3.4: Synthetic example of one-dimensional diffusion filtering. The original signal is shown above and is shown below filtered between scales $s = 1$ and $s = 120$.

dimensional convolution may therefore be more efficiently implemented as two one-dimensional convolutions along the rows and down the columns of the image.

$$y_s(n,m) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} T_s(k,l) y(m-k, n-l) \tag{3.8}$$

where

$$T_s(k,l) = e^{-s} I_k(s) e^{-s} I_l(s) \tag{3.9}$$

As these kernels are used to filter images to increasing scales it is important to limit any arithmetic errors due to finite precision. Because of the high level of blurring that occurs as a result of diffusion filtering, image features are generally traced through the the scale space by following the extrema points of each scale representation. False extrema points caused by factors such as kernel truncation errors can confuse tracking algorithms and so it is helpful to minimise these effects. If possible the kernel truncation

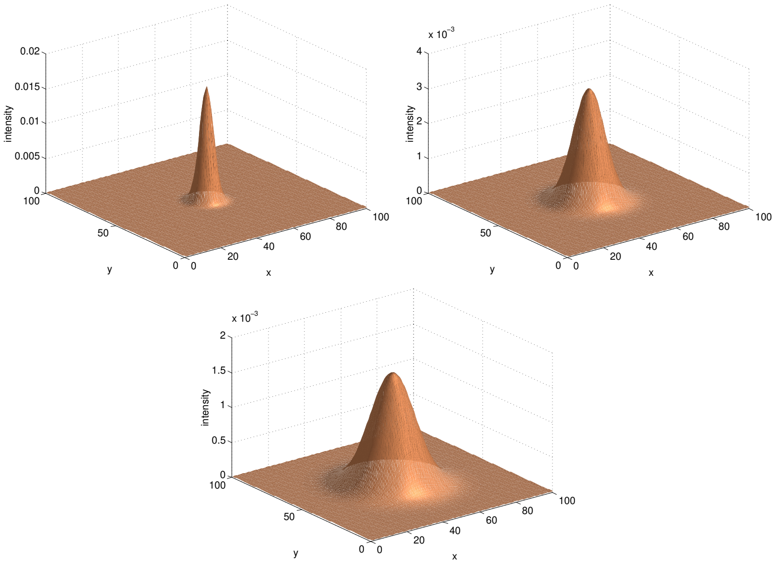


Figure 3.5: Two-dimensional linear diffusion kernels at scales $s = 10$ (top-left), $s = 50$ (top-right) and $s = 100$ (below). Note the differences in intensity between the three kernels: all kernels must sum to 1.

length should be greater than the length of the longest image side. This will contain the truncation errors outside of the image boundarys. However, for very large images this becomes impractical. For speed, the FFT/IFFT algorithms that are used to implement the linear diffusion convolution throughout this thesis pad both the kernel (of length L) and the image (longest side length N) to

$$\text{zero-padding length} = 2^n \geq (L + N + 1) \qquad n > 0 \tag{3.10}$$

As image sizes increase, convolving with a kernel whose length is longer than the largest image length is only practical for large scales; where the minimum requirement of 8 standard deviations of kernel are greater than N. As an alternative, Lindeberg [71] suggests that for all convolutions the upper bound of the relative truncation error ε, must be sufficiently small (typically 10^{-6} or 10^{-7}).

3.2.2 Boundary conditions

Increasing the length of the kernel to limit the truncation error and to include as many standard deviations as possible produces a further side effect. The kernel is then sensitive to changes outside of the image confines and so the boundary conditions for the image must therefore be considered.

There are various options for the choice of image boundary. These include:

- *Mirroring* at the boundary to produce a reversed image at each edge.
- *Wrapping* each border to meet the opposite image boundary, creating an "infinite" image.
- *Extending* the image boundary by continuing the edge pixels up to the length required to pad the FFT/IFFT.
- *Zero-padding* the image, this usually involves subtracting the image mean to zero to reduce the possible edge effects.

Of these solutions, the mirroring and wrapping options are discarded. They create an artificial structure to the space outside of the image that cannot be assumed. If there are significant (high intensity) features towards the edges of the original image, this artificial boundary may intrude upon the diffused image through filtering. Of the third and fourth boundary options, extending the image boundary is discarded for efficiency reasons and the remaining zero-padding option is used to implement all examples of linear diffusion in this thesis. In addition to this the mean is subtracted from all experimental examples in an attempt to reduce the step effect at the boundary edges. Nevertheless, it should be emphasised that the boundary conditions for diffusion systems are rarely discussed but have far more serious consequences than, for example, the method of discretising the kernel.

3.2.3 Properties of linear diffusion

The linear diffusion scale-space, as described by Lindeberg, has the following properties.

- **Causality:** In fact, the original causality definition of "no creation of new extrema" (described in Section 2.3.1 of the previous chapter) does *not* hold for all cases with the linear diffusion scale-space. There are, (albeit pathological) cases in two dimensions where the number of extrema within an image will increase with increased filtering. A popular example of this is presented in [67]. The paper describes a two-dimensional image consisting of two hills connected by a thin isthmus. This image is described as having one extremum. The top panel

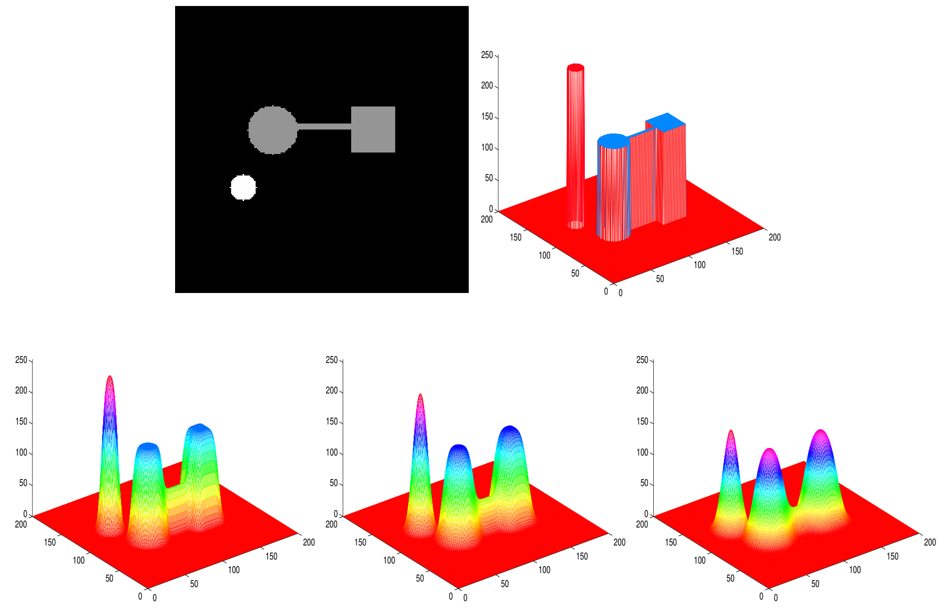


Figure 3.6: New local extrema may be created by the diffusion filter in the two-dimensional case. Top panel: Original image of *one* extremum and its surface intensity plot. Bottom panel left: original image filtered to scale 10, middle: image filtered to scale 20, right: image filtered to scale 40. all filtered images show two extrema.

of Figure 3.6 shows a synthetic example of such an image, with the hills being replaced by a sharp-edged disc and square. The right hand side of the top panel shows the corresponding surface intensity plot.

When this image is filtered by the discretised two-dimensional diffusion processor, the thin isthmus diffuses away more rapidly than the two connected areas. This leads to the development of two separate extrema after filtering – a violation of the previously mentioned causality axiom. In fact, the creation of new local extrema by linear smoothing is *"inherent and inescapable"* [71] when working with two or more dimensions. As a result of this there is a re-definition of the causality axiom. This simply requires that "*if for some level t_0 a point x_0 is a local maximum for the scale-space representation at that level (regarded as a function of the space coordinates only) then its value must not increase when the scale parameter increases.*" [71] (If x_1 were a minimum then its value must not decrease for increases in scale.)

- **Isotropy and Homogeneity:** The diffusion kernel is isotropic and homogeneous, having the same effect in all directions over the image, i.e. it is space invariant. However, these two properties serve no real practical purpose, they were simply introduced in [63] to simplify the analysis that showed the only linear solution to causal scale-space analysis to be the linear diffusion equation.

3.2.4 Example linear diffusion scale-spaces

Figure 3.7 shows a sample of signals from a one-dimensional scale-space representation where the original image (top panel) is 331 elements in length and has intensities in the range $[0-255]$. Scales $s = 5,\ 25,\ 50,\ 100,$ and 500 are shown, and the detail of the original signal is blurred as scale increases. The signal mean is subtracted before diffusion (See Section 3.2.2).

Example images from the two-dimensional diffusion processor (3.8) are shown in Figure 3.8, the original image is shown in the top panel followed by scale-representations at $s = 2,\ 10,\ 20$ and 50. As in the one-dimensional case, the increasing scale filters bring increased suppression of detail. However the disadvantages of the high levels of blurring are more apparent in two dimensions. At large scales it becomes difficult to distinguish between different objects. This can lead to localisation problems when estimating the position of features at a coarse scale. There are also further problems associated with drifting extrema. The 'infinite' or, in practice, 'wide-spread' influence of the kernel means that any pixel in the image is affected by all other pixels. As a signal is filtered to increasing scales, the extrema positions of the signal are influenced by surrounding features; the more significant (higher intensity) those features are, the more influence they exert. For this reason it is necessary to track the movement of features through scale and space in order to relate feature positions at a coarse scale to the

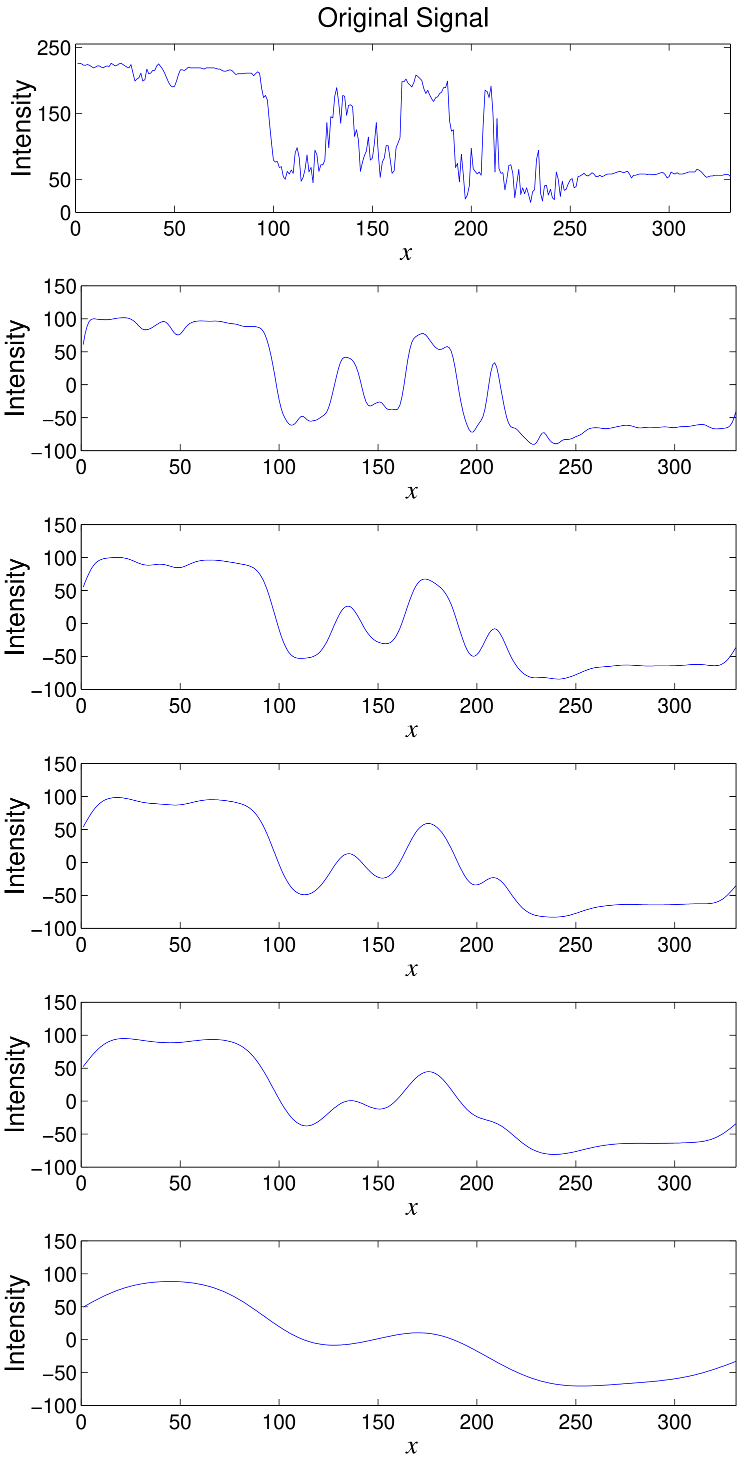


Figure 3.7: One-dimensional scan-line (top panel) simplified via linear diffusion at scales 5, 25, 50, 100, and 500. The mean is subtracted from the original signal before filtering.

Figure 3.8: Images simplified via linear diffusion at scales 2, 10, 20 and 50 (clockwise).

positions of similar features at a small-scale. It is the problem of this blurring effect that led to the development of an alternative anisotropic diffusion filter. There are many variations of this filter which is based on allowing the rate of diffusion to vary as a function of the image gradient – a few of the more popular diffusion functions are discussed in detail in the following section.

In Section 2.3 of the previous chapter, one of the suggested properties of any practical scale-space was the development of a scale-selection surface. In other words, a means of identifying objects or features within a signal as being of a particular scale. There is such a surface for the linear diffusion filter. One of the suggestions in [71] is to study the curvature of the scale-space. The positions of peaks in this measure are said to indicate significant scales of image features. This is described in more detail in Section 3.7 where the different scale-measures and scale-selection surfaces of all processors described in this thesis are discussed.

3.3 Anisotropic diffusion

If the homogeneity and isotropy axioms of Section 2.3.2 are discarded then the conductivity, c, may vary so that c becomes $c(x, y, s)$. In this case the diffusion equation becomes

$$\frac{\partial I(x, y, s)}{\partial s} = \text{div}\left(c(x, y, s)\nabla I\right)$$

Non-linear or *anisotropic* diffusion, as described by Perona and Malik [91, 92], is one example of a numerical solution to the diffusion function $c(x, y, s)$. Anisotropic diffusion, which works by allowing the structure of the input image at any particular point to determine the local rate of diffusion, has the advantage of limiting the problematic blurring that is inherent to the linear diffusion system. At the beginning of this chapter, Figure 3.1 showed a simple synthetic example image filtered using anisotropic diffusion. The scale $s = 10$ and $s = 20$ representations show the reduction in noise from the original image while the edges of the image features remain unblurred. In this implementation, scale refers to the number of iterations required to generate an output image.

A general framework for anisotropic diffusion may be formulated via the study of robust statistics [23]. In the following subsection this framework is used to illustrate some common anisotropic diffusion schemes.

3.3.1 Robust statistics

Robust statistics is concerned with outliers within a signal. If the input signal is assumed piecewise-constant, corrupted with zero-mean Gaussian noise with a small variance. Then outliers within this signal are identified as points with relative gradients

$$\nabla I_{i,p} = I_p - I_i^s, \qquad p \in \eta_i \tag{3.11}$$

that are significantly higher than the allowable variance of the assumed noise. This *noise threshold* or cutoff point is labelled σ. In (3.11), s denotes scale, i is pixel position and p represents the spatial neighbourhood of each pixel; η_i being the North, South, East and West pixels for a four-connected neighbourhood of i.

The nature of the diffusion will depend on σ and the behaviour of the conductivity for gradients under and near the σ threshold. Points with gradients that are within this limit are smoothed towards piecewise-constant regions and points outside the limit, those identified as outliers, are rejected and left relatively untouched by the diffusion.

The continuous diffusion equation of (3.6) may be rewritten as

$$\frac{\partial I(x,y,s)}{\partial s} = \text{div}\left(\rho'(\| \nabla I \|, \sigma)\frac{\nabla I}{\| \nabla I \|}\right) \tag{3.12}$$

where

$$c(x,y,s) = \frac{\rho'(\| \nabla I \|, \sigma)}{\| \nabla I \|} \tag{3.13}$$

and $\rho(x)$ is chosen to minimise the effects of outliers, $(I_p - I_i)$. The strength of the function $\rho'(\| \nabla I \|, \sigma)$ compared to $\| \nabla I \|$ controls the conductivity. In anisotropic diffusion the idea is to reduce the conductivity at large $\| \nabla I \|$ so one would expect $\rho'(x, \sigma)$ to tend to zero for large x.

If the continuous linear diffusion equation is implemented in this manner, where no attention is given to minimising the diffusion of outliers, then equation (3.6) may be re-written as

$$\frac{\partial I(x,y,s)}{\partial s} = \text{div}\,(c\nabla I) \tag{3.14}$$

which, when compared to (3.12), leads to

$$\rho'(x,\sigma) = cx \tag{3.15}$$

and so for large x, there is increased smoothing, as expected for the case of linear diffusion.

Figure 3.9 shows the result of selecting a *Lorentzian* function,

$$\rho_{_{Lor}}(x,\sigma) = \log\left(1 + \frac{1}{2}\left(\frac{x}{\sigma}\right)^2\right) \tag{3.16}$$

$$\rho'_{_{Lor}}(x,\sigma) = \frac{2x}{2\sigma^2 + x^2} \tag{3.17}$$

to control the degree of diffusion. In this figure the diffusion-rate is divided into three zones. In zone (**I**), for smaller intensity differences, the diffusion rate (vertical axis) is

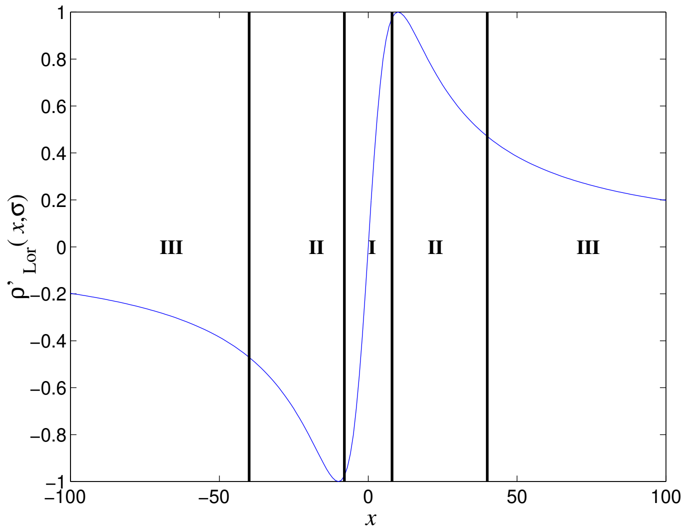


Figure 3.9: Derivative of the Lorentzian diffusion function, $\rho'_{Lor}(x, \sigma)$, (3.17) for increasing x (increasing gradients). The function is scaled into the range [-1,1] and the outlier cutoff is 10.

almost linearly related to the gradient magnitude (horizontal axis) and the system acts like a linear diffusion system. In zone (**II**) the diffusion is decreasing, the turning point being the specified noise cutoff σ, and in zone (**III**), the diffusion tends towards zero for larger magnitude gradients.

The Lorentzian function is proportional to one of the diffusion functions or *stopping functions* given by Perona and Malik in [91]. Their notation differs slightly, the stopping function being described as $g(x)$ in the diffusion equation

$$\frac{\partial I(x, y, s)}{\partial s} = \text{div}\,(g(\parallel \nabla I \parallel)\nabla I) \tag{3.18}$$

However, this links in easily with the earlier robust statistics diffusion equation of (3.28), if

$$g(x) = \frac{\rho'(x)}{x} \tag{3.19}$$

Perona and Malik's stopping function (3.20) is designed to limit the amount of blurring across wide regions.

$$g(\nabla I) = \frac{1}{1 + \left(\frac{\|\nabla I\|}{K}\right)^2} \tag{3.20}$$

The value of the constant, K, determines the level of preservation of image structure. Smaller values of K cause smoothing around the lower contrast regions of the image whereas larger values quickly break down the structure of an image. K may be set by hand or, alternatively, obtained using the Canny, [30], *noise estimator*. In [107], this involves setting K to the 90% value of the integral of a histogram of the absolute values of the gradient throughout the image (3.21).

$$K = i_{val} \qquad \text{such that} \qquad \frac{\sum_{i=0}^{i_{val}} \| \nabla I \| (i)}{\sum_{i=0}^{\max(\|\nabla I\|(i))} \| \nabla I \| (i)} = 0.9 \tag{3.21}$$

where K is the index along the histogram baseline. It is also suggested that K be recalculated at every iteration, but this involves more computation than simplifying the image so a compromise is to compute the value of K every 5 or 10 iterations.

In the case of (3.20), $g(x)$ is directly related to the Lorentzian robust diffusion function shown in Figure 3.9. If the constant K is set to $K^2 = 2\sigma^2$, then using (3.19), equation (3.20) may be rewritten as

$$g(\nabla I)\nabla I = \frac{2\nabla I}{2 + \left(\frac{\| \nabla I \|}{\sigma}\right)^2} \tag{3.22}$$

This may then be rearranged to give

$$g(\nabla I)\nabla I = \frac{2\nabla I \sigma^2}{2\sigma^2 + \| \nabla I \|^2} \propto \rho'_{Lor}(\nabla I, \sigma) \tag{3.23}$$

Which is found in [23], to be proportional to the derivative of the *Lorentzian* diffusion function (3.17).

Perona and Malik also suggest a second stopping function in [91] that in [23] is reported to be related to a diffusion function of Leclerc [66]. This stopping function,

$$g(\nabla I) = e^{-\left(\frac{\|\nabla I\|}{K}\right)^2} \tag{3.24}$$

is designed to limit the blurring across higher contrast edges and is shown in Figure 3.10, plotted in the form $\rho'(x) = g(x)x$ so that it may be easily compared with the Lorentzian-based function of Figure 3.9.

As with the previous diffusion function, the diffusion rate for small (≤ 10) gradients is approximately linear and the point of maximum diffusion is determined by the noise cutoff. Outliers with gradients above this point suffer less diffusion and are preserved. Unlike the Lorentzian function, the second Perona-Malik edge-based function has a diffusion-flow that approaches zero for realistic intensity differences. In the case of the Lorentzian of eqn (3.17), $\rho'_{Lor} \to 0$ only for $\nabla I_{s,p} \to \infty$.

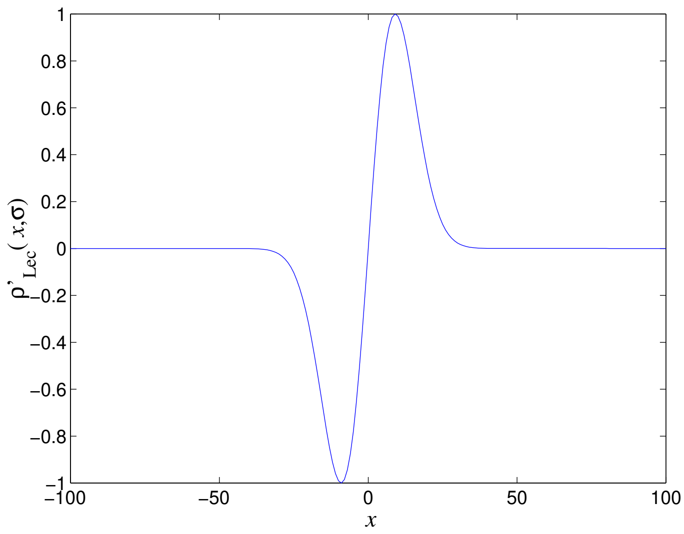


Figure 3.10: Plot of the normalised Perona-Malik edge-based diffusion function, (3.24) in the form of $\rho'(x) = g(x)x$. The function is scaled into the range [-1,1] and the outlier cutoff is 10.

The value of the noise-threshold constant σ is based on the calculation of a *robust scale*, σ_e, obtained from robust statistics, (3.25).

$$\sigma_e = 1.4826 \text{ median}_i \left(\mid (\parallel \nabla I \parallel - \text{median}_i(\parallel \nabla I \parallel)) \mid\right) \tag{3.25}$$

This represents the median of the deviation of the absolute deviation of the original signal, from the median of the absolute deviation, and the constant is derived from the fact that the median absolute deviation of a zero-mean normal distribution with unit variance is $0.6745 = 1/1.4826$. The relationship between the threshold constant σ and value of σ_e depends on the choice of diffusion function. In the examples of Figure 3.9 and Figure 3.10, the value of σ_e is set to 10. For the Lorentzian function to begin rejecting outliers at a gradient value of σ_e, a value of $\sigma = \sigma_e/\sqrt{2}$ is required and, $K = \sigma = \sigma_e\sqrt{2}$ appears to be a suitable relation for the diffusion function of (3.24). Both plots show how the influence of large-value outliers, in this case $((-10 > \nabla I_{s,p})$ or $(\nabla I_{s,p} > 10))$, is limited. This, in turn, limits the local diffusion-flow around the outliers.

Since the introduction of the Perona and Malik non-linear diffusion scale-spaces many variations have been developed. In addition to providing a formal robust statistics analysis of some existing systems, Black [23], presented a new system that also obeys scale-space causality and that is reported as being robust to some types of image noise.

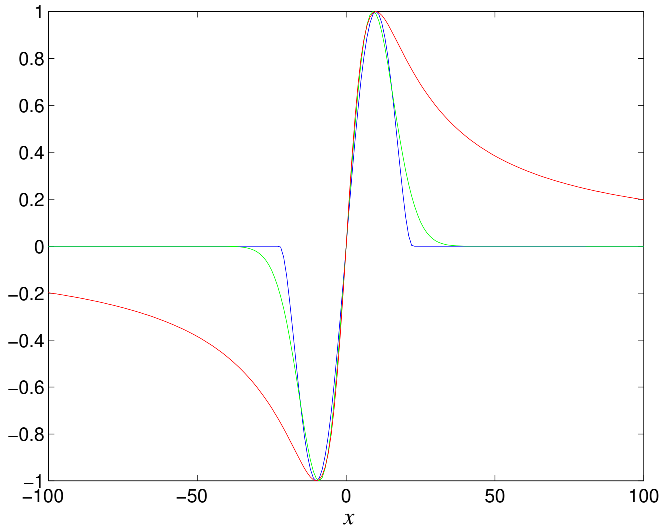


Figure 3.11: The normalised Tukey-biweight derivative function, $\rho'_{Tuk}(x,\sigma)$ (3.27), in *blue*, shown with the Lorentzian diffusion function, $\rho'_{Lor}(x,\sigma)$, of (3.17) in *red* and the Perona-Malik edge-based diffusion function of (3.24) in *green*. Each function is scaled into the range [-1,1] and the outlier cutoff is 10.

The *Tukey-biweight* based function (3.26) that is used in Black's anisotropic system has a derivative function, (3.27), that descends to zero for realistic image gradients.

$$\rho_{Tuk}(x,\sigma) = \begin{cases} \frac{x^2}{\sigma^2} - \frac{x^4}{\sigma^4} + \frac{x^6}{3\sigma^6} & |x| \le \sigma \\ \frac{1}{3} & \text{otherwise} \end{cases} \tag{3.26}$$

$$\rho'_{Tuk}(x,\sigma) = \begin{cases} x(1-(x/\sigma)^2)^2 & |x| \le \sigma \\ 0 & \text{otherwise} \end{cases} \tag{3.27}$$

Figure 3.11 shows a plot of the Tukey derivative function (in blue) along with the Lorentzian and (3.24) functions. All of the functions are normalised to be within the range $[-1, 1]$ and in all cases $\sigma_e = 10$, σ being calculated as $\sigma = \sigma_e\sqrt{5}$ for the Tukey function.

The behaviour of the Tukey derivative is similar to that of the edge-based function of Perona and Malik. For gradient values $I_p - I_i < 10$ pixel intensity differences are smoothed towards flat-zones. The maximum amount of smoothing occurs for gradients $I_p - I_i = 10$, and for steeper gradients, the rate of smoothing rapidly decreases until, for outliers of $I_p - I_i \geq 22$, there is no diffusion, preserving the edges of the flat-zones.

In practice, the continuous diffusion equation given in (3.12) may be implemented through the iterative discretisation

$$I_i^{s+1} = I_i^s + \frac{\lambda}{\mid \eta_i \mid} \sum_{p \in \eta_i} \rho'(\nabla I_{i,p}, \sigma) \tag{3.28}$$

Where λ is a scalar (usually set to 1) that determines the rate of diffusion scaling. I_i^{s+1} is the value of the current pixel, i, after filtering to a scale of $s + 1$; scale being marked by the number of iteration steps, and $\mid \eta \mid$ is equal to 4 for a four-connected neighbourhood.

3.3.2 Boundary conditions

All examples on anisotropic diffusion in this thesis use a four-connected neighbourhood. For this reason, all that is required as a boundary condition is the extension of the current boundary intensities for one pixel in the North, South, East and West directions.

3.3.3 Properties of anisotropic diffusion filtering

- **Causality:** All of the non-linear diffusion functions described in this thesis adhere to the causality criterion; i.e. no artificial extrema points are created for increasing scale images.

- **Immediate localisation:** For the three listed diffusion schemes, the rate of diffusion is restricted at regions of high local contrast. Since high contrast points are linked with edges and region boundaries in images, non-linear diffusion provides a good basis for accurately locating these boundaries.

- **Piecewise smoothing:** If the local contrasts within each region are are small compared to the region boundaries and are below the noise-cutoff level, then as scale increases intra-region smoothing will occur preferentially over inter-region smoothing.

3.3.4 Examples of non-linear diffusion scale-spaces

Examples of some scale-spaces for real one-dimensional signals, produced by the three functions are shown in Figures 3.12, 3.13 and 3.14. In all cases the value of σ_e is calculated using eqn (3.25).

In Figure 3.12 we see the simplification of the one-dimensional signal using the Lorentzian based diffusion or "stopping" function. As scale increases the filter removes the small-scale noise present in the original signal, scale being related to the local contrast. Signal elements with low contrast are removed, retaining a series of "flat-zones". These "flat-zones" are also present in the edge-based diffusion of Figure 3.13 along with additional lower-contrast edges considered significant by the value of σ. The faster fall-off or drop in the edge-based diffusion rate for gradients above the cutoff limit, (seen in Figure 3.11), causes more edges to remain untouched by the diffusion.

The result of the even-sharper falloff of the Tukey diffusion function in seen in Figure 3.14. The value of σ_e as calculated by eqn (3.25) for this one-dimensional example is 3.7065. Since this is so small, much of the higher contrast points are left untouched. Only the lower-contrast areas are smoothed to flat zones.

The differences between linear and non-linear diffusion filtering are more easily observed in two dimensions. Figures 3.15, 3.16 and 3.17 show the effects of the three anisotropic diffusion functions on a real image. In each case the original image is shown in the top panel, followed by images filtered to scales $s = 10$ (i.e. 10 iterations), 50, 100 and 200.

The differences between these images and those, from the linear diffusion system shown in Figure 3.8, are clear. Both the linear and non-linear diffusion scale-spaces are causal, however with the anisotropic system the property of *immediate localisation*, introduced in Section 2.3.3, is observed. Structures in the filtered image have sharp edges that coincide with similar feature edges in the original image. For instance, in all three anisotropic diffusion systems, the white rail above the glass panels is visible, and

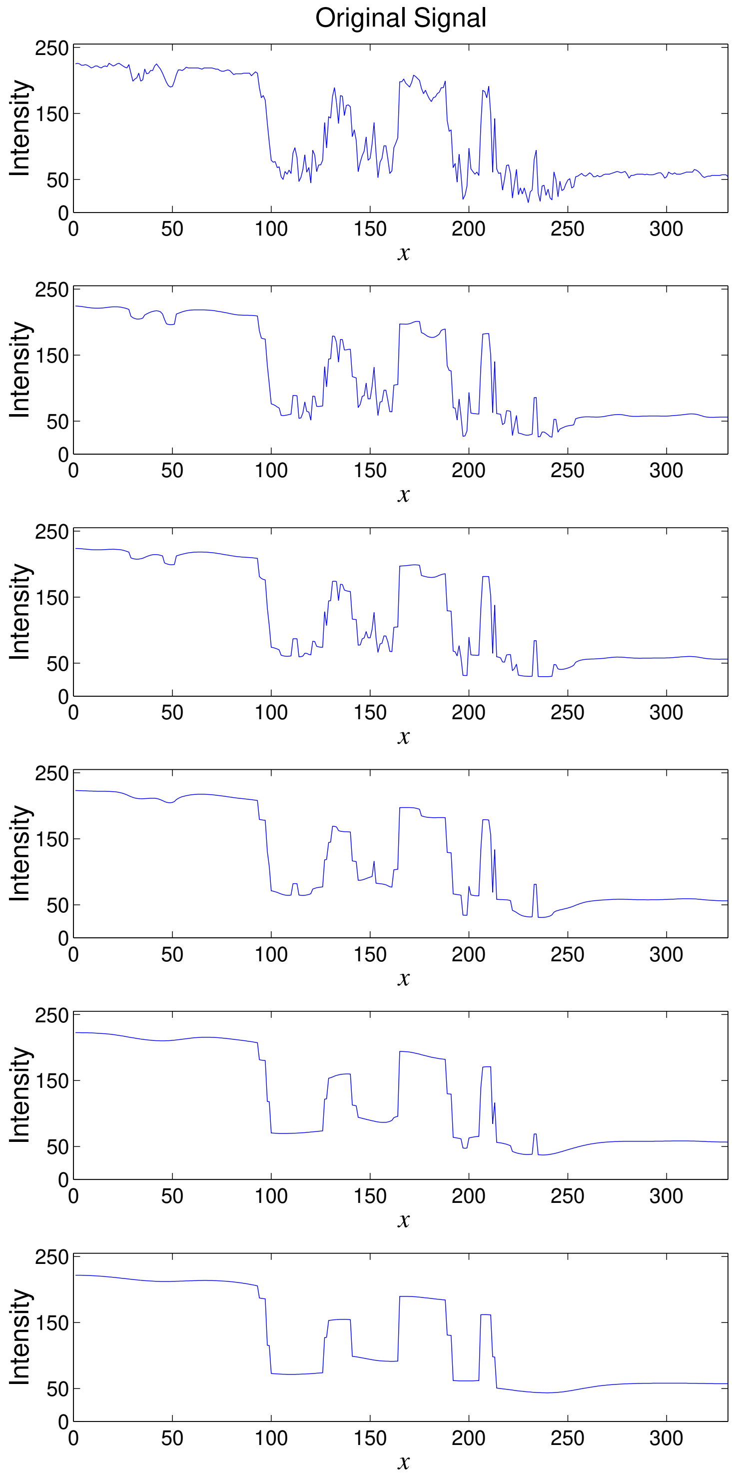


Figure 3.12: Scan line simplified via Lorentzian-based anisotropic diffusion (3.17); Scales 25, 50, 100, 250, 500 and 1000 are shown. The value of σ_e is calculated from (3.25)

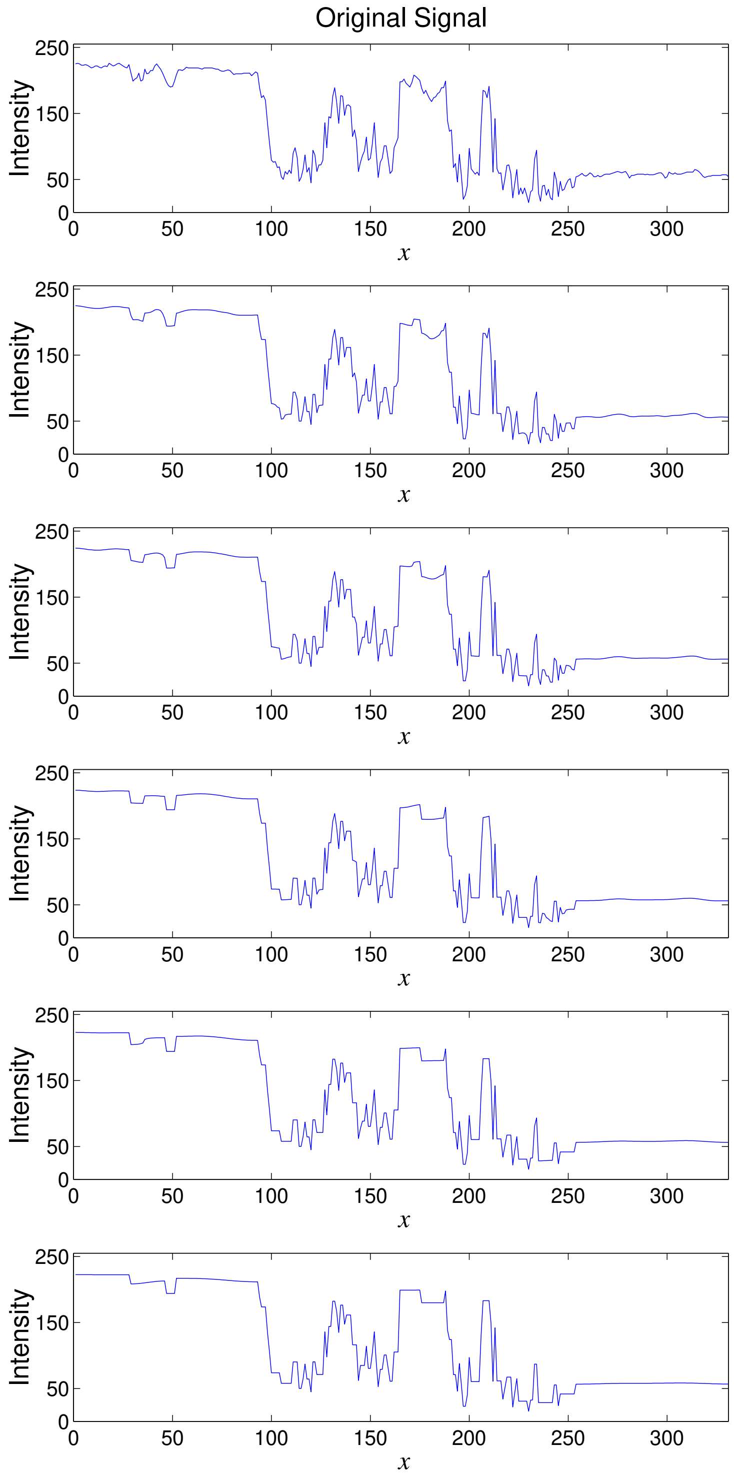


Figure 3.13: Scan line (top panel) simplified via Perona and Malik edge-based anisotropic diffusion: (3.24). Scales 25, 50, 100, 250, and 500 are shown. The value of K was set using $K = \sigma = \sigma_e\sqrt{2}$.

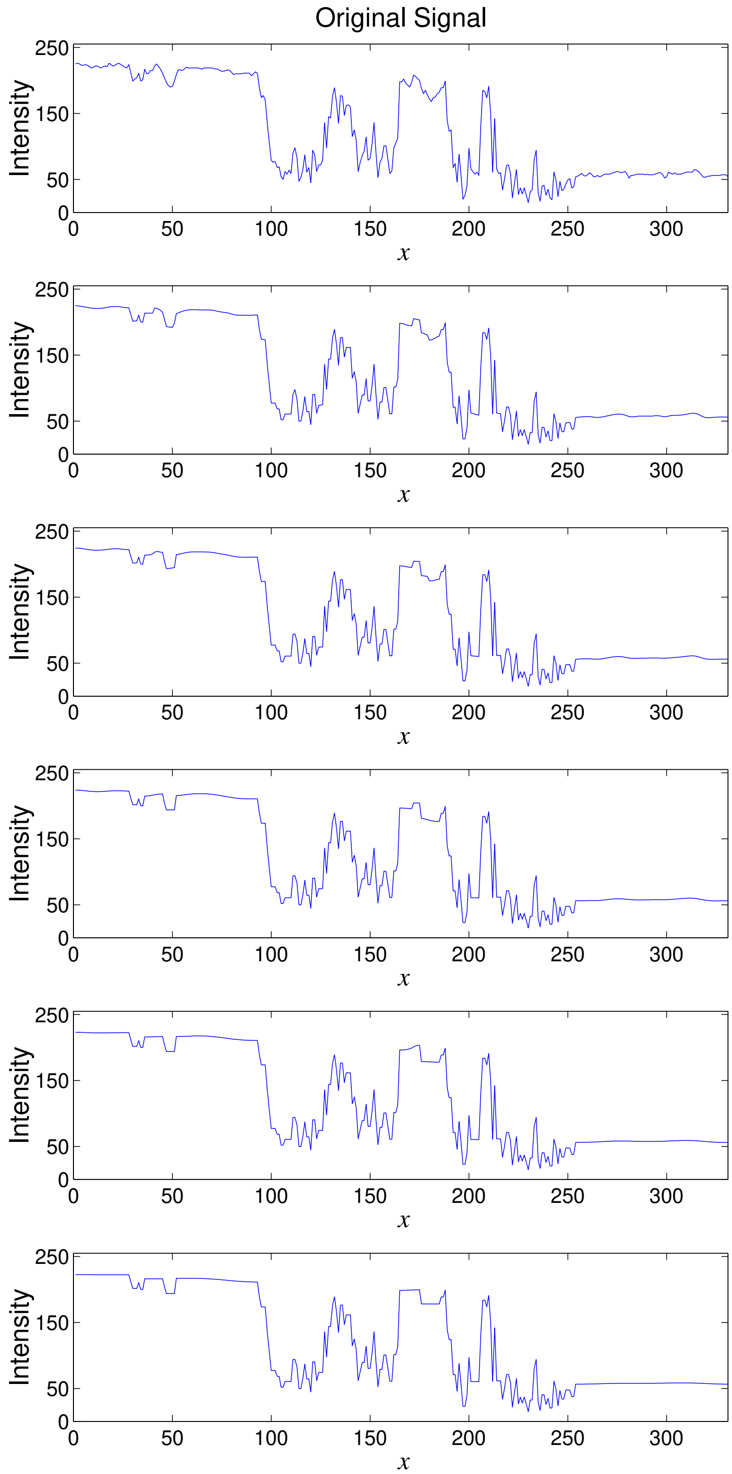


Figure 3.14: Scan line (top panel) simplified via Tukey-based anisotropic diffusion (3.27); Scales 25, 50, 100, 250, 500 and 1000 are shown. The value of σ_e is calculated from (3.25)

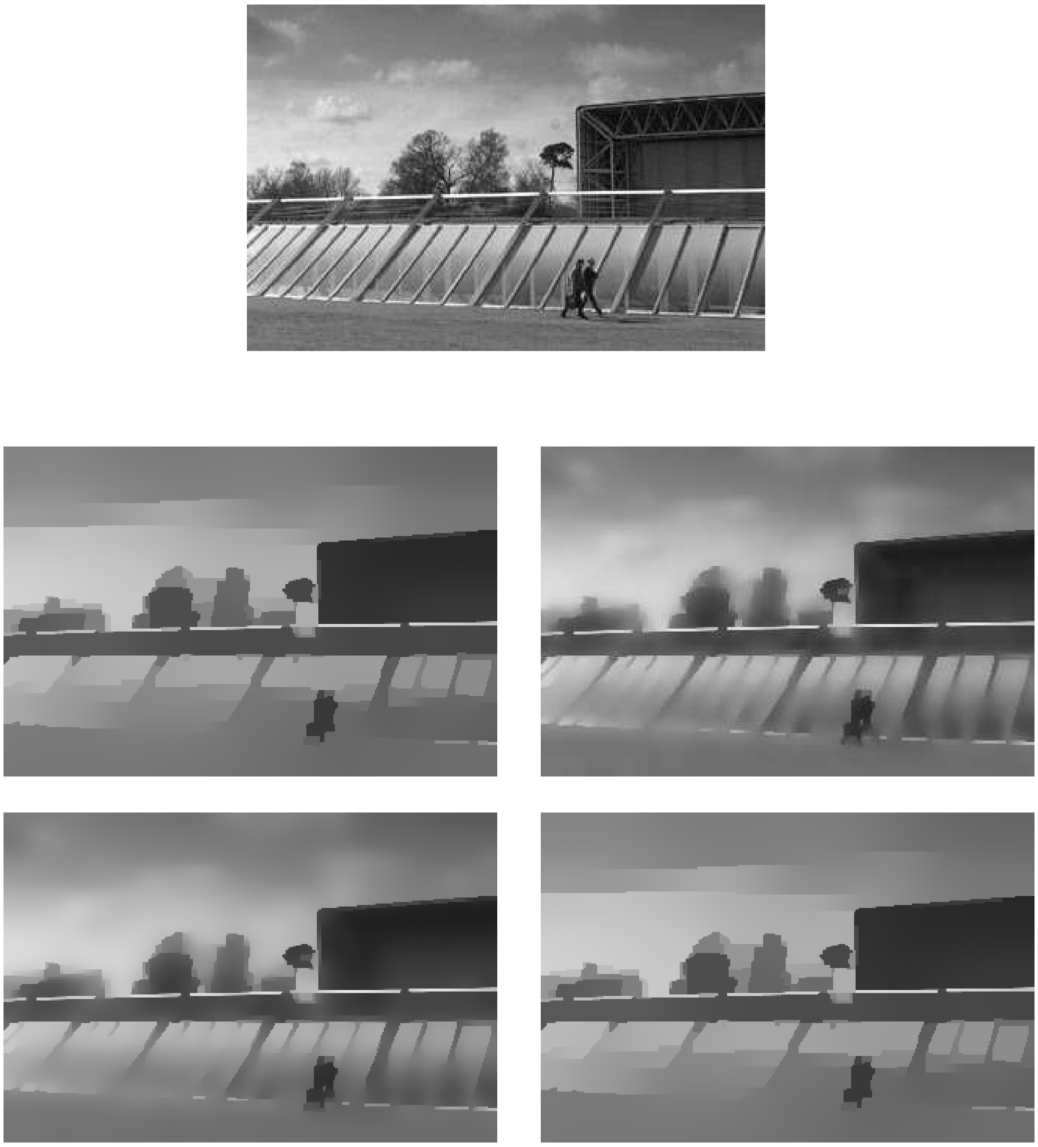


Figure 3.15: Images simplified via anisotropic diffusion using the diffusion function given in (3.17). The scale-space is made up of images filtered to scales 10, 50, 100 and 200, the scale indicating the number of iterations of eqn (3.18)

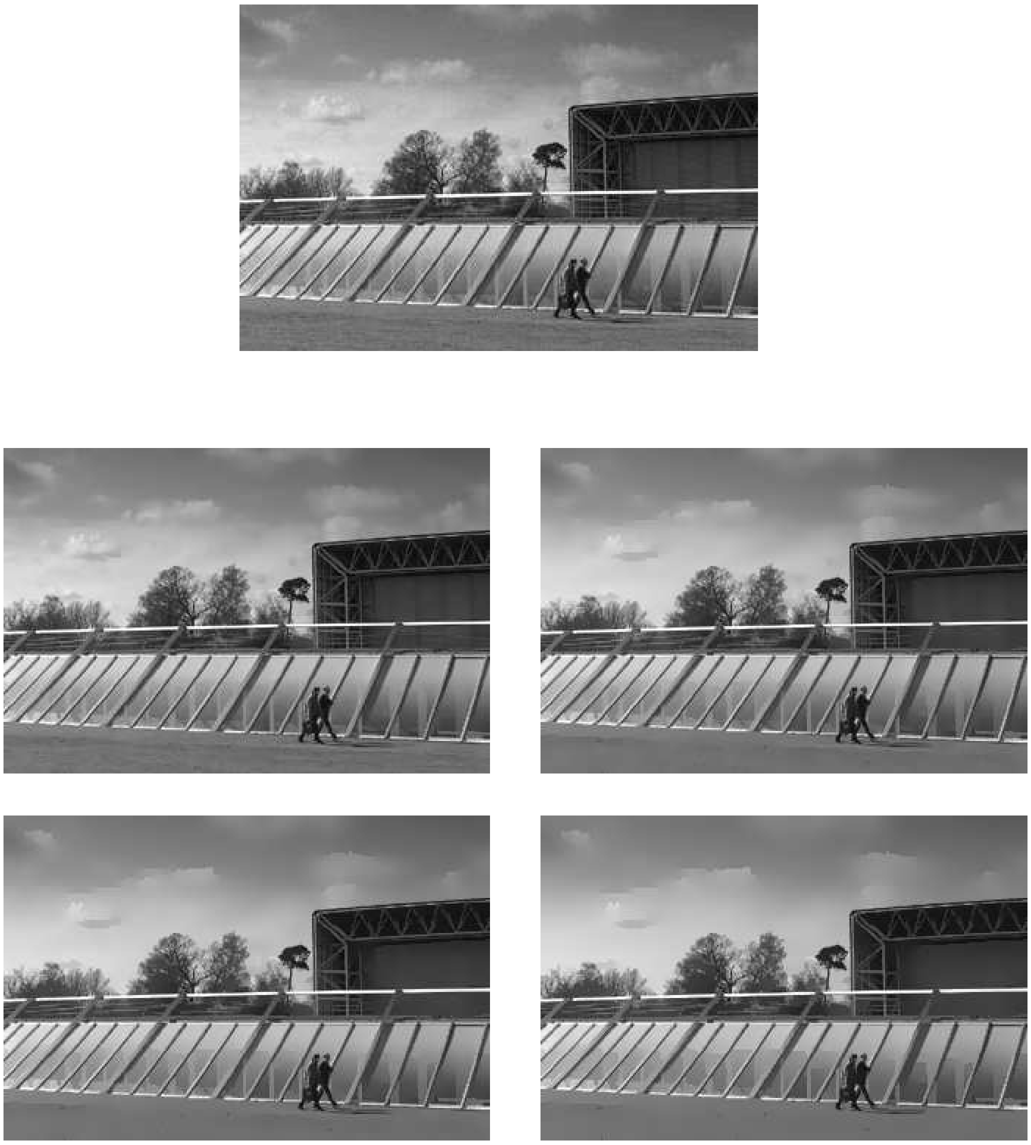


Figure 3.16: Images simplified via anisotropic diffusion using the diffusion function given in (3.24). The scale-space is made up of images filtered to scales 10, 50, 100 and 200, the scale indicating the number of iterations of eqn (3.18)

Figure 3.17: Images simplified via anisotropic diffusion (3.28) with the Tukey-biweight diffusion function of (3.26) at scales 10, 50, 100 and 200

remains in the same position throughout filtering. This same feature in the linear diffusion example is blurred in the image at scale 20 and indistinguishable (by eye) from the background at scale 50. The ease with which coarse-scale features in an image may be located in the original signal means that the tracking that is required to locate and record the movement of objects in a linear scale-space is not required. The difference between each of the non-linear systems is also more apparent in two dimensions, after a large number of filtering iterations, the edge detail of the Lorentzian system in Figure 3.15 begins to diffuse into the background, this is not the case with the second Perona and Malik function and the Tukey function as diffusion is stopped completely around higher contrast points. Indeed, the small value of σ_e calculated for this image allows for very little diffusion in the case of the Tukey filter.

Much of the fine-scale detail of the original image is retained in the anisotropic systems. For instance, the branches of the trees and the wire structure on the right-hand building are all visible in the coarse-scale representations because of the high contrast between these features and their backgrounds. Only the low contrast features are affected by the anisotropic diffusion filter. This is even more apparent in the Tukey results where the diffusion is limited to the low-contrast cloud and grass areas of the image, giving the final filtered image the appearance of a 'cleaned' and 'sharpened' version of the original. The filter has the effect of smoothing away any low-contrast noise with minimal damage to the underlying structure of the image. Raising the value of σ has the effect of increasing the level of diffusion in the higher-contrast features.

3.3.5 Discussion

There are many published variations on these anisotropic diffusion systems, all aimed at limiting the blurring of the linear diffusion processor, but they have significant problems. The processing time taken for these filters far exceeds that taken for the linear diffusion counterpart. Furthermore since they are iterative schemes, the computation time is image dependent; a major problem for larger images. Also, although some of these filters are based on robust statistics (for instance the Tukey-based filter), they operate by retaining the outliers. Hence low amplitude white noise is removed provided the local contrast measures do not exceed the noise cutoff. For larger amplitude white noise, or impulsive noise spikes the filters are not as robust. Noise spikes with higher local gradients either diffuse too slowly or remain untouched by the filter. Low-pass filtering the original image before applying the anisotropic diffusion will remove unwanted noise spikes, however this has the drawback of introducing a new parameter into the filtering procedure and also, the accuracy of the anisotropic filter is diminished. The papers [6, 31] discuss this problem in more detail and describe a further solution whereby the diffusion function $c(x, y, s)$, is a function of the gradient of the image smoothed by a Gaussian kernel of variance σ, $c(x, y, s) = g(|\nabla(G_\sigma * I)|)$.

A more significant problem is that it is difficult to associate objects within an image to a particular scale. This will be discussed in more detail in Section 3.7, but briefly, the notion of scale changes with this implementation; scale is now associated with the number of filtering iterations rather than the dimensions of the filtering element. This creates problems when attempting to generate a scale-selection surface for the filter. The scale-space representations of the Figures 3.12 to 3.17 show that there are objects of different sizes at a particular scale iteration. Objects are not removed from the image at a set scale that corresponds to their size, their duration through the scale-space is instead a function of the local image gradients at those object locations. Section 3.7 will discuss this and other problems of scale-selection for all of the studied scale-space systems.

3.4 Morphology based scale-spaces

Mathematical morphology is a method of filtering signals, particularly images, with respect to shape and it is often associated with Serra [99, 100, 102] and Matheron [79] but also derives from the work of others such as Blum [24]. Morphological methods are widely used in commercial image processing packages where they appear as greyscale, or more usually, binary operations, such as erosions, dilations, opening, closing, tophat or watershed [82, 104] algorithms. Indeed, in many applications, the morphological filters perform better than the convolution-based alternatives because of the emphasis on shape matching.

Scale-related morphological operations, (where increasing size operators are applied to a signal), have been in use since the early days of mathematical morphology. For instance, Matheron [79] used binary image openings with scaled structuring elements to investigate the distribution of particle sizes within an image. Since then further developments have arisen in the form of multiple scale erosions with fixed-shape structuring elements [55], increasing scale sequences of open/closings or alternating sequential filters [106] with flat structuring elements [98] and alternating sequential filters with reconstruction, "reconstruction filters" [96, 97, 103]. Reconstruction filtering involves filtering with a fixed-shape structuring element. If image objects are small enough to fit within the structuring element they are removed through filtering. Objects that are too large to be removed will, instead, be distorted. This distortion is then subsequently *corrected* and the remaining objects are restored to their original state.

Matheron's technique, called *granulometry*, is based on observing the number of objects at a particular scale of filtering, with no position information retained and has since been used in further scale-related morphology, including the sieve algorithm described later. Granulometry has been used by [75, 93] to characterise pattern spectrums and analyse the shape-size content of a signal. The granulometry theory is also extended to greyscale analysis [34] and there are methods for choosing the relevant granulometry in a principled fashion [42]. These granulometry algorithms have the advantage of being particularly fast in one dimension [9, 10, 115].

The general field of morphology may be divided into three sub-fields, into which the majority of morphological filters will fall. Firstly, there is the more popular filtering with a rigid structuring element, where the signal is fitted to the surface of the structuring function. In terms of scale-related morphology, this simply implies filtering with an increasing size (but fixed shape) function. Secondly, there exists a newer approach which is based on non-rigid connected-sets. Here the signal is filtered with an emphasis on spatial size (length or area) rather than shape. In a scale-related system, the signal is simply filtered to increasing area. Lastly, the final sub-field is rank filters which are based on order-statistics. The signal is filtered according to some ordering scheme (mean, median etc.) over a window which can either be rigid or based on connected sets. Again, scale-related morphology involves increasing the size of the filtering window.

It is the first two of these fields that have received the most attention for extending the scale-related morphology into a scale-space morphology system. The rank filter approach suffers since it is difficult to filter over a rigid window in two or more dimensions and not introduce new extrema into the signal. However, it is possible to implement rank filtering over a connected-set and this is discussed briefly in a later section.

Many of the current scale-related morphological approaches do not take the causality constraint into account and they cannot therefore be classified as scale-space systems. In one dimension a multi-scale opening or closing operation may preserve scale-space causality [32] (with careful choice of structuring element) but this does not necessarily hold for higher dimensions [85]. Although in [32] Chen and Yan filtered binary images with openings by discs of scale equal to the disc radius, they were only concerned with the introduction of extrema over the contour of shapes within the images and not over the image as a whole. This work on *shape-based* scale-spaces started with work from Asada and Brady [7] and continued with Mokhtarian [83], Chen and Yan [32] and Jang and Chin [61], Chen and Yan having produced the first proof of any type of causal morphological multi-resolution system. However, this thesis is more concerned with the synthesis of *grey-level* based morphological scale-spaces, which developed along two paths; one involving filtering with *rigid* structuring elements and the other, filtering with *connected-sets*.

The two systems that are studied in the following sections emphasise these different approaches to scale-space. Multi-scale filters which use rigid greyscale structuring elements to match features [102, 106] are said to be a powerful analogue of linear matched filters, and the first filter discussed is described by Jackway [57, 59, 60] and van den Boomgaard [111] and is based on decomposition using a fixed structuring element with a parabolic shape. The second filter is based on the more recent advances in connected set filters [48, 112]. Alternating sequential filters that do not impose this rigid geometry on objects, and that use connected sets have been described in [97, 103, 109, 113, 114] and the *sieve* [14, 15, 17, 18], which implements an area-based decomposition, is one example of this different approach.

This first section provides a brief description of some basic morphological operations The definitions cover binary signals and images, and are extended to three-dimensional space for use in greyscale image filtering. A more in-depth analysis of these operations may be found in tutorial texts such as [35]. The definitions given here relate directly to the descriptions of the Boomguard/Jackway filters, and although, within the sieve definition, there are no fixed shape structuring elements, the basic morphological operations are still applicable.

3.4.1 Basic morphological operations

The most basic of morphological operations are erosions and dilations. These involve passing a structuring element along the signal and at every successive point, attempting to *fit* the signal to the structuring element. Secondary operations, such as openings and closings, may be defined in terms of erosions and dilations. The definitions of these operations (for binary signals) are described in terms of the *set* notation of Dougherty and Astola [35].

Erosion

The binary *erosion* of an image described by the set f, by a structuring element described by the set A, is denoted by $f \ominus A$ and is defined as

$$f \ominus A = \{u : A + u \subset f, u \in f\} \tag{3.29}$$

where $u : (A + u)$ indicates the *translation* $(A + u)$, of the structuring element by each pixel position (u) in the original image.

Treating A as a template, $f \ominus A$ consists of *all* the template origin positions for which the template, translated by u, fits inside f. This is the same as an intersection or AND'ing operation between the binary image and the translated template.

This definition is easily extended into greyscale erosion by taking the max in discrete space. The greyscale erosion of an image, $f(x)$, by a structuring element, $a(x)$, is defined as

$$f(x) \ominus a(x) = \min\{f(x+y) - a(y) \mid (y \in A)\} \tag{3.30}$$

In the binary case the translated template of the structuring element must be a spatial-subset of the original image. For greyscale images the structuring element must be *spatially*, and *vertically*, within the confines of the original image.

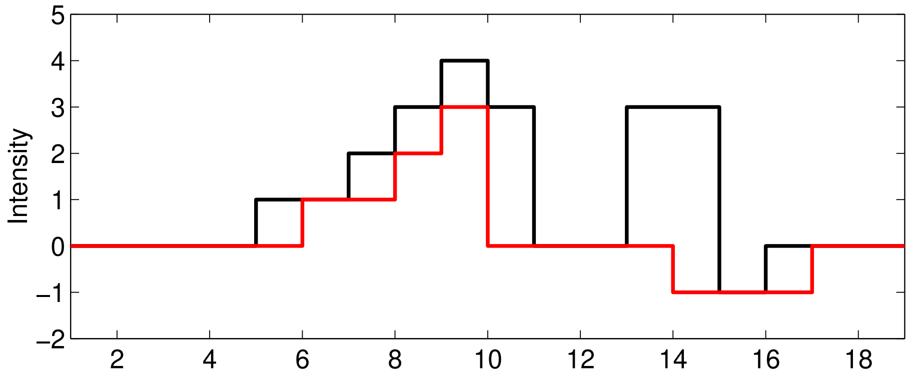


Figure 3.18: Example of one-dimensional erosion. Original signal in *black*, output after eroding with a flat 3-point structuring element is shown in *red*.

Figure 3.18 shows an example one-dimensional erosion. The input signal (in black) is eroded by a flat 3-point structuring element (origin in the centre) and the result is shown in red. The usual analogy of this operation is to imagine pushing the structuring element up to the underlying surface of the image, $f(x)$. The eroded image is represented by the path of the structuring element's origin point as the structuring element is pushed up as far as possible into the image.

Dilation

The binary *dilation* operation is the direct complement of the binary erosion operation. Given the signal described by the set f, and a structuring element described by the set A, the opening of f by A is denoted by $f \oplus A$ and is defined as

$$f \oplus A = \{A + u : u \in f\} \tag{3.31}$$

If the template of the structuring element, A, is translated to all points of the input image, f, the union of these translates make up $f \oplus A$. The same result is achieved through an OR'ing operation between the binary image and the translated template.

Again, this definition is extended to greyscale images by taking vertical translation into account. In discrete space the dilation of the image, $f(x)$, by the structuring element, $a(x)$ is achieved through a $\min$ operation,

$$f(x) \oplus a(x) = \max\{f(x-y) + a(y) \mid (y \in A)\} \tag{3.32}$$

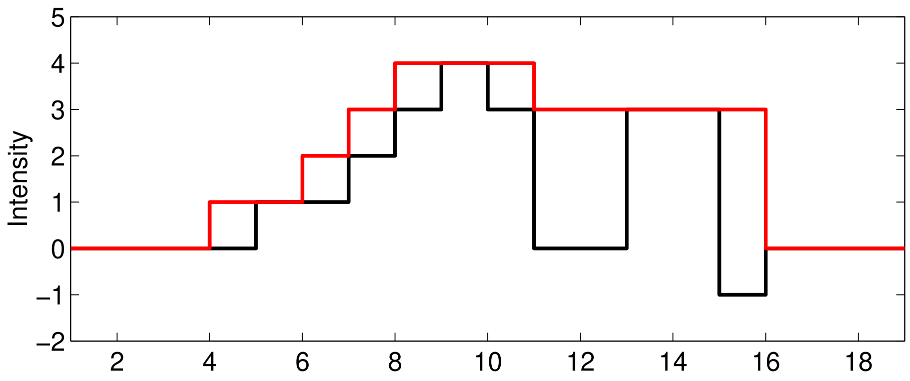


Figure 3.19: Example of one-dimensional dilation. Original signal in *black*, output after dilating with a flat 3-point structuring element is shown in *red*.

Figure 3.19 shows an example one-dimensional dilation. The input signal (in black) is dilated by a flat 3-point structuring element (origin in the centre) and the result is shown in red. The simplest description of the greyscale dilation operation is to imagine the reflected structuring element, $a(x)$, being pressed down onto the above-surface of the image, $f(x)$. The dilated image is represented by the path of the structuring element's origin point as the structuring element is pushed-down as far as possible into the image.

Opening

In practice, erosions and dilations tend to be applied in pairs. Filtering a signal using purely erosions or purely dilations produces a large bias in the filtered output. This bias, towards signal maxima or minima, is caused by repeated min or max operations.

Following an erosion by a dilation (or visa versa) helps to reduce this bias slightly and create a more robust filter output. An *opening* is one such secondary morphological operation. The opening of a greyscale signal, $f(x)$, by a structuring element, $a(x)$, is denoted by $\psi_{a(x)}f(x)$, and is defined as an iteration of erosion followed by dilation,

$$\psi_{a(x)}f(x) = (f(x) \ominus a(x)) \oplus a(x) \tag{3.33}$$

Different notations for an opening operation include the following: $f \circ a$ and $O(f, a)$.

For a binary image, an opening involves taking the union of all translations of the structuring element that fit inside the input image, f. The result of iteratively applied openings is the elimination of image detail that is smaller than the structuring element. For instance, the opening of an image with a disc-shaped structuring element will eliminate small positive-going image features and smooth object edges and contours. There is still a bias in the filtering process (removing only positive-going features), however it is possible to further reduce this bias through more complex filtering iterations that will be described later. Figure 3.20 shows an example one-dimensional opening. The input signal (in black) is opened by a flat 3-point structuring element (origin in the centre) and the result is shown in red. Note that all positive going extrema with width less than three are removed.

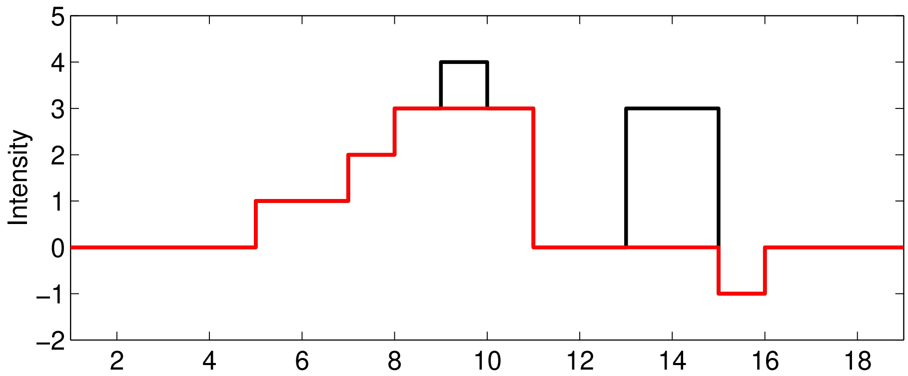


Figure 3.20: Example of one-dimensional opening. Original signal in *black*, output after opening with a flat 3-point structuring element is shown in *red*.

Closing

The closing operation is the dual of the opening operation and is defined as an iteration of dilation followed by erosion. The closing of the image, $f(x)$, by the structuring element, $a(x)$, is denoted by $\gamma_{a(x)}f(x)$ and is described by,

$$\gamma_{a(x)}f(x) = (f(x) \oplus a(x)) \ominus a(x) \tag{3.34}$$

Figure 3.21 shows an example one-dimensional closing. The input signal (in black) is closed by a flat 3-point structuring element (origin in the centre) and the result is shown in red. Here all negative-going extrema with widths less than three points are removed.

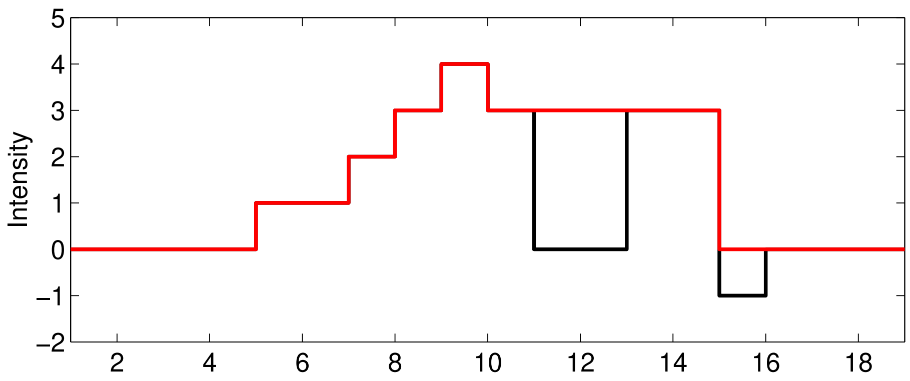


Figure 3.21: Example of one-dimensional closing. Original signal in *black*, output after closing with a flat 3-point structuring element is shown in *red*.

Using the disc example from the openings description, closing an image with a disc-shaped structuring element will eliminate negative-going features that are smaller than the structuring element and smooth image edges and contours.

Alternating sequential filters

An obvious solution to reducing the opening and closing filters' respective bias towards maxima and minima is to filter by alternating the two operators. These third-level operations are referred to as *Alternating filters*, or *Alternating Sequential filters* [101, 106] when applied repeatedly with increasing size structuring elements.

The $\mathcal{M}$ filter is defined as an opening followed by a closing operation

$$\mathcal{M}_{a(x)}f(x) = \gamma_{a(x)}(\psi_{a(x)}f(x)) \tag{3.35}$$

This has the effect of removing both positive and negative going extrema that are smaller than the structuring element.

Its dual operation is the $\mathcal{N}$ filter, defined as a closing followed by an opening operation

$$\mathcal{N}_{a(x)}f(x) = \psi_{a(x)}(\gamma_{a(x)}f(x)) \tag{3.36}$$

Again, this removes both minima and maxima that are smaller than the structuring element. However because of the differing orders of filter application, ($\mathcal{M} = (\min, \max,$-$\max, \min)$ and $\mathcal{N} = (\max, \min, \min, \max)$), the resulting $\mathcal{M}$ and $\mathcal{M}$ filter outputs still retain a small bias towards either signal minima or maxima.

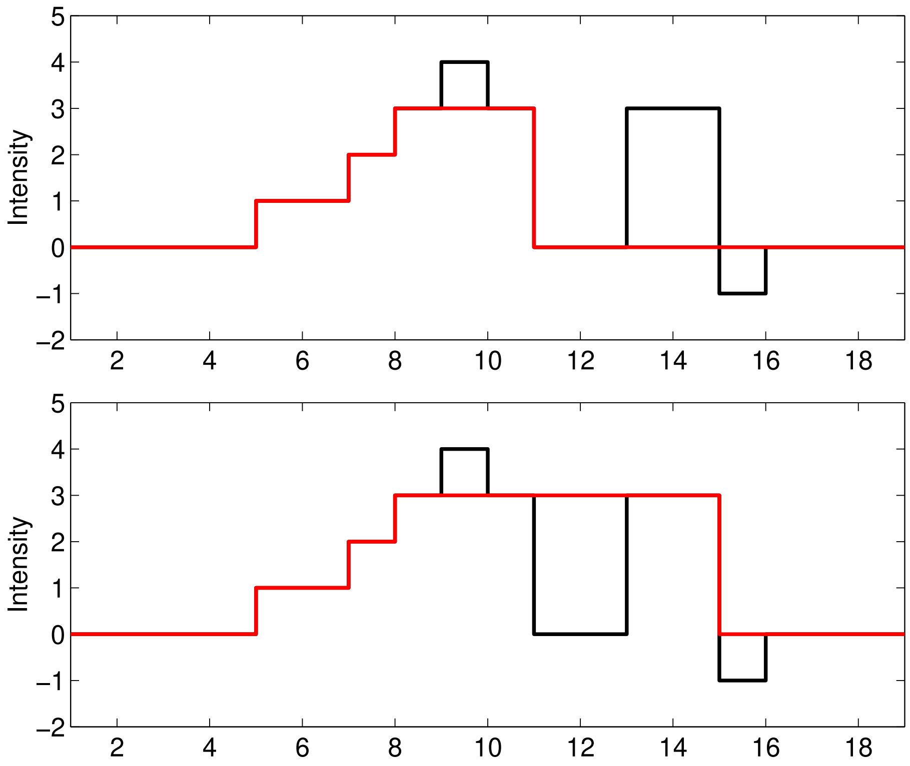


Figure 3.22: Example of one-dimensional $\mathcal{M}-$ and $\mathcal{N}-$filtering. Original signal in *black*, top panel shows output after $\mathcal{M}-$filtering and bottom panel shows output after $\mathcal{N}-$filtering with a flat 3-point structuring element.

Figure 3.22 shows an example of one-dimensional $\mathcal{M}$ and $\mathcal{N}$ filtering. As before the input signal (in black) is filtered with a flat 3-point structuring element (origin in the centre). The above panel shows the result of the $\mathcal{M}-$filter output and the bottom panel shows the $\mathcal{N}-$filter output. The two operations do not give the same output because of the small extrema bias that exists for these filters.

The repeated application of these alternating operators with increasing size structuring elements will remove larger and larger features in the filtered image.

Rank filters

If an image is badly corrupted with noise, it may be difficult to segment features within the image from the noise. In this situation the simpler morphological operations or the alternating sequential filters may not give the best results. A separate stream of development has been that of rank filters. Rank-order methods aim at estimating the grey value of a pixel from the values of the surrounding pixels that fit within some window. The simplest estimate would be to return the average over the window, however, this linear filter blurs sharp edges within an image. This blurring is avoided if the median value over the window is returned. Median filters and their derivatives are commonly used techniques in image processing, variations include the median [108], root median [40], recursive median [81] and, more generally, stack filters [33].

These types of filters were developed primarily to remove random noise from signals and images, however, there are suggestions for using them for tasks such as shape recognition [95]. Although the nature of a window-based filter is to ignore shape and form within an image, there is a close relationship between rank filters and morphological filters [76].

Figure 3.23 shows an example one-dimensional median filter. The input signal (in black) is filtered by a $3-$point median window (origin in the centre) and the result is shown in red. A median filter of window length N will remove extrema of length $(N-1)/2$, where N is odd.

3.4.2 Morphological filter properties

Some of the properties that are used to describe the morphological filters are defined below in terms of a general morphological operator $\mathcal{Q}$.

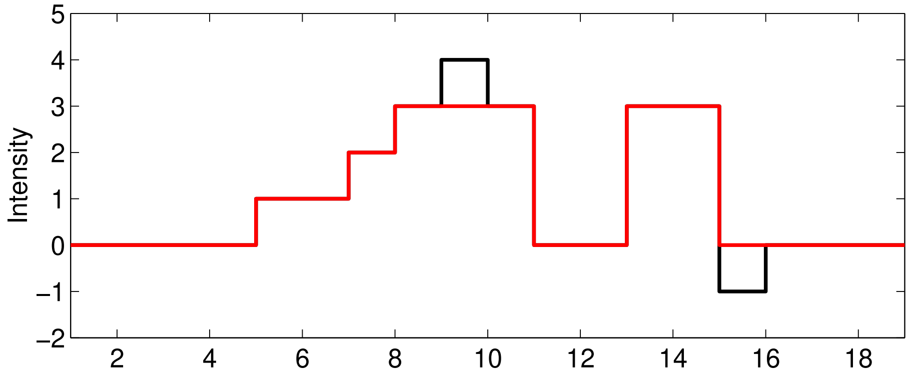


Figure 3.23: Example of one-dimensional median filtering. Original signal in *black*, output after filtering with a 3-point median window is shown in *red*.

- An operator $\mathcal{Q}$ is *idempotent* if $\mathcal{Q}(f) = \mathcal{Q}(\mathcal{Q}(f))$. In words, an operator is idempotent if applying the operator to the signal repeatedly produces no further change. Opening, closing, $\mathcal{M}-$ and $\mathcal{N}-$filters are all idempotent.

- The *dual* of the operation $\mathcal{Q}(f)$ is defined as $-\mathcal{Q}(-f)$. Erosion is the dual of dilation, which means that erosion may be expressed in terms of dilation operations, and dilation may be expressed in terms of erosion operations. Similarly, closing operations are the dual of openings and $\mathcal{M}$ operations are the dual of $\mathcal{N}$ operations. If $\mathcal{Q}(f) = -\mathcal{Q}(-f)$ then the operator is *self dual*

- An operator $\mathcal{Q}$ is *anti-extensive* if $\mathcal{Q}(f) \leq f$ and is *extensive* if $f \leq \mathcal{Q}(f)$. If the filtered output is contained by (or is a subset of) the original signal then the operator is anti-extensive. If the original signal is contained by the filter output, then the operator is extensive. The opening operation is anti-extensive and the closing operation is extensive.

- An operator $\mathcal{Q}$ is *monotonically increasing* if it preserves the order of signals such that if $f_1 \leq f_2$, then $\mathcal{Q}(f_1) \leq \mathcal{Q}(f_2)$. For a fixed-shape structuring element erosion, dilation, opening and closing are monotonically increasing. This property is central to designs involving multiple erosions (or dilations); if, instead of two signals there is one fixed signal, f, and two operators where $\mathcal{Q}_1 \leq \mathcal{Q}_2$, then $\mathcal{Q}_1(f) \leq \mathcal{Q}_2(f)$ for monotonic increasing systems.

3.5 Morphology using poweroids

As with linear systems there are several schemes for the morphological simplification of images but only a few that obey the scale-space causality principle, and, as with Gaussian filtering, schemes that produce scale-spaces from one-dimensional signals may need rather careful extension to more than one dimension [57].

Figure 3.24 shows a particular structuring function and its point of origin. For the primary erosion and dilation morphological operations the output is defined as the path of the origin point as the structuring element is pushed into the the surface of the image. In the case of erosion the element is pushed up to the underside of the image and in the case of dilation the element is reflected in the xy plane and then pushed down onto the above surface of the image. In either case there is no vertical or horizontal translation involved. Offsets in the filter output will only occur when the origin point of the structuring element is changed. For a scaled structuring element these offsets may be scale dependent and can cause problems in tracking image features between scale outputs. For this reason there are no vertical or horizontal offsets in the filtering operations described in this thesis.

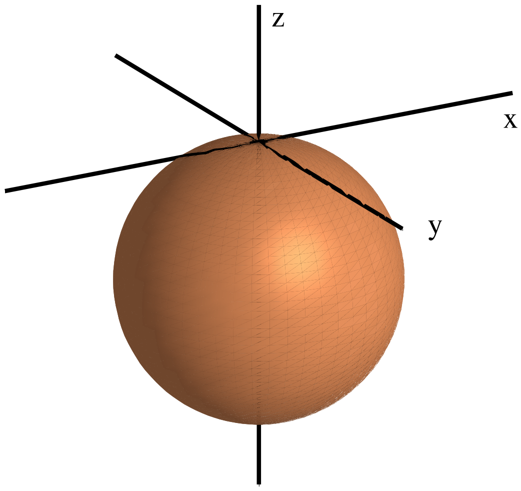


Figure 3.24: Point of origin of a structuring element.

A simple example of erosion and dilation in one-dimension is shown in Figure 3.25. The input signal is a scan line from a real image and the structuring elements are one-dimensional poweroid functions. The details of the structuring function will be described later, the figure simply shows the effect of the erosion and dilation operation on the scan line. The blue line shows the result of erosion with a function of scale 0.1. It smoothes the scan line slightly and removes small-scale extrema (biasing the signal towards maxima). The red line shows the result of dilation with the same shape structuring element, this time of scale 0.5. This larger function causes more smoothing and reduces a higher number of large-scale extrema (biasing the signal towards minima). Once removed these extrema will not reappear at larger scales. In the limit, as scale approaches zero, the original signal is recovered and as scale approaches ∞ the output is a flat line.

The figure shows that the number of extrema in a signal is reduced as the signal is filtered to increasing scales. It is therefore reasonable to consider this morphological method when building a scale-space system.

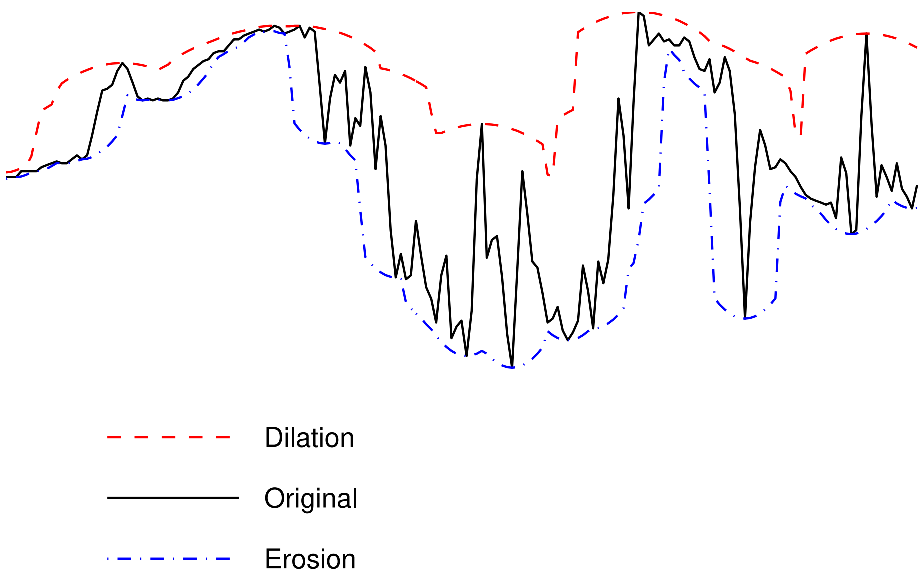


Figure 3.25: Example of one-dimensional erosion and dilation with general poweroid functions of scale 0.1 and 0.5 respectively

There are types of scaled two-dimensional structuring elements, such as flat functions, spherical functions and poweroid functions, that may be used in morphological filtering but only a few have been shown to be of use in a scale-space based approach. Jackway [57] and van den Boomguard [111] have developed a morphological filter based on erosions and dilations with fixed circular-poweroid structuring elements.

The general equation of this two-dimensional poweroid function, as defined by [57], is given in (3.37) below.

$$g_s(x,y) = -|s| \left(\frac{\sqrt{x^2+y^2}}{|s|} \right)^{\alpha} \qquad \alpha \geq 0,\ s \neq 0. \tag{3.37}$$

The variable s is the *scale* of the element, however, it is α that determines the overall shape of the function. Examples of various values of alpha are shown in Figure 3.26 For $\alpha = 1$ the structuring element is a cone, for $\alpha = 2$ a paraboloid, for $\alpha = 4$ a quartoid and for $\alpha = \infty$ a cylinder.

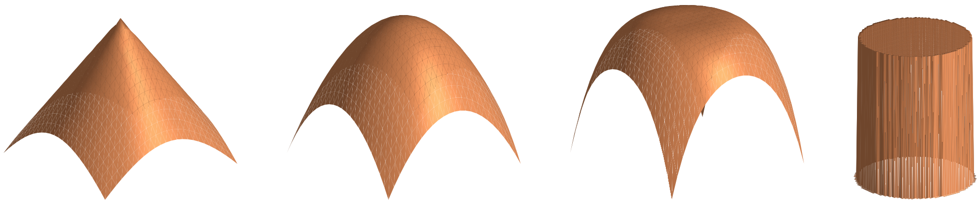


Figure 3.26: Parabolic structuring elements (3.37) with different values of α. From left to right: $\alpha = 1$ for a cone, $\alpha = 2$ for a paraboloid, $\alpha = 4$ for a quartoid and $\alpha = \infty$ for a cylinder.

This section discusses only erosions and dilations. Jackway discusses and provides a full proof of the causality of the morphological erosion and dilation operations in [55,57,59,60] and also discusses and makes corrections on the previous causality proofs of Chen and Yan [32], and Nacken [85] in [58].

Of the choices of α in (3.37) and from Figure 3.26, it is observed that, for any fixed scale, the more “pointed” (low α) structuring elements tend to give more emphasis to the local shape of the original signal. Filtering with flat structuring elements, such as cylinders ($\alpha = \infty$), gives rise to problems in localisation. The filtered greyscale image contains large flat regions imposed onto the signal by the structuring element and the position and shape of the local extrema are distorted. This creates difficulties in linking extrema at different scales, making it unreliable as a scale-space system.

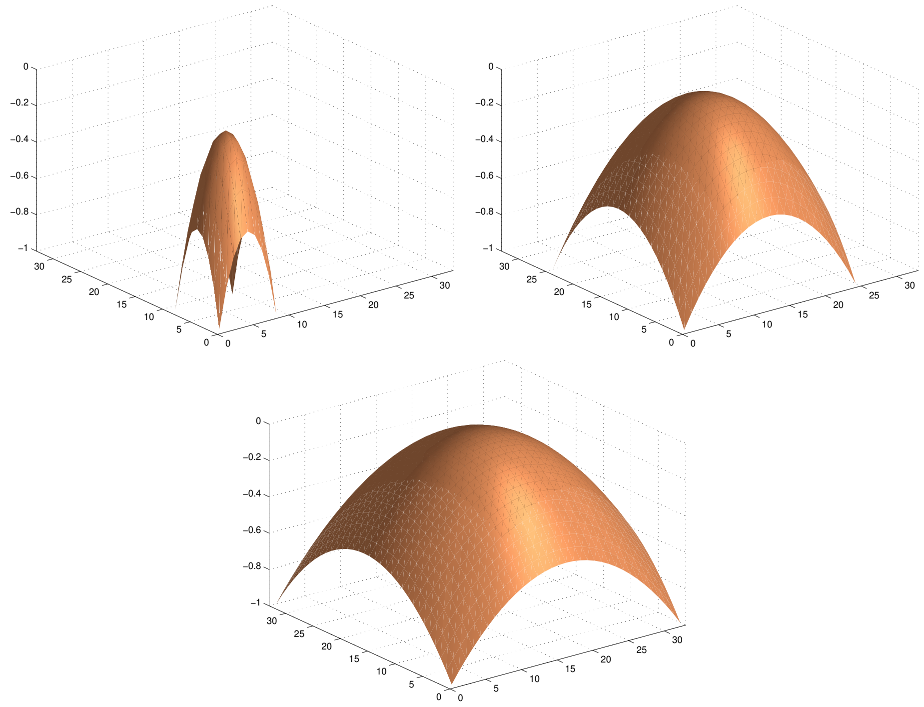


Figure 3.27: Paraboloid structuring elements of scale 0.05, 0.5 and 1.0

Setting $\alpha = 2$, produces the paraboloid function (3.38). Figure 3.27 shows three examples of this paraboloid, at scales $s = 0.05$, 0.5 and 1.0. As scale increases, so does the volume of the paraboloid.

$$g_s(x, y) = -\frac{(x^2 + y^2)}{|s|} \tag{3.38}$$

Jackway shows that these functions are advantageous because of (i) their second order characteristics (the point of maximum curvature is found at the origin), (ii) morphological erosions and dilations from this function obey the semi-group property which allows scale representations to be calculated from previous smaller scale outputs

$$I \ominus g_{s+u} = (I \ominus g_u) \ominus g_s$$

$$I \oplus g_{s+u} = (I \oplus g_u) \oplus g_s$$

and (iii) these functions are computationally less intensive than the alternatives since the paraboloid is separable. A two-dimensional filtering operation can therefore be performed more efficiently as two one-dimensional operations.

Figure 3.28 shows a synthetic image containing four impulses that is eroded and dilated by a paraboloid function. The effect of the erosion is to suppress all maxima that are smaller than the element and to blur the minima by imposing the shape of the structuring element onto the signal. Vice versa for the dilation operation.

Since the combined erosion-dilation (opening) or dilation-erosion (closing) operations have not yet been fully proven to obey causality [57], filtering to a scale s produces two images, one erosion and one dilation, leading to a different scale-space structure.

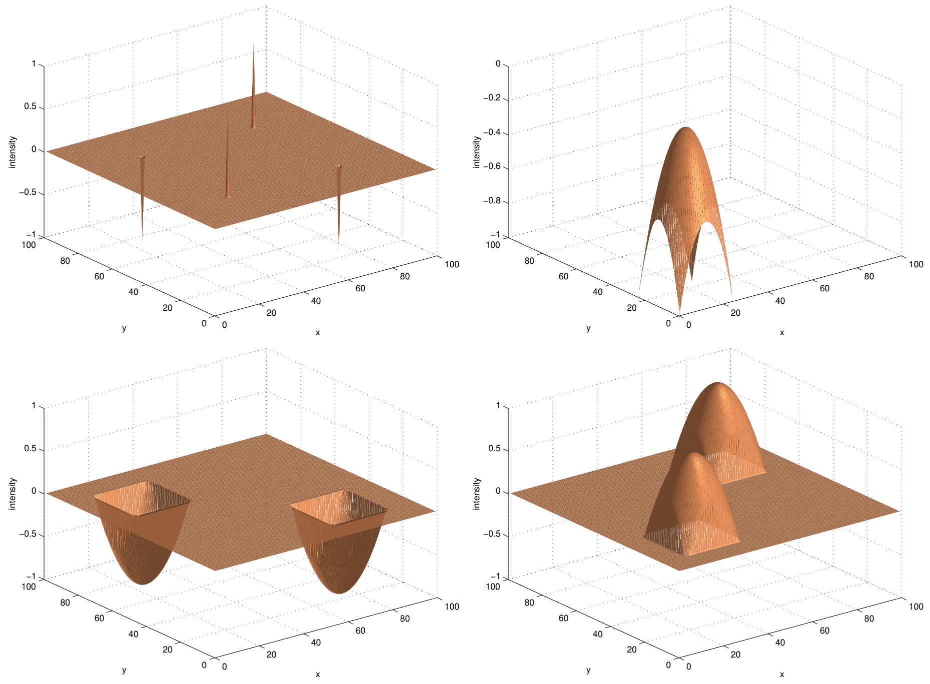


Figure 3.28: Example two-dimensional erosion and dilation filtering. Original image (top left), structuring element (scale 0.5) (top right), result of erosion (bottom left), result of dilation (bottom right)

The major differences between the scale-space structure used by both Jackway and van den Boomgaard [111], and the previous scale-spaces that originate from the work of Witkin [118], is that the morphological erosion-dilation scale-space describes the signal at both negative and positive scales. The diffusion scale-space is based on the premise that the scale zero image is the original signal and the positive scale representations of this image are the result of diffusion to a particular scale s. In the case of morphological filtering the dual erosion and dilation operations are used to create the scale-space, so filtering to scale s is slightly ambiguous, involving either an erosion or a dilation of scale s. The scale-space is therefore divided into three regions, where the image at scale, s, is calculated as

$$I_s(m,n) = \begin{cases} (I \oplus g_s)\,(m,n) & s > 0 \\ I(m,n) & s = 0 \\ (I \ominus g_s)\,(m,n) & s < 0 \end{cases} \tag{3.39}$$

Thus the input signal I is subject to either a greyscale dilation for positive scales or a greyscale erosion for negative scales. In [57] the *non-negative* aspect of the Gaussian scale-space is described as a "limitation", however, having to combine the information from the pairs of images that correspond to each scale of this erosion-dilation scale-space is in itself problematic.

3.5.1 Boundary conditions

The input signal is assumed to be $-\infty$ outside its boundaries, this region is described as the *unidentified* signal region in [35]. For erosion operations the structuring element must fit entirely within the spatial confines of the original signal, this results in a narrowing of the output signal after filtering (shown in Figure 3.29). The degree of narrowing is equal to the horizontal distance between the origin of the structuring element and its outermost point in the filtering direction.

For dilation operations the filtered signal is wider than the original, however it is usual to show only the signal-region that was occupied by the original signal. The mirrored structuring element is 'pressed' to the above surface of the input signal and at the boundary points the filtered signal is wider by a length equal to the horizontal distance between the structuring elements origin and its edge.

3.5.2 Properties of the paraboloid scale-space

This section lists the scale-space properties of Section 2.3 that are held by the paraboloid erosion-dilation scale-space. The system also holds many of the various morphological properties as defined in Section 3.4.2. The erosion operation is the *dual* of the dilation operation, erosion is *extensive* while dilation is *anti-extensive* and both operations are *monotonic increasing*. The paraboloid filtering operation is not, however, idempotent; filtering a signal twice with the same structuring element will produce two different outputs. The filter, instead, obeys the *semi-group* criterion (3.4) also mentioned earlier.

- **Causality:** In [55,57] the causality of the paraboloid erosion-dilation scale-space is proven such that given the local set of maxima and minima,

$$E_\sigma^{max} = \{t : I_\sigma(t) \text{ is a local maximum}\} \tag{3.40}$$

$$E_\sigma^{min} = \{t : I_\sigma(t) \text{ is a local minimum}\} \tag{3.41}$$

  then, for all

$$\sigma_1 > 0, \quad \text{if } \sigma_2 > \sigma_1 \quad \text{then } E_{\sigma_2}^{max} \subseteq E_{\sigma_1}^{max} \tag{3.42}$$

  and likewise, for all

$$\sigma_1 < 0, \quad \text{if } \sigma_2 < \sigma_1 \quad \text{then } E_{\sigma_2}^{min} \subseteq E_{\sigma_1}^{min} \tag{3.43}$$

3.5.3 Example paraboloid scale-spaces

Figure 3.29 and Figure 3.30 show examples of the scale-space created from a real one-dimensional signal taken as a scan-line from an image. Figure 3.29 shows the signal eroded by one-dimensional paraboloid structuring elements (3.38) of scale $s = -0.05$, -0.2, -0.5,-2 and -4. The original signal is assumed to be at $-\infty$ at the edges and so the eroded output is narrowed as the scale of the structuring element increases. As expected the erosion removes increasing scale maxima. Figure 3.30 shows the corresponding positive scale-space signals, filtered by one-dimensional paraboloid structuring elements (3.38) of scale $s = 0.05$, 0.2, 0.5, 2 and 4. Here the minima of the signal are removed at increasing scales causing the overall amplitude of the filtered signal to increase.

These results are more easily observed in two dimensions, where problems with the filtering scheme also become apparent. Figure 3.31 and Figure 3.32 show four examples

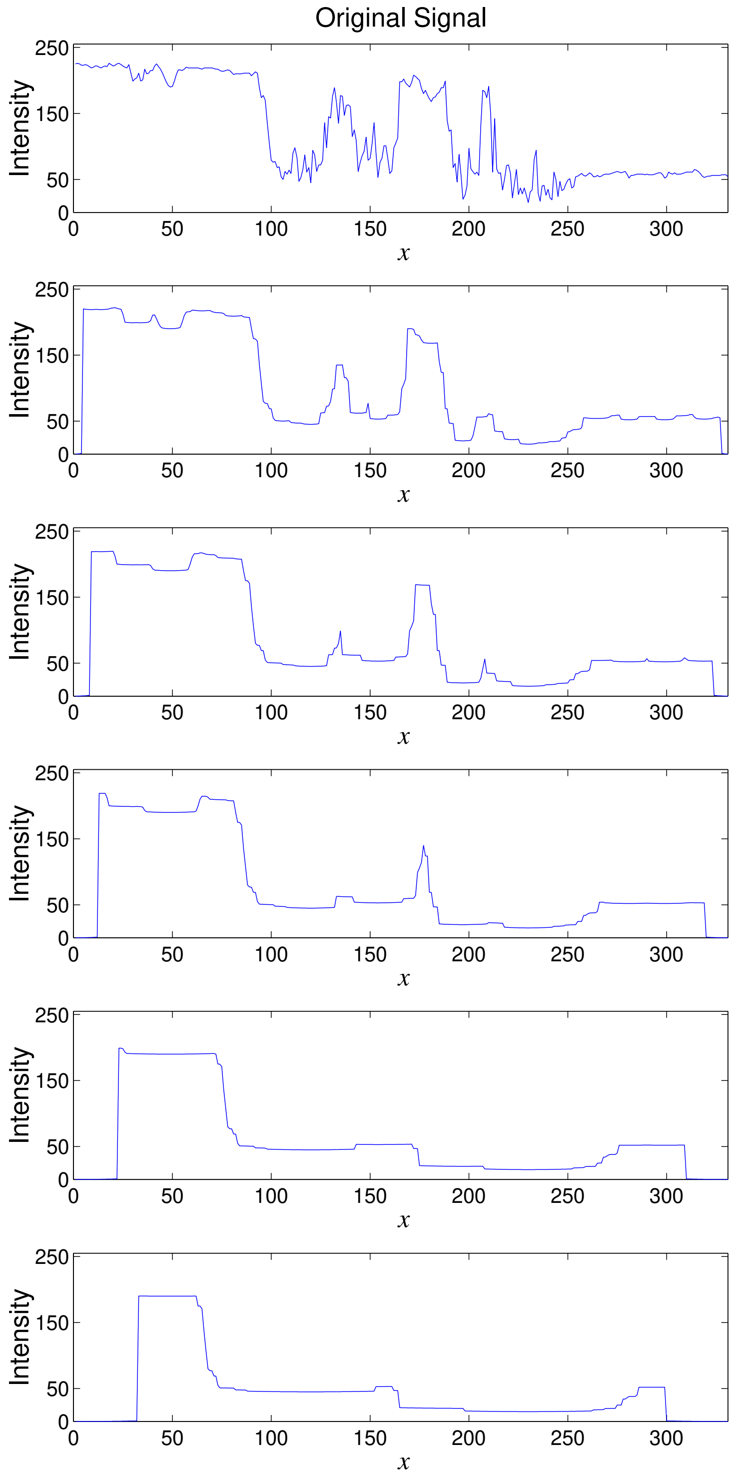


Figure 3.29: Scan line (top panel) simplified via erosion with a one-dimensional paraboloid function (3.38); Scales -0.05, -0.2, -0.5, -2 and -4 are shown.

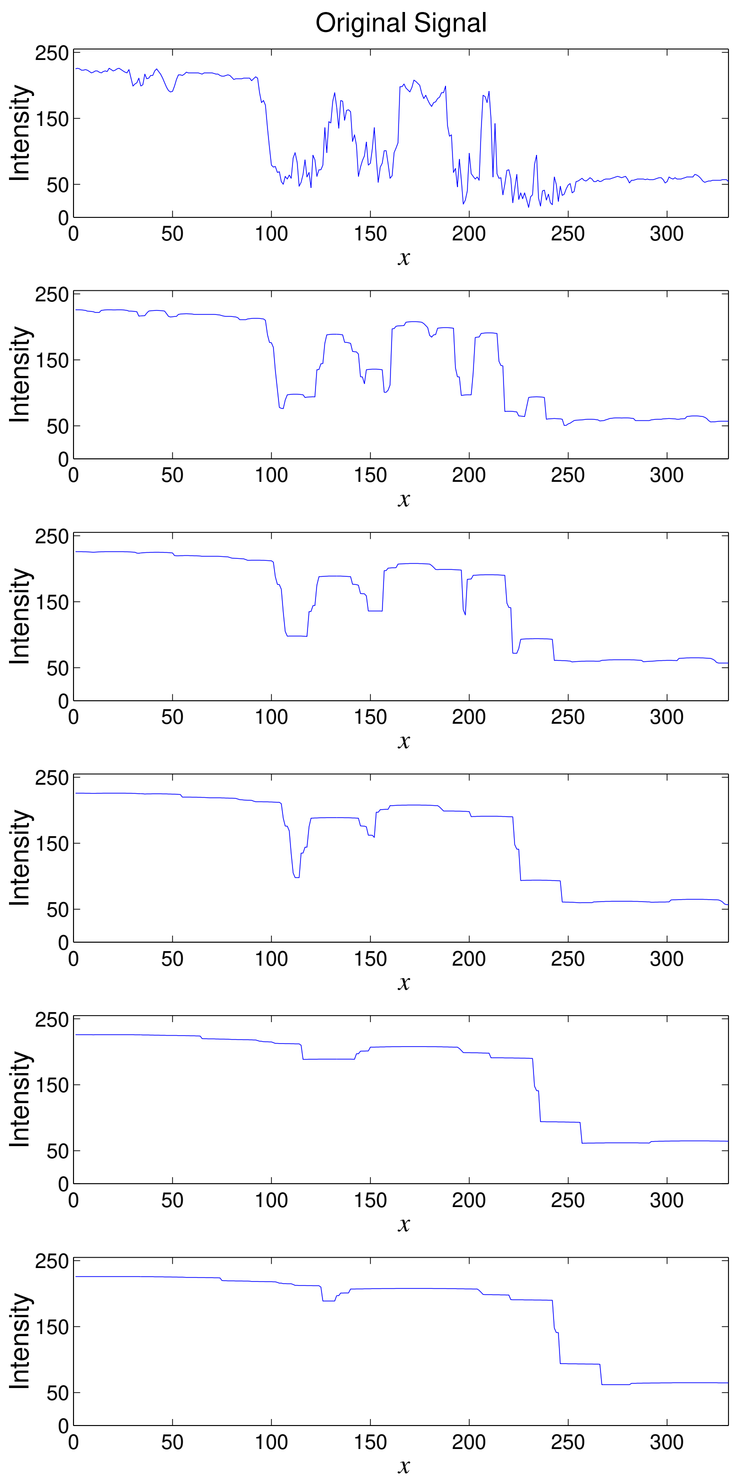


Figure 3.30: Scan line (top panel) simplified via dilation with a one-dimensional paraboloid function (3.38); Scales 0.05, 0.2, 0.5, 2 and 4 are shown.

of the test image smoothed using increasing scale paraboloid structuring elements of equation (3.38). Figure 3.31 shows the original image in the top panel, followed by the image eroded to scales $s = -0.05$, -0.15, -0.35 and $s = -1.0$ using equation (3.39). As scale increases the images grow darker as the erosion causes the maxima regions of the images to be removed. For instance the railing disappears at scale -0.15 and the clouds are gradually eroded. One obvious problem here is that the erosions cause considerable distortions. The image is *fitted* to the curved surface of the structuring element, giving the filtered images a *bubble* effect. This causes considerable problems in the analysis of the scale-space.

Figure 3.32 shows the corresponding positive-scale (dilation) outputs and these too show the bubble effect of the paraboloids. Increasing size minima are removed by increasing scale structuring functions so that, for example, the walking couple are completely removed by scale 1. Since the dilation operation removes the minima from the signal the images become brighter for increasing scale.

The main problems associated with this system are apparent in the example figures. Firstly, the scale-space is divided into positive and negative scales, with the original images as the zero scale representation. The positive scales represent the maxima region information and the negative scales represent the minima regions, and since regions of importance can not necessarily be segmented into purely maxima or minima regions, results have to be fused from the two representations. Secondly, since erosions and dilations are bias towards minima and maxima, the operations are very sensitive to noise spikes. A solution to this would be to consider a scale-space based on the less-sensitive open and close operations, however this is not yet proven as a causal system [56, 57]. Thirdly, the bubble-effect distortion in the images causes problems in tracking features through the scale-space and, lastly, there is, as yet, no method to associate given image features with a particular scale, in other words there is no scale-selection surface for this filter. This last point will be discussed again in Section 3.7.

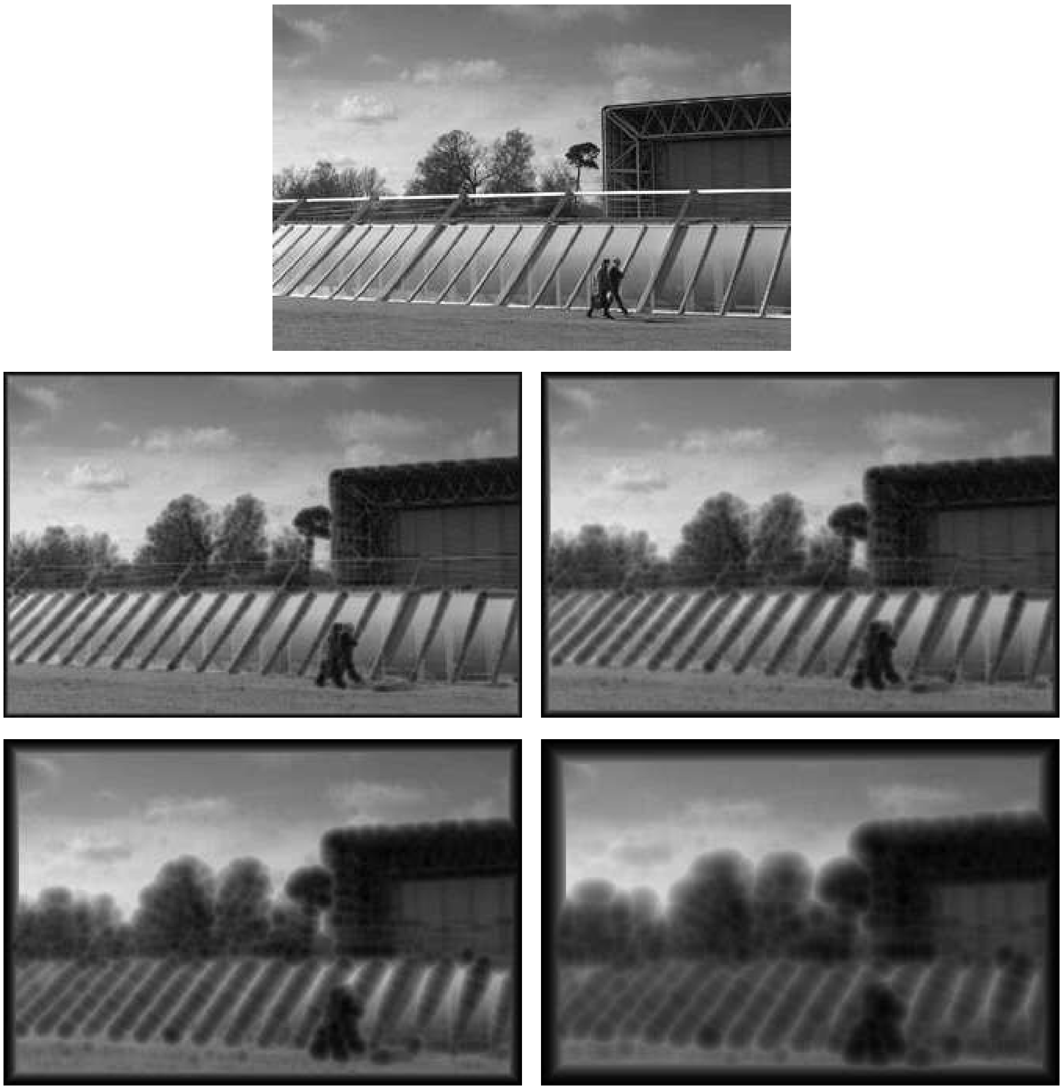


Figure 3.31: Images simplified via erosion with a parabolic structuring element of scale -0.05, -0.15, -0.35 and -1.0

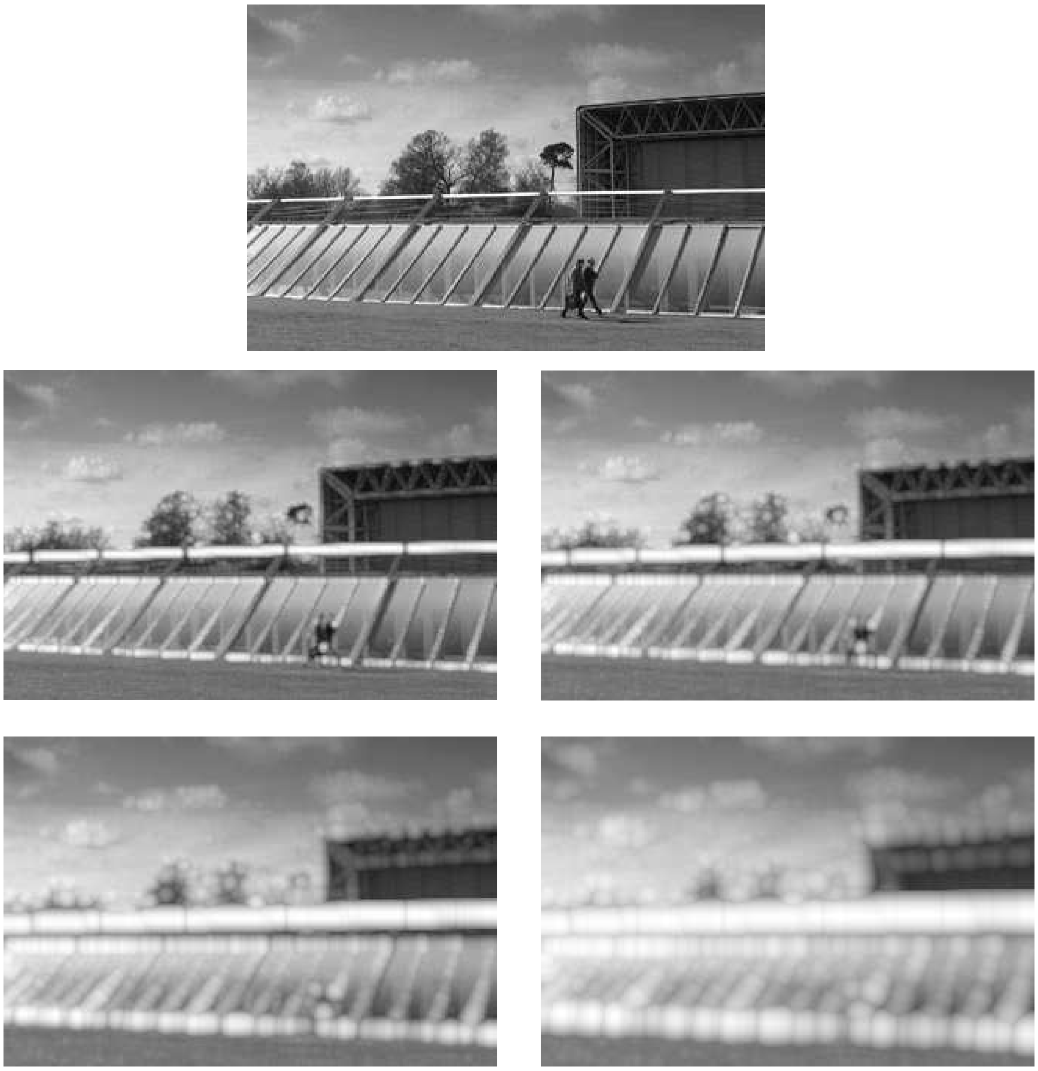


Figure 3.32: Images simplified via dilation with a parabolic structuring element of scale 0.05, 0.15, 0.35 and 1.0

3.6 The sieve

The sieve is a non-linear decomposition algorithm with applications in low-level signal and image processing. It uses morphological operations to remove increasing scale detail from a signal, in any number of dimensions, without destroying the basic structure of the signal. This signal simplification is causal and so the sieve satisfies the principle axiom of scale-space theory. A detailed mathematical description of the sieve and its theorems is given elsewhere [15], this section provides a basic description of the algorithm and its properties along with examples in one and two dimensions.

The sieve works by applying a flat structuring element to an input signal but this structuring element is different to those described in the previous section because it has a fixed size but a varying shape. This means that the sieve decomposes a signal based on size: sample length in one-dimensions, pixel area in two-dimensions, voxel volume in three-dimensions and so on. The significant feature of the sieve is that the filtering operations are designed to remove extrema of a set size or *scale* rather than a set shape.

In the two-dimensional examples of Section 3.5.3 it was shown that filtering with a fixed structuring element deforms the output image since the shape of the structuring element is imposed onto the filtered image. The sieve operates differently, using a *flexible* structuring element that fits to the shape of the greyscale image. The usual explanation for the sieve decomposition is based on *Graph theory* [48,65,112] where the input signal is represented as a graph, $G = (V, E)$. The vertices or the set of elements of the graph are described by V and the set of edges, E, describes the connectivity of these elements. It is impossible to perform a *true* erosion or dilation operation over the graph, however it is possible to perform redefined opening, closing, $\mathcal{M}$ and $\mathcal{N}$ operations which will be described later.

Through the use of graph notation it is possible to describe the sieve in any number of dimensions. In one dimension the graph notation of a signal results in a list. For the example in Figure 3.33 the graph (G_{1D}) vertices are the element labels $V = \{1, 2, 3, ..., 10, 11\}$, and the set of edges, E, store the connectivity information based on two-connected samples, $E = \{\{1, 2\}, \{2, 3\}, ..., \{10, 11\}\}$,

This notation is easily extended into two dimensions as shown in Figure 3.34. The vertices of the graph are now the pixel elements, and the edges, E, represent the four-connected neighbourhoods of each pixel. The new graph representation, $G_{2D} = (V, E)$,

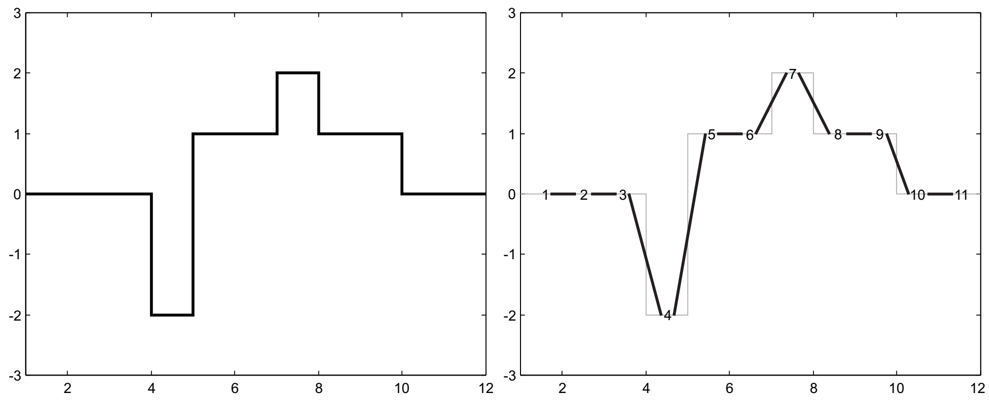


Figure 3.33: Graph representation in one dimension (G_{1D}), left: original one-dimension signal, right: vertex labels (V), and edges (E).

is composed of

$$V = \{1, 2, 3, ..., 24, 25\} \tag{3.44}$$

and

$$E = \{\{1, 2\}, \{1, 6\}, \{2, 3\}, ..., \{20, 25\}, \{24, 25\}\} \tag{3.45}$$

where the image intensities are defined as a function, f, of the vertex number, $f(v)\ \ v \in V$,

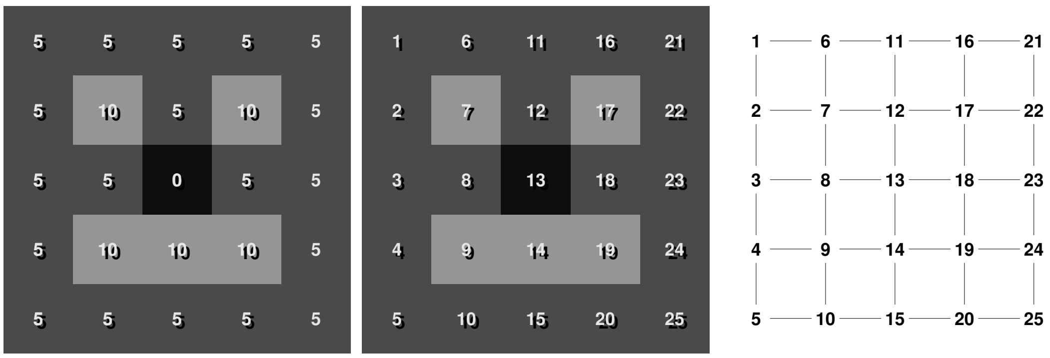


Figure 3.34: Graph representation in two dimensions (G_{2D}), left: original two-dimensional signal showing pixel intensities, $f(v)\ \ v \in V$, middle: vertex labels ($v \in V$), right: edge labels (E).

The sieve filter performs a *decomposition by area* which is seeded at the extrema points of the original signal. The sieve removes these signal peaks, starting at scale $s = 1$, (maxima or minima of length one unit or area one pixel) and increasing to the maximum scale, m. This is represented in one dimension by the length of the input signal and in two dimensions by the pixel area of the input image.

The signal peaks are removed by applying morphological operations over the graph. Defining $C_r(G)$ as the set of connected subsets of r elements, within G, allows the definition of $C_r(G, v)$ as those elements of $C_r(G)$ that contain the vertex v.

$$C_r(G, v) = \{\xi \in C_r(G) | v \in \xi\} \tag{3.46}$$

This may be explained in simpler terms as the set of connected regions of size r that contain the vertex v. For example, in the one-dimension signal of Figure 3.33, the set of regions of size 2 elements that contain the vertex 7 is

$$C_2(G_{1D}, 7) = \{\{6, 7\}, \{7, 8\}\} \tag{3.47}$$

similarly, for the two-dimensional, four-connected graph of Figure 3.34

$$C_2(G_{2D}, 7) = \{\{2, 7\}, \{6, 7\}\{7, 12\}, \{7, 8\}\} \tag{3.48}$$

The removal of extrema over the graph is achieved through the application of a morphological operator $\mathcal{Q}_s$ of scale, s, over $C_r(G)$. $\mathcal{Q}$ may be any of the opening, closing, $\mathcal{M}$ or $\mathcal{N}$ operators or, alternatively, $\mathcal{Q}$ may be a *recursive median* sieve operator (ρ), defined later.

The redefined, $N-$dimensional, graph-based opening and closing operations are given in equations (3.49) and (3.50) respectively. They are simply ordered $\min$ and $\max$ operations. The graph-based $\mathcal{M}$ (3.51) and $\mathcal{N}$ (3.52) operations are also shown; these remain as ordered opening and closing functions.

$$\begin{aligned}
\psi_r f(x) &= \max_{\xi \in C_r(G,x)} (\min_{u \in \xi} f(u)) && (3.49)\\
\gamma_r f(x) &= \min_{\xi \in C_r(G,x)} (\max_{u \in \xi} f(u)) && (3.50)\\
\mathcal{M}_r f(x) &= \gamma_r(\psi_r(f(x))) && (3.51)\\
\mathcal{N}_r f(x) &= \psi_r(\gamma_r(f(x))) && (3.52)
\end{aligned}$$

For the opening, closing, $\mathcal{M}$ and $\mathcal{N}$ sieve operators the relationship between the region-size subscript r, and the scale subscript s, is $s = r - 1$. So, features of scale s may be removed via a sieve of region-size $s + 1$ elements.

For instance, the effect of applying an *opening* of scale $s = 1$, ($\mathcal{Q}_1 = \psi_2$), to any signal would be the removal of all maxima of unit length or area. (A scale $s = 0$ opening (ψ_1) operates on individual pixels and makes no change to the overall signal.) The values of these removed vertices after filtering are lowered to equal that of their highest intensity neighbours. For example, the middle panel of Figure 3.35 shows the result of a scale *one* opening on the simple example image from Figure 3.34. A ψ_2 operator is applied to every pixel in the image. This involves:

1. applying a $\min$ operation to all sets of $C_2(G_{2D}, v)$ for all v, followed by
2. applying a $\max$ operation to all sets of $\min(C_2(G_{2D}, v))$

The effect of (1) on, for example, vertex 7 of the image would produce

$$\begin{aligned} \min(C_2(G_{2D}, 7)) &= \{\min\{2,7\}, \min\{7,8\}, \min\{7,12\}, \min\{6,7\}\} \\ &= \{5,5,5,5\} \end{aligned}$$

and the effect of (2) on this set would be

$$\max(\min(C_2(G_{2D}, 7))) = 5$$

Therefore, the value of the vertex 7 and similarly the value of the vertex 17 (the only other unit area maximum of the image) is five after filtering with a ψ_2 operator.

Conversely, the effect of applying a scale 1 closing filter, γ_2, to the image would be the removal of all minima of unit area (i.e. vertex 13). The values of the filtered points would then equal those values of the lowest intensity neighbours. See the right-hand panel of Figure 3.35.

Figure 3.36 shows the differences between these sieve outputs for a simple synthetic one-dimension signal. The signal is filtered to scale *one*, using each of the sieve-operators. The differences between the $\mathcal{M}_2$ and $\mathcal{N}_2$ outputs arise because of the ordering of the maxima and minima along the signal. Figure 3.36 also shows the sieve-based

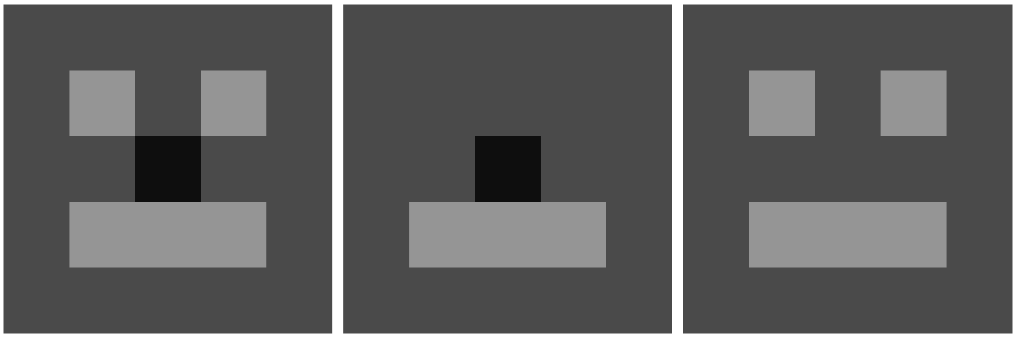


Figure 3.35: left: Original image, middle: Opening to scale 1 (ψ_2), right: closing to scale 1 (γ_2)

recursive median sieve output, (ρ_r) [19]. In one-dimension this is described as a one pass approximation to the root median filter, a median filter applied to the input signal until no further change occurs. The definition of a one-dimensional recursive median sieve is

$$\rho_r f(x) = \begin{cases} median\{ & \rho_r f(x - \frac{r-1}{2}), \rho_r f(x - \frac{r-1}{2} + 1), \ldots, \\ & \ldots, \rho_r f(x-1), f(x), f(x+1), \ldots, \quad x \geq 0 \\ & \ldots, f(x + \frac{r-1}{2} - 1), f(x + \frac{r-1}{2})\} \\ 0 & \qquad\qquad\qquad\qquad x < 0 \end{cases} \tag{3.53}$$

In practice, $\rho_r f(x)$ will remove extrema, of scale $s = (r-1)/2$ where r must be odd. These elements are removed in the order that they occur as the signal is parsed. This sieve operator may also be implemented in two dimensions, where the image is simply scanned from top to bottom, as in a raster scan and extrema are processed as they occur. This is not an implementation of a conventional two-dimensional recursive median filter – the filters are only the same in one dimension. It should also be noted that this recursive median operator does not always preserve causality. In Figure 3.36 the result of the one dimensional recursive median sieve, (ρ_3), gives the same output as the $\mathcal{N}$ filter, this is because the signal is scanned from left to right and the extrema are processed in the order that they occur. If the input signal is flipped from left to right, the output would equal that of the $\mathcal{M}$ filter. For more complicated interactions between the signal extrema, the recursive median filter does not necessarily match the other operators.

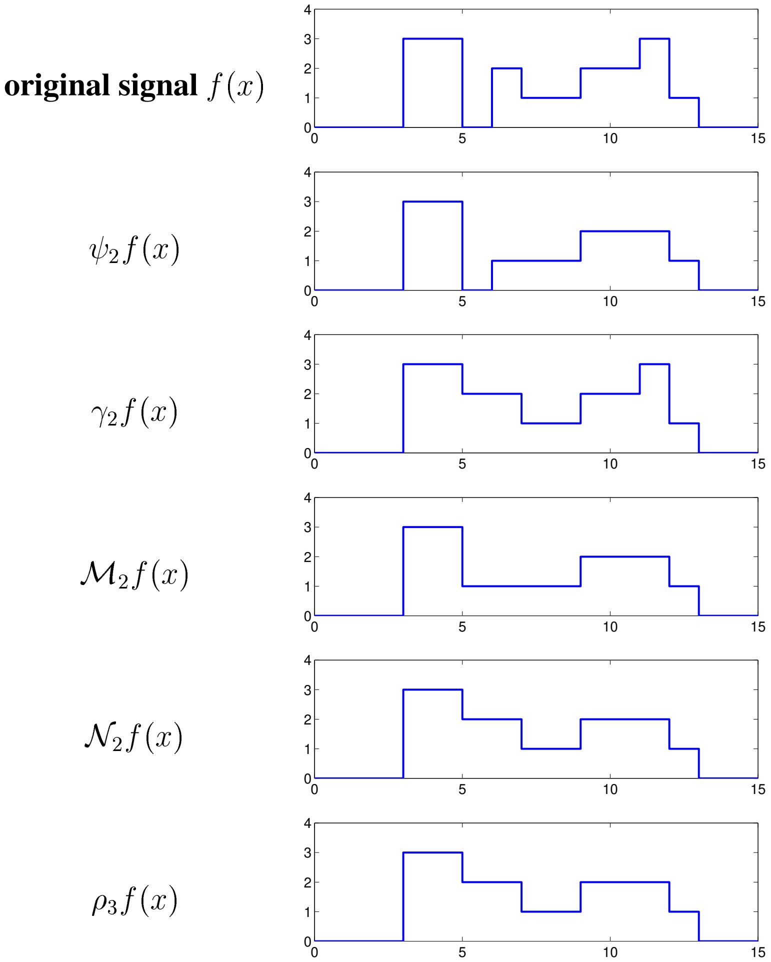


Figure 3.36: Simple one-dimensional example showing the differences between the scale 1 opening ($\psi_2 f(x)$), closing ($\gamma_2 f(x)$), $M-$ ($\mathcal{M}_2 f(x)$), $N-$ ($\mathcal{N}_2 f(x)$) and recursive-median ($\rho_3 f(x)$) sieve operators.

The structure of a full sieve decomposition of an input signal is shown in Figure 3.37. Increasing scale operators are applied to the previous scale sieve outputs, producing a signal that is simplified by scale. The result of applying a sieve operator of scale s to an input signal f is defined as

$$f_s = \mathcal{Q}_s(f_{s-1}) \tag{3.54}$$

where the original zero scale signal is $f_0 = \mathcal{Q}_0(f) = f$

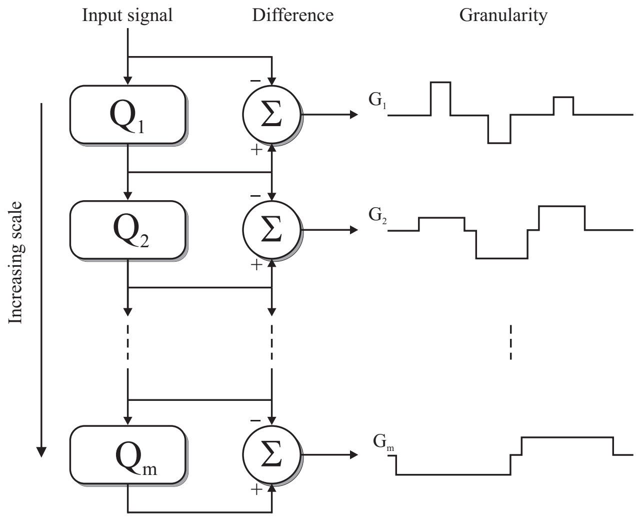


Figure 3.37: Structure of a sieve decomposition. Q denotes a sieve operator; either ψ (opening),γ (closing),$\mathcal{M}$, $\mathcal{N}$ or ρ

The cascaded operators remove increasing size maxima or minima. Therefore the result of applying a scale-two opening, ψ_3, to $\psi_2(f(x))$ would be the removal of all maxima of size ≤ 2, and so on. The development of sieves as scale-space processors relies on the repeated application of the alternating sequential filters ($\mathcal{M}_r$ and $\mathcal{N}_r$), to increasing scales.

$$M_s f(x) = \mathcal{M}_r M_{s-1} f(x) \quad \text{and} \quad N_s f(x) = \mathcal{N}_r N_{s-1} f(x) \tag{3.55}$$

with the initialising condition $M_0 f(x) = \mathcal{M}_1 f(x) = f(x)$ and $N_0 f(x) = \mathcal{N}_1 f(x) = f(x)$. Since no new extrema are introduced through the filtering process [15], the cascade operations of (3.55) produce causal scale-space representations of the original signal. As scale increases the detail of the input signal is gradually removed until, at the maximum scale, the filtered output is flat.

Figure 3.37 also shows the *Granularity* domain of the sieve decomposition. This too is essential to the sieve's scale-space representation. The *granule functions*, ($\mathcal{G}_s$), of the sieve are defined as the differences in the successive low-pass sieve outputs (3.54)

$$\mathcal{G}_s = \mathcal{Q}_{s-1}(f_{s-2}) - \mathcal{Q}_s(f_{s-1}) = f_{s-1} - f_s \tag{3.56}$$

The non-zero regions of these granule functions are the connected sets of elements that have been removed from the original image at a particular scale. These non-zero regions of $\mathcal{G}_s$ are called *granules*, (g_s), and they have the useful property of occupying the same position within the granule function as they did in the original signal. This allows the sieve decomposition to be fully invertible; the original signal may be reconstructed by summing all the granule functions over all scales

$$f(x) = \mathcal{G}_1 + \mathcal{G}_2 + \ldots + \mathcal{G}_{m+1} \tag{3.57}$$

where

$$\mathcal{G}_{m+1} = f_m \tag{3.58}$$

This property is shown more clearly in the synthetic two-dimensional example of Figure 3.38. The 100×100 pixel image is filtered to a single flat-zone (top row) and all features that are removed from the image are stored as granules in the corresponding granule functions (bottom row).

The small rectangle in the bottom left-hand corner of the image has an area of 30 pixels and so is removed at scale 30; filtering below scale 30 produces no change. Once removed from the lowpass signal, the rectangle is stored as a granule, shown in the second panel of the bottom row. As the lowpass signal is filtered with increasing scale $M-$sieves, there is no change until scale 121, when the centre, smaller square is removed. Again this is stored as a granule in the corresponding granule function. At scale 900 all detail is removed from the original signal and the final large-scale granule is recorded.

Only those scales where changes occur are shown in the figure. For an image of this size there are 10000 possible decomposition images. Since it is infeasible to display and store such vast amounts of data it is possible to sum all of the outputs over a range of scales into a *channel*. In this case features of different scales may appear in the same channel and so only an approximate scale estimate may be calculated. If a more accurate scale estimate is required, a channel may be sieved to a finer level of quantisation.

3.6.1 Sieve properties

Below is a brief summary of the more useful sieve properties – with respect to implementation as a practical scale-space processor.

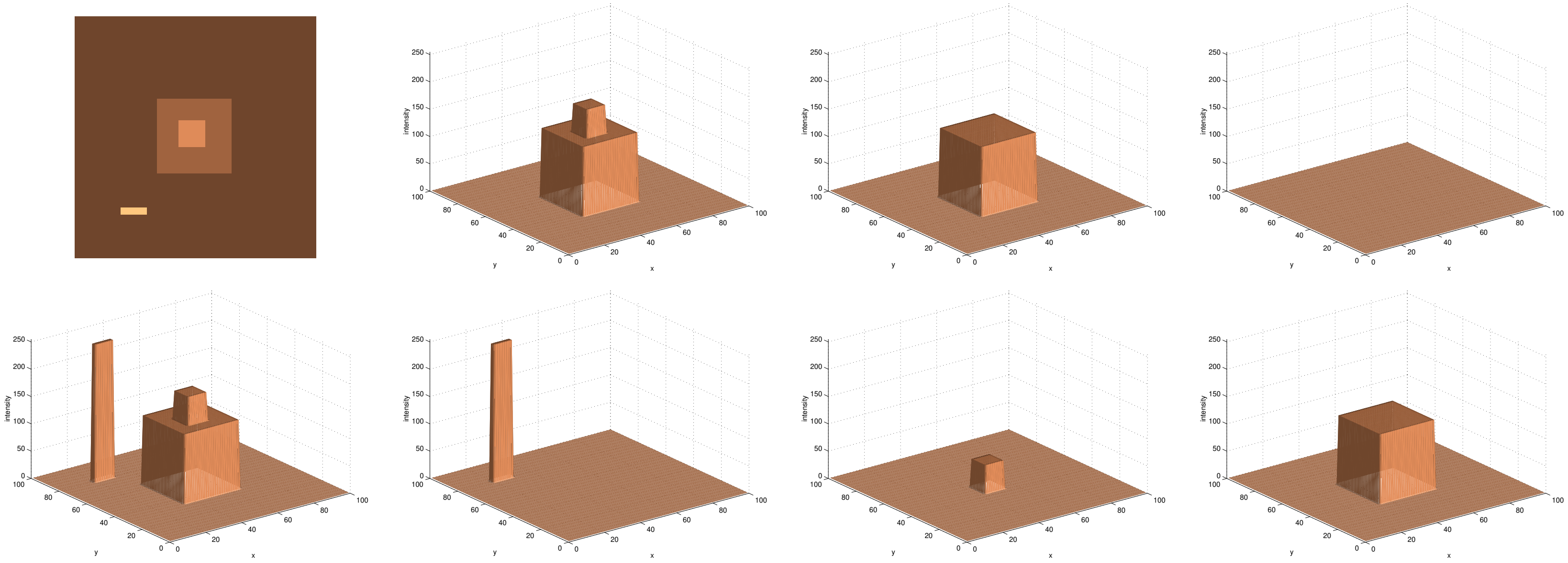


Figure 3.38: Plot showing the low-pass $\mathcal{M}$-sieve decomposition of an image (top row) shown as a surface intensity plot, and the corresponding granule surfaces (bottom row).

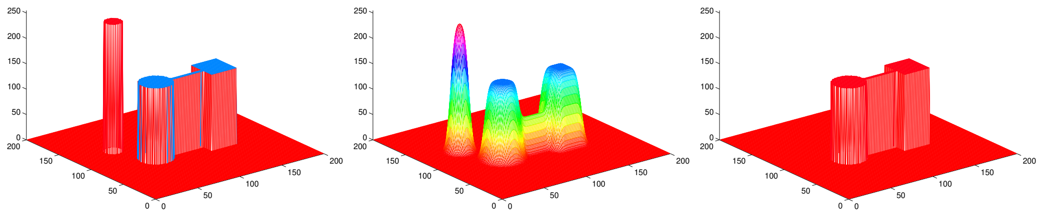


Figure 3.39: Surface intensity plot of a simple image. Left: Original image, middle: image diffusion filtered to scale 10, right: image $\mathcal{M}-$sieved to scale 260.

- **Idempotency:** Sieving a signal to a scale s will remove all extrema of size s. Further filtering operations to a scale less than or equal to s will produce no change in the sieve output. The sieve is therefore *idempotent*.

$$f_s = \mathcal{Q}_s(f_{s-1}) = \mathcal{Q}_s(\mathcal{Q}_s(f_{s-1})) \tag{3.59}$$

- **Invertibity:** Once a signal has been sieved to increasing scales it is possible to reproduce the original signal by summing the differences between successive lowpass sieve outputs ($\mathcal{G}_s$) (See (3.57)). It should be noted that, to fully reproduce the original signal from a complete sieve decomposition, the signal should be bounded on all sides by zeros. See Section 3.6.2 for more details.

- **Causality:** Filtering a signal with the opening-, closing-, $\mathcal{M}-$ or $\mathcal{N}$-sieve operators will not introduce any artificial extrema into the signal [15]. Figure 3.39 shows a popular example, introduced by Lifshitz and Pizer [67], discussed in Section 3.2.3, that illustrates the problem of diffusion based systems obeying the causality axiom in two dimensions. The increase in the number of extrema points at increasing scales for the diffusion filter caused the re-evaluation of the causality axiom to that stated in Section 2.3.1. The sieve algorithm creates no new extrema and so obeys both the original and the re-evaluated causality axiom.

- **Immediate Localisation:** As proposed by Perona and Malik, (see Section 2.3), the region boundaries of the sieves output are sharp and are easily related back to the original signal.

3.6.2 Boundary conditions

Since the sieve-decomposition is invertible, the sum of the sieve's granule functions should replicate the original signal. However, this is only guaranteed if the boundary conditions of the input image are controlled. In one-dimension, this involves padding with enough zero elements to equal to the original signal length, (m), at the start and end points of the signal. In two-dimensions, a full boundary of zeros whose area is greater than the area of the image is needed. Without this zero-padding the baseline or maximum scale (m) intensity value cannot be preserved.

3.6.3 Example sieve scale-spaces

A scale-space decomposition of a more realistic, noisy signal is shown in Figure 3.40. Here the input is the same as that used to illustrate the one-dimensional linear and non-linear diffusion systems and the morphological poweroid scale-space, shown in Figures 3.7, 3.12, 3.14 and 3.29 – 3.30. The figure shows the low-pass $\mathcal{M}$-sieve decomposition with signal representations at scales 3, 10, 20, 40 and 100. Figure 3.41 shows the granules as they are removed during filtering. Filtering to scale 3 removes the vast majority of small scale detail from the signal, and the remaining, larger extrema are reduced to flat regions before being removed at the appropriate scale. Here, it is necessary to filter to scale 331 (the signal length) to reduce the output signal to a single zero-amplitude flat-zone (not shown).

Figure 3.42 shows the result of $M-$sieving the more realistic image that was used to demonstrate the performance of the other systems. As detail is removed by the sieve the extrema regions merge into their local backgrounds. The couple walking in front of the building are removed between scales 500 and 1000. In comparison to the diffusion systems, the sieve output shows no blurring and the extrema points of the image are not subject to spatial movement. The sieve output has some resemblance to the non-linear diffusion systems, but has an intrinsic scale-calibration: extrema of pixel-area s-pixels are removed at scale s. The anisotropic systems have no such scale calibration. This is discussed in more detail in the following section.

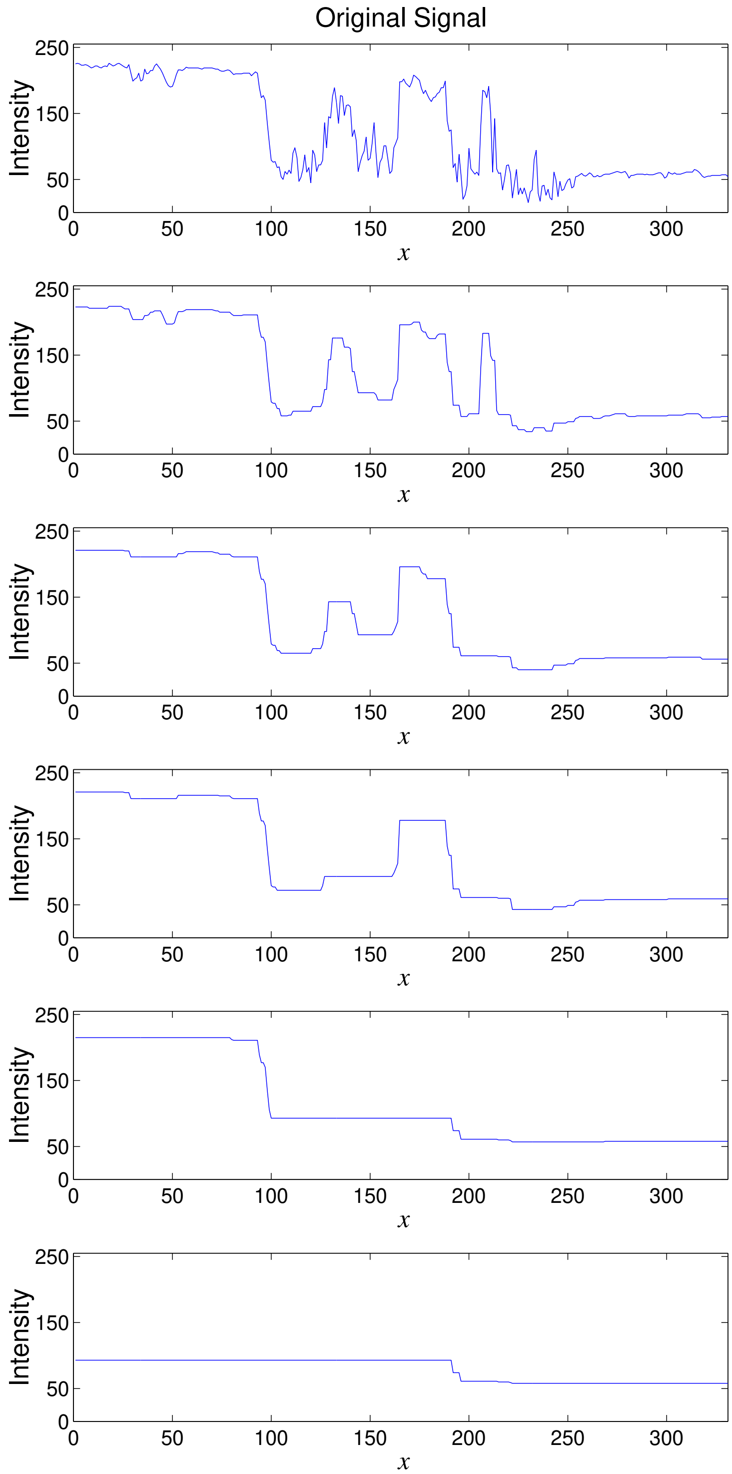


Figure 3.40: Scan line (top panel), simplified via $\mathcal{M}-$sieving. Results after sieving to scales 3, 10, 20, 40 and 100 are shown.

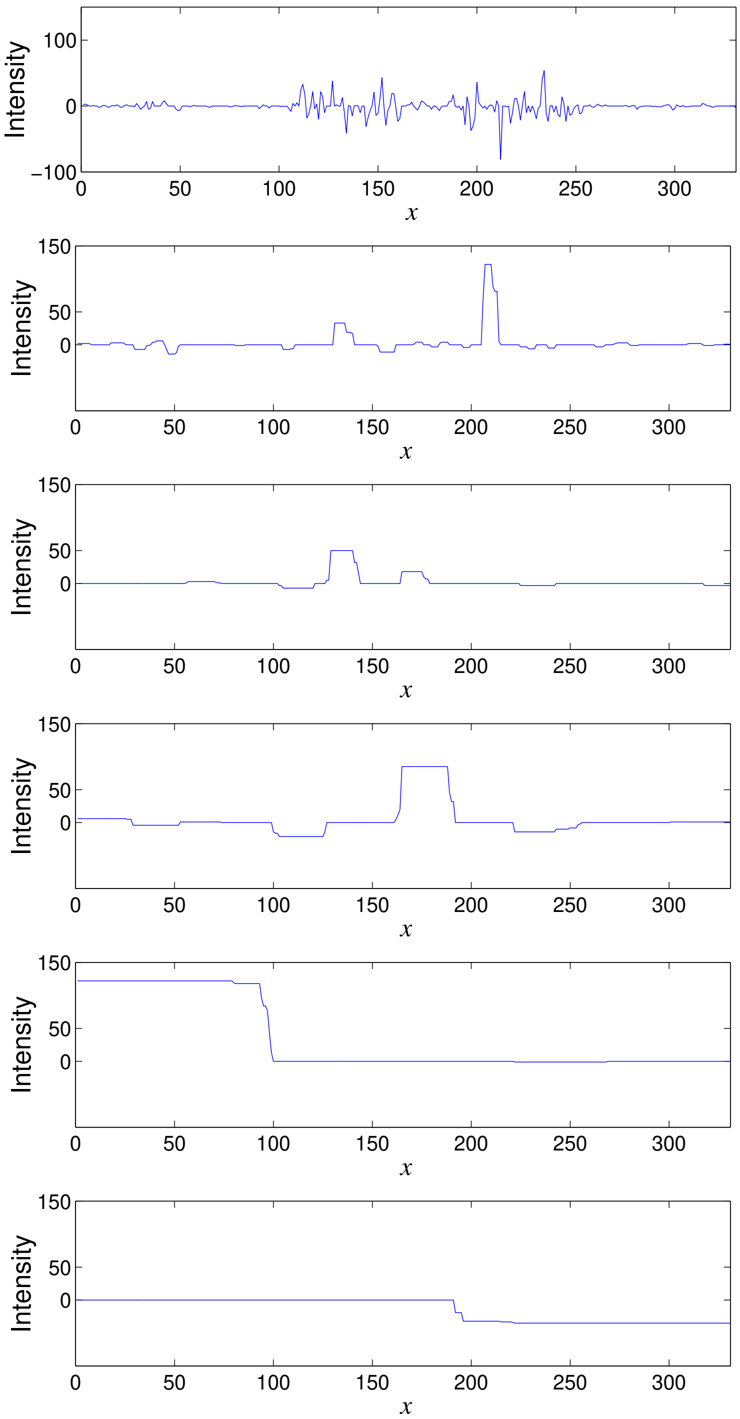


Figure 3.41: Granule functions ($\mathcal{G}_s$) corresponding to the $\mathcal{M}-$sieved signal of Figure 3.40, granule functions $\mathcal{G}_{1\to3}$, $\mathcal{G}_{3\to10}$, $\mathcal{G}_{10\to20}$, $\mathcal{G}_{20\to40}$, $\mathcal{G}_{40\to100}$ and $\mathcal{G}_{100\to331}$ are shown.

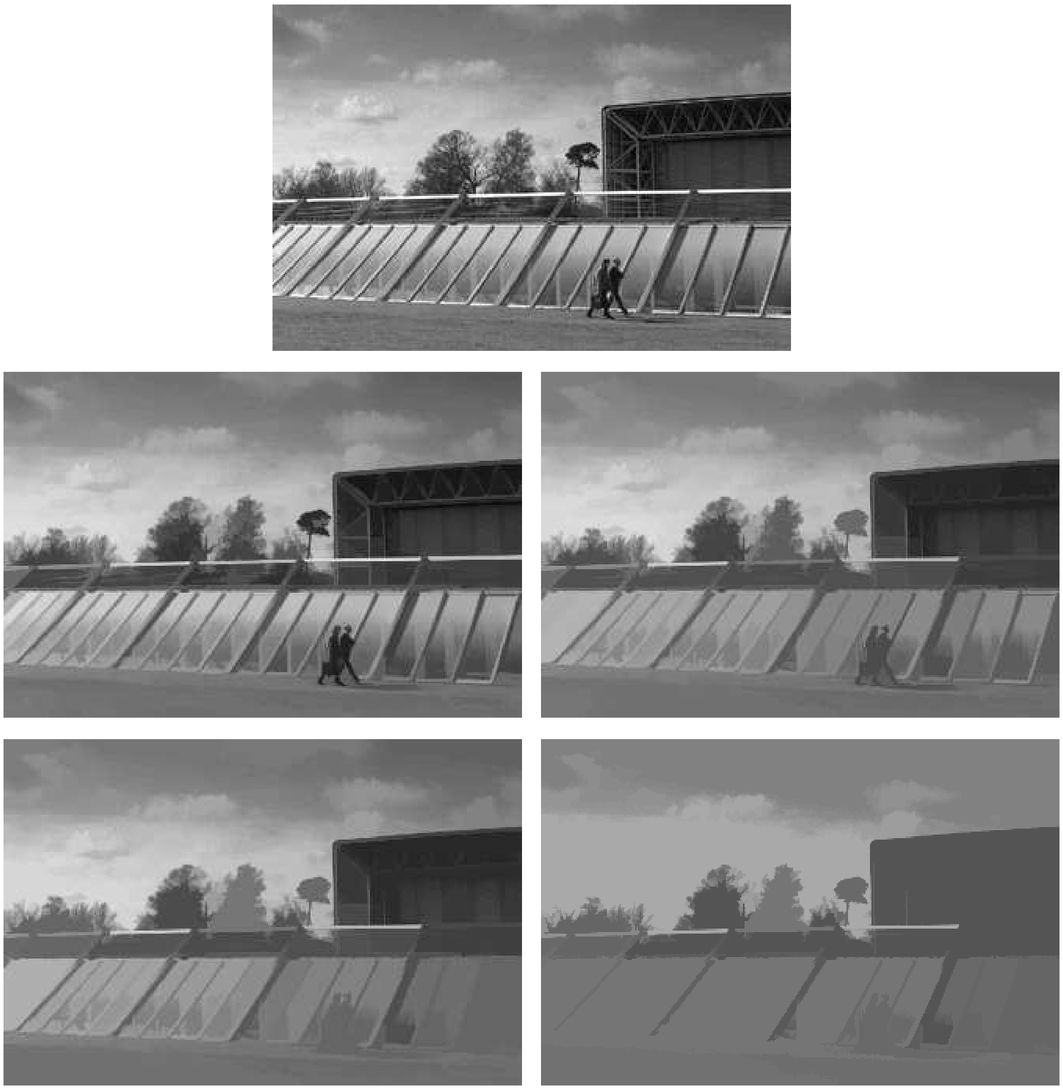


Figure 3.42: Images simplified via $M-$sieving to scales 100, 500, 1000 and 10000

3.7 Scale-selection and scale calibration

The well-published requirements and aims [63, 71, 118] of scale-space processors were discussed briefly in Section 2.3. Each of the filters described in this chapter obeys the necessary axiom of causality along with other selected axioms from the list. However, it is the final addition to the property list; the property of a *scale-selection surface* that is discussed here. The aim of this thesis is to study and compare the performances of scale-space systems as estimators and the experiments that are described in Chapters 4 and 5 show how well these systems may cope with distortions such as image noise. However, to perform these experiments there is the need for a scale-selection surface and a method for associating objects in an image with their own particular scale.

A general multiscale representation contains no explicit information on the significance of features within an image or signal and also no information on the appropriate scales at which to view these, as yet unknown significant features. Unless there is some *a priori* knowledge available, the low-level multi-scale analysis must begin by treating all scales as having equal importance.

The definition of exactly what constitutes *significant features* within an image is by no means fixed. Edge-boundaries and corner-points are sometimes mentioned, as are extremal regions [73, 118]. This thesis concentrates on extremal-regions, since the scale-selection surfaces for the other features are not so well developed.

If these systems are then able to use the scale-space information to label image features as being of a particular scale, even if these features exist over a range of scales within the scale-space, then this information may be passed on to any higher-level systems, and the delay that can exist between the low-level and high-level processors may be reduced.

The definition of scale alters for the different scale-space representations, varying from the dimensions of a rigid filter to the number of filtering iterations to the area of a non-rigid connected-set. The next section shows the methods by which some of these scale-spaces produce scale estimates, and introduces the problem of comparing these methods.

3.7.1 Linear diffusion

At any scale representation, the linear diffusion scale-space may contain features of multiple scales. The example images of Figure 3.8 show that the small-scale walking couple are still visible in the largest scale image. The blurring caused by the filter makes objects indistinguishable as they are reduced to smoothed peaks and troughs that eventually merge at larger scales in the scale-space. A simplified example of this, and the problems that it can cause is shown in Figure 3.43.

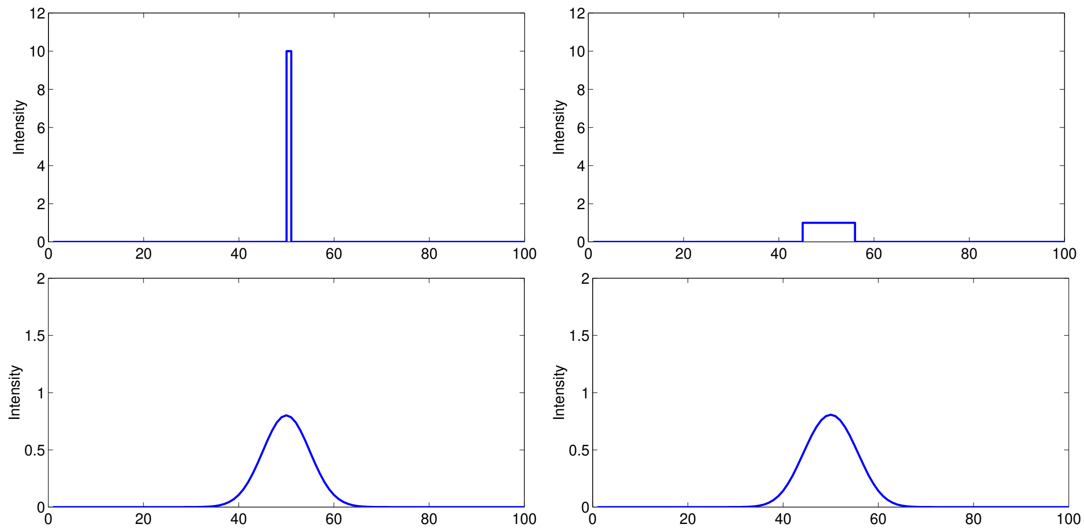


Figure 3.43: Top left: signal containing impulse of amplitude 10, top right: signal containing pulse of width 10 units and amplitude 1. The signals are filtered by linear diffusion (eqns (3.8) and (3.9)) to scale $s = 25$ for the top-left signal (output shown bottom-left) and $s = 18$ for the top-right signal (output shown bottom right)

The figure shows two signals, in the top left panel, an impulse of length 1 unit and amplitude 10, and on the right a pulse of length 10 units and amplitude 1. The signals are both of length 100 units. When filtered to different scales (scale 25 bottom left and scale 18 bottom right) these signals become practically indistinguishable. Given that there is no prior information as to the *correct* scale at which to view these signals, it is impossible to make any attempt at scale estimation from the low-pass blurred output. Instead, a second level of computation is needed.

In [71–73] Lindeberg investigates methods of automatic scale selection. The equation given in (3.60) is reported as one of the options for scale estimation. This *scale-*

selection surface gives a measure of the *curvature* of the scale-space at any position, and peaks in this output are thought to represent the presence of possible significant features in the corresponding low-pass filtered image.

$$d_s = s\nabla^2 I_s = s\left(\frac{\partial^2 I_s}{\partial x^2} + \frac{\partial^2 I_s}{\partial y^2}\right) \tag{3.60}$$

In (3.60) I_s is the image I filtered to a scale s. In practice, the square of d_s is used to remove the polarity effects of the signal, and a search for maxima over this surface is used to find the *significant* points. For the processing of larger images, the idea of storing both the low-pass scale-space and the corresponding scale-selection surface is costly in terms of memory requirements. In this situation the *five-point* discretised version of this equation (3.61) [4] may be applied to the low-pass surface's extrema points (x), and their North (x_N), South (x_S), East (x_E) and West (x_W) neighbouring amplitudes. (h is a measure of the separation between adjacent points)

$$d_s = s\left|\frac{1}{h^2}(x_N + x_E + x_S + x_W - 4x)\right| \tag{3.61}$$

Figures 3.44 and 3.45 show the result of applying (3.60) to the simple 1D pulse-signals from the top panel of Figure 3.43, after diffusion to increasing-scales. The original signals are given in the top panels and below, the left-hand panel shows the low-pass diffused signals at scales $s = 5$, 10, 25 and 70. The right-hand panel shows the scale-selection surface for the corresponding scales. For the filtered impulse signal (Figure 3.44), the curvature of the diffused signal is higher at small scales, whereas for the wider-pulse signal (Figure 3.45), the curvature of the scale-space is low at small scales, and at increasing scales the curvature increases, peaking around scale 25 before reducing again for large scales.

In [71] it is the position of these scale-selection surface peaks that provides scale estimates for the pulses. In these simple examples the pulse's position estimate may also be taken directly from the position of peak, however for signals containing more than one feature (the majority of non-synthetic images) tracking is needed to estimate position. This is due to the influence of the diffusion kernel covering a wide area of the input signal; all features with positions that fall into the range of the filtering kernel directly affect the filter output at that point. For instance, a signal containing two positive impulses filtered to a large enough scale will produce a scale-space representation

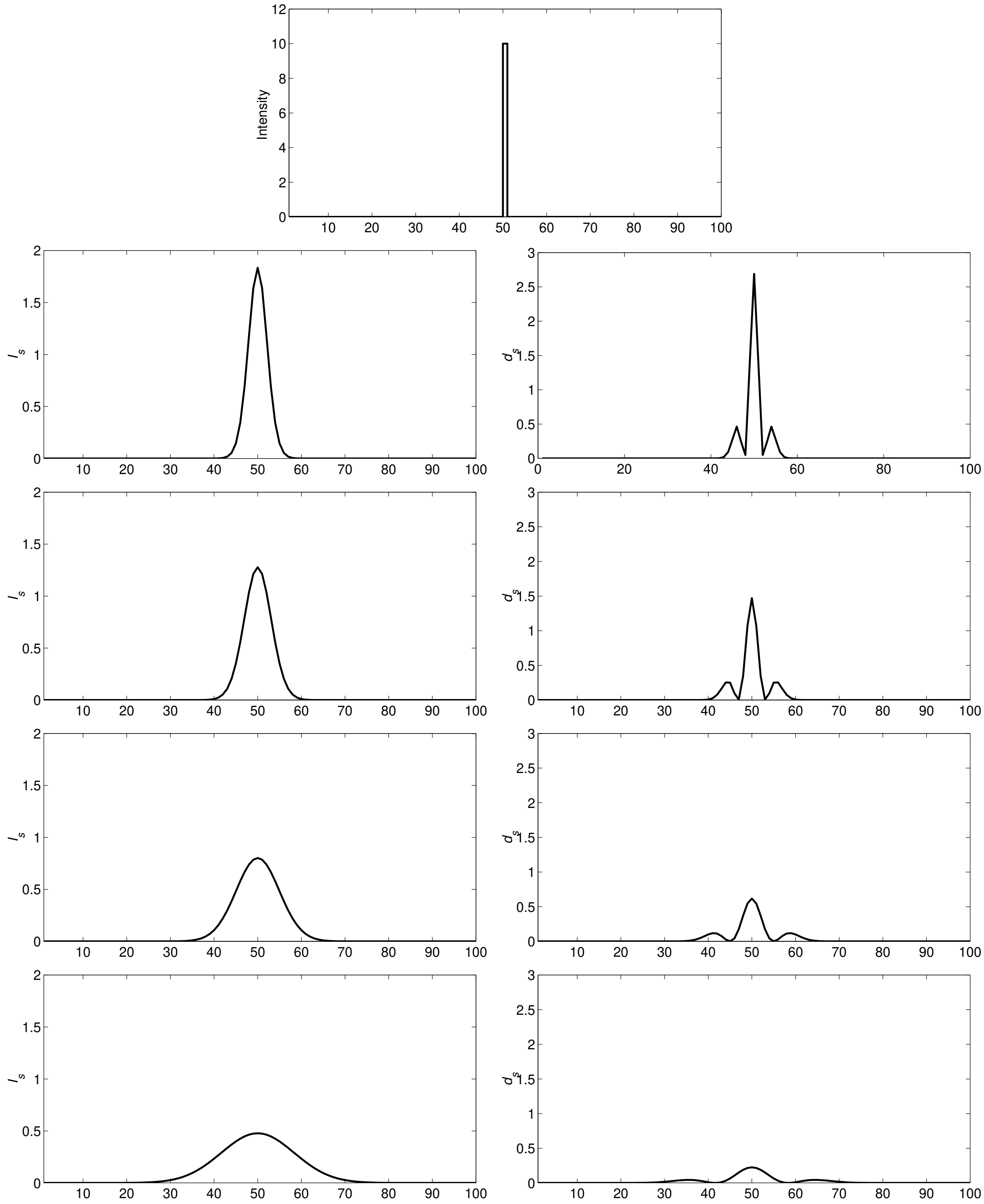


Figure 3.44: Scale-selection for the linear diffusion filter. Top panel: Original image, left-hand panel: the linear diffusion scale-space at scales $s = 5$, 10, 25, and 70. The right-hand panel shows the corresponding squared scale-selection surface created using equation (3.60)

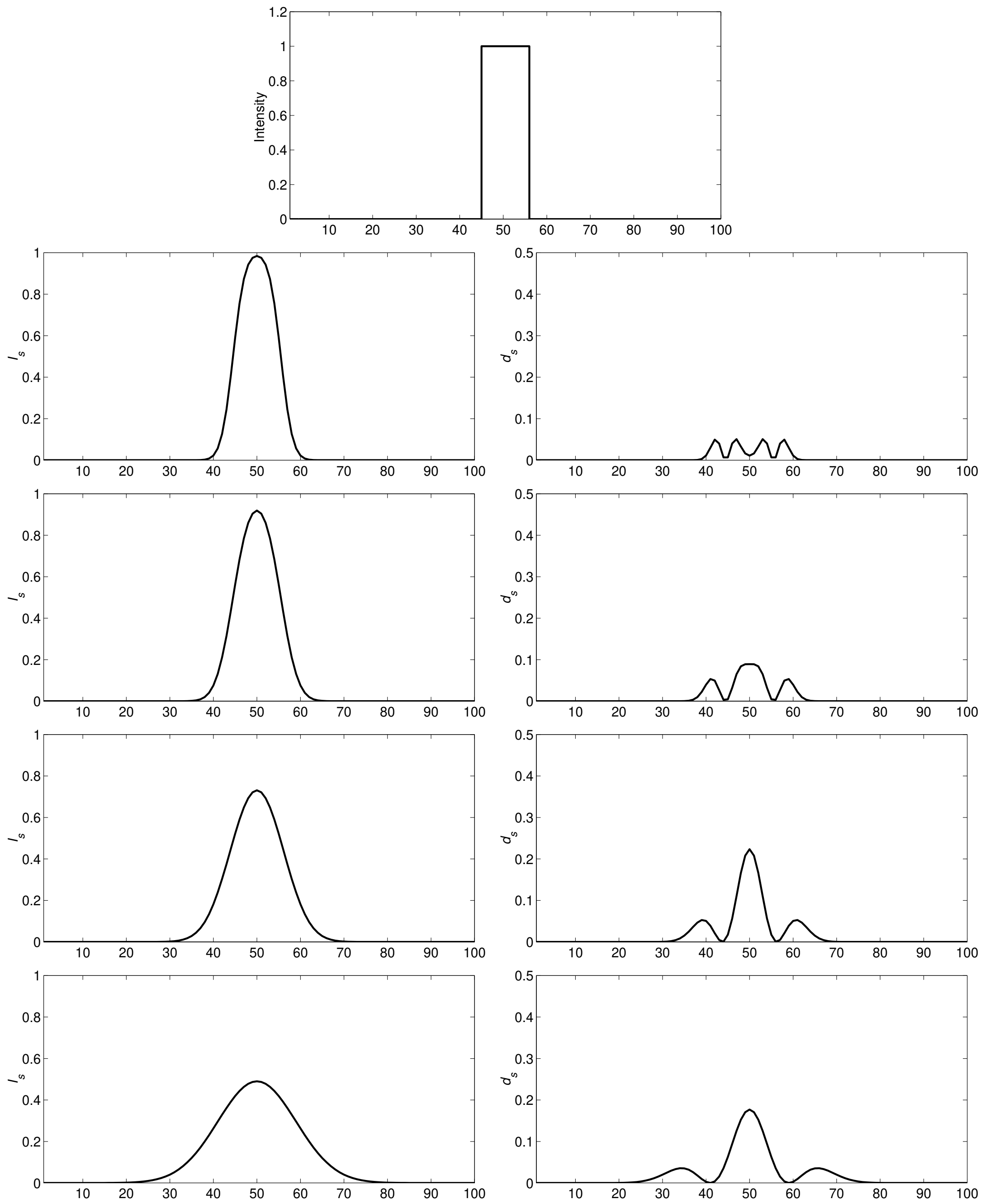


Figure 3.45: Scale-selection for the linear diffusion filter. Top panel: Original image, left-hand panel: the linear diffusion scale-space at scales $s = 5$, 10, 25, and 70. The right-hand panel shows the corresponding squared scale-selection surface created using equation (3.60)

containing a single extremum. The maxima in the scale-selection surface will provide the scale-estimates, however, to recover the position information these maxima must be traced back over the scale-space surface to the original image at scale-zero.

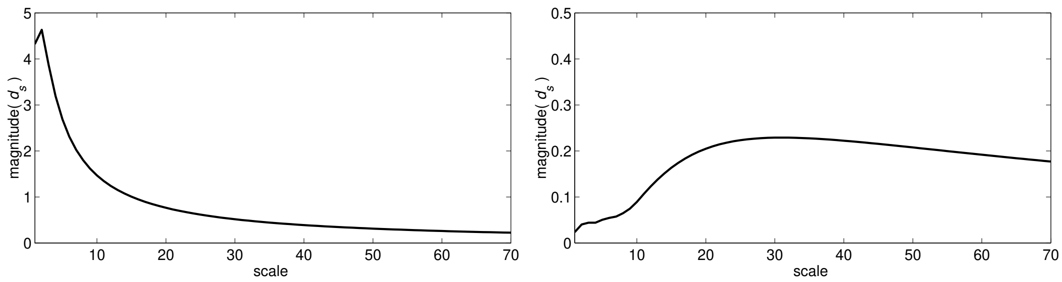


Figure 3.46: Plot of the maximum points of the scale-selection surfaces of the two signals from Figure 3.43 plotted against increasing scale. Left panel: scale-selection surface of the impulse signal - estimated scale $s = 2$, right-panel: scale-selection surface of the pulse signal - estimated scale $s = 28$.

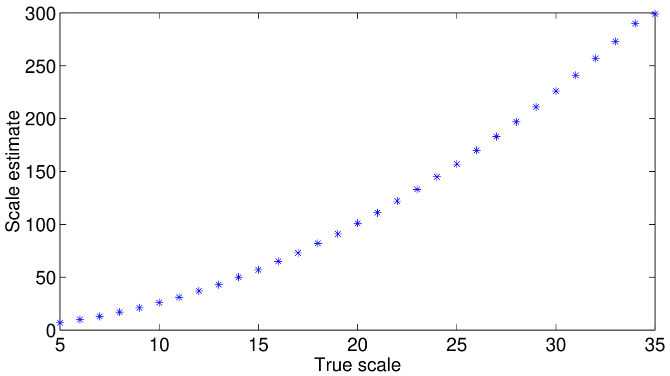


Figure 3.47: Relationship between estimated scale of simple pulse signals and true scale (pulse width) for linear diffusion

Figure 3.46 shows examples of the maxima of the scale-selection surfaces plotted as a function of scale. The input signals were, again, the impulse and the width-10 pulse of Figure 3.43. The scale estimates for the two signals are taken at the peaks of the scale-selection curvature, and so the impulse (on the left) has an approximate scale of $s = 2$ and the pulse (on the right), $s = 28$. It is possible to calibrate these results and find an explicit relationship between the recorded scale estimates and the width of a signal pulse. This relationship is indicated in Figure 3.47, where the scale estimates of

31 pulses are plotted against the true pulse width. The relationship is invariant to pulse amplitude, which simply causes a vertical offset in the scale-selection surface.

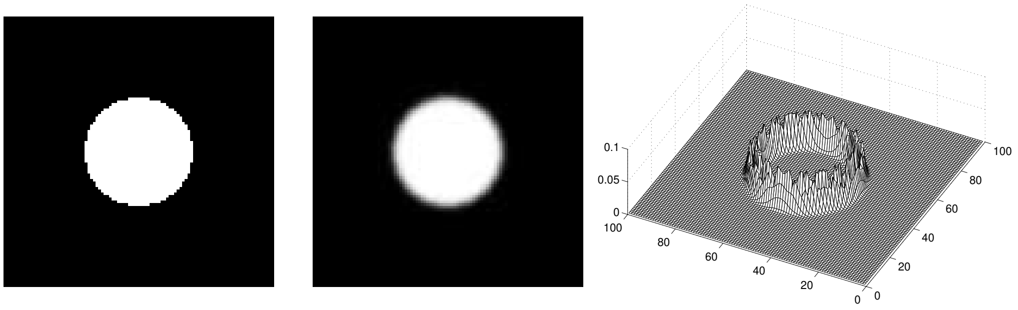


Figure 3.48: The *crown of thorns* effect. The original image (left), is diffused to scale $s = 2$ (middle), and the resulting scale selection surface is shown (right). These extrema points are caused by the sharp edges of the synthetic disc image.

There are further problems associated with tracking the curvature points of the diffusion scale space. At small scales, for sharp-edged features the scale-selection measure (3.60) produces sharp extrema points. Figure 3.48 shows an example of this effect where the original image, a synthetic disc (left), is filtered to scale $s = 2$ (middle). The right-hand panel of Figure 3.48 shows the *crown of thorns* effect that comes of applying (3.60) to the filtered disc. The size of these extrema points are small compared to the maxima point associated with the scale of the disc, but the distance between the true-scale points (centre of the disc) and the points form the crown of thorns (edge of the disc) can cause problems with tracking and result in false scale labelling if not treated carefully. One simple solution to this problem is to avoid searches over the small-scale region of the scale-selection surface, however, if small scale features are contained within the image, the restriction to the search range may cause these features to remain undetected.

3.7.2 Anisotropic diffusion

All of the anisotropic diffusion systems described in this chapter are implemented iteratively, where the diffusion of any point in an image is a function of the gradients to the four-connected neighbours. The idea behind the anisotropic system was to improve on the blurring that is inherent to the linear-diffusion system and as a result, produce images that are have "semantically meaningful" region boundaries when compared to the original image. However, the solution to the blurring is to limit the diffusion flow for points with contrasts above some threshold and these high contrast points then remain throughout the scale-space. This becomes a problem when attempting to associate particular features with their own scale.

To develop a scale-selection surface for a scale-space system, the filter *must* remove image features as scale increases. At scale $s = \infty$, the final image must be flat. This is not the case for the majority of anisotropic systems. The filters were not designed to provide a scale-selection surface, in fact the Tukey-based system was designed with just the opposite aim; to keep image features from being removed. Once high contrast points are identified as outliers they remain unaffected as the scale of filtering increases. The system was designed to robustly retain the high contrast points of an image as the lower-contrast point are blurred towards flat-zones, and it does this successfully. However, this thesis aims to test the reliability of these scale-space filters for use in position and scale estimation and a scale-selection surface is required. Even the Lorentzian system, which may output a flat image at scale $s = \infty$ cannot produce a valid scale-selection surface since the rate at which features are removed from an image depends purely on the local gradients of the pixels. There is no *size* aspect to the scale parameter and so two extrema regions of equal size will not be removed at the same rate if the contrast around the regions is dissimilar. As a result of these problems current anisotropic systems cannot be used for the practical estimation of scale.

3.7.3 Morphological scale-space from poweroids

The morphological erosion-dilation scale-space described in Section 3.5 does produce a flat image at scales $s \to \infty$. Also, features are removed as a function of the size of the structuring element that is fitted to the image. However there is no published scale-selection surface available. The curvature measure employed in the linear-diffusion system is impractical here, since the curvature in the scale-images is always imposed directly from the structuring element.

3.7.4 The sieve

The granule functions of the sieve decomposition provide the scale-selection surface of the sieve. These granule functions ($\mathcal{G}_s$) are the differences between the low-pass sieve outputs and contain features that are removed from the input signal for increasing scales.

$$\mathcal{G}_s = f_{s-1} - f_s$$

The non-zero elements of the granule function are called *granules*, (g_s), and these regions occupy the same space in the granule function that they occupied in the low-pass sieved image. Figure 3.38 showed this clearly. The granules are localised in scale and position, so a search over all granule functions from a sieve decomposition will reveal the scale and position of all features from the original image.

The input image of Figure 3.38 is ideal for the sieve decomposition because of the sharp edges of its features. The regions are removed cleanly and do not spread over multiple scale granule functions. However, this is not always the case and the top left-hand panel of Figure 3.49 shows a slightly less than ideal image and its intensity plot (bottom left-hand panel). The disc in the images gives a more complicated area-decomposition because it has blurred edges, the tip of the disc having an area (804 pixels) that is significantly smaller area than that of the base. The shape *cannot* be removed at a single scale and instead it is removed in a series of layers, the 2nd, 3rd and 4th plots in the top panel of Figure 3.49 show the disc's intensity plot as it is $\mathcal{M}$-sieved to increasing scales ($s = 1000$, 1400 and 2000). The corresponding images in the lower panel show the differences between these increasing scale outputs, $\mathcal{G}_{0\to1000}$, $\mathcal{G}_{1001\to1400}$ and $\mathcal{G}_{1401\to2000}$. These functions make up the sieve's scale-selection surface and estimates of scale and position may be recovered directly. However, in this example for display purposes, the granule functions shown are the result of sieving to *channels* rather than single scales. This is because the disc may be visible in granule functions between scales $s = 804$ and $s = 1804$ and it is easier to view these as three channelled-granule functions rather than one thousand separate granule functions. Channels equate to the sum of successive single-scale granule functions, for example, the channel output $\mathcal{G}_{1001\to1400}$ in Figure 3.49, corresponds to the sum of individual granule functions between scales $s = 1001$ and 1400, i.e. $\mathcal{G}_{1001\to1400} = \mathcal{G}_{1001} + \mathcal{G}_{1002} + \mathcal{G}_{1003} +, \ldots, + \mathcal{G}_{1400}$.

Whether features are confined to one granule function or spread over a wide range,

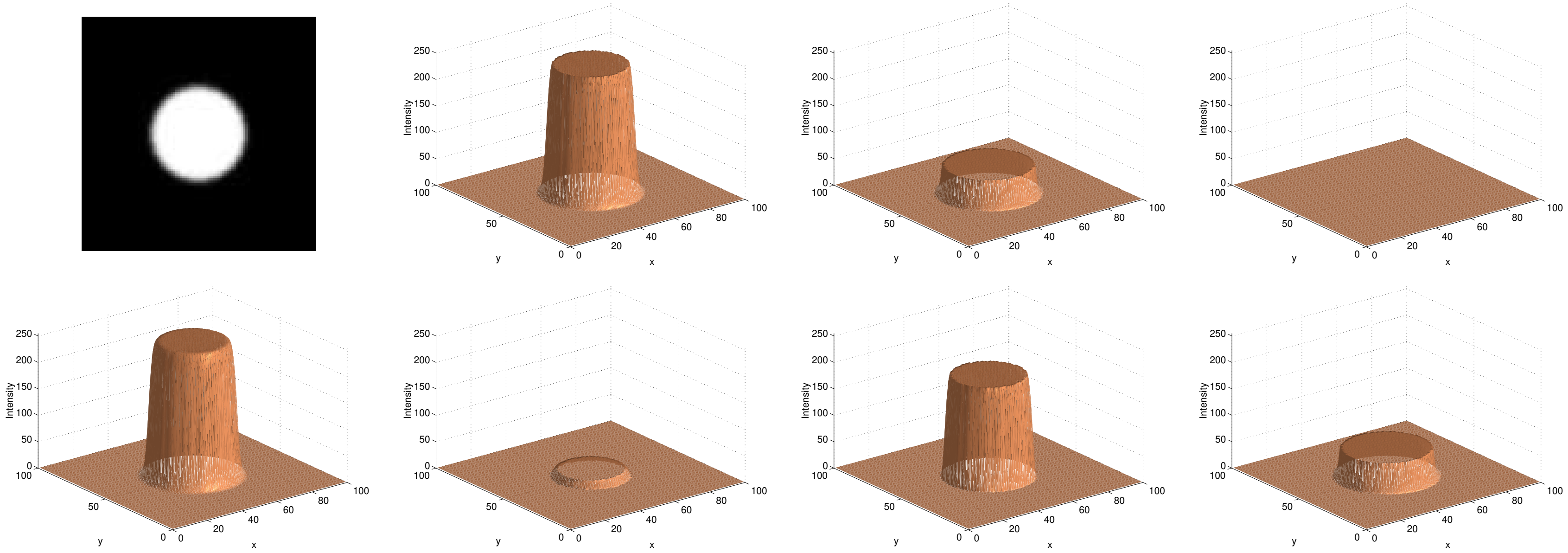


Figure 3.49: Filtering a blurred disc with an $\mathcal{M}$-sieve. Top left-hand panel is the original image, with the intensity plot directly below. $2nd$ image in top panel shows the intensity plot of the disc filtered to scale $s = 1000$, $3rd$ image has disc filtered to scale 1400 and $4th$ image has disc filtered to scale 2000. The three further images in the bottom panel show the granule functions (scale-selection surface) $\mathcal{G}_{0\rightarrow 1000}$, $\mathcal{G}_{1001\rightarrow 1400}$ and $\mathcal{G}_{1401\rightarrow 2000}$

the *correct* scale of any feature is recorded as the scale at which the largest amplitude granule that constitutes that feature is found. For some features, this may involve tracing the granules across scales, however, the sieve introduces no spatial translation and so simple tracking algorithms will suffice. Once the scale of an object is found, the position estimates may be taken directly from the positions of the granules, tracing the features back to the original image is not required. Further calibration of the sieve's scale estimates are also not required since they relate directly to the pixel areas of objects.

3.8 Discussion

This chapter has introduced a series of systems that each obey the scale-space axiom of causality, but that stem from different approaches, including the classical linear diffusion system which was the basis to the original concept of scale-space, the alternative non-linear diffusion and the newer morphological methods.

The chapter considers the practical aspects of implementing these systems. The example scale-spaces that are produced highlight the differences between the filters and indicate some of the problems that would be encountered were these systems to be used at the low-level stage of some more complex vision system. The actual scale-space representations contain minimal information. Each level in the scale-space simply contains less information than its predecessor and no new extrema are created for increasing scale. Because of this, none of the systems produce results that quantify signal or image structures directly from the scale-space. This requires a second level of computation, producing a *scale-selection surface*

Obviously, it was not the aim of some of these systems to produce quantifying scale-estimates and so there are scale-spaces that cannot produce a scale-selection surface. These systems such as those based on anisotropic diffusion, cannot resolve scale-estimates because the duration or lifetime of a structure is not purely dependent on scale but also on the local gradients surrounding that feature. Other systems, such as the poweroid based morphological filter, have not reached this stage of development and more work is required if this is ever to produce a scale-selection surface.

Of the filters studied, only the linear diffusion system and the sieve are capable of performing a complete scale-based decomposition and providing a scale-selection surface to give scale-estimates for all image structures. For complex images, because of the nature of these selection surfaces, a high degree of processing may still be required to label structures. For example Figures 3.50 and 3.51 show the scale-selection surfaces of both the linear diffusion filter and the $\mathcal{M}$-sieve, for scales that correspond to the *low-pass* filter outputs of Figures 3.8 and 3.42. The original image in this example contains multiple structures of many scales and as a result the selection surface may be cluttered at some scales.

The remaining chapters of this thesis deal with the comparison of the performances of these two remaining filters. The majority of signals and images that are used are synthetic, and so the signal parameters may be easily controlled and varied to determine the limits of the filter's performance. In particular we investigate the robustness of the scale-space systems and the effects of signal noise on the filter performance.

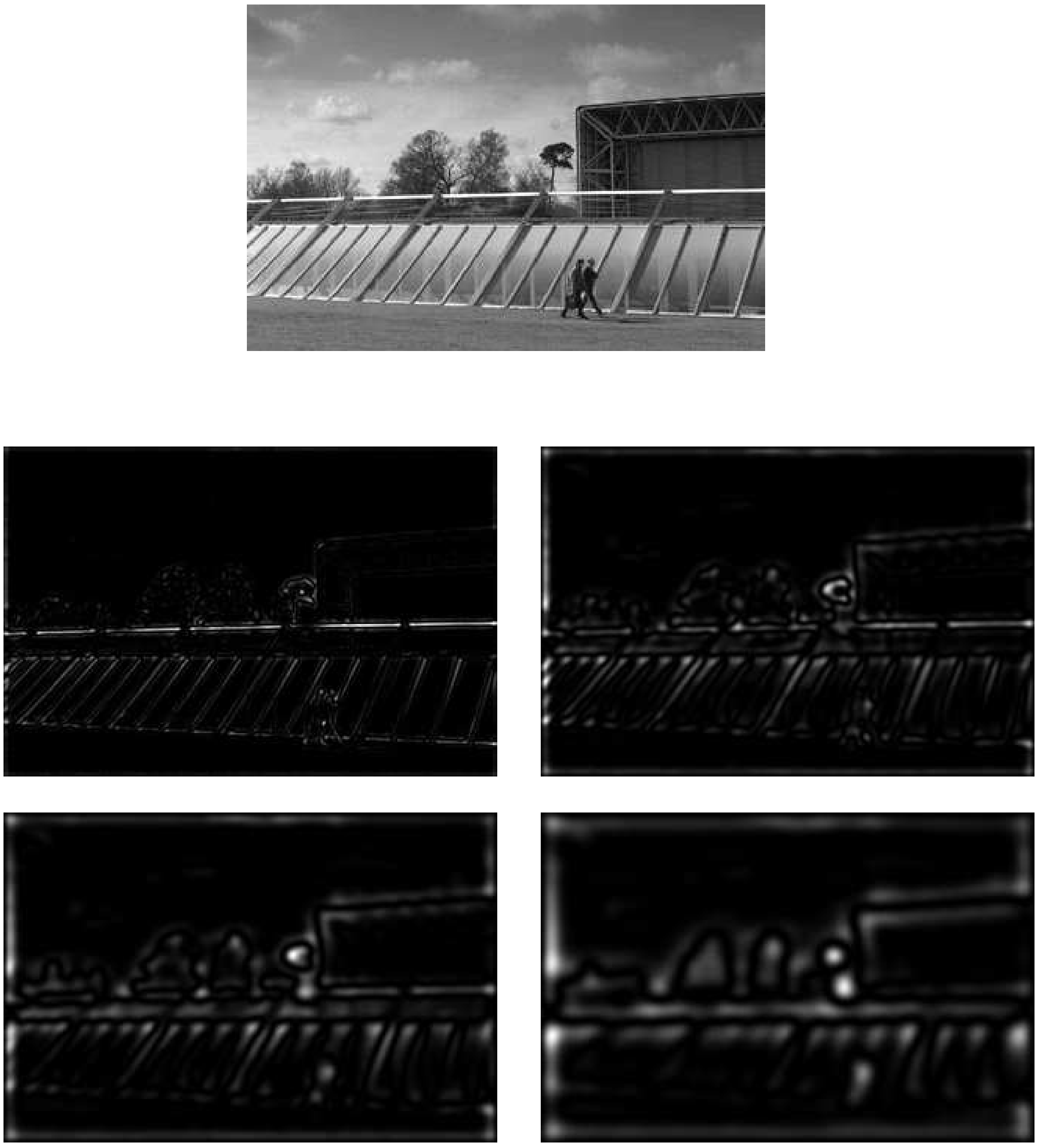


Figure 3.50: Scale selection images for the linear diffusion system scale $s = 2$, 10, 20 and 50. For display purposes the images are all normalised such that the maximum intensity point of each image equals 255.

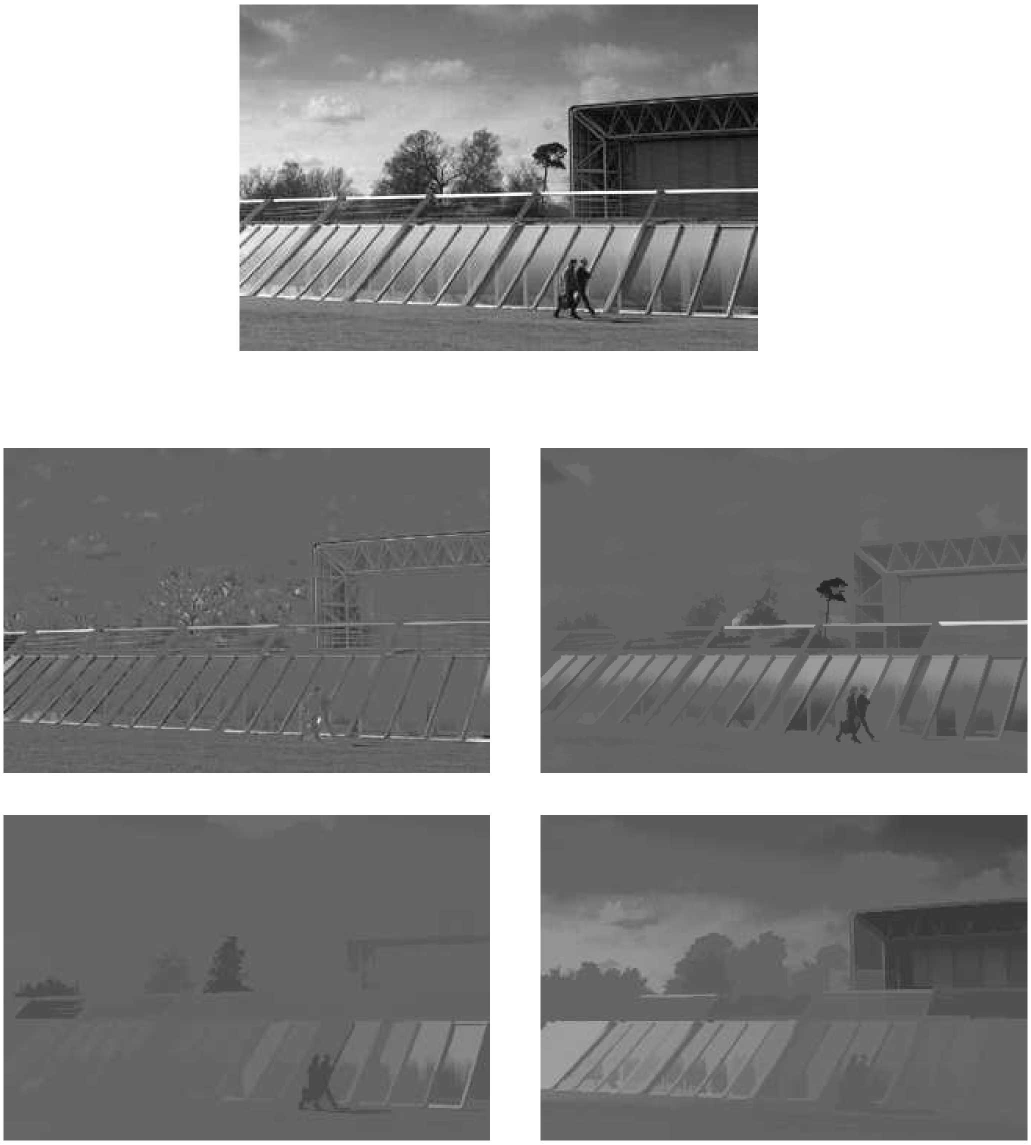


Figure 3.51: Scale-selection images for the sieve. Channels at scales $\mathcal{G}_{0\rightarrow 100}$, $\mathcal{G}_{101\rightarrow 500}$, $\mathcal{G}_{501\rightarrow 1000}$ and $\mathcal{G}_{1001\rightarrow 10000}$ are shown. The absolute range of the granules of these functions is [-255,255] and so, for display purposes, the images show the granule functions $+100$, this converts all images into the range [0,255]

Chapter 4

One-Dimensional Robustness Experiments

Previous chapters have presented a selection of scale-space processors and showed examples of the scale-spaces that are produced from real signals and images. It is easy to find examples of scale-space literature where the quantitative results of a filter's performance are not given and the only indication of system performance is shown with example output images. It is unsatisfactory however, to compare filtering systems based on a simple inspection of these images. A quantitative approach is needed to reliably establish performance under different signal and noise conditions.

This chapter describes the testing procedure from [46] that is used to evaluate the performances of the linear diffusion processor and the different sieve operators in one dimension. The alternative anisotropic and poweroid-based scale-space systems discussed in Chapter 3 are not included as a scale-selection surface is required in the testing procedure.

The systems are tested using synthetic pulse signals that are corrupted with different types of noise. The synthetic nature of the data allows for extensive testing under a wide range of conditions. These experiments are controlled and are easily repeatable. More significantly, it is possible to vary individual parameters of the test signals and observe the effects of these changes on the filtered signal.

The robustness tests are set up as a parametric estimation problem. Given a noise-corrupted test signal containing a pulse, the task of the two systems is to return an estimate of the size, or *scale* of the pulse and an estimate of the position of the pulse along the signal.

The effect of increasing or decreasing different parameters of the noise distributions on the behaviour of the processors is then investigated. The results form the basis for the two-dimensional tests reported in the following chapter.

4.1 One-dimensional synthetic test signals

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*. In Chapter 3, Figures 3.45 and 3.38 show examples of this for both the linear diffusion and $M-$sieve scale-selection surfaces. For one-dimensional signals, the addition of random noise to the test signal will produce small deviations in the location of this maximum and it is the aim of this chapter to investigate this deviation-error for increasing levels of noise.

Figure 4.1 shows two examples of the synthetic noise-corrupted target models that are generated. The signals are 100 samples in length, and the target, a simple pulse, is 20 samples long. A vertical offset leaves a background amplitude of 128 and a pulse intensity of 150. The noise corruptions are *Gaussian* and *impulsive* noise. Gaussian noise corruption is a reasonable model for CCD noise [25, 47] and the top panel of Figure 4.1 shows a typical test-signal where additive Gaussian noise with a mean of zero and a standard deviation of 16 is used to corrupt the underlying pulse, the corruption almost completely masks the original signal.

The second noise type is replacement impulsive noise which is also a common problem in image processing through factors such as 'glint' and pixel drop-out. For this model, random samples along the noiseless pulse signal are replaced with a probability of 0.1. The values of these replacement samples are uniformly distributed in the range $[0, 255]$. The bottom panel of Figure 4.1 shows a typical impulsive noise test signal.

Figures 4.2 and 4.3 show the Gaussian noise signal of Figure 4.1 filtered to increasing scales along with the corresponding scale-selection surfaces. In the linear diffusion scale-space (Equations (3.3) and (3.5)) of Figure 4.2 (top panel), the small-scale noise is quickly smoothed away leaving the smoothed pulse. The signal has been scaled to have zero mean to minimise edge effects. The scale-selection surface (3.60) is shown

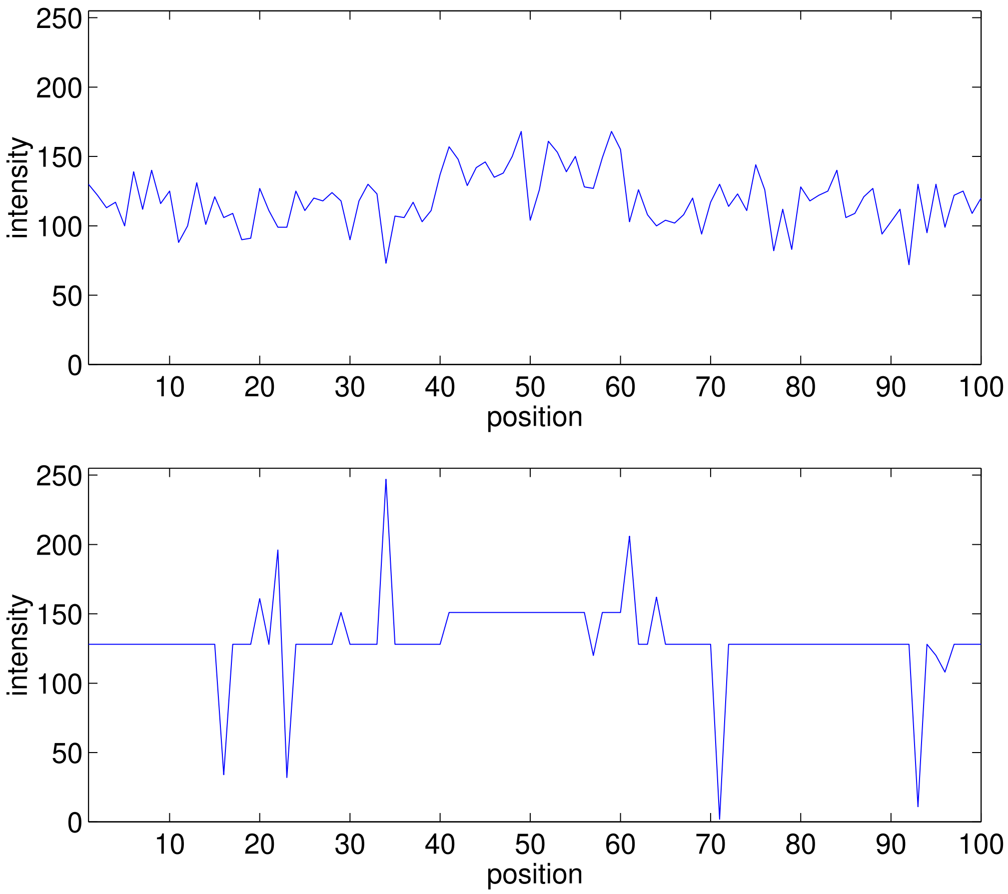


Figure 4.1: One-dimensional target pulses with added Gaussian ($\mu = 0$,$\sigma = 16$) above, and replacement impulsive noise (noise density = 0.1) below.

in the bottom panel. Comparing this figure to the simple-pulse selection surface of Figure 3.45 shows difference that the added noise makes to the signal.

Figure 4.3 shows the same signal filtered by the $M-$sieve defined in (3.52) and (3.55). The scale-space (top panel) shows the gradual removal of signal detail, the signal isn't smoothed as it is in the diffusion case, but is reduced to a series of flat-zones that are removed as scale increases until the signal is completely flat. Note, to reduce the low-pass sieve to a zero-intensity flat zone the signal must be filtered to scale 100, the length of the signal. The sieve's scale-selection surface (3.7.4), in the bottom panel, simply shows the differences between the successive rows of the scale-space representation.

Figures 4.4 and 4.5 show the corresponding images for the impulsive noise target

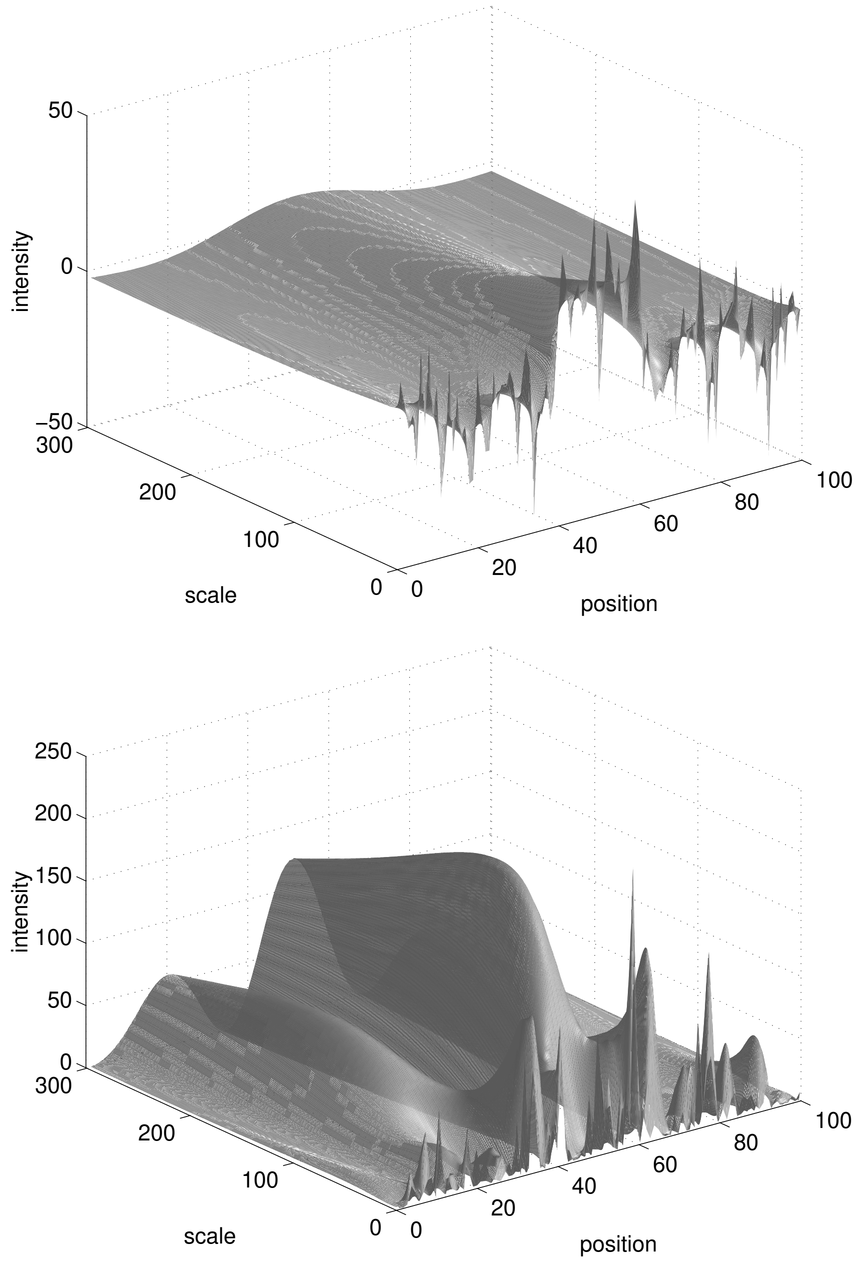


Figure 4.2: One-dimensional target pulses with added Gaussian noise - lowpass filter output for the linear diffusion filter and the corresponding scale-selection surface.

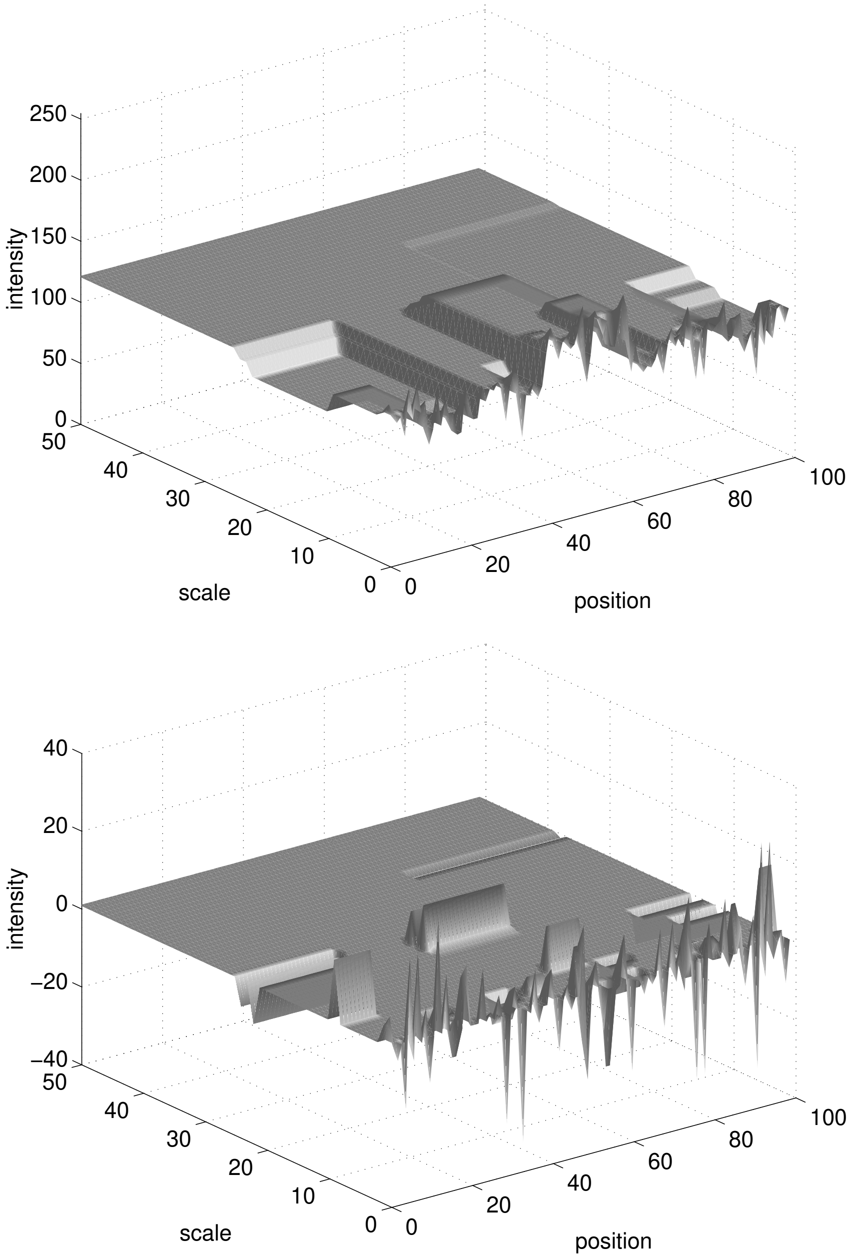


Figure 4.3: One-dimensional target pulses with added Gaussian noise - lowpass filter output for the $M-$sieve filter and the corresponding scale-selection surface.

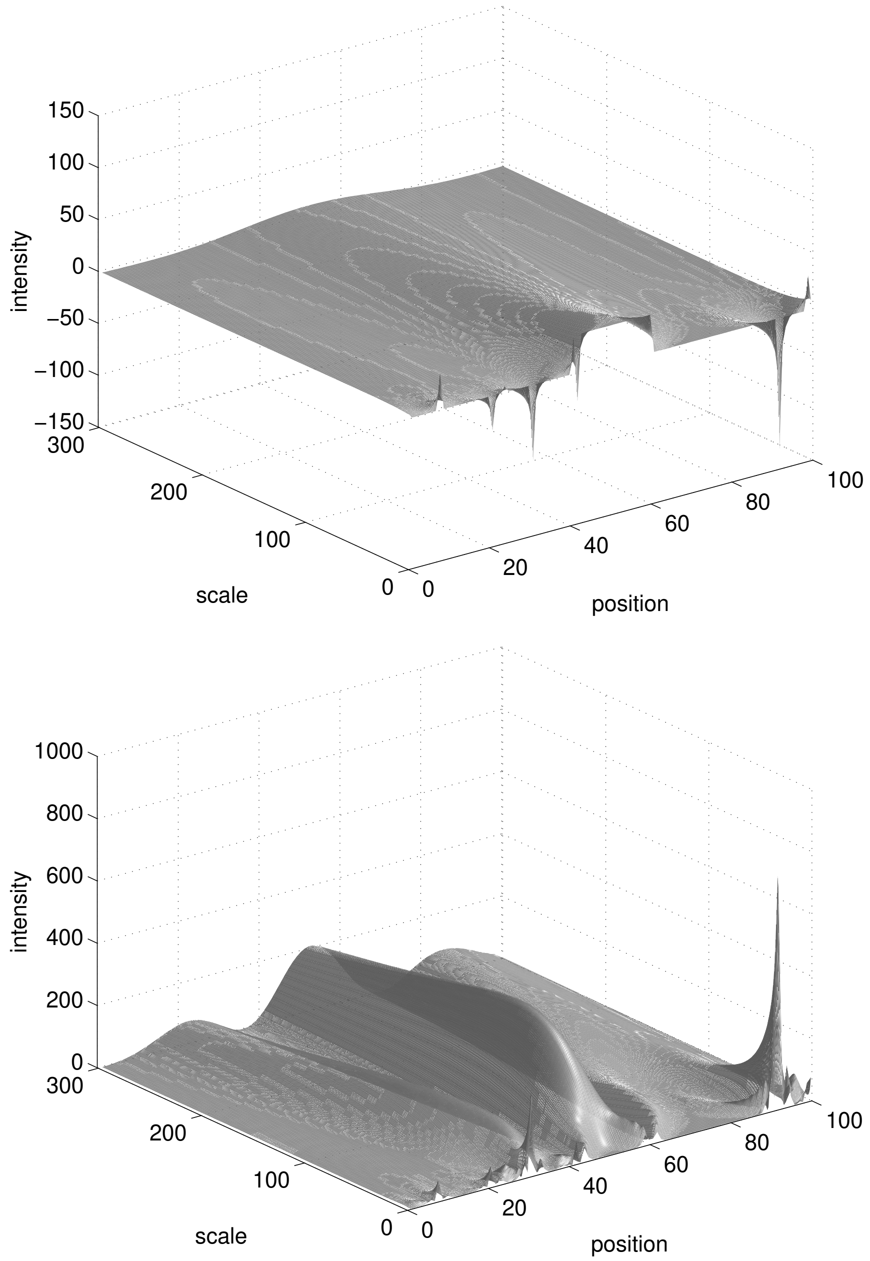


Figure 4.4: One-dimensional target pulse with replacement impulsive noise - low-pass filter output for the linear diffusion filter and the corresponding scale-selection surface.

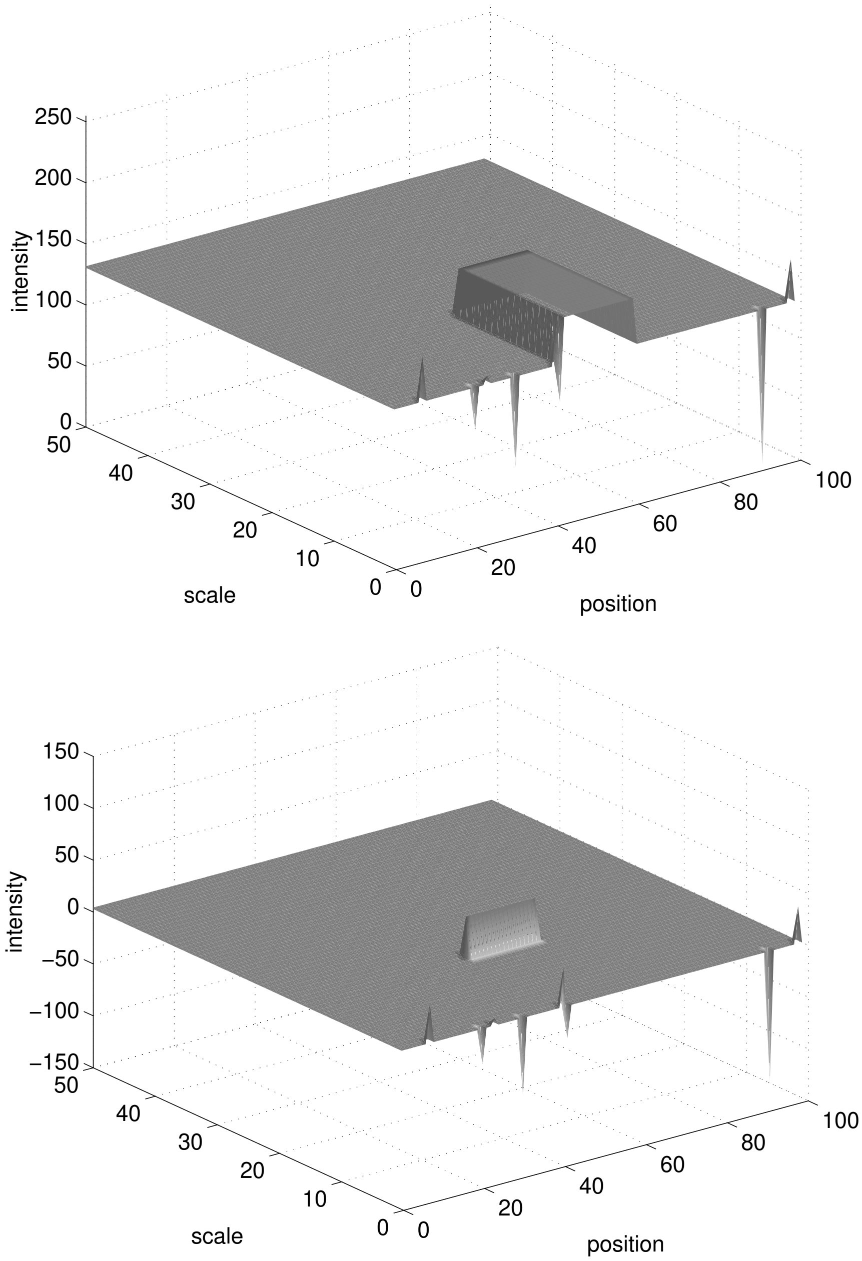


Figure 4.5: One-dimensional target pulse with replacement impulsive noise - low-pass filter output for the $M-$sieve filter and the corresponding scale-selection surface.

signal of Figure 4.1. Here the noise has a large effect on the structure of the linear diffusion scale-space (top panel of Figure 4.4), the large spikes taking a long time to diffuse into the background. The scale-selection surface below contains multiple maxima, and does not give a clear indication of the scale of the target pulse.

The sieve on the other hand removes the majority of the impulsive noise by scale 2. This is shown in the top panel of Figure 4.4, where only the target pulse remains from scale 2 upwards. The scale-selection surface (below) shows the pulse cleanly removed around scale 20.

4.2 Scale and space estimation in one dimension

To quantify the effects of these noise corruptions, the task of the two systems is to make an estimate of the location and scale of the pulse from the scale-selection surfaces. As the previous figures show, the noise that corrupts the signal is mainly limited to small-scales. Selecting search ranges for both processors that avoid the distraction of the noise simplifies the analysis; it is only the deviation in the pulse's maxima point that is of interest here. Without these small-scale extrema, the search simply involves locating the coordinate of the maximum amplitude point over the entire selection surface. The position estimate of the pulse is represented by the position of the maxima along the signal and the scale estimate is taken as the level of filtering required to produce that selection surface maxima.

Repeating this search for multiple test signals gives a good basis for comparisons between, not only the classical linear approach and the sieve method, but also allows us to investigate the differences between the different sieve operators defined in Section 3.6. Searches are made more efficient by dividing the scale range, sampling at set points and centring in on the maximum. A typical search algorithm is given in Appendix A, Figure A.1. The total search range of the filters is set to $[60 - 140]$ for the linear diffusion system and $[12 - 92]$ for the sieve, the *true* scale of the pulse being 102 (see Figure 3.47) and 20 for the respective systems. Examples of some of the results of these searches are shown in Figure 4.6. The probability density functions indicate the spread of the position estimates of a pulse in Gaussian noise for the linear diffusion system and the $M-$sieve filter after 1000 trials.

To improve the precision of the diffusion processor's position estimates, a quadratic

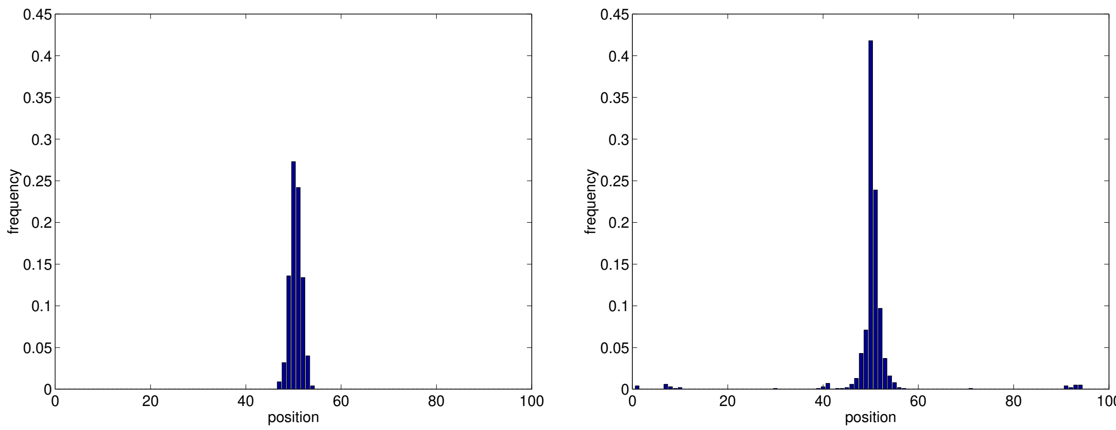


Figure 4.6: Histograms of position estimates for 1000 target signals with added Gaussian noise for the Diffusion processor (left) and M-sieve (right).

surface is least-squares fitted to the selection surface's maximum coordinate and its two-connected neighbours. The maximum of this new surface is recorded as the final position estimate. Since the sieve's scale-selection surface contains flat extrema, the position estimates are recorded as the mean x-coordinate of all samples in the selected granule.

The frequency difference between the two plots of Figure 4.6 is due to the fact that over 10% of the scale estimates from the diffusion filter are outside of the designated search range. These estimates are discarded and so the diffusion system's position estimates in the above figure show

$$\sum_{x=1}^{100} p_{X_{\text{diff}}}(x) = 0.87 \tag{4.1}$$

whereas the sieve-based system returns 1000 estimates,

$$\sum_{x=1}^{100} p_{X_{\text{sieve}}}(x) = 1 \tag{4.2}$$

The variable $p_X(x)$ being the probability that the experimental value of the variable X is equal to x on a single outcome.

The results indicate that 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. This is shown more fully in Table 4.1 where the standard deviation of the search results are recorded. For the diffusion system all estimates that fall at the limits of, or outside of the search range are discarded and the scales of the remaining estimates are normalised to the sample-length of the pulse.

	Gaussian noise		Impulsive noise	
	σ_x	σ_s	σ_x	σ_s
cc	0.9767	1.8091	14.46	4.628
diffusion	1.209	3.867	6.948	4.536
M-sieve	7.01	2.98	3.235	1.595
N-sieve	6.88	2.97	1.797	1.669
ρ-sieve	7.12	2.84	3.164	1.694
ψ-sieve	24.63	9.72	31.85	13.16

Table 4.1: Standard deviations of position, x, and scale, s, estimates for all systems. The normalised cross-correlation, (cc), results are shown for comparison, this system is not a scale-space processor.

The table also shows the results of the *normalised cross-correlation* system (cc). The target signals, (I), are matched with pulses, (W), of increasing scale (length),

$$cc(i) = \sum_{p=-m}^{+m} W(p)I(i+p) \tag{4.3}$$

and are then normalised

$$cc'(i) = \frac{cc(i)}{\left[\sum_{p=-m}^{+m} W^2(p)\right]^{1/2}\left[\sum_{p=-m}^{+m} I^2(i+p)\right]^{1/2}} \tag{4.4}$$

The scale with the highest correlation is recorded, along with a least-squares estimate around the maximum in scale to give the position of the pulse. Note that this is *not* a scale-space system, and is merely presented for comparison with the other processors. Cross-correlation is known to be optimal for position estimation in Gaussian noise, however, as the table shows, the system also gives the best performance of scale estimation in Gaussian noise.

In impulsive noise the systems behave very differently. The $M-$, $N-$ and $\rho-$sieve based systems are hardly affected because the noise is localised to one scale and is therefore removed cleanly. The scale-selection surface is almost completely noise free since the search range is limited so as not include small-scale features. The localisation in scale of the impulsive noise is clearly seen in the lower panel of Figure 4.5. 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 diffusion system, on the other hand, has poor performance in impulsive noise. Limiting the search range to avoid the noise distortion does not help, the sharp noise spikes diffuse slowly and destroy the structure of the entire scale-space. Again, this is seen clearly in the scale-selection surface of Figure 4.4

The opening sieve, (ψ), performs very badly in both noise types since it is biased towards positive-going pulses. The opening operation only removes maxima, leaving the minima noise-spikes untouched. 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.

4.3 Different noise parameters

Table 4.1 only shows results at one standard deviation of Gaussian noise and at one noise density of impulsive noise and is therefore inconclusive. To investigate the effect of altering the noise distributions the experiments are repeated for different noise levels.

For the Gaussian noise test, the standard deviation of the noise is varied $(2 \leq \sigma \leq 26)$, and for the impulsive noise tests the noise density (probability that a signal element is corrupted) is tested in the range $[0, 0.1]$. Running these 1000-trial tests an extra six times gives some indication of the variation between the results. The standard deviation of these extra tests are plotted as error bars for the different levels of noise (only ± 1 standard deviation is shown). The results are shown in Figures 4.7 and 4.8; the position estimate results are in the top panels and the scale estimates are in the bottom panels.

In Gaussian noise the sieve proves to be better at scale estimation than the diffusion filter for all but the highest noise levels. However in position estimation the diffusion 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 impulsive noise the sieve consistently betters the performance of the diffusion filter. The noise density of 0.1 that was used in previous tests was too high for the linear

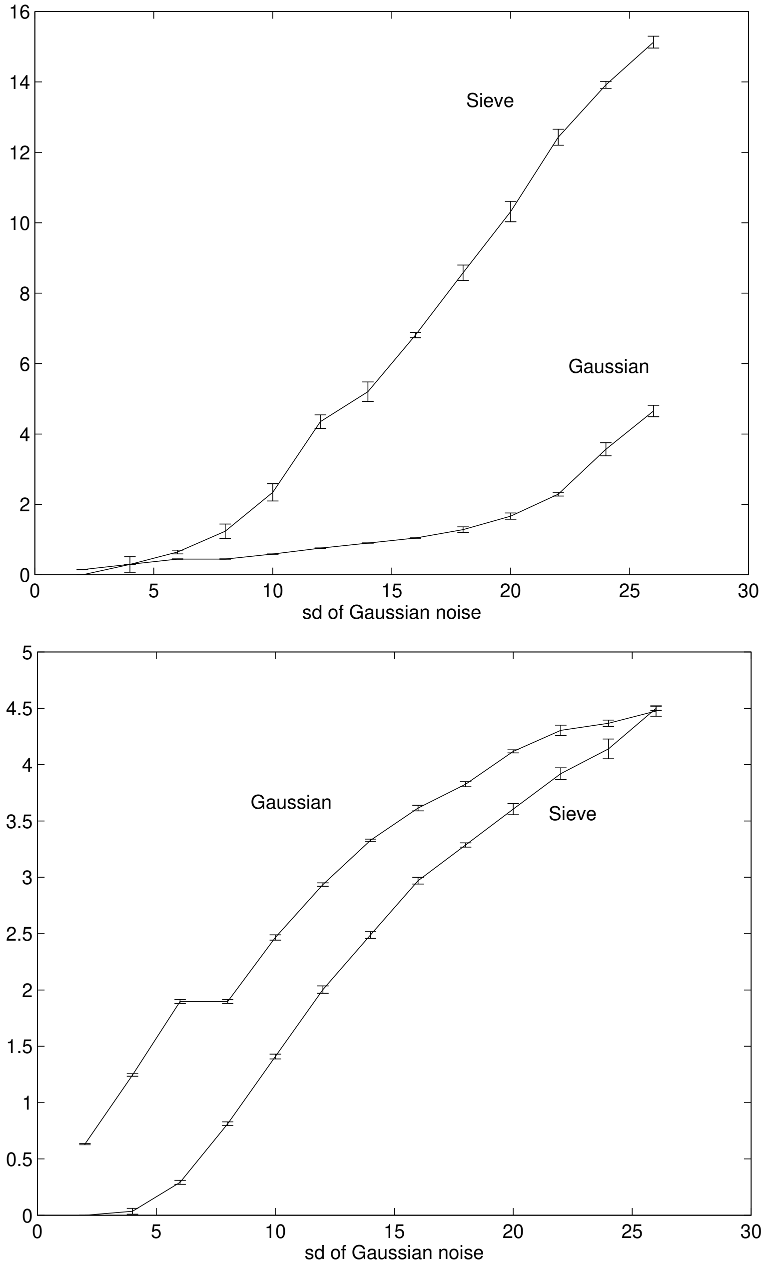


Figure 4.7: Effect of varying σ of the Gaussian noise, above are the position estimates and below are the scale estimates. Vertical axis corresponds to the standard deviation of the estimates. The sieve results are those of the M-sieve.

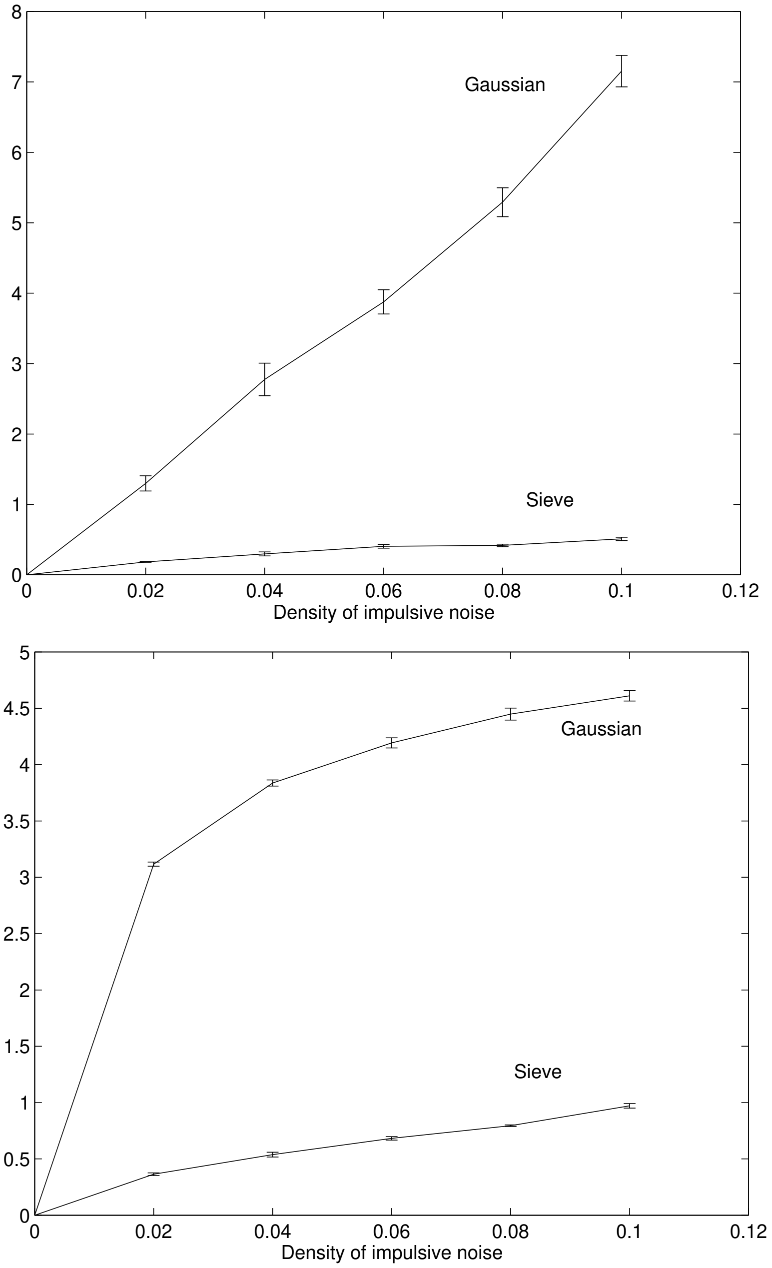


Figure 4.8: Effect of varying the noise density of the impulsive noise, above are the position estimates and below are the scale estimates. Vertical axis corresponds to the standard deviation of the estimates. The sieve results are those of the M-sieve.

system. Testing at these lower noise levels, the rapid deterioration of the diffusion system is apparent in both scale and position estimation. For noise levels above this range the scale estimates of the diffusion system are flat across the search range and the target cannot be located.

4.4 Summary

This chapter has compared the robustness of the linear diffusion filter and the different sieve operators in one dimension. The tests involved the location and scale estimation of a noise-corrupted target pulse-signal. Multiple versions of these signals were then processed to give a statistically reliable degree of accuracy.

The results show that in the presence of Gaussian noise all of the systems give a similar performance, whilst in the presence of impulsive noise the M, N, and the recursive median ρ- type systems out-perform both the linear diffusion-based filter and the less robust opening and closing sieve filters. These results augment and strengthen the results obtained in two-dimensions which are presented in the following chapter.

The two-dimensional tests that follow are the natural extension of those completed in one dimension since the two-dimensional diffusion processor is used in many practical applications [28] and it is important to understand how the system may be affected by common image noise. Chapter 5 also focuses on an additional, and often overlooked, problem in image processing; that of occlusion.

Chapter 5

Two-Dimensional Robustness Experiments

This chapter is divided into two main parts. The first is follows on directly from the previous chapter and extends the one-dimensional robustness tests into two dimensions, again comparing the performances of the scale-space processors under conditions of altering degrees of image noise. The second part of this chapter broaches the problem of occlusion in images. Image occlusions are a major problem to many computer vision systems and, depending on the degree of the occlusion, can even cause the human vision system to misinterpret a scene. This second section aims to investigate the problem of occlusion, using simplified image models, and attempts to evaluate the effect of known occlusions on the overall scale-space decomposition. This is a review of the work that was first presented by the author in [27, 43–45]

5.1 Two-dimensional noise experiments

All of the scale-space processors that are discussed in this thesis are easily extended from one to two dimensions and so it is only natural to continue the robustness investigation that considered waveforms in the previous chapter, to two-dimensional images.

5.1.1 Two-dimensional synthetic test signals

The two-dimensional target signals that are used to test the diffusion and sieve-based systems are similar to those used in one dimension. The 100×100 pixel image models are shown in Figure 5.1. In each case there is an underlying image of a disc of area 1264 pixels with intensity 144 on a flat background of intensity 112. This image is then corrupted with two types of noise. On the left is is an example of additive Gaussian noise with parameters $\mu = 0$ and $\sigma = 24$, and on the right an example of replacement impulsive noise in the range $[0, 255]$ with a noise density of 0.1. All images are clipped into the range $[0, 255]$, however the minimal degree of clipping required does not significantly alter the distribution of the noise.

Again, as in the previous chapter it is the task of the linear diffusion-based and sieve-based systems to make an estimate of the scale and the position of the target, which is in this case the central disc. The images are filtered to increasing scales. For the diffusion-based scale-space the images are filtered between the scales $[74 - 410]$ and for the sieve operators (scale now referring to pixel area) $[800 - 4384]$, the filtering ranges being restricted to avoid the distractions of the small-scale image noise. The scale-selection surfaces of these images are then searched for maxima, the maxima over the entire surface representing the location of the target disc.

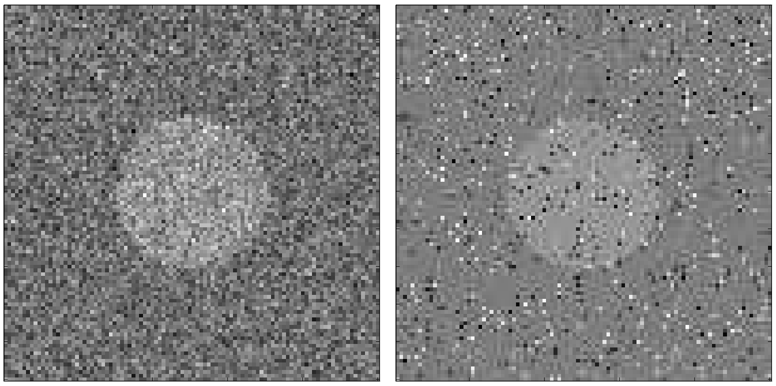


Figure 5.1: Two-dimensional noisy target images. Left: additive Gaussian noise, $(\mu = 0, \sigma = 24)$, right: replacement impulsive noise in the range $[0 - 255]$. Image size is 100×100 pixels

Figures 5.2 and 5.3 show the results of these searches for 150 test images. Each search returns one estimate of the scale and position of the target disc. The scale estimate is determined by the peak in the scale-selection surface over all scales and the position estimate is the spatial position of this peak. Since the granules of the sieve's selection surface are flat zones, the spatial estimate is recorded as the centroid, or the centre of mass of the maximum amplitude granule. In the case of the linear diffusion system the precision of the spatial estimate is improved through a quadratic least squares fit to the maximum coordinate of the selection surface and its 8-connected neighbours. The mean of the scale estimates from the linear diffusion filter are then normalised to the true scale of the disc (1264 pixels) to help compare the two systems.

The actual scale and position estimates are marked on the three-dimensional plots by black circular markers. To help judge the spread of the estimates in each dimension these points are projected onto the three walls of the figure. The projections are shown by the green dots. The first panel of Figure 5.2 shows the scale-space of the diffusion processor for the Gaussian noise test images and below are the corresponding sieve results. The two systems perform similarly under these conditions; the sieve has slightly more accurate position estimates and the diffusion system has more accurate scale estimates.

In the case of the impulsive noise tests (Figure 5.3), the sieve (lower panel) outperforms the diffusion processor in both scale and position estimation. The linear system's estimates deteriorate as the noise spikes diffuse over the image. However, there is still a large improvement from the linear system's performance in one dimension. This is due to the increased spatial averaging causing the noise spikes to diffuse more quickly. In impulsive noise the sieve repeats its performance in one dimension. Since the noise is localised to one scale, and the search ranges are purposely set to avoid these scales, the only deviations in the scale and position estimates arise when the replacement noise spikes fall on or next to the the edge pixels of the disc. In this situation the max and min operations either raise the pixel intensity to that of the disc or lower it to the intensity of the background. These discrepancies make little change to the overall result from 150 test images.

Table 5.1 shows the standard deviation of the results of Figures 5.2 and 5.3 along with the results of the additional sieve operators. As in the one-dimensional experi-

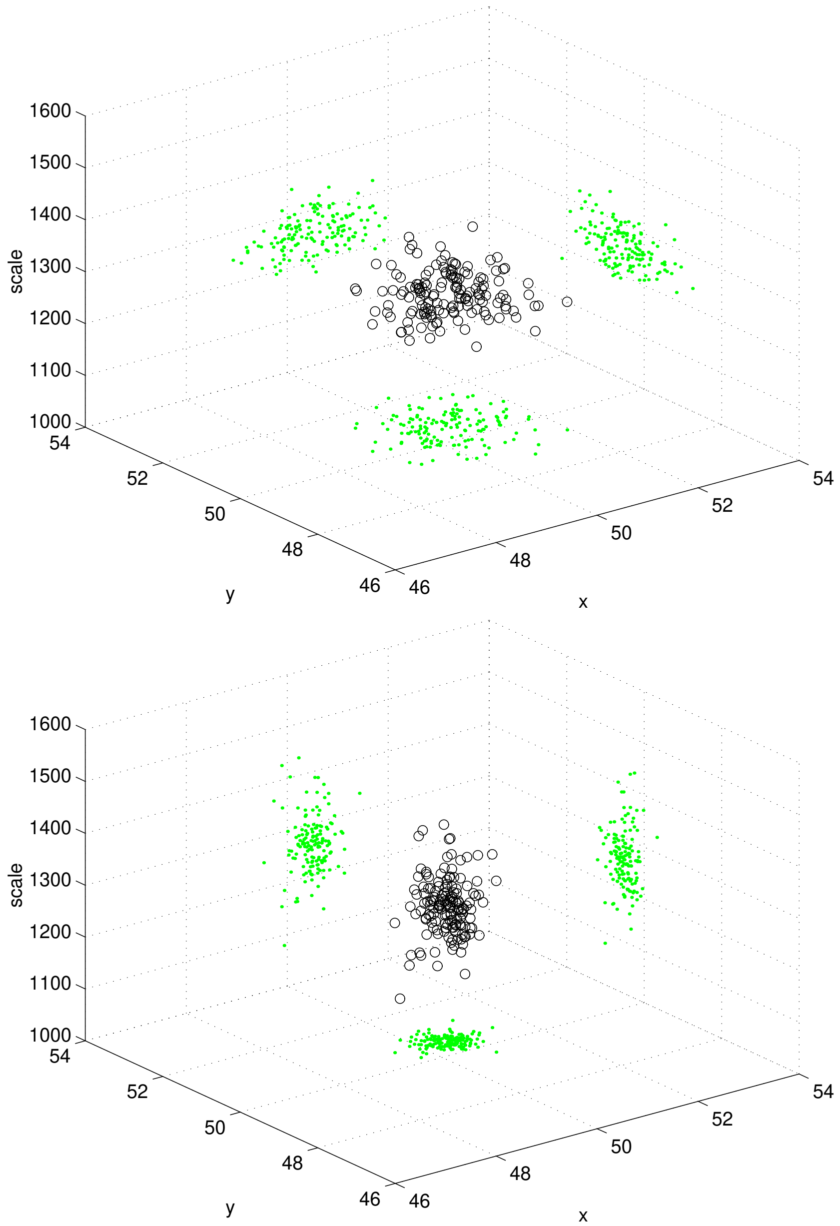


Figure 5.2: Scale-space estimates of disc for 150 synthetic images with added Gaussian noise with parameters ($\mu = 0, \sigma = 24$). (See left hand panel of Figure 5.1). Top panel linear diffusion results, bottom panel $M-$sieve results.

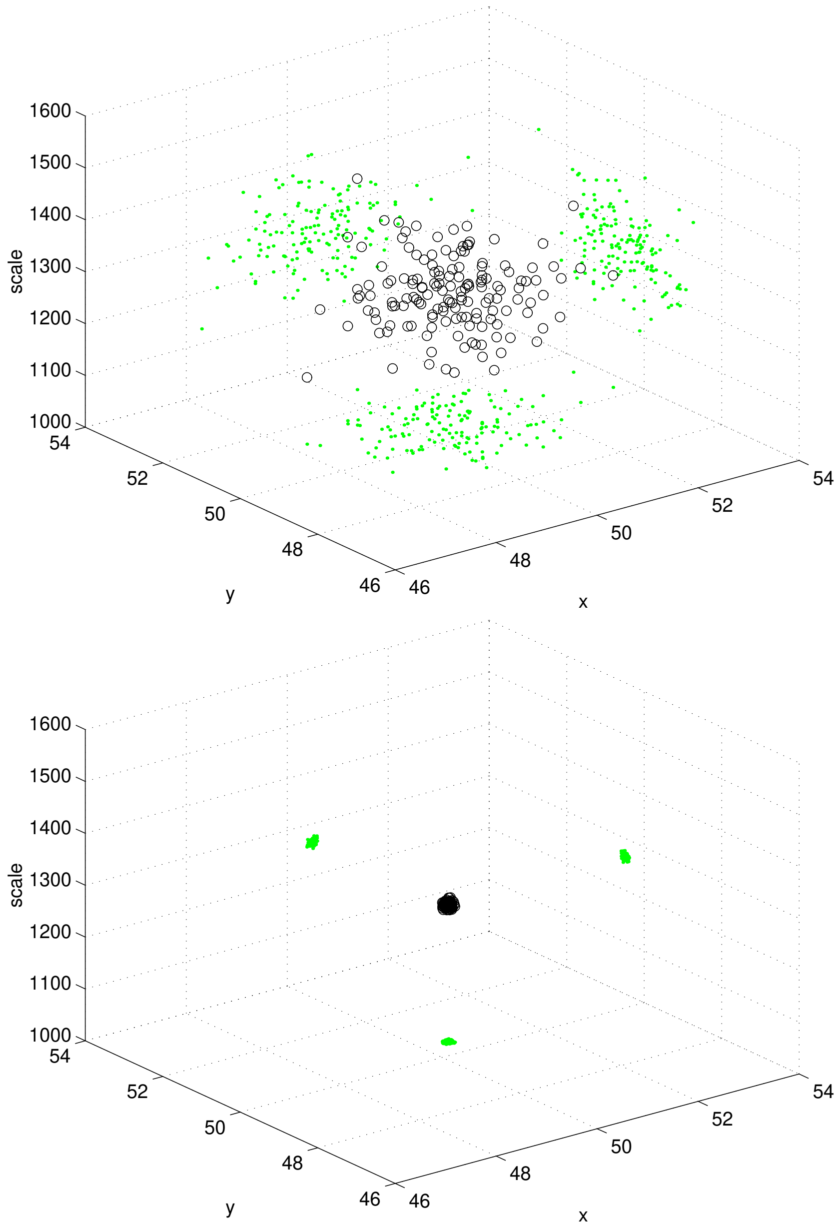


Figure 5.3: Scale-space estimates of disc for 150 synthetic images with replacement impulsive noise in the range $[0, 255]$ with a noise density of 0.1. (See the right-hand panel of Figure 5.1). Top panel diffusion results, bottom panel $M-$sieve results.

	Gaussian noise			Impulsive noise		
	σ_x	σ_y	σ_s	σ_x	σ_y	σ_s
cc	0.21	0.20	30.3	0.27	0.33	35.9
diffusion	0.64	0.63	32.4	0.93	0.79	59.9
$M-$sieve	0.24	0.24	58.9	0.046	0.038	3.90
$N-$sieve	0.26	0.25	48.7	0.040	0.045	4.30
$\rho-$sieve	0.28	0.24	55.0	0.043	0.042	3.91
$\psi-$sieve	2.88	1.14	105.8	0.076	0.071	8.894

Table 5.1: Standard deviations of 150 estimates of the position (x, y) and scale, s of a disc in Gaussian and impulsive noise for normalised cross-correlation (cc), linear diffusion, M-, N-, ρ- and ψ-sieve.

ments, the normalised cross-correlation system,

$$cc(i,j) = \sum_{p=-m}^{+m} \sum_{q=-n}^{+n} W(p,q) I(i+p, j+q) \tag{5.1}$$

$$cc'(i,j) = \frac{cc(i,j)}{\left[\sum_{p=-m}^{+m} \sum_{q=-n}^{+n} W^2(p,q)\right]^{1/2} \left[\sum_{p=-m}^{+m} \sum_{q=-n}^{+n} I^2(i+p, j+q)\right]^{1/2}} \tag{5.2}$$

is included simply as a benchmark guide since the system is not causal and cannot be recognised as a scale-space processor. For this cross-correlation system the target images are matched to a series of increasing scale discs, where scale is represented by the disc's radius. The disc that produces the highest correlation provides the scale estimate and an 8-connected least-squares fit around the maximum of the correlation surface provides a spatial position estimate.

The results of Table 5.1 show that, in Gaussian noise, the diffusion system and the $M-$, $N-$, and $\rho-$sieves approach the performance of the benchmark normalised cross-correlation system. Opening- or $\psi-$sieves, in which increasing scale greyscale openings are repeatedly applied, have the worst performance. This is to be expected since it is known that opening operations are sensitive to noise. The result of multiscale closing is not shown since the closing operator cannot resolve a positive disc under these conditions. (The closing operation removes minima, leaving maxima intact.)

In the case of impulsive noise, all sieve operators can produce better scale and position estimates than those from the diffusion system. Again, the asymmetry of multiscale closings means that the experiment would have to be specifically tuned to handle positive targets.

5.1.2 Target-shape dependency for scale-space estimation

In moving from one- to two-dimensional signals the idea of *target-shape* is introduced. The choice of a disc shaped target is not accidental; the disc is radially symmetrical and therefore has a minimal effect on the filter outputs. This is not the case for most target shapes.

It is the diffusion filter that is particularly sensitive to the shape of the target function. The different sieve operators filter purely with respect to area and so morphological changes in the target simply result in a possible change in the pixel area or scale of the target and a change in the target's centre of mass or position. Changing the shape of the target will not affect the localisation properties of the sieve. However, for the diffusion filter, as defined in Section 3.2, the target's shape *is* important for position and scale estimation. The system is implemented using separable one-dimensional kernels of similar scale and so, accurate estimates are only obtained for targets that are invariant through a rotation of $90°$. If the target is non-invariant through this rotation, such as in the case of a rectangle, then a diffusion kernel that is tuned to the correct scale for the width of the rectangle will not be correctly tuned for the length of the rectangle.

Figure 5.4 shows an example of a tuned two-dimensional diffusion kernel that may be used to locate a rectangle of specified width and length. The final kernel (bottom left and right) is the product of two one-dimensional kernels (top right) that are tuned to locate pulses of length equal to the rectangle's length and width. Although this two-dimensional kernel is easily generated if the dimensions of the rectangle are known, it is infeasible to locate and estimate the scale of a rectangular (or indeed any non-symmetrical) target of unknown dimensions. Also, there is an additional problem of labelling the scale of these tuned kernels since they are composed of one-dimensional kernels of different scales, this scale-label is required for the scale-selection surface in (3.60).

For these reasons, all of the targets used in this thesis, whether synthetic or real, are radially symmetrical. These targets may therefore be located using separable kernels of similar scale.

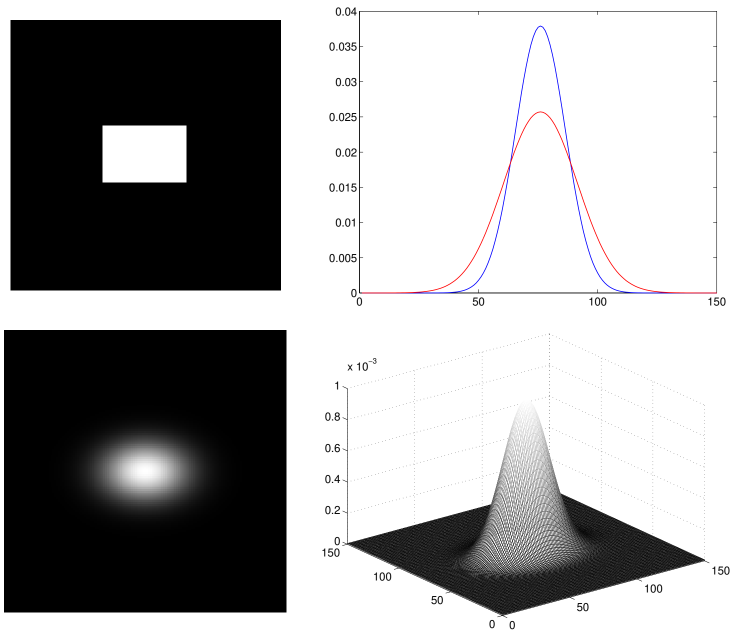


Figure 5.4: Shape dependency for the linear diffusion system. Top-left: image with rectangular target, top-right: Diffusion kernels of scale $s = 111$ and $s = 241$. These are the *optimal* scale one-dimensional kernels for the width and length of the rectangle. Bottom-left shows the two-dimensional kernel created by multiplying the one-dimensional kernels, bottom-right shows the surface intensity plot of this tuned two-dimensional kernel.

5.1.3 Different noise parameters

As in the one-dimensional experiment chapter it is not sufficient to test the reliability of these two-dimensional systems under a single set of noise conditions. Figure 5.5 shows the effect of varying the standard deviation of the Gaussian noise between $[2, 26]$ for both the linear diffusion filter (shown in blue) and the $M-$sieve operator (shown in red). The plots show that at all of the tested noise levels, the errors in the (x, y) position estimates for the sieve are less than those for the diffusion filter. However, the results of the scale estimates show that there is a crossover point in the behaviour of the systems. For target images with a noise standard deviation above 14 the linear diffusion filter returns the more accurate scale estimates, however, for noise standard deviations below 14 the sieve-based system is the more accurate system.

The effects of altering the parameters of the impulsive noise target images are shown in Figure 5.6. The noise density is varied between $[0.05, 0.275]$ and, again, the standard

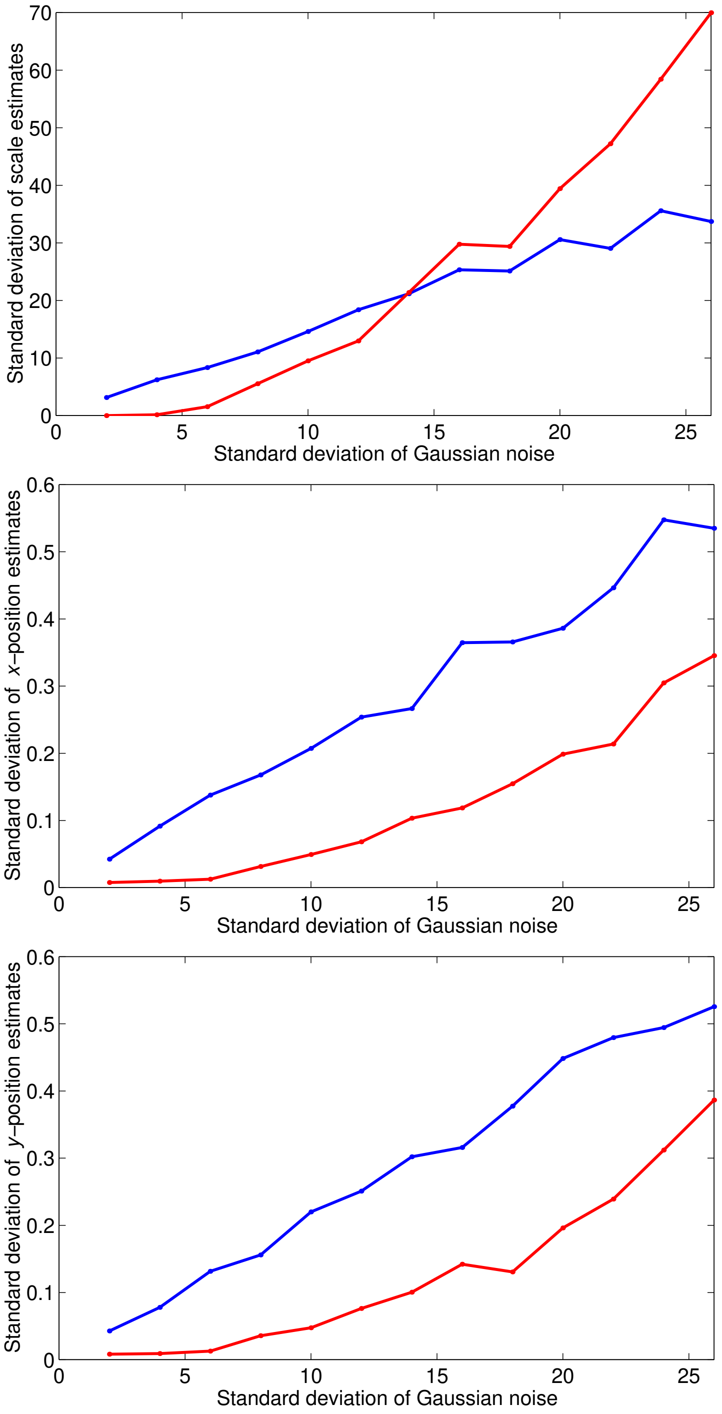


Figure 5.5: The effect of varying the standard deviation of the additive Gaussian noise in the target images. The plots show the standard deviation of the estimates against the standard deviation of the noise. The linear diffusion results are shown in *blue* and the $M-$sieve results are in *red*. Top panel: scale estimates, middle panel: x-position estimates, lower panel: y-position estimates.

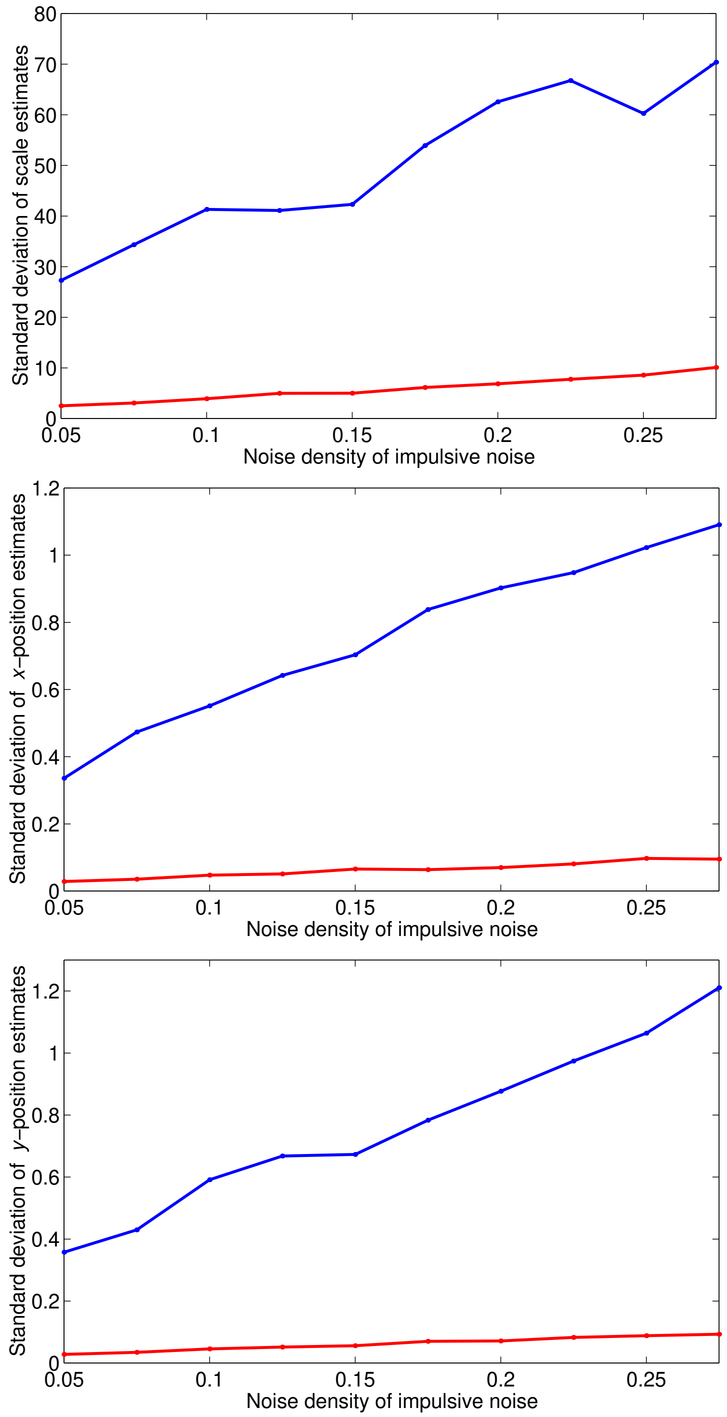


Figure 5.6: The effect of varying the noise density of the replacement impulsive noise in the target images. The plots show the standard deviation of the estimates against the density of the noise. The linear diffusion results are shown in *blue* and the $M-$sieve results are in *red*. Top panel: scale estimates, middle panel: x-position estimates, lower panel: y-position estimates.

	Gaussian noise (n)	$\sigma(\sigma_x)$	$\sigma(\sigma_y)$	$\sigma(\sigma_s)$
Diffusion	$\sigma_n = 2$	0.0029	0.0012	0.1214
	$\sigma_n = 26$	0.0268	0.0197	2.1756
Sieve	$\sigma_n = 2$	0.0002	0.0003	0.0002
	$\sigma_n = 26$	0.0232	0.0327	4.4754
	Impulsive noise	$\sigma(\sigma_x)$	$\sigma(\sigma_y)$	$\sigma(\sigma_s)$
Diffusion	density $= 0.05$	0.0186	0.0290	1.0828
	density $= 0.275$	0.8994	0.9097	43.7731
Sieve	density $= 0.05$	0.0018	0.0015	0.1514
	density $= 0.275$	0.0076	0.0083	0.8718

Table 5.2: Error in the standard deviation of position and scale estimates for multiple trials with Gaussian noise corrupted images ($\sigma = 2, 26$). Errors are given to ± 1 standard deviation for 6 trials.

deviation of the estimates from the diffusion filter and the $M-$sieve are recorded. The plots show that in both (x, y) position estimation and in scale estimation the sieve filter gives the better performance. This is to be expected based on the sieve's performance in the one-dimensional noise tests. The noise, which is localised to one scale, is removed cleanly and is only apparent in the scale-selection surface as a slight perturbation in the pixel area of the disc. It is this change in area that causes the discrepancy between the actual scale and position of the disc and the returned estimates.

The diffusion filter does have a more reliable performance in two dimensions. This is mainly due to the increased spatial averaging. However, the impulsive noise still badly corrupts the structure of the diffusion system's scale-selection surface. At a noise density of 0.275 (shown in Figure 5.6) 10% of the filter's estimates are unresolvable, in these instances the scale of the disc is estimated as being outside of the designated search range. For noise densities above 0.275 this percentage increases quickly until, for densities of 0.5, under 4% of the estimates are valid. To increase the range of the searches would serve no useful purpose since the correct scale of the disc, obtained from a noiseless target, is well within the allowable search range and an increase in this range also involves an increase in the processing time.

Adding error-bars to the plots of Figures 5.5 and 5.6 involves large amounts of computation, however, to give an idea of the variance of the plotted points Table 5.2 shows estimates of the error in the standard deviation for 6 trials (each over 150 images) at Gaussian noise standard deviations of 2 and 26 and at impulsive noise densities of 0.05

and 0.275. These results reinforce the evidence of the worsening sieve and diffusion performance in high levels of Gaussian noise and the unreliability of the diffusion system estimates in high-density impulsive noise.

Varying the target intensity

As an additional test, these noise experiments are repeated for different target amplitudes, and also random target amplitudes. Until this time the parameters of the underlying target signal have remained unchanged, however in real images this will rarely be the case. Changes in lighting and surroundings may cause significant intensity fluctuations in an image.

Disc amplitude	Processor	Gaussian noise			Impulsive noise		
		σ_x	σ_y	σ_s	σ_x	σ_y	σ_s
130	diffusion	1.05	0.86	69.0	0.99	1.02	67.7
130	$M-$sieve	1.19	3.65	144.4	0.05	0.05	4.22
144	diffusion	0.64	0.63	32.4	0.92	0.79	57.9
144	$M-$sieve	0.24	0.24	58.9	0.05	0.04	3.90
150	diffusion	0.42	0.42	31.3	0.44	0.47	30.3
150	$M-$sieve	0.21	0.19	40.1	0.05	0.05	4.31
160	diffusion	0.35	0.35	23.8	0.40	0.37	26.5
160	$M-$sieve	0.13	0.11	25.7	0.04	0.04	4.56
170	diffusion	0.29	0.27	17.4	0.34	0.34	21.6
170	$M-$sieve	0.09	0.10	16.9	0.04	0.04	4.13
Random	diffusion	1.33	1.34	99.28	1.41	1.41	110.18
Random	$M-$sieve	9.60	9.88	157.66	0.04	0.04	3.96

Table 5.3: Standard deviations of estimates in Gaussian noise and impulsive noise for target discs of different amplitude. The images were used to test the linear diffusion system and the M–sieve based system.

Table 5.3 shows the effect of repeating the noise tests of Section 5.1.1 for additional target disc intensities of [130, 150, 160 and 170] and for a separate random-amplitude test. The distributions of the noise corruptions for these tests are ($\mu = 0, \sigma = 24$) for the Gaussian noise and impulsive noise of density (0.1) in the range $[0, 255]$.

The results of Table 5.3 are shown in Figures 5.7 and 5.8. In Gaussian noise (Figure 5.7), for small intensity targets, the diffusion processor is superior to the $M-$sieve in both position and scale estimation. As the intensity of the disc increases the spatial estimates of the sieve improve, and the scale estimates begin to approach those of the diffusion processor, until, for the largest intensity discs the processor's performances are similar.

	Gaussian noise	$\sigma(\sigma_x)$	$\sigma(\sigma_y)$	$\sigma(\sigma_s)$
Diffusion	Target amp $= 130$	0.0547	0.0233	2.8761
	Target amp $= 170$	0.0105	0.0054	0.5780
Sieve	Target amp $= 130$	0.9548	0.9467	11.3214
	Target amp $= 170$	0.0069	0.0064	0.7415
	Impulsive noise	$\sigma(\sigma_x)$	$\sigma(\sigma_y)$	$\sigma(\sigma_s)$
Diffusion	Target amp $= 130$	0.0676	0.0380	2.1809
	Target amp $= 170$	0.0223	0.0201	0.9101
Sieve	Target amp $= 130$	0.0029	0.0037	0.3074
	Target amp $= 170$	0.0013	0.0039	0.2196

Table 5.4: Error in the standard deviation of position and scale estimates for multiple trials with Gaussian noise corrupted images ($\sigma = 2, 26$). Errors are given to ± 1 standard deviation for 6 trials.

For discs corrupted with impulsive noise (Figure 5.8) the amplitude of the target has no effect on the sieve-based filter. The image decomposition is invariant to relative changes in amplitude. Only changes that affect the ordering of the merging extrema will change the decomposition. The impulsive noise is localised to small-scales and is easily removed and, although the performance of the diffusion processor improves with increasing disc amplitude, the sieve remains superior.

As with the previous varying noise distribution experiments, lengthy computation time prevents the use of error-bars on all points of the plots of Figures 5.7 and 5.8. However, Table 5.4 gives an estimate of the error in the standard deviation for 6 trials (each for 150 images) with target amplitudes of 130 and 170 in both Gaussian and impulsive noise.

The random-amplitude disc tests simply re-enforce the point that the sieve is invariant to the amplitudes of extrema points within an image; the results for the Gaussian noise estimates deteriorate, as do the diffusion results in impulsive noise. Only the sieve-based estimates in impulsive noise remain unchanged.

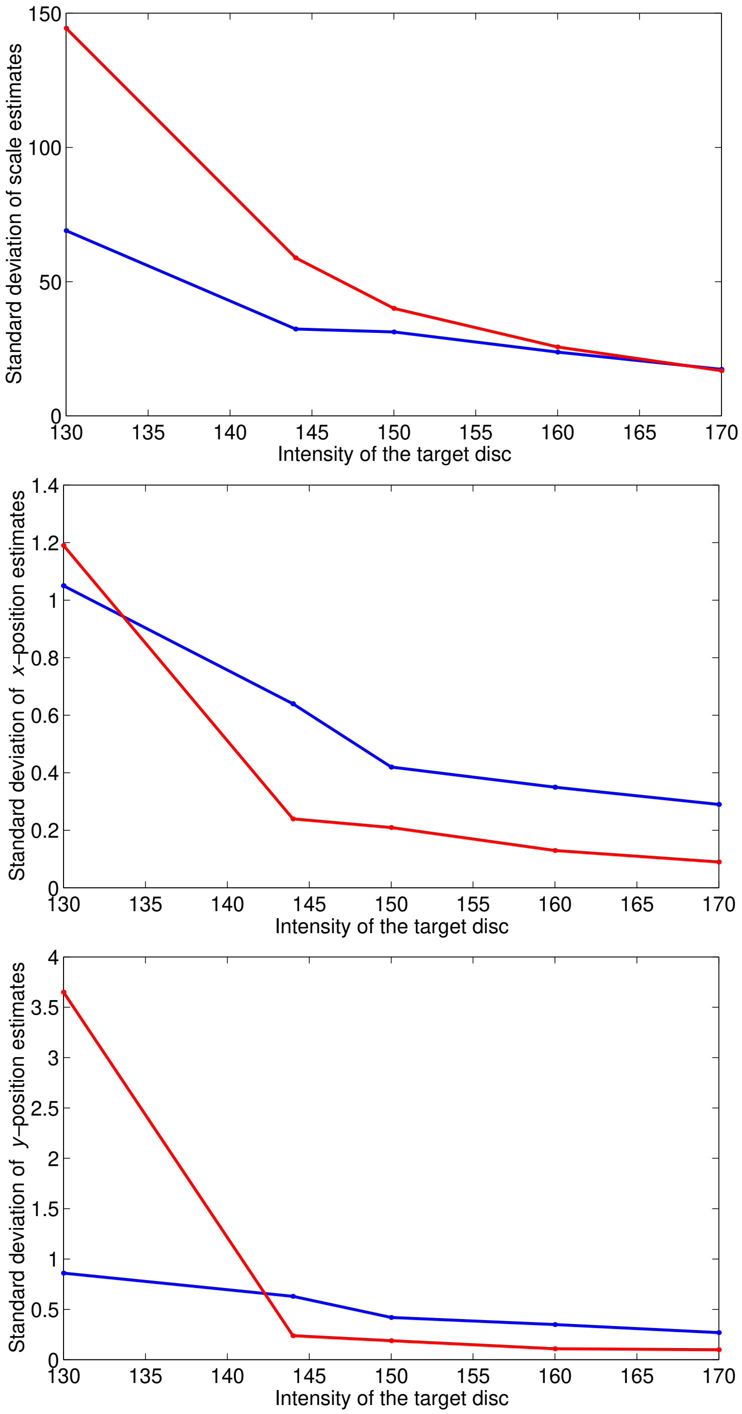


Figure 5.7: The effect of varying the intensity of the target disc in the Gaussian noise target images. The plots show the standard deviation of the estimates against the disc intensity. The linear diffusion results are shown in *blue* and the $M-$sieve results are in *red*. Top panel: scale estimates, middle panel: x-position estimates, lower panel: y-position estimates.

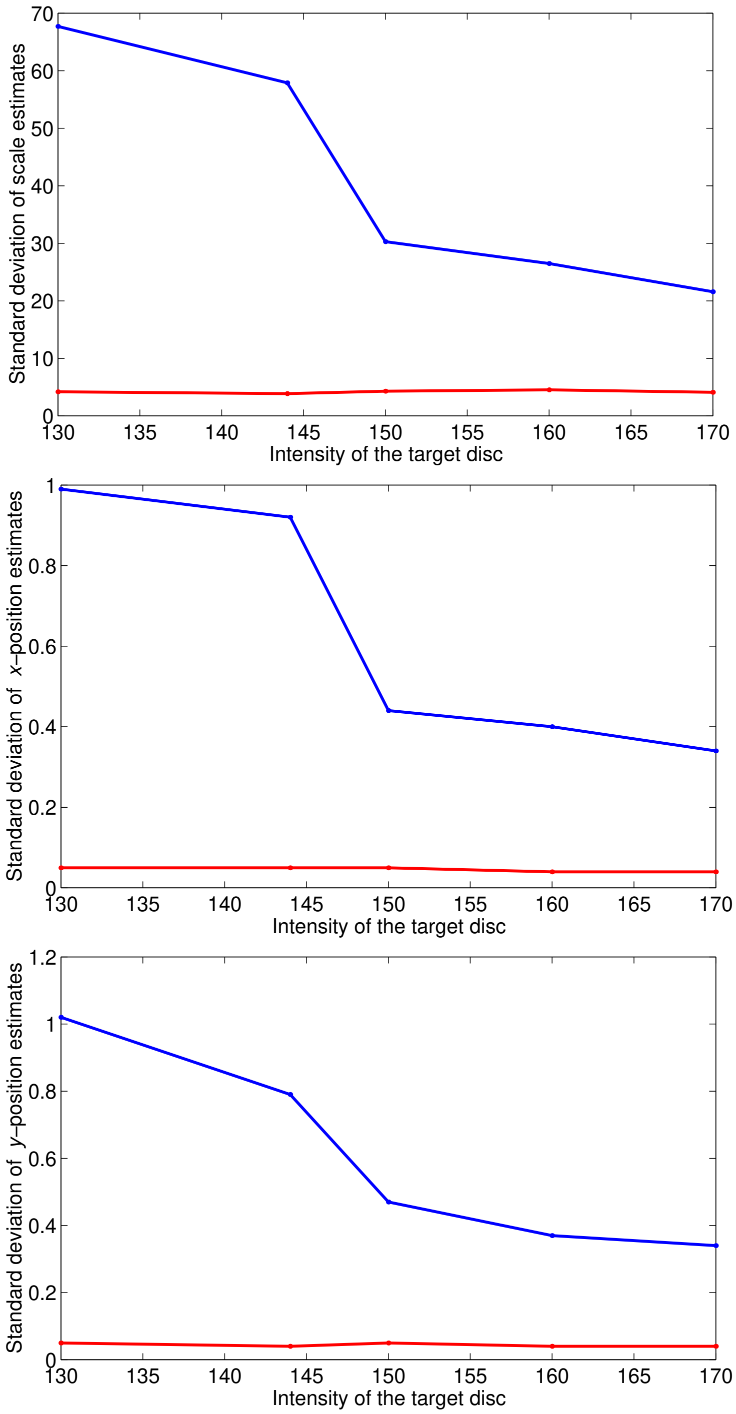


Figure 5.8: The effect of varying the intensity of the target disc in the impulsive noise target images. The plots show the standard deviation of the estimates against the disc intensity. The linear diffusion results are shown in *blue* and the $M-$sieve results are in *red*. Top panel: scale estimates, middle panel: x-position estimates, lower panel: y-position estimates.

5.1.4 Anti-aliasing the discs

It could be argued that the synthetic *two-level* nature of the disc favours the sieve processor and so presents an unfair advantage over the diffusion system. Indeed the sieve removes a noiseless disc cleanly and stores it in a granule function at a scale equal to the pixel area of the disc. The diffusion processor, on the other hand, is more closely matched to Gaussian-shaped features and its performance suffers slightly with sharp edged features.

With this in mind the two systems are tested with *anti-aliased discs*, corrupted with the original noise distributions (Gaussian noise with ($\mu = 0, \sigma = 24$) and impulsive noise with a density of (0.1), see Figure 5.1). Anti-aliasing provides a means of reducing the *staircasing* effect on the edge pixels of the original disc, where any pixel that touches the area of the theoretical-disc is set to full disc intensity and all other pixels are left untouched. Anti-aliasing is usually implemented by some low-pass filter, cutting off the higher frequencies and so limiting the jagged-edge effect by changing the gradations in intensity of the disc-edge pixels.

There are many anti-aliasing algorithms available [39], the four examples shown in Figure 5.9 range from simple diffusion, smoothing the high-contrast gradients, to the more computationally expensive Gupta-Sproull algorithm [41]. The four methods are described briefly below:

- **Diffusion filtering:** The noiseless 100×100 pixel image is filtered with a discrete analogue to the Gaussian kernel of (3.3) with a scale of 1. (Shown in Figure 5.9 second row, left-hand panel).

- **Supersampling followed by block averaging:** The noiseless image is supersampled to a size of 800×800 pixels. This is followed by a block-averaging operation (using non-overlapping blocks of size 8×8 pixels). Every eighth pixel is then used to produce a 100×100 anti-aliased image. (Shown in Figure 5.9 second row, second panel).

- **Weighted averaging filter:** A supersampled high resolution 201×201 image is filtered with the mask of (5.3), allowing only the edges of the filter-block to overlap. The filtered image is then sub-sampled to a 100×100 image. (Shown in

Figure 5.9 second row, third panel).

$$\frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array} \tag{5.3}$$

- **Gupta-Sproull algorithm:** Each of the pixels in the 100×100 noiseless image is considered to be a cone of unit volume and radius r, centred at the pixel centre. The volume of cone cut off by the theoretical disc-edge is used calculate the intensity of the disc-edge pixels. For instance, if over half of the cone lies inside of the disc boundary, then the pixel intensity will be 1. If a quarter of the cone's volume lies inside the disc, then the corresponding pixel amplitude is 0.5. (Shown in Figure 5.9 second row, right-hand panel). The algorithm is explained in more detail in Appendix C.

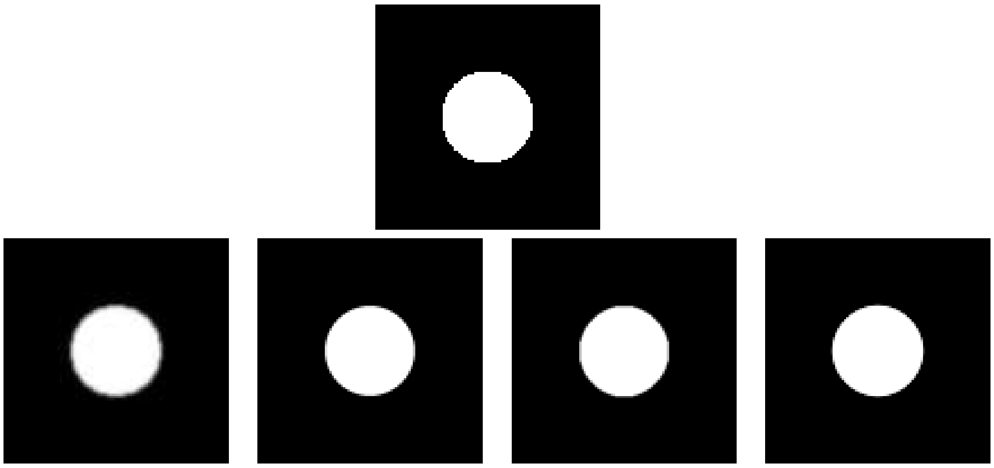


Figure 5.9: Original disc and four anti-aliased discs. Top panel: Original disc. Second row, left-hand panel: diffusion filtering to scale 1, second panel: supersampling followed by block averaging third panel: weighted averaging filter, right-hand panel: Gupta-Sproull algorithm (cone radius of 1.25).

Of the four anti-aliasing methods, the more *realistic* discs are produced by the supersampling block-averaging algorithm and the more complex Gupta-Sproull algorithm.

These discs are then corrupted with either Gaussian or impulsive noise and are used to test the diffusion and sieve-based filters, Table 5.5 shows the results. Comparisons

	Filter	Gaussian noise			Impulsive noise		
		σ_x	σ_y	σ_s	σ_x	σ_y	σ_s
Diffusion	diffusion	0.600	0.569	37.606	0.686	0.641	43.355
	$M-$sieve	0.397	0.360	76.532	0.044	0.048	4.110
	$N-$sieve	0.427	0.393	89.765	0.044	0.045	3.746
	$\rho-$sieve	0.418	0.337	85.025	0.046	0.045	3.710
	$\psi-$sieve	0.491	0.451	140.51	0.078	0.072	8.312
Block-average	diffusion	0.570	0.565	39.740	0.635	0.682	42.673
	$M-$sieve	0.332	0.340	64.928	0.040	0.046	21.380
	$N-$sieve	0.348	0.363	61.556	0.046	0.043	20.456
	$\rho-$sieve	0.325	0.338	62.143	0.047	0.045	19.125
	$\psi-$sieve	2.427	4.083	130.56	0.071	0.071	20.701
Weighted-average	diffusion	0.570	0.552	39.415	0.631	0.605	39.044
	$M-$sieve	0.865	3.396	78.257	0.045	0.046	3.885
	$N-$sieve	0.374	0.362	78.79	0.046	0.045	3.642
	$\rho-$sieve	0.369	0.352	68.256	0.044	0.044	3.895
	$\psi-$sieve	3.708	2.719	127.61	0.071	0.076	8.115
Gupta-Sproull	diffusion	0.570	0.576	41.935	0.519	0.616	45.847
	$M-$sieve	0.353	0.319	67.679	0.041	0.049	11.237
	$N-$sieve	0.395	0.364	80.611	0.046	0.040	10.972
	$\rho-$sieve	0.363	0.332	69.161	0.043	0.045	11.043
	$\psi-$sieve	0.410	0.384	137.94	0.075	0.068	11.970

Table 5.5: Standard deviations of system estimates for anti-aliased discs corrupted with additive Gaussian ($\mu = 0, \sigma = 24$) and replacement impulsive noise (density 0.1)

with the original results from Table 5.1 show that in Gaussian noise there is no discernible difference in the performance of the diffusion processor, while the performance of the sieve filter suffers for scale estimation. In impulsive noise there is both an improvement in the diffusion systems estimates and a degradation in the sieve's estimates, but the overall relative system performances are unchanged.

5.1.5 Are noise tests relevant?

There are many types of imaging systems, and these in turn produce many types of images. Some typical examples are shown in Table 5.6. The quality of these images varies greatly and it is important to be able to predict the behaviour of any low-level signal processing system under a given set of signal and noise conditions. Once this

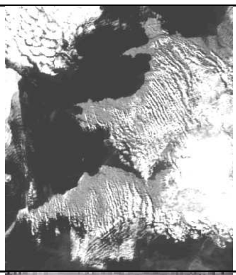
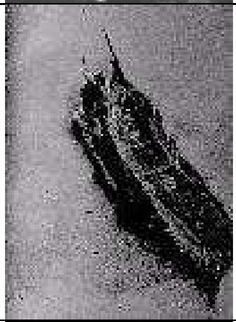
Satellite image	
Sonar image	
MRI image	
Ultrasound image	
Hubble Space Telescope Image	

Table 5.6: Example satellite [3], sonar [49], MRI [1], ultrasound [89] and Hubble space telescope [2] images.

behaviour is understood, the reliability of the system may be determined.

So far, this chapter has described the testing of two different systems under a wide range of noise levels using synthetic signals, but the similarity between these testing signals and real images has not been established.

It is well known that specialised images such as medical ultrasound scans and sonar images, for example, have high noise content, however, the majority of camera images of natural scenes are not subject to this type of distortion. For the scanned camera image that is used to demonstrate the differences between the scale-spaces of Chapter 3, the noise levels appear to be low; this is confirmed in noise-level tests. In [54], Immerkær reports on a method for estimating the level of noise in an image, calculating the variance of additive zero-mean Gaussian noise for an image with intensities in the range $[0, 255]$. See Appendix B for a description of this algorithm.

Using this algorithm to test the Gaussian noise synthetic images of Section 5.1.1 gives estimates of the noise variance that are consistently within 0.5% of the set noise levels. When the example images of the previous chapters (see Figure 3.8 for example) are tested, along with the set of real images that are to be used later in this chapter, the average estimated noise variance is around 0.5, much lower than the level set for the Gaussian noise robustness tests.

It appears that, while the camera-images do contain a certain level of Gaussian noise, the more significant problems occur through *replacement noise*, of which impulsive noise is one example. Replacement noise occurs when the values of a continuous region of pixels are replaced by values drawn from a distribution that is different to the original distribution or background. The most common form of replacement noise in images is occlusion or clutter. Objects that overlap in an image significantly alter the image intensity distribution, confusing many imaging systems, especially those that rely on locating image features by finding a close match to a fixed intensity distribution template.

Alternative forms of replacement noise may occur on a smaller scale; glint and pixel drop-out are some examples. Glint occurs when a scene's illuminant reflects on surfaces with complex texture. These surfaces are often non-matt and the reflections can cause sharp intensity differences in the captured image. The discontinuities in image surface intensities that arise from this phenomenon and from pixel drop-out are well modelled by the impulsive noise signal corruption that was used to test the scale-spaces. The concluding pages of this chapter consider the problems of larger scale occlusions, using both synthetic and real images that contain multiple overlapping targets.

5.2 Robustness to image occlusions

Occlusion can be modelled as a form of replacement noise. There are obviously many complicated cases of occlusion in images that even the human vision system cannot resolve and so this section describes a much simpler situation that is an extension of the previous work. Target images containing multiple occluded image features are generated. Since the linear diffusion system is not invariant to the shape of image features, only discs are used and, as before, scale-space estimates of the discs are required. The following sections describe work that was reviewed in the following publications [26, 45].

5.2.1 Synthetic occlusion image model

Image occlusion is modelled via a set of synthetic test images that are generated by placing a random selection of four different-area discs (areas 49, 253, 481 and 709 pixels) onto a 100×100 pixel image containing a central target disc. The central disc has an area of 901 pixels and intensity 190 with a background intensity of 127. The four cluttering discs have random position (although they are forced to be entirely within the image confines), random intensity (uniform range $[0, 255]$) and they randomly occlude one another. Figure 5.10 shows four typical example images.

The task of the systems is to return scale-space estimates of all target discs. If the five discs did not occlude one another, the scale-space systems would produce five scale-selection surface peaks at the true positions of the objects. With the occlusion of these image features these estimates are degraded. As in the robustness experiments of the previous section, the degradation may be quantified by repeatedly generating test images and returning all *significant* scale-space estimates. Prior knowledge is used to determine this significance; the five largest amplitude peaks in the scale-selection surface are returned, hopefully corresponding to the five discs within each image. By this means we avoid the order estimation problem.

The search algorithm that is used to locate a single scale-selection surface peak

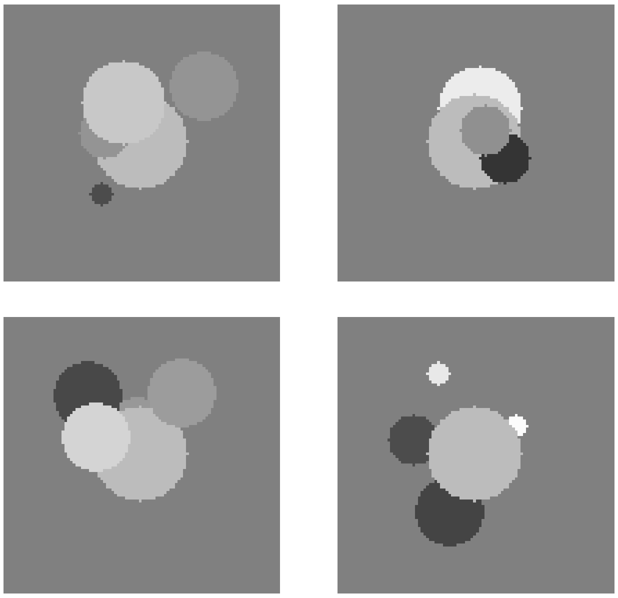


Figure 5.10: Examples of the occlusion model for synthetic images. The large central target disc partially occludes and is partially occluded by other objects.

previously (described in Appendix A) cannot be used for this task since the images contain multiple targets that are of multiple scales. Instead, an exhaustive search over the entire scale-space is required. In the case of linear diffusion, each test image is filtered via equations (3.8) and (3.9) up to a scale of twice the scale of the largest disc. Again, some prior knowledge is required for this step, but this does not affect the results it merely reduces computation time. If no prior scale-knowledge is available the image should be filtered until it is reduced to a single extrema point. The scale-selection point-measure of equation (3.61) is then applied to every extrema point (and their four-connected neighbours) of the low-pass scale-space surface. This curvature measure saves on computation time and also avoids the *crown of thorns* effect described in Section 3.7. The scale-selection equation is only applied to the extrema points of the scale-selection surface and so the sharp edges of the discs are avoided.

The image-mean is normalised to zero for these diffusion experiments. This reduces the step-effect at the image boundary, thereby minimising the boundary effects on the diffusion of the image features.

For the sieve-based system, the test images are again filtered up to a scale of twice the area of the largest image feature. If this area is unknown, the image is filtered up to a scale at which the low-pass sieve output is flat. As scale increases the centroid position, amplitude and scale of all non-zero regions (granules) of the granule functions are recorded.

Figure 5.11 shows examples of the diffusion and sieve-based scale-selection surfaces for the synthetic test image shown in the bottom right-hand panel of Figure 5.10. The accompanying images to these three-dimensional plots show the plan view.

The positions of the extrema points of each scale-space are plotted as a function of scale. In the linear diffusion plot (top panel) the points are seen as tracks of *blue* dots. Obtaining an estimate of the scale of each disc entails tracing along each individual track and finding the scale-position of the maximum over the scale-selection surface. The figure also shows how the extrema points of the image move around in space with increased filtering. This is due to the wide-spread influence of the convolution kernel and as a result, finding the spatial position (described in [8, 71]) involves tracing to the base of each track and recording the position at scale zero. The final estimates for this image are the five most significant scale-space estimates (shown as large *red* dots), chosen as those points (from *separate* paths) with the largest values of scale-space curvature (3.61). The additional *error tracks* (named as error tracks since they do not appear to be associated with any of the synthetic discs) describe extrema points that are caused by, either regions in between disc areas, or by the small differences between the image background and the image's zero-padded boundary. In practice the scale-selection surface values associated with these tracks are small and the returned estimates are of small scales. These points may therefore be removed through thresholding techniques.

For the $M-$sieve example, (bottom panel of Figure 5.11), the synthetic discs are removed cleanly and no tracks are formed in the scale selection surface. The centroids of the five largest-amplitude granules of the scale-selection surface are recorded and plotted as a function of scale (large *red* dots).

Figures 5.12 and 5.13 show the results of processing 100 synthetic occlusion images in this manner. Five scale-space estimates are returned per image. The (x, y) position estimates (horizontal axis) are shown as a function of scale (vertical axis). In the top panel of Figure 5.12 the linear diffusion system's scale and position estimates are plotted as *black dots* and their projections on the position and scale planes are plotted as *green dots*. Control experiments on the five different area discs show that the diffusion system labels the discs as being of scales 7, 40, 76, 112 and 145 (marked as dashed lines

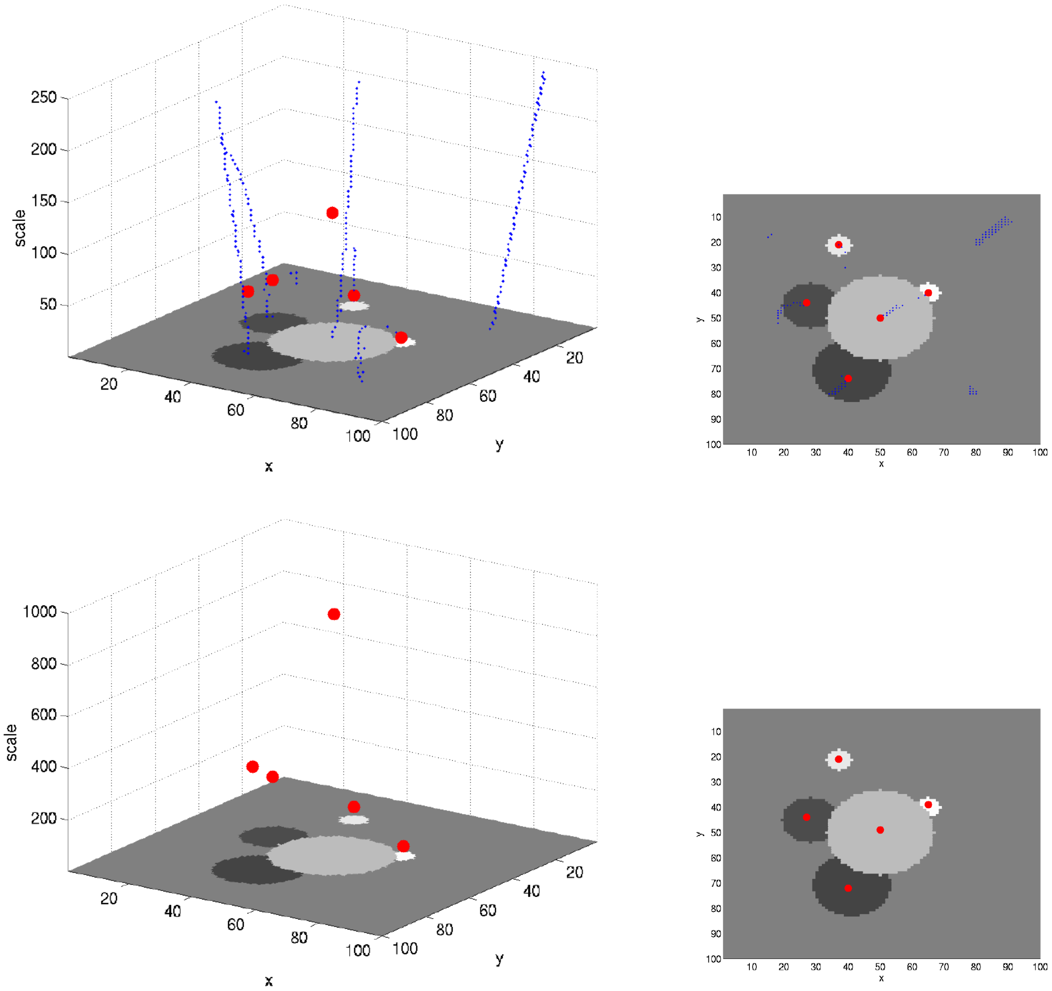


Figure 5.11: the paths of the scale-space estimates for all scales of a synthetic image. Top panel: diffusion results, bottom panel: sieve results

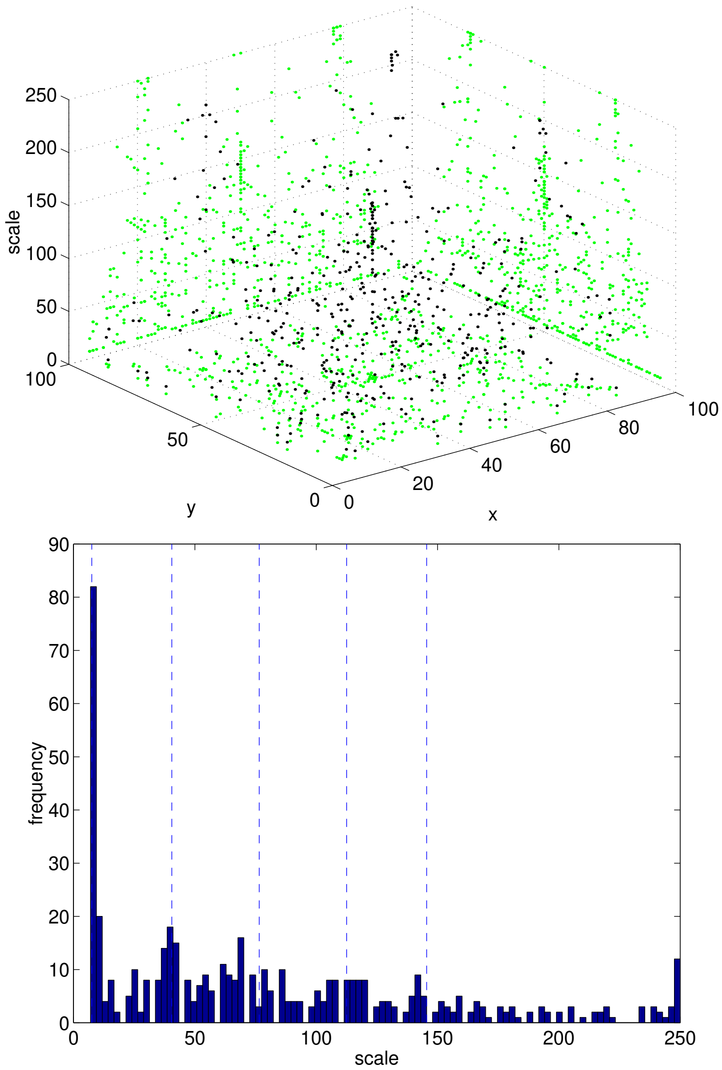


Figure 5.12: Diffusion system's scale-space estimates for synthetic occluded images. Below is the corresponding scale-estimate histogram.

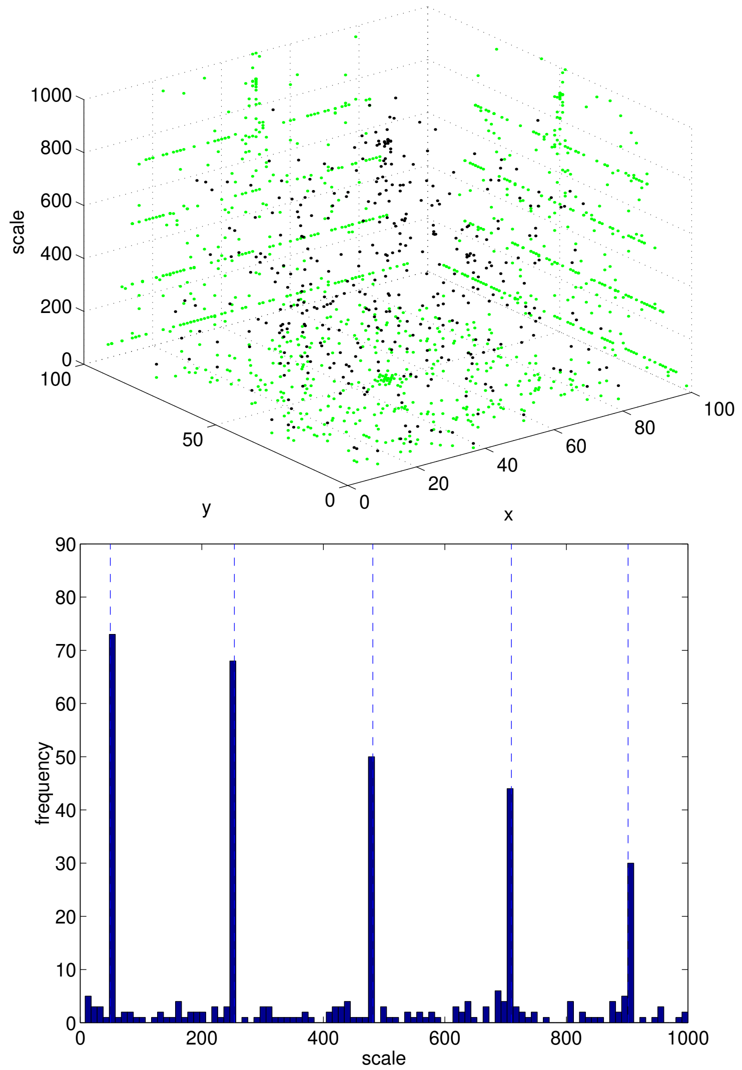


Figure 5.13: $M-$sieve system's scale-space estimates for synthetic occluded images. Below is the corresponding scale-estimate histogram.

on the histogram plot). However, looking at the green projections onto the $y = 100$ and $x = 100$ planes, there are no discernible peaks around these scales and overall there is very little structure to the scale estimates.

The large number of small-scale estimates from the linear diffusion filter are caused by the system requirement to return five estimates. In those cases when all five discs are not visible, the additional estimates are obtained from scale-selection peaks in the *error tracks* (such as the one shown in the top left-hand panel of Figure 5.11). The plot in the bottom panel is a histogram of the scale-estimates. This clearly indicates the lack of structure in the diffusion results, the sharp edges of the target discs are not a good match for the diffusion system.

The top panel of Figure 5.13 shows the sieve's scale and position estimates. Here, the green dots show a clustering. There are bands corresponding to the scales of the clutter objects (49, 253, 481, and 709 pixels – marked as dashed lines on the histogram plot) and there is a cluster at the centre corresponding to the 901 pixel stationary target. This structure in the scale dimension is obvious in the histogram of the sieve's scale estimates (bottom panel).

These plots of the scale-space estimates go a small way in comparing the performances of the two systems, however, to quantify the results it is helpful to know the ground-truth for the test images. For each image, the centroid position and scale of the five discs are recorded. The errors in the position estimates are then taken as the sum (E_n) of the minimum distances between the estimated, ($\hat{x}$) and actual, (x) positions for all possible permutations (p_n).

$$E_n = \sum_{i=1}^{N} ||\hat{x}_{p_n(i)} - x_i|| \tag{5.4}$$

For five discs $N = 5$, and the number of permutations is $n = 5! = 120$. The final error is chosen from the permutation p_n that minimises E_n. Figure 5.14 shows the results of this error estimation. For each system the standard deviation and mean of the errors are returned for an increasing number of synthetic images. The two error plots that are given for the diffusion system show the differences between taking the position estimate at the location of the scale-selection surface peak (no-tracking – top panel) and taking the position estimate from the scale-zero image (tracking – middle panel). All plots show little variation in mean or standard deviation for iterations of over 40 images,

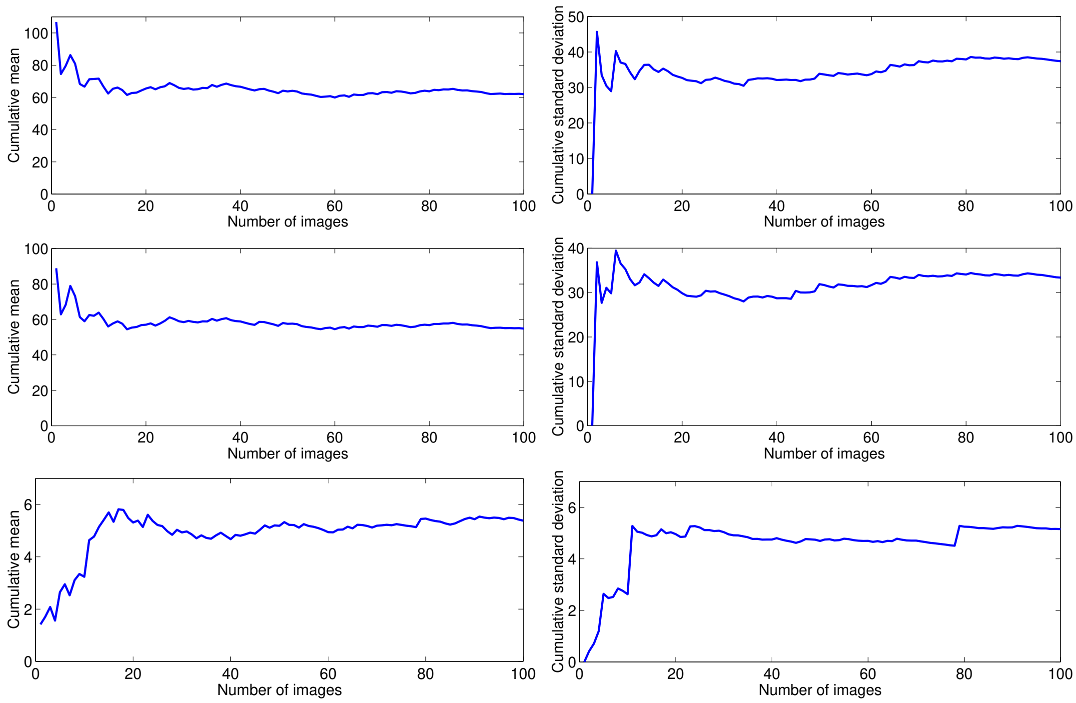


Figure 5.14: Mean (left-hand column) and standard deviation (right-hand column) of the errors in the position estimates over the 100 synthetic images. The top row shows the diffusion system results when the position estimates are taken as the position of the scale-selection surface peak (i.e. no tracking). The middle row shows the results of the diffusion system when the position estimates are taken by tracking down to the scale-zero image. The third row shows the results of the $M-$sieve system.

showing that the 100 images that are processed give statistically reliable results. As expected from the histogram of Figure 5.13 and the general performance of the sieve with sharp-edged objects, the sieve gives the better performance under these conditions (bottom panel), the final mean and standard deviation of the errors (over 100 images) being $(\mu, \sigma) = (62.02, 37.41)$ for the diffusion system (no tracking), $(54.87, 33.37)$ for the diffusion system (with tracking) and $(5.38, 5.15)$ for the $M-$sieve system.

Several observations follow from this experiment. The sieve clearly performs well in scale and position estimation and is suited to the sharp edges of the discs. The diffusion filter, on the other hand, does not perform so well because the sharp-edged, piecewise constant regions of the occluded images do not provide a good match with the diffusion filter and lots of small scale errors occur due to error tracks. The differences between position results from the tracking to scale zero and the non-tracking experiments are small, however the more accurate estimates are from the system with the position estimates taken at scale zero.

5.2.2 Anti-aliased synthetic occlusion image model

The synthetic occlusion images of Figure 5.10 are developed from a very simple image model that does not take into account the noise and texture features that are present real images. The discs in the images have both uniform intensities and sharp edges, making them particularly easy to segment through the sieve-based decomposition. In Figure 5.15 a more complex occlusion model is shown. This is developed in an attempt to mimic the distortions of real images.

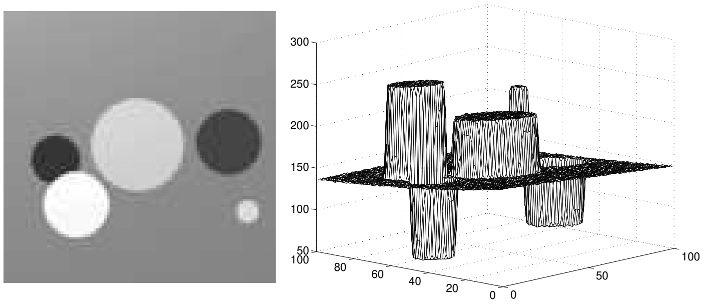


Figure 5.15: Example of the synthetic, anti-aliased occlusion images with added noise and lighting skew. On the right is the corresponding surface intensity plot.

As in the previous synthetic model, the images contain one large central disc and five smaller occluding discs of four possible areas. These disc-images are anti-aliased (using the weighted-average method described in Section 5.1.4) to smooth the sharp edges. The image, $F(x, y)$, is then subjected to a simple lighting-skew, $E(x, y)$ such that the skewed image is the product of the noise-corrupted image and the illumination function $I(x, y) = E(x, y) \times F(x, y)$, and the resulting image is then corrupted with additive Gaussian noise with a mean of $\mu = 0$ and a standard deviation of $\sigma = 0.5$. The small levels of noise and the lighting skew are more apparent in the surface intensity plot on the right-hand side of Figure 5.15.

These images are processed in a similar manner to the basic synthetic images of the previous section. Figure 5.16 shows the results of searching for five scale-space

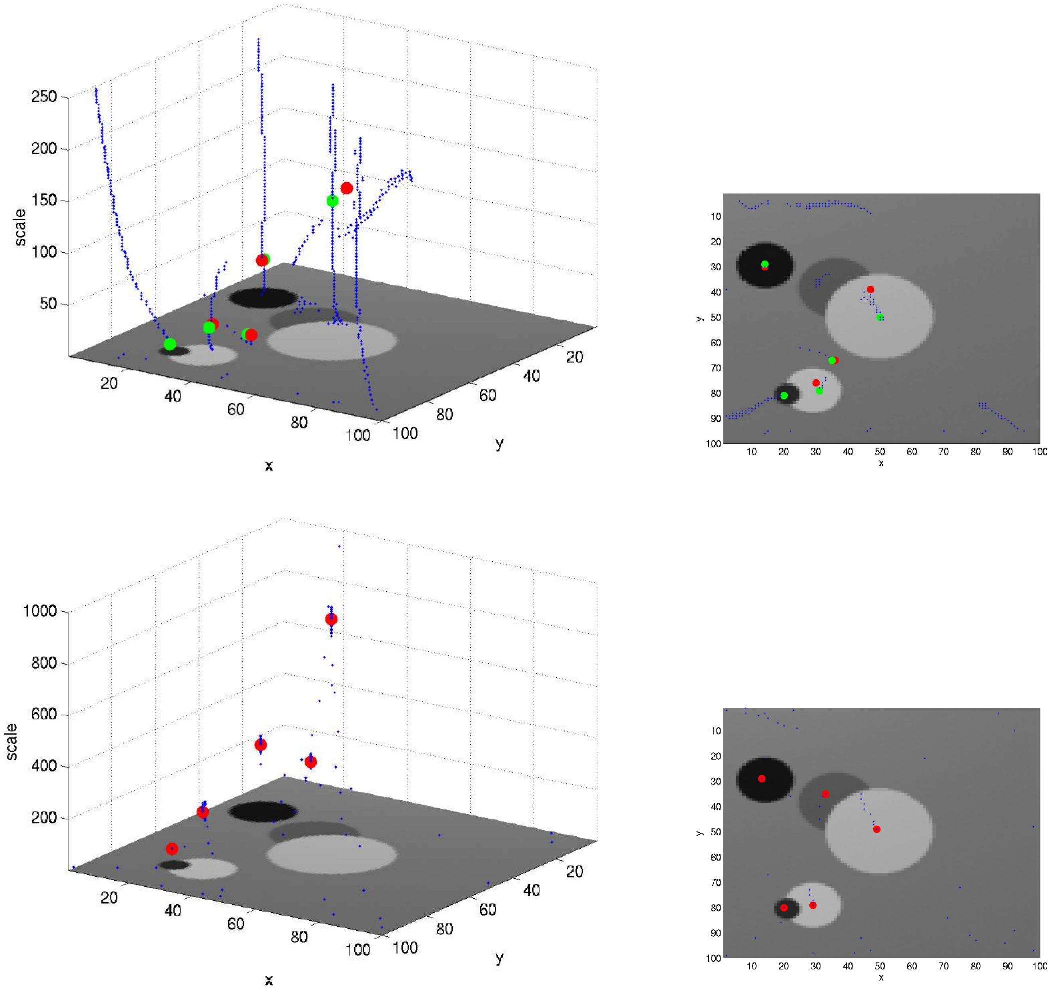


Figure 5.16: The paths of the scale-space estimates for all scales of an anti-aliased synthetic image. Top panel: diffusion results, bottom panel: sieve results

estimates over a particular image. In the top left-hand panel, the blue points show the movement of the scale-space extrema points as the image is diffused to increasing scales. The scale-selection measure of equation (3.61) is then applied to all extrema points and the large red dots show the scale-space estimates, taken as the five largest selection surface peaks. The spatial estimate of these results are calculated using the process described for the synthetic images; the estimate is taken as the (x, y) position at the base of each path. However, this is not necessarily the best approach in these conditions. The figure also shows five alternative scale-space estimates shown as large *green* dots. These points have the same scale estimates as the original results, however the spatial estimates are simply the (x, y) coordinates of the selection surface peaks, *without* tracking down to scale-zero. It is observed that tracking to scale zero under noisy conditions may return a position estimate that is affected by the texture-detail of the image. This is reported in more detail in the following section, however, the plan view of the diffusion plot does show that the position estimate of the central target disc has improved.

The lower panel of Figure 5.16 shows the $M-$sieve results for the same image. The blue points show the positions of all granules over the scale-selection surface. Since the edges of the discs are no longer sharp, the sieve cannot remove them at a single scale. Instead, increasing area layers of the discs are removed (see Figure 3.49) and the centroid positions of these layers are plotted as a function of scale (*blue* dots). The scale-space estimates are calculated from these tracks as the centroids of the maximum amplitude granules within each path, and of these, the five largest amplitude granules are recorded (large *red* dots). The problems caused by the texture variations are negligible, since the texture is limited to small scales and has a small amplitude in comparison to the granule amplitudes at the scale-selection surface peaks.

The final scale-space estimates over 100 anti-aliased synthetic images are shown in Figures 5.17 and 5.18. The diffusion results show some improvement over the previous synthetic clutter tests. There are some peaks in the scale histogram that correspond to the true scales of the smaller discs. These true scales (for the diffusion system only) are 10, 40, 73, 112, and 145 and are marked as dashed lines. In the histogram of Figure 5.18 the scale estimates of the sieve-based system are more clear. The true scale estimates are 52, 256, 452, 708 and 910 and the histogram shows peaks around all of these scales.

The plots in Figure 5.19 show the errors (5.4) in the position estimates of the two

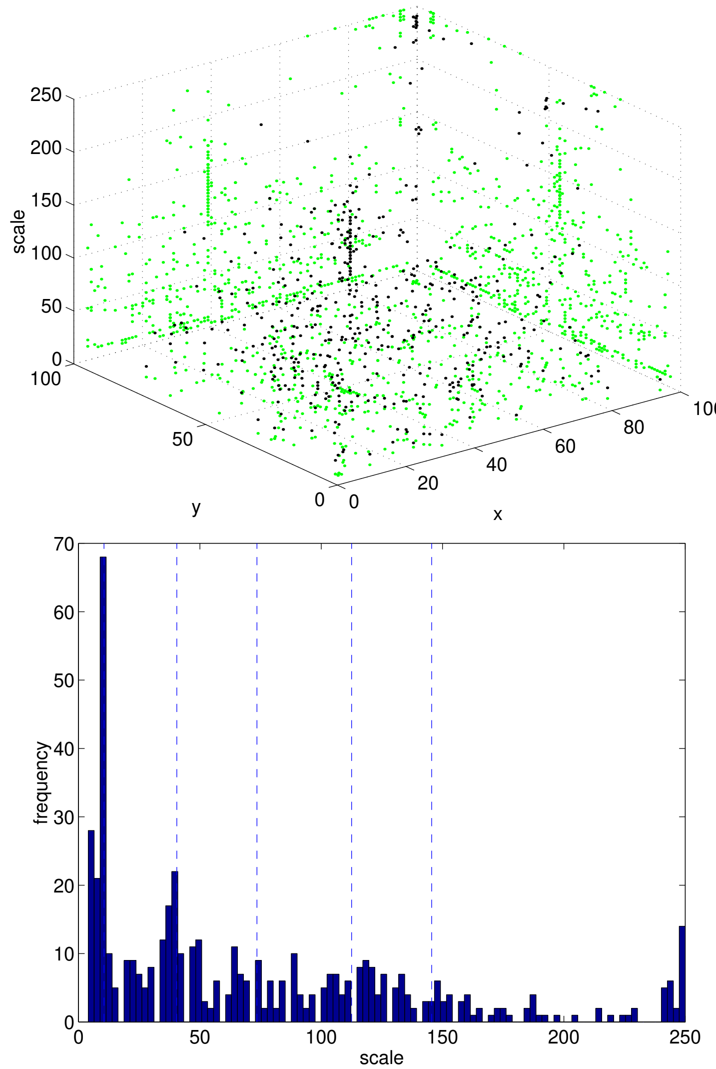


Figure 5.17: Scale-space estimates for anti-aliased, noise corrupted and lighting skewed synthetic occluded images. Diffusion scale-space estimates, below are the corresponding scale-estimate histograms.

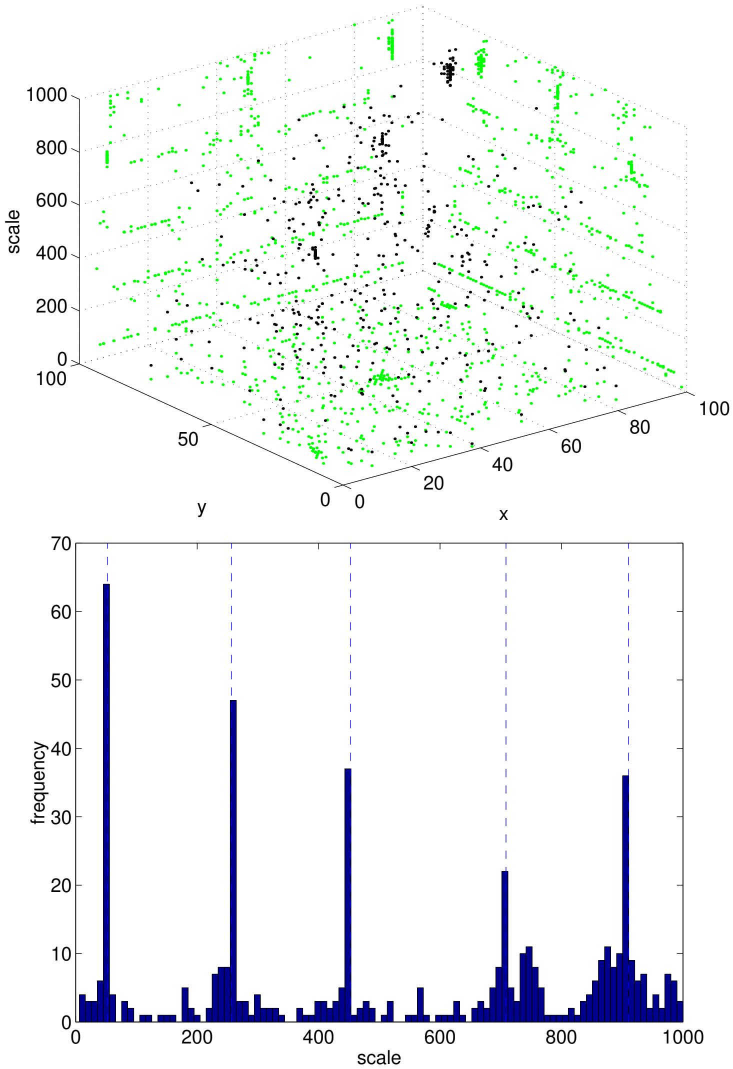


Figure 5.18: Scale-space estimates for anti-aliased, noise corrupted and lighting skewed synthetic occluded images. $M-$Sieve scale-space estimates, below are the corresponding scale-estimate histograms.

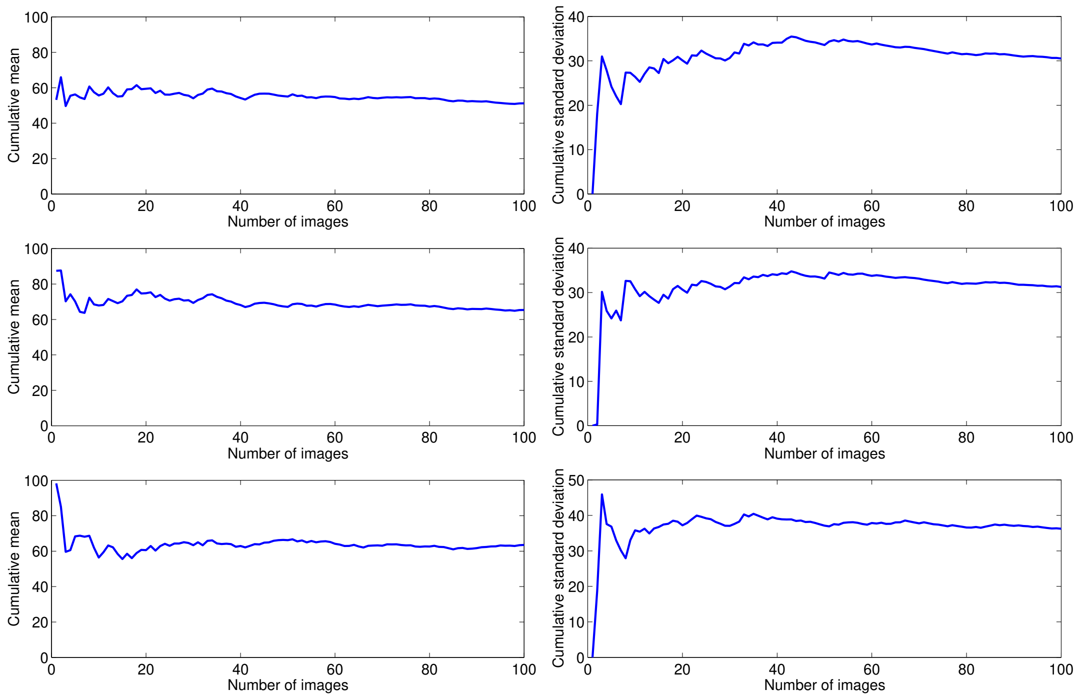


Figure 5.19: Mean (left-hand column) and standard deviation (right-hand column) of the errors in the position estimates over the 100 anti-aliased synthetic images. The top row shows the diffusion system results when the position estimates are taken as the position of the scale-selection surface peak (i.e. no tracking). The second row shows the results of the diffusion system when the position estimates are taken by tracking down to the scale-zero image. The third row shows the results of the $M-$sieve system.

filters. As before, the mean and standard deviation are recorded for increasing numbers of test images. In the bottom panel, the sieve's localisation-performance suffers with the added noise and anti-aliasing, however it is still as accurate as the estimates from the diffusion system. Although the two sets of diffusion estimates are similar, it is the non-tracking estimates that return the lowest mean and standard deviation, suggesting that the small-scale texture does have a detrimental effect on the diffusion systems localisation performance. The overall mean and standard deviation of the errors (for 100 test images) are $(\mu, \sigma) = (51.21, 30.51)$ for the non-tracking diffusion system, $(65.36, 31.26)$ for the tracking diffusion system and $(63.53, 36.23)$ for the sieve-based system.

5.2.3 Real occlusion image model

The results of the anti-aliasing experiments of Sections 5.1.4 and 5.2.2 show that there is some degradation in the performance of the sieve-based processor for more realistic disc-shapes. However the anti-aliased images are still only synthetic and the effects of real image distortions have yet to be studied. Given the improvement of the diffusion processor for the filtered synthetic discs, a further improvement may be possible for real images.

Figure 5.20 shows an example of the *real* image model that is to be used to compare the scale-space systems. The model consists of discs of card placed onto a black-card background and the location of the discs (i.e. the image sector, the distance from the image-centre and the layering position (foreground, background, etc.)) were determined using a random number generator. The largest disc is the stationary, central-target disc. This is randomly occluded by (and randomly occludes) four smaller target discs (selected from a set of four possible areas and six possible intensities). The images are greyscale, of size 192×255 pixels, and in the range $[0, 255]$.

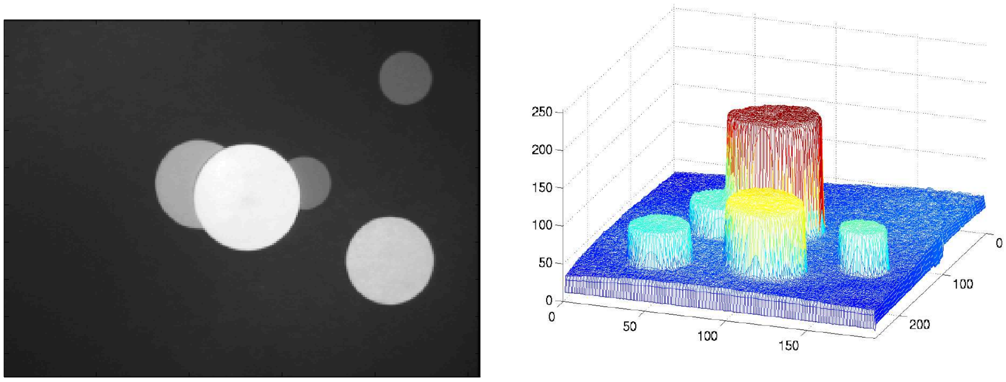


Figure 5.20: Occlusion model for real images. The central target partially occludes and is partially occluded by other objects. On the right is the corresponding surface intensity plot

Again, the real images are processed in a similar manner to the synthetic images. However, due to the increase in the size of the image, a minimum diffusion-kernel length of 260 elements (greater than the longest side-length of the image) is used. This is the simplest way of avoiding the finite precision errors caused by kernel-truncation and therefore eliminates the need to threshold the extrema points of the scale-space surface.

Figure 5.21 shows the diffusion and sieve-based scale-space estimates for the example image of Figure 5.20. Both of the systems produce scale-space tracks that must be followed to reveal the final estimates. The sieve's estimates are processed as before, returning the positions and scales of the largest amplitude granules. In the diffusion system however, the differences between choosing to track to scale-zero for the position estimates (*red* dots) or returning the position of the scale-space peak (*green* dots) are more apparent; the non-tracking estimates having the more central disc positions.

These images contain texture detail and intensity variations that are due to complex non-uniform lighting. If the discs are viewed through a surface intensity plot (see the middle panel of Figure 5.22), then these intensity variations are seen more clearly. In filtering to small scales, it is this texture variation that is dominant. Once this detail is removed through diffusion, the scale of the disc is found. As scale then increases further the image features begin to merge and the detail of the individual discs are lost. The three-dimensional plot on the right-hand side of Figure 5.22 shows how the positions of the scale-space peaks are affected by the small-scale lighting variations and it is this added variation that is avoided if the spatial estimates of the discs are not based on the peaks at scale-zero.

From the plan views of Figure 5.21 it is apparent that neither the diffusion or sieve-based filtering system is able to accurately resolve the position or scale of the disc that is occluded by the large central target. This is because the occluded disc is not an extrema point, and both of the filtering systems have a bias towards labelling image extrema as significant features. In the case of the sieve, the fifth, and non-accurate scale-space estimate is based on the merged area of the central and occluded disc, whilst the fifth estimate of the diffusion system determines the minimum region between the central disc and the separate surrounding discs to be significant.

Figures 5.23 and 5.24 show the scale-space estimates of the filters for 100 real occlusion images. The results of the linear diffusion processor (Figure 5.23) show a marked improvement over the previous synthetic and anti-aliased experiments indicating that the smoother edges of the real discs provide a closer match to the Gaussian than the sharper-edged synthetic discs. The actual scales of the discs (for the diffusion system only) are 46, 103, 174, 298 and 418 (shown as dashed lines on the histogram plot), and

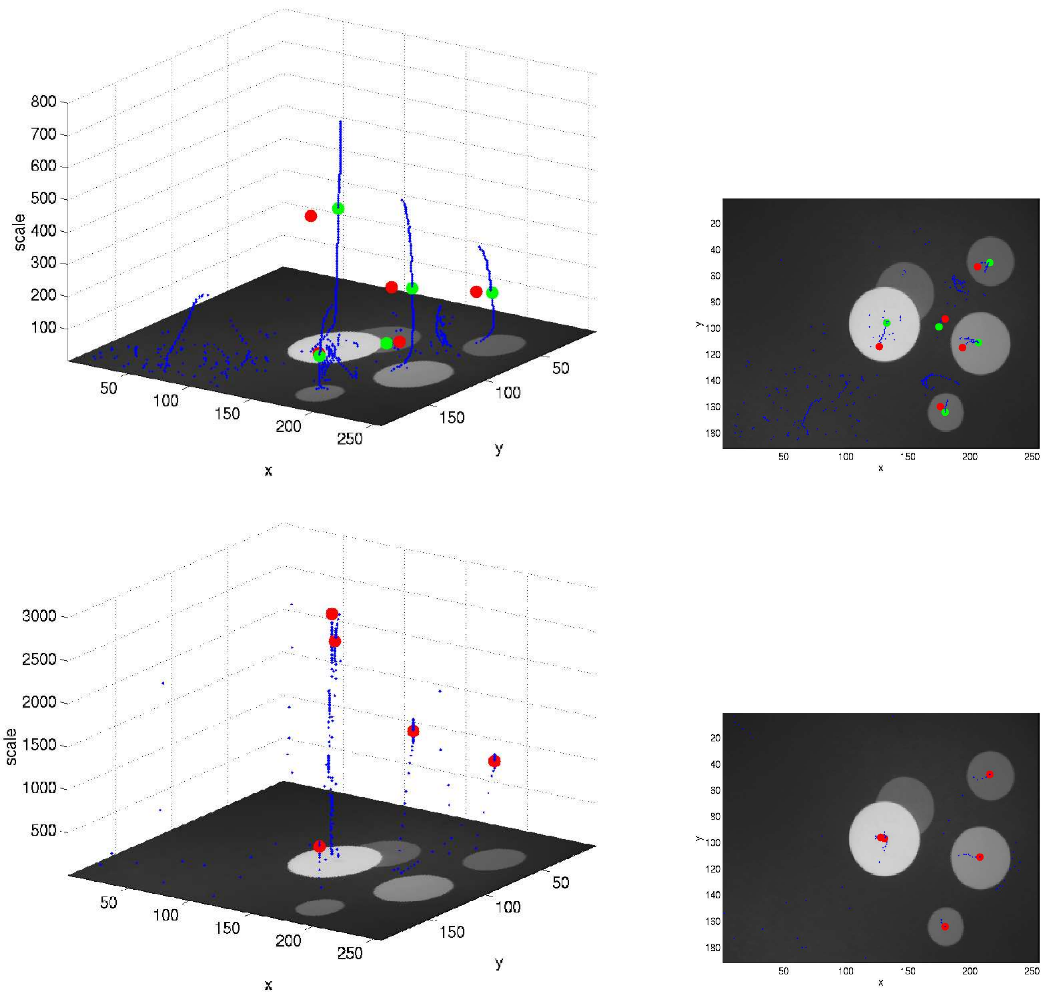


Figure 5.21: The paths of the scale-space estimates for all scales of a real image. Top panel: diffusion results, bottom panel: sieve results

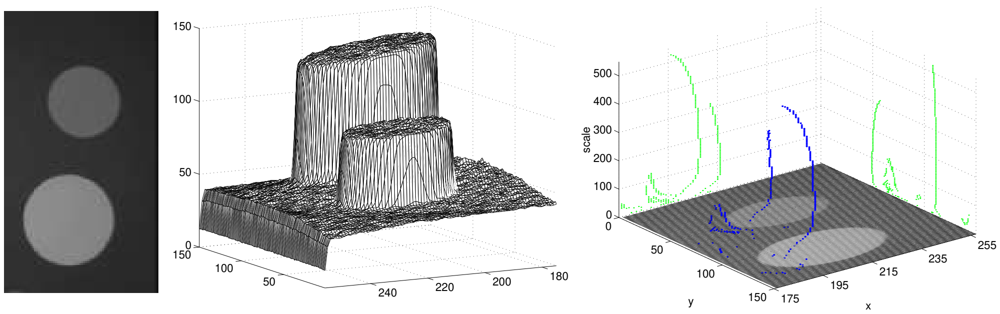


Figure 5.22: Left-hand panel: section of image shown in Figure 5.20, middle-panel: surface intensity plot of image, right-hand panel: the corresponding scale-space tracks shown in blue, and their point-projections onto the two side walls shown in green. Note that these tracks are the tracks that are produced when filtering the *entire* image and not simply this section.

there are bands of points on the walls of the 3D figure that correspond to the scales of the smaller cluttering discs. Also the central stationary target is located with a peak of estimates around scale 400.

The sieve results are shown in the top panel of Figure 5.24 and there is some degradation over the anti-aliased synthetic results since these real disc images are subject to more complex image distortions, especially due to non-uniform lighting conditions that cannot be modelled by a simple lighting-skew. However, even allowing for these degradations the sieve scale estimates are still clearer than those of the diffusion processor and the peaks in the scale-histogram correspond to the sieve-based scale estimates of the real discs which are 268, 658, 1060, 1822 and 2632.

For these real images there is no ground-truth available to show the true positions of the discs and so it is not possible to make an accurate estimate of the errors in the position estimates. However based on the results of the anti-aliased images, it is unlikely that the position estimates of either filter would deteriorate further since the real images are less cluttered than the synthetic images when the size of the discs are compared to the total image area.

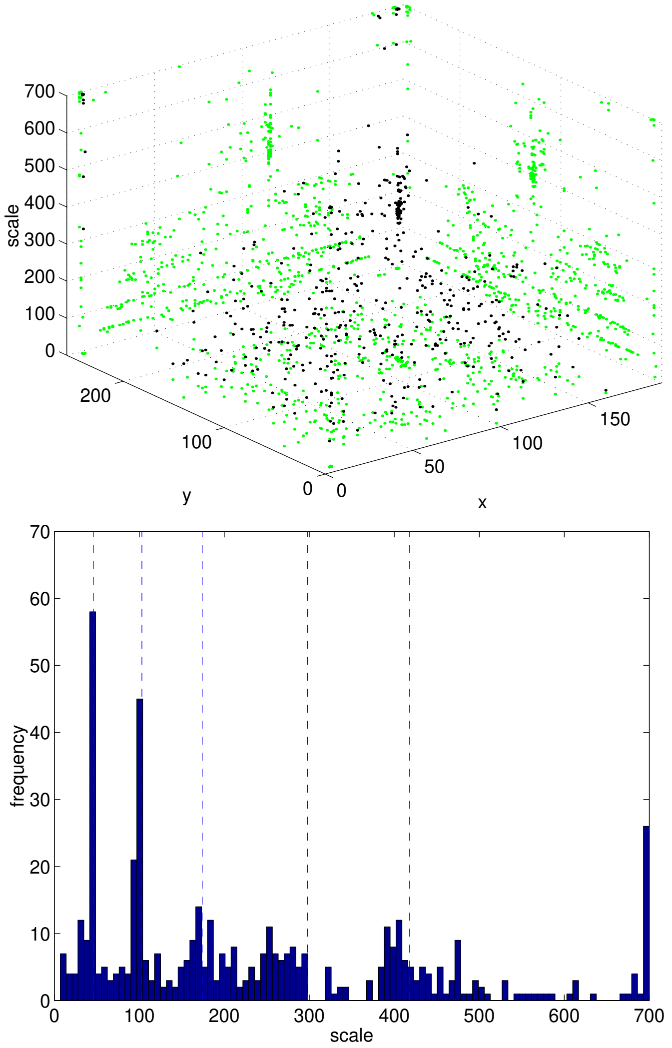


Figure 5.23: Scale-space estimates of the diffusion filter for real occluded images. Below are the corresponding scale-estimate histograms.

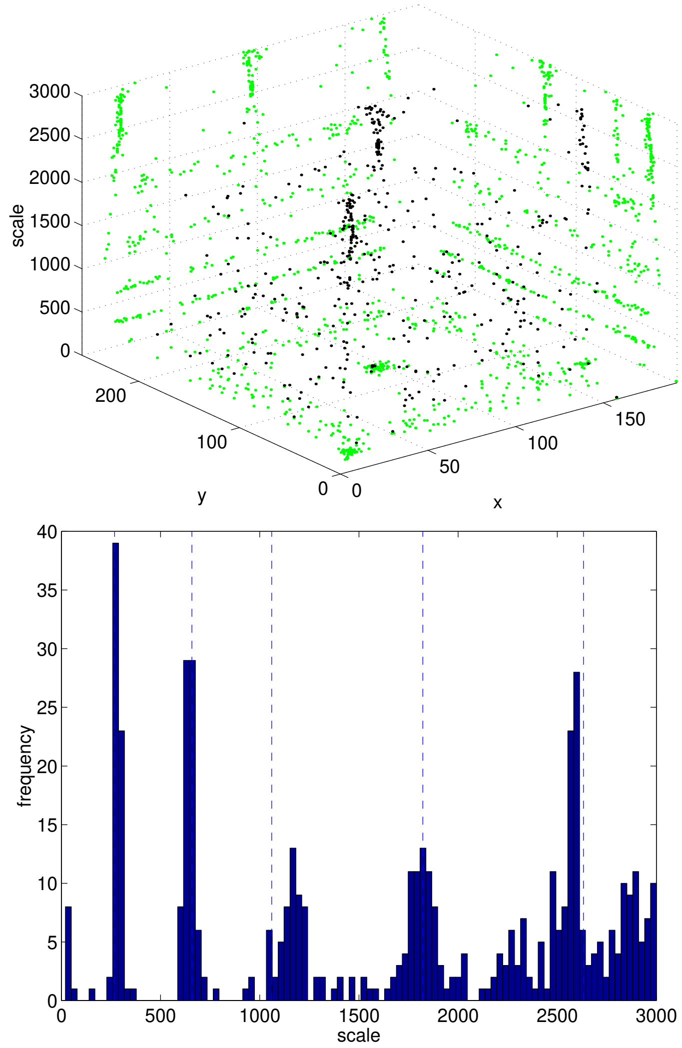


Figure 5.24: Scale-space estimates of the sieve filter for real occluded images. Below are the corresponding scale-estimate histograms.

5.3 Summary

This chapter reviewed the results of a number of robustness tests that were designed to compare the reliability of the linear-diffusion filter and the sieve-based operators. The test signals were all two-dimensional and ranged from purely synthetic noise images containing single image features to more complex real images containing multiple target features. In all cases the task of the processors was to return scale and space estimates of image objects, using as little prior knowledge as was practical.

For the Gaussian noise corrupted images of Section 5.1.1 the systems performed similarly. It was in the presence of replacement impulsive noise that the differences in performance became apparent, the sieve showing good localisation in scale and space without any real degradation in performance for increasing noise density or reduction in target amplitude.

Following on from these single target images, multiple feature occlusion images were created and used to evaluate the robustness of the systems. Image occlusions are a common problem in computer vision and it is important to understand the reliability of any system under such conditions.

For simple synthetic occlusions the sieve performs well, as expected given the nature of the sieve decomposition and the clean edges of the target discs. As the images become more complicated, the accuracy of the sieve's position estimates diminishes slightly but it remains comparable with the position estimates of the linear diffusion system. The accuracy of the scale estimates of the sieve system, again, diminishes slightly as the images become more complex, however, it is the diffusion system scale-results that show the least structure. These results are only partly due to the synthetic nature of the images, they are also due to the unreliability of the scale-selection measure for this task. From (3.60), peaks of the scale-selection curvature are recorded, however, these are not always the result of individual filtered discs – they may be due to disc unions that are formed at the blurring stage. As only one estimate per scale-space-track is allowed, the errors from the union-scale estimates become all the more significant. If the search range of the sieve were wider then it may also suffer from the same disc-union problem since merged discs are treated as single granules. Limiting the search range to have a maximum that is close to the largest-size image feature is a simple solution, unfortunately there is no equivalent solution for the linear diffusion case.

Chapter 6

Discussion

In this thesis a new scale-space filter, called a *sieve*, has been described. The sieve decomposes an image according to a scale parameter that is linked to the area of signal extrema. There are a series of operators that are used in conjunction with the sieve and these determine the bias of the filter output. The opening operator, (ψ_s), will only remove signal maxima and the closing operator, (γ_s), will only remove signal minima. These primary operations may be alternated to give less sensitive filter outputs; the $\mathcal{M}_s$ filter is defined as an opening followed by a closing and an $\mathcal{N}_s$ filter is defined as a closing followed by an opening.

In Chapter 3 of this thesis the theory and implementation of the sieve filter and its different operators were compared with other scale-space systems. These filters were the linear diffusion approach that is described by Lindeberg [71], the anisotropic diffusion approach that is described by Perona and Malik [23,91], and the mathematical morphology system of Jackway [57] and van den Boomgaard and Smeulders [110] that involves erosions and dilations with a paraboloid structuring element.

The priority of all of the filters is to remove signal detail in a *causal* manner, but they achieve this in different ways as shown in Figure 3.1. The linear diffusion system follows the classical approach to scale-space analysis [51, 118] where the image features are suppressed by blurring. The larger the degree of blurring, the more detail is removed. The anisotropic diffusion approach was developed as a means to remove unwanted and confusing detail in an image whilst retaining an accurate picture of the structure of the image. As a solution to this the degree of blurring is controlled primarily by the local contrast of the image. Regions of high contrast (such as edges) are retained and intra-region areas of low contrast are blurred to flat zones. Lastly, the morphological erosion and dilation scale-space of [57] removes detail by fitting the surface of a curved (paraboloid) structuring element to the input image and tracing the path of the structuring element's origin point to give a filtered output.

Articles on scale-space systems often adopt the convenient format of (*i*) expounding some axioms and (*ii*) presenting some example images (although several avoid even this last step). Such a format is not conducive to comparing different systems – example images are not a replacement for quantitative experiments. The aim of this thesis was to set up such experiments and to investigate their behaviour and reliability for different conditions of input signal, especially those corrupted with noise. The framework we use is that of signal estimation theory. Such a framework is useful since it demands estimates of signal parameters. For example, a method of labelling the scale of image features is required. This is done through the use of a *scale-selection surface* - extrema points over this surface point to the locations of significant features within the input image. Not all of the systems considered in Chapter 3 can produce this scale labelling and even fewer are scale calibrated. The anisotropic diffusion filter does not remove or suppress image detail based on size - it filters the image using local gradients and so similar sized objects may not have the same scale lifetime or longevity over scales if their local contrasts are dissimilar. The erosion-dilation system with paraboloid structuring elements also cannot produce scale labels for features at this time (private communication - Jackway). The remaining sieve-based and linear diffusion based filters were tested using the noise corruption experiments described in Chapters 4 and 5.

In Chapter 4 simple synthetic pulse signals were corrupted with either Gaussian noise or impulsive noise. It was the task of the sieve-based systems and the linear diffusion system to make an estimate of the position and scale of the pulse. The results of these experiments for different distributions of noise are shown in Figures 4.7 and 4.8. The figures show that in one-dimension, for Gaussian noise corrupted signals the linear diffusion filter is better at position estimation and the sieve is better at scale estimation, whilst in impulsive noise, the sieve is superior at both position and scale estimation.

The first sections of Chapter 5 extend the noise-corruption experiments to two dimensions (see Figures 5.5 and 5.6). Here the performances of the two systems are similar in the presence of Gaussian noise (more accurate scale estimation from the diffusion filter and more accurate position estimation from the sieve), but the sieve still remains superior under the corruption of impulsive noise. The additional experiments using anti-aliased discs did not significantly alter the relative behaviour of the systems but, as expected, the diffusion filter did improve slightly and the performance of the sieve-based filters worsened.

The results of the recursive median sieve operation were shown throughout the noise corruption experiments as a comparison to the other sieve operators. This filter is known to be robust to image noise, but, it does not obey the scale-space axioms in our current implementation; the recursive median sieve filter will return different outputs if the input signal is rotated and these outputs may not always be causal.

Chapter 5 also studies the performance of the two filters for occluded images. Synthetic images of different sized occluding and occluded discs are generated and the task of the filters is to return scale and position estimates of *all* significant features within these images. These experiments are then repeated for anti-aliased and noise corrupted synthetic images and finally, for real images created from randomly placed discs of card on a black background. The results of these experiments show that the position estimates of the sieve-based system deteriorate for more realistic images, however they always have a localisation performance that is either better than or equal to that of the linear diffusion system. The more significant result for these occlusion experiments shows that the scale estimates from the sieve decomposition are *not* significantly corrupted by the image occlusions. The scale estimates returned for both the synthetic and real occlusion images correspond to the scales of the individual discs and since the sieve is scale calibrated, this corresponds to the pixel areas of the discs.

The experiments described in this thesis covered a wide range of noise corruptions, some of which, such as the high levels of Gaussian noise, would not necessarily be found in digitised camera images of natural scenes. However, the tests do expose the weaknesses of these filters under certain signal conditions. The tests are simple, stylised scale and position estimation problems which could be solved easily using a processor that has more prior information. For example, if it is assumed that the shape of the target image is known. However, what is required from these low-level systems is a basic signal analysis that makes no assumptions on the appropriate scale at which to view an image and that receives as little detail on the importance of image features as is practical. Indeed, what is required is a primary vision system that obeys Marr's "principle of least commitment" [77] but preserves as much structure as possible. It appears that the sieve preserves structure, and does so more robustly than the diffusion system.

Returning to the seminal publications on the scale-space description [8, 51, 63, 118–120], as mentioned earlier in Chapter 2, the aims and requirements of such an analysis are described in detail. Many of these early approaches to scale-space analysis followed the linear diffusion scheme where the original signal was Gaussian filtered to increasing scales, suppressing the image detail by blurring. The attractiveness of this approach is said to lie in its "well behaved" properties [119]. The Gaussian kernel has good integration and differentiation properties and has limited bandwidth. It is also symmetric and strictly decreasing about the mean, causing signal values to decrease smoothly with distance. The Gaussian convolution is also well behaved, approaching the original signal for small scales and approaching the signal's mean for large scales.

All of these reasons help to explain the prevalence of the Gaussian and diffusion based scale-space decomposition, but the underlying problems of this method of analysis have not gone unnoticed. Elsewhere [23, 31, 91], the localisation problems caused by these blurring systems were criticised and alternative non-linear diffusion solutions were developed along with mathematical morphology based systems. This thesis has shown that localisation problems and the need for tracking are not the only difficulties that are faced through the use of the linear diffusion system. A summary of these problems follows:

The blurring that is inherent to all linear diffusion systems makes it difficult to localise features at a coarse scale. This results in a need to track extrema points through the scale-space to the scale-zero image; not an easy task and one that it would be better to avoid. The tracking itself does not always lead to an accurate localisation performance; non-uniform lighting and texture detail have been shown to cause misleading position estimates (see Figure 5.22). However, not tracking can also lead to errors caused by extrema points moving around in space with increased filtering.

The Gaussian kernel has infinite support at all scales causing image boundaries to need careful treatment and introducing unpredictability in the behaviour of image features. Filtering an image that has a single extrema point, E_1, will produce a track through the scale-space, but the addition of a second extrema point, E_2, to the image will produce two scale-space tracks, both of which may be different to the first track.

In many cases they are not *significantly* different, but this introduces a new problem – that of determining significance. In practice all Gaussian scale-space tracking systems rely on a number of hidden thresholds to determine significance. The sieve will only produce a different scale-space decomposition at the scale where the second extrema point connects to the first point on the graph representation.

The scale-selection measurements of the diffusion filter are not easy to interpret and are affected by shape. Scale-space estimates are only accurate for simple, regular shapes that are invariant under a rotation of $\pi/2$. Tuned kernels are required to match regular non-rotation invariant shapes (such as rectangles) and there are no kernels to match irregular features. In the case of tuned kernels scale measurements are not possible from the equations (3.60) and (3.61) because the two-dimensional kernel is composed of one-dimensional kernels of different scales. Even when regular, rotation-invariant shapes are used, image occlusion can destroy the structure of the scale-space, blurring occluding features together so that the scale-selection measure estimates the scale of the union of image features.

Scale-selection measures at small scales are also difficult to obtain. Images containing sharp-edged features produce significant curvature measurements at their region boundaries at small scales, (see the crown of thorns effect in Figure 3.48). Since image features are traced through the maximum points of the scale-selection surface, small-scale objects in an image may be difficult to locate as their scale-selection estimates become confused with the scale-selection maxima that are associated with sharp edges in the original image.

If scale selection estimates are obtained, relating this to the true scale of the image features in terms of pixel dimensions is difficult. The scale-calibration relationship that is required for noiseless one-dimensional pulse signals was shown in Figure 3.47, this relationship becomes increasingly complex when two-dimensional signals are considered.

As shown in Chapters 4 and 5, the linear diffusion filter is relatively robust to signal corruptions of additive Gaussian noise in both one and two dimensions. However, in the case of impulsive noise and general image occlusions the filter is not as reliable, particularly so in scale estimation. In one dimension impulsive noise is shown to destroy the structure of the scale-space (see Figure 4.4). The random amplitude spikes of noise are slow to diffuse and produce errors in the scale estimates. In two dimensions these results improve since increased spatial averaging allows the noise spikes to diffuse more quickly, but there are still many inaccuracies in the scale-estimation, especially for higher noise densities. The occlusion experiments that followed this noise-corruption testing highlighted further scale-estimation problems that have already been mentioned; errors occur when the image features cannot be separated through the scale-space and instead the unions of features are evaluated. This inability to separate features can also cause problems in tracking and hence, feature localisation.

Finally, the diffusion scale-space decomposition is slower to compute, requiring floating point arithmetic compared to the sieve's integer arithmetic.

When compared to the linear diffusion system, the implementation of the sieve and the results of the sieve analysis usually prove to be less problematic. The sieve's area-decomposition removes extrema directly from an image, replacing these points with the nearest neighbouring intensity and this produces no blurring effect. Tracking is required for objects with non-sharp boundaries, however this task is not difficult given the lack of spatial movement for increasing-scale filtering.

The scale-selection surface of the sieve is easily computed as the difference between consecutive low-pass sieve outputs and extrema granules in this surface represent the scale and position of image features. One of the worst-case scenarios for the sieve decomposition would be the scale calibration of a particularly diffuse image (especially when corrupted with high levels of Gaussian noise!). In such a case the scale-selection surface would show a uniform spread granule decomposition over all scales and the best scale-estimate, instead of the scale-position of the largest amplitude granule, would be the mean-scale granule. In this situation the most suitable filter choice would be to match the diffuse image with the linear diffusion system. However, it is rare to find camera images of this nature and for the majority of natural-light images, region boundaries and edges are generally sharp enough for the sieve decomposition to provide accurate scale and position estimates. Changes in target shape have no effect on the scale-analysis and since scale corresponds directly to pixel area calibration is straightforward.

This thesis has shown the sieve to be a useful tool for scale-space analysis and image decomposition. The robustness of the sieve to different types of signal noise and occlusion means that it has potential applications in low-level pattern recognition. Preliminary investigations into simple one-dimensional pattern recognition tasks are already published [20], however, more complex forms of signal must be considered, extending these investigations into two dimensions. The sieve analysis also lends itself to a hierarchical approach to image analysis based on trees. These tree structures develop directly from the sieve decomposition; the entire image forming the root node of the tree and the decreasing scale granules forming the children and the children's children of the root node etc. until the smallest-scale granules are represented by the leaves of the tree. This new sieve description, called a *scale-tree*, is described in more detail in [21, 94] and it has the advantages of being concise and easily navigated, avoiding the issues of sieving to granule channels and then searching within these channels for significant features.

Appendix A

Search Algorithm

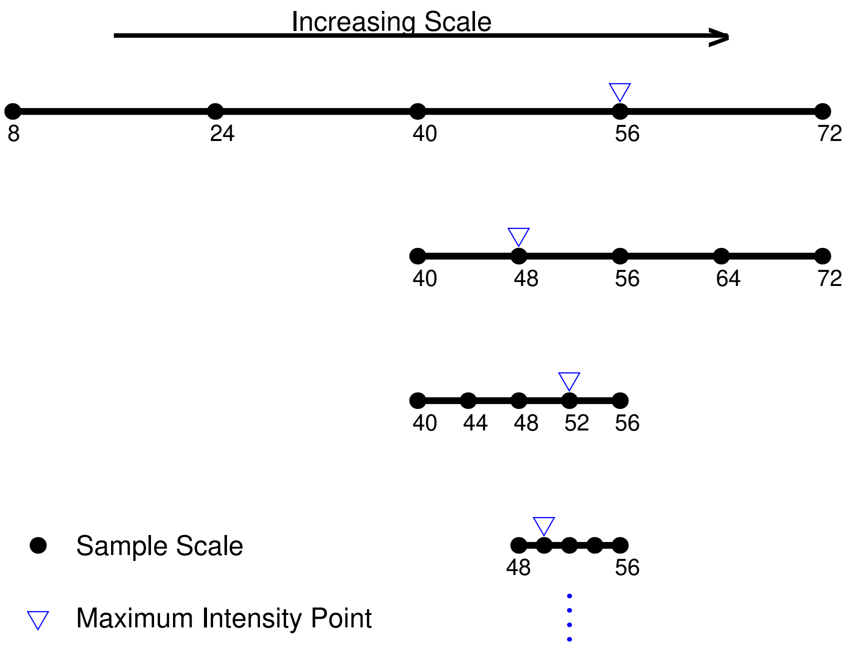


Figure A.1: Search algorithm for finding maxima over scale selection surface

The search algorithm shown in Figure A.1 is used to locate the maximum point over the scale selection surfaces of the linear diffusion filter and the sieve for the single target case. The search space is first divided into equally spaced regions (scale-selection surface representations $s = 8, 24, 40, 56$ and 72 in the figure example) and the scale-value of the maximum intensity point over these images is found. This first estimate is then used as a centre point for the next resolution of scale-sampling ($s = 40, 48, 56$, etc. in the example figure). The maximum intensity point over this new set of images provides the second scale estimates and this successive approximation continues until the maximum intensity point over the entire search range is located.

Appendix B

Level of Gaussian noise in images

In [54] an algorithm is described to calculate the variance of additive zero-mean Gaussian noise in an image. The image has a width, W, and height, H and must be in the range $(0 - 255)$.

The method applies a 3 by 3 element mask, called a noise estimation operator, N, to the image.

$$N = \begin{array}{|c|c|c|} \hline 1 & -2 & 1 \\ \hline -2 & 4 & -2 \\ \hline 1 & -2 & 1 \\ \hline \end{array} \tag{B.1}$$

The operator is intended to be as insensitive as possible to the underlying structure of the image by calculating the noisy *deviations* of the actual image. N is formed from the difference of two masks that each approximate the Laplacian of an image. This allows the estimator to be insensitive to the strong second order differential components caused by edges in the image.

Using this operator, the standard deviation of Gaussian noise in the image, I, may be calculated as

$$\sigma_n = \sqrt{\frac{\pi}{2}} \frac{1}{6(W-2)(H-2)} \sum_{\text{image} I} |I(x, y) * N|. \tag{B.2}$$

Appendix C

The Gupta-Sproull Anti-aliasing Algorithm

The jagged or staircasing effect that is visible at the edge-pixels of many synthetic images is due to the on-off pixel intensities that are used to define objects. In a greyscale image this can be avoided by displaying edge-pixels using grey-level values that are between these two *on* and *off* extremes. To determine these intermediate intensities the image must be filtered.

The Gupta-Sproull anti-aliasing algorithm [41] is based on the proposal that each pixel may be represented by a cone of unit volume, centred on the pixel-centre. If 50% or more of a cone's volume is within an object's region boundary then the pixel has the full object intensity. However, if at a region boundary, less than half of the cone is covered then the final pixel intensity is directly related to the cone volume that lies within the object. If for instance, 25% of the cone's volume lies inside of the object boundary, then the final grey-level value is half of the full object intensity.

Figure C.1 shows an example of this method, used to anti-alias a solid line. Pixels (row,column) $(2,3)$,$(3,4)$,$(3,5)$, etc. are more than half covered by the object and will have the full object intensity. However, the intensities of pixels such as $(3,3)$ (less than half covered) need further attention. If the distance, (p), between the centre of the cone and the normal point of the dissecting line is calculated, then the volume of dissected cone may be found from a lookup table given in [41].

The disadvantage of this method is that it may only be used to anti-alias non-

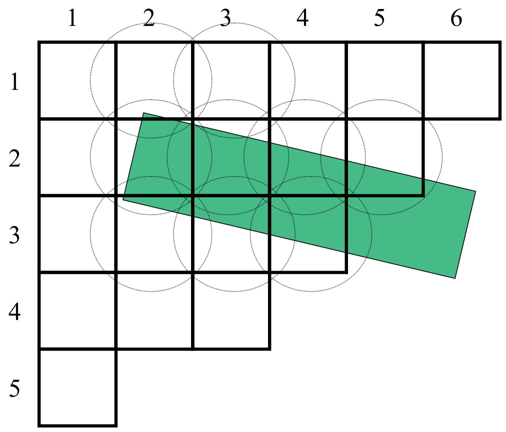


Figure C.1: The Gupta-Sproull anti-aliasing method. Each pixel is represent5ed by a cone of unit volume centred on the pixel-centre. The volume of cone dissected by the true object boundary translates to the final pixel grey-level intensity. The discs indicate the bases of the cones. A cone base-radius of 1.25 of a pixel length gives a good quality smoothed edge.

complicated edges. The algorithm is defined by Gupta and Sproull as a way of filtering edges that may be extended to smooth the edges of polygons. More complex images containing overlapping features of different intensities are not considered and would need more careful treatment.

Bibliography

[1] Processing of MRI images. The Bowman Gray School of Medicine, North Carolina State University, http://www2.ncsu.edu/ncsu/CIL/med_imaging/Stevestuff.html.

[2] Hubble Space Telescope images. http://oposite.stsci.edu/pubinfo/pr/1998/18/greatest-hits-gallery.html, 1987.

[3] Dundee satellite receiving station sample images. http://www.sat.dundee.ac.uk/demo.html, February 1998.

[4] Milton Abramowitz and Irene A. Stegun, editors. *Handbook of Mathematical Functions*. Dover publications inc., New York, 1972.

[5] L. Alvarez, F. Guichard, P.-L. Lions, and J.-M. Morel. Axioms and fundamental equations in image processing. In *Archive for Rational Mechanics and Analysis*, volume 123, pages 199–257, 1996.

[6] Luis Alvarez, Pierre-Louis Lions, and Jean-Michel Morel. Image selective smoothing and edge detection by nonlinear diffusion II. *Society for Industrial and Applied Mathematics*, 29(3):845 – 866, 1992.

[7] H. Asada and M. Brady. The curviture primal sketch. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:2–14, 1986.

[8] J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda. Uniqueness of the Gaussian kernel for scale-space filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:26 – 33, 1986.

[9] J. A. Bangham. Data-sieving hydrophobicity plots. *Anal. Biochem.*, 174:694–700, 1988.

[10] J. A. Bangham. Properties of an improved variant on the median filter. In *Communication, Control and Signal Proc.*, pages 1591–1597. Elsevier press, 1990.

[11] J. A. Bangham. Using the root datasieve to recognise and locate patterns in a data sequence. In *Proc. EUSIPCO'92*, pages 983–987, 1992.

[12] J. A. Bangham, T. G. Campbell, and R. V. Aldridge. Multiscale median and morphological filters used for 2d pattern recognition. *Signal Processing*, 38:387–415, 1994.

[13] J. A. Bangham, P. Chardaire, and P. D. Ling. The multiscale morphology decompostion theorem. In J. Serra and P. Soille, editors, *Mathematical Morphology and its Applications to Image Processing*, pages 179–184. Kluwer Academic Publishers, 1994.

[14] J. A. Bangham, P. Chardaire, C. J. Pye, and P. D. Ling. Multiscale nonlinear decomposition: The sieve decomposition theorem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5):529–539, 1996.

[15] J. A. Bangham, R. Harvey, P. D. Ling, and R. V. Aldridge. Morphological scale-space preserving transforms in many dimensions. *Journal of Electronic Imaging*, 5(3):283–299, July 1996.

[16] J. A. Bangham, R. W. Harvey, P. D. Ling, and R. V. Aldridge. Nonlinear scale-space from *n*-dimensional sieves. *Procceedings of the European Conference on Computer Vision*, 1:189–198, 1996.

[17] J. A. Bangham, P. D. Ling, and R. Harvey. Scale-space from nonlinear filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(5):520–528, 1996.

[18] J. A. Bangham, P. D. Ling, and R. W. Harvey. Scale-space from nonlinear filters. In *5th International Conference on Computer Vision*, pages 163 – 168, 1995.

[19] J. A. Bangham, P. D. Ling, and R. Young. Multiscale decomposition by extrema: the recursive median datasieve and its relation to other sieves. In *IEEE Workshop on nonlinear signal and image processing*, pages 1038–1041, 1995.

[20] J. Andrew Bangham, Richard W. Harvey, and Graham Tattersall. Recognition of image patterns generated by stochastic object replacement. In *IEEE International Conference on Image Processing*, Chicago, 1998.

[21] J. Andrew Bangham, Javier Ruiz Hidalgo, Richard W. Harvey, and Gavin Crawley. The segmentation of images using scale-space trees. In *British Machine Vision Conference*, volume 1, pages 33–42, 1998.

[22] M. J. Black and G. Sapiro. Robust anisotropic diffusion. *IEEE Transactions on Image Processing*, 7(3):421 – 432, 1988.

[23] M. J. Black and G. Sapiro. Robust anisotropic diffusion: Connections between robust statistics, line processing, and anisotropic diffusion. In *Proc. of First Int. Scale-space Conf.: Scale-space theory in Comp. vision*, volume 1, pages 323–326, 1997.

[24] H. Blum. Biological shape and visual science (part 1). *J. Theoret. Biol.*, pages 205–287, 1967.

[25] Robert A. Boie and Ingemar J. Cox. An analysis of camera noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(6):671 – 674, 1992.

[26] Alison Bosson and Richard W. Harvey. Using occlusion models to evaluate scale-space processors. In *IEEE International Conference on Image Processing*, 1998.

[27] Alison Bosson, Richard W. Harvey, and J. Andrew Bangham. Scale-space pattern processors: are they robust to noise and occlusion? In *IEE Colloquium on Pattern Recognition, no.018*, pages 8/1–8/6, Savoy Place, London, 1997.

[28] L. Bretzner and T. Lindeberg. On the handling of spatial and temporal scales in feature tracking. In *Proc. First Int. Conf. on Scale-space theory*, pages 128–139. Springer, 1997.

[29] F. W. Campbell and J. G. Robson. Contrast sensitivity, Fourier analysis and vision. *Journal of physiology*, 197:551– 566, 1968.

[30] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[31] Francine Catté, Pierre-Louis Lions, Jean-Michel Morel, and Tomeu Coll. Image selective smoothing and edge detection by nonlinear diffusion. *Society for Industrial and Applied Mathematics*, 29(1):182 – 193, 1992.

[32] Ming-Hua Chen and Ping-Fan Yan. A multiscaling approach based on morphological filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):694–700, 1989.

[33] E. J. Coyle, J. H. Lin, and M. Gabbouj. Optimal stack filtering and the estimation and structural approaches to image processing. *IEEE Trans. Acoust. Speech, Signal Processing*, 37:2037–2066, 1989.

[34] E. R. Dougherty. Euclidean gray-scale granulometries: Representation and umbra inducement. *Journal of Mathematical Imaging and vision*, 1(1), 1992.

[35] E. R. Dougherty and J. Astola. *An Introduction into Nonlinear Image Processing.* SPIE Optical Engineering Press, 1994.

[36] L. M. J. Florack. Data, models and images. In *Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 469–472, 1996.

[37] L. M. J. Florack, B. M. terr Harr Romeny, J. J. Koenderink, and M. A. Viergever. Scale and the differential structure of images. *Image and Vision Computing*, 13(4):376 – 388, 1992.

[38] L. M. J. Florack, B. M. terr Harr Romeny, J. J. Koenderink, and M. A. Viergever. Linear scale-space. *Journal of Mathematical Imaging and Vision*, 4(4):325 – 351, 1994.

[39] James Foley, Andries van Dam, Steve Feiner, and John Hughes. *Computer Graphics - Principles and Practice*. Addison-Wesley Publishing Company, 1990.

[40] N. C. Gallagher and G. N. Wise. A theoretical analysis of the properties of median filters. *IEEE Trans. Acoust. Speech, Signal Processing*, 21:pp 1136–1141, 1981.

[41] S. Gupta and R. F. Sproull. Filtering edges for gray-scale displays. *Computer Graphics*, 15(3):1–5, 1981.

[42] R. M. Haralick, P. L. Katz, and E. R. Dougherty. Model-based morphology: the opening spectrum. *Graphical models and image processing*, 57:1–12, 1995.

[43] Richard W. Harvey, J. Andrew Bangham, and Alison Bosson. Scale-space filters and their robustness. In *Proceedings of the First International Scale-space Conference: Scale-space Theory in Computer Vision*, pages 341 – 344, 1997.

[44] Richard W. Harvey, Alison Bosson, and J. Andrew Bangham. A comparison of linear and nonlinear scale-space filters in noise. *Signal Processing VIII: Theories and Applications*, III:1777–1780, September 1996.

[45] Richard W. Harvey, Alison Bosson, and J. Andrew Bangham. The robustness of some scale-spaces. In *British Machine Vision Conference*, pages 11 – 20, 1997.

[46] Richard W. Harvey, Alison Bosson, and J. Andrew Bangham. One-dimensional scale-space preserving filters. *Signal Processing IX: Theories and Applications*, pages 821–824, 1998.

[47] G. E. Healey and R. Kondepudy. Radiometric CCD camera calibration and noise estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16:267–275, 1994.

[48] H. J. A. M. Heijmans, P. Nacken, A. Toet, and L. Vincent. Graph morphology. *Journal of Visual Computing and Image Representation*, 3(1):24–38, March 1992.

[49] Sture Hultqvist. Sonar project. http://www.tsl.uu.se/ stureh/1997.htm, 1997.

[50] T. Iijima. Basic theory of pattern observation. *Papers of Technical Group on Automata and Automatic Control, IECE*, 1959.

[51] T. Iijima. Basic theory of pattern normalization (for the case of a typical one-dimensional pattern. *Bulletin of the Electrotechnical Laboratory*, 26:368–388, 1962.

[52] T. Iijima. Observation theory of two-dimensional visual patterns. *Papers of Technical Group on Automata and Automatic Control, IECE*, 1962.

[53] T. Iijima. Basic theory on normalization of two-dimensional visual pattern. *Studies on Information and ControlPattern Recognition Issue, IECE*, 1:15–22, 1963.

[54] J. Immerkær. Fast noise variance estimation. *Computer Vision and Image Understanding*, 64(2):300 – 302, 1996.

[55] P. T. Jackway. Morphological scale-space. In *Proc. 11th IAPR International conference on pattern recognition*, pages C252–255. IEEE, 1992.

[56] P. T. Jackway. Scale-space properties of the multiscale morphological closing-opening filter. In *Proceedings of 2nd Singapore International Conference on Image Processing*, pages 278–281, 1992.

[57] P. T. Jackway. *Morphological Scale-space with application to three-dimensional object recognition*. PhD thesis, School of Electrical & Electronic Eng. Queensland University of Technology, 1994.

[58] Paul T. Jackway. On the scale-space theorem of Chen and Yan. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):351–352, 1998.

[59] Paul T. Jackway and Mohamed Deriche. Scale-space properties of the multiscale morphological dilation-erosion. In *Proceedings of the 11th IAPR Conference on Pattern Recognition*, 1992.

[60] Paul T. Jackway and Mohamed Deriche. Scale-space properties of the multiscale morphological dilation-erosion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:38–51, 1996.

[61] Ben. K. Jang and Roland. T. Chin. Shape analysis using morphological scale-space. In *Proc. 25th Annual conference on Information Sciences systems*, pages 41–44, 1991.

[62] A. Klinger. Pattern and search statistics. In J. S. Rustagi, editor, *Optimizing Methods in Statistics*. Academic Press, 1971.

[63] J. J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363 – 370, 1984.

[64] Jan. J. Koenderink. Scale in perspective. In J. Sporring, M. Nielsen, L. Florack, and P. Johansen, editors, *Gaussian Scale-space theory*. Kluwer Academic Publishers, 1997.

[65] Longin Jan Latecki. *Discrete Representation of Spatial Objects in Computer Vision*. Klewer Academic Publishers, 1998.

[66] Y. G. Leclerc. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, 3(1):73–102, 1989.

[67] Lawrence M. Lifshitz and Stephen M. Pizer. A multiresolution hierarcrchical approach to image segmentation based on intensity extrema. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6):529 – 540, 1990.

[68] Tony Lindeberg. Scale-space for discrete signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3):234 – 254, 1990.

[69] Tony Lindeberg. Scale-space behaviour of local extrema and blobs. *Journal of mathematical Imaging and Vision*, 1(1):65 – 99, 1992.

[70] Tony Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of applied statistics*, 21(2):224 – 270, 1994.

[71] Tony Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, 1994.

[72] Tony Lindeberg. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, 30:117 – 154, 1998.

[73] Tony Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30:77 – 116, 1998.

[74] Tony Lindeberg. A scale selection principle for estimating image deformations. *Image and Vision Computing*, 16(14):961 – 977, 1998.

[75] P. Maragos. Pattern spectrum and multiscale shape representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):701–716, 1989.

[76] P. Maragos and R. W. Schafer. Morphological filters - part 1: Their set theoretic analysis and relations to linear shift-invariant filters. *IEEE Trans. Acoust, Speech, Signal Processing*, 35:1153–1169, 1987.

[77] D. Marr. *Vision*. W. H. Freeman and Company, New York, 1982.

[78] D. Marr and E. Hildreth. Theory of edge detection. In *Proceedings of Royal Society, London*, volume B207, pages 187–217, 1980.

[79] G. Matheron. *Random sets and integral geometry*. Wiley, 1975.

[80] Iain Matthews, J. Andrew Bangham, Richard Harvey, and Stephen Cox. A comparison of active shape model and scale decomposition based features for visual speech recognition. In *European Conference on Computer Vision*, volume II, pages 514–528, 1998.

[81] M. P. McLoughlin and G. R. Arce. Deterministic properties of the recursive separable median filter. *IEEE Acoustics, Speech and Signal Processing*, 35:98–108, 1987.

[82] F. Meyer and S. Beucher. Morphological segmentation. *Journal of Visual Communication and Image Representation*, 1:21–46, 1990.

[83] F. Mokhtarian. Silhouette-based isolated object recognition through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:539–544, 1995.

[84] Kimberly Moravec, Richard Harvey, and J. Andrew Bangham. Improving stereo performance in regions of low texture. In *British Machine Vision Conference*, volume 1, pages 822–831, 1998.

[85] P. F. M. Nacken. Openings can introduce zero crossings in boundary curvature. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16:656–658, 1994.

[86] M. Nielsen, L. Florack, and R. Deriche. Contrast sensitivity, Fourier analysis and vision. *Proceedings of the European Conference on Computer Vision 1996*, II, 1996.

[87] M. Nielsen, L. Florack, and R. Deriche. Regularization, scale-space and edge detection filters. In B. Buxton and R. Cipolla, editors, *Proc. of European Conference on Computer Vision*, volume II, pages 70–81, 1996.

[88] Wiro J. Niessen and Robert Maas. Optic flow and stereo. In J. Sporring, M. Nielsen, L. Florack, and P. Johansen, editors, *Gaussian Scale-space theory*. Kluwer Academic Publishers, 1997.

[89] Lars S. Nyland. Ultrasound images. http://www.cs.unc.edu/ nyland/andrew/ultrasound.html.

[90] N. Otsu. *Mathematical Studies on Feature Extraction in Pattern Recognition*. PhD thesis, Electrotechnical Laboritory, 1-1-4, Umezono, Sakura-mura, Niihari-gun, Ibaraki, Japan, 1981. In Japanese.

[91] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629 – 639, 1990.

[92] P. Perona and J. Malik. Scale-space edge detection using anisotropic diffusion. In *Proc. of IEEE Computer Society Workshop on Computer Vision*, pages 16–22, 1997.

[93] I. Pitas and A. N. Venetsanopoulos. Morphological shape decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12:38–45, 1990.

[94] Javier Ruiz Hidalgo. The representation of images using scale trees. Master's thesis, University of East Anglia, School of Information Systems, 1999. In preparation.

[95] J. M. Sacks. Target discrimination utilizing median filters, 1984. US Patent number, 4,603,430.

[96] P. Salembier and M. Kunt. Size sensitive multiresolution decomposition of images with rank order based filters. *Signal Processing*, 27:205–241, 1992.

[97] P. Salembier and J. Serra. Flat zones filtering, connected operators and filters by reconstruction. *IEEE Transactions on Image Processing*, 8(4):1153–1160, August 1995.

[98] D. Schonfeld and J. Goutsias. Optimal morphological filters for pattern restoration. In *Proc. SPIE*, volume 1199, pages 158–169, 1989.

[99] J. Serra, editor. *Image analysis and mathematical morphology*, volume 1. Academic Press, London, 1982.

[100] J. Serra. Introduction to mathematical morphology. *Computer vision, graphics and image processing*, 35:283–305, 1986.

[101] J. Serra. Alternating sequential filters. In J. Serra, editor, *Image analysis and mathematical morphology Volume 2: Theoretical Advances*, volume 2. Academic Press, London, 1988.

[102] J. Serra, editor. *Image analysis and mathematical morphology Volume 2: Theoretical Advances*, volume 2. Academic Press, London, 1988.

[103] J. Serra and P. Salembier. Connected operators and pyramids. In *Proceedings SPIE*, volume 2030, pages 65–76, 1994.

[104] P. Soille and L. Vincent. Determining watersheds in digital pictures via flooding simulations. In M. Kunt, editor, *Visual Communications and Image Processing '90*, volume SPIE-1360, pages 240–250, 1990.

[105] J. Sporring, M. Nielsen, L. Florack, and P. Johansen, editors. *Gaussian Scale-space theory*. Kluwer Academic Publishers, 1997.

[106] S. R. Sternberg. Grayscale morphology. *Computer Vision, Graphics and Image Processing*, 35:333–335, 1986.

[107] Bart M. ter Haar Romeny, editor. *Geometry-Driven Diffusion in Computer Vision*. Kluwer Academic Publishers, 1994.

[108] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley, London, 1971.

[109] Corinne Vachier and Fernand Meyer. Extinction value: a new measurement of persistence. In Ionas Pitas, editor, *Proceedings of 1995 IEEE Workshop on nonlinear signal and image processing*, volume 1, pages 254–257, June 1995.

[110] R. van den Boomgaard and A. Smeulders. The morphological structure of images: the differential equations of morphological scale-space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(11):1101–1113, November 1994.

[111] Reinier van den Boomgaard. *Mathematical Morphology: Extensions Towards Computer Vision*. PhD thesis, University of Amsterdam, 1992.

[112] L. Vincent. Graphs and mathematical morphology. *Signal Processing*, 16:365–388, 1989.

[113] L. Vincent. Morphological area openings and closings of greyscale images. In *Workshop on Shape in Picture*. NATO, 1992.

[114] L. Vincent. Grayscale area openings and closings, their efficient implementation and applications. In J. Serra and P. Salembier, editors, *Proceedings of the International Workshop on Mathematical Morphology and its Applications to Signal Processing*, pages 22–27, May 1993.

[115] Luc Vincent. Fast opening functions and morphological granulometries. In *SPIE*, volume 2300, pages 253–267, 1994.

[116] Marco Ferretti Virginio Cantoni. *Pyramidal Architectures for Computer Vision*. Plenum Press, 1994.

[117] J Weickert, S. Ishikawa, and A. Imiya. Linear scale-space has first been proposed in japan. *Journal of Mathematical Imaging and Vision*, 10(3):237 – 253, 1999.

[118] A. P. Witkin. Scale-space filtering. In *8th. International Joint Conference on Artificial Intelligence*, pages 1019 – 1022, 1983.

[119] A. P. Witkin. Scale-space filtering: a new approach to multi-scale description. In *Image Understanding*, chapter 3, pages 79–95. Ablex, 1984.

[120] A. L. Yuille and T. A. Poggio. Scaling theorems for zero crossings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:15 – 25, 1986.