Algorithms and Art

J. Andrew Bangham (around 2005)

Algorithms

1

Algorithms for Art

2

Shape

3

Algorithms: MSER's and sieves

Figure 1.1: *Linear blurring, top right, simplifies at the expense of smoothed and indistinct edges. Median filters, bottom left, have clearer edges but imprint their shape (here a disc) onto the image. Whereas, bottom right, shows that the sieve ('buZZ' Photoshop plugin, Fo2PiX, Cambridge, UK, 2001) simplifies without changing the shapes of regions it preserves.*

Algorithms and scale-space

The one dimensional sieve algorithm was devised as an answer to the challenge of distinguishing noise from the tiny, picoamp, currents flowing through single protein molecules. Following their discovery by Neher and Sackmann we were measuring them in frog lens cells.

Scale Space Filters

Classical signal and image filters are linear and, for these, there are well developed theories. For example, in the 1980's Witkin [86], Koenderinck [49] and others [76, 52] see also [40] drew attention to the particular importance of scale-space preserving, diffusion, filters. The goal is to simplify the image without introducing any new features (in this context: extrema). The idea is best illustrated by example.

Imagine projecting a black and white image onto a screen for an hour. Now turn off the projector and turn on a thermal imager. White areas will be warmer than dark areas and the image will be visible. Over time heat diffuses from the warm to cool areas and as locally hot and cold spots are smoothed out so the image becomes blurred and, with fewer thermal maxima and minima, simpler. At no time during the diffusion process are new warm and cold regions created and this property is known as scale-space causality preserving.

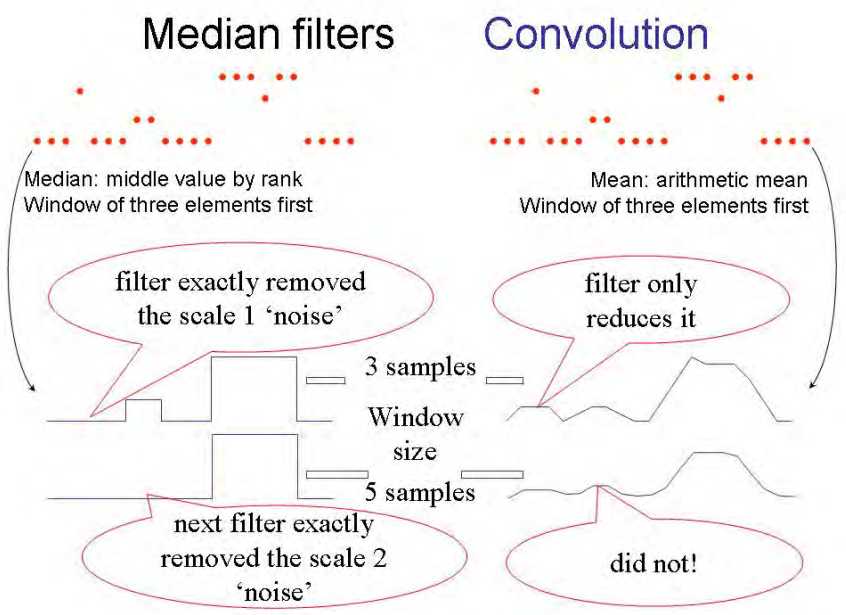


Figure 1.2: *Showing that median filters robustly reject impulsive noise where linear convolution filters do not. The signal, shown as points at the top, is either filtered using a window of 3 samples, upper trace, or 5 samples, lower trace. Whereas the running mean filter smoothes the signal, right column, the median filters do not. Instead, median filters completely eliminate short lived outliers, i.e. medians are robust. However, neither median nor mean filters preserve scale space.*

Rank filters Note, there is an implicit assumption that important features are associated with local maxima and minima (extrema). Gaussian filters uniquely among linear filters have this property [46]. In the early 1990's the library of causality preserving filters was extended to include non-linear diffusion filters [64, 59] that produce attractive results but confound scale and contrast. Other non-linear filters that were explored included morphological dilation (max) and erosion (min) filters [48]. Like median filters, Figure 1.1, they have structuring elements that imprint their shape on the image. Unlike medians however, they are very sensitive to noise (a characteristic that is exploited elsewhere to generate 'painterly' effects). By contrast non-linear sieves [2, 6](best conference paper and see Figure 1.1 bottom right) also use rank ordering properties but are extremely robust to noise and do not have structuring elements.

Median filters are widely discussed in the literature, and used in industry and image editors. One of the claimed advantages over linear filters in two dimensions is that they preserve the edges of objects, Figure 1.1 and in one dimension they preserve the transients (edges) of pulses (see Figure 1.2).

However, this is only roughly true. Despite the publication of numerous algorithms for optimally weighting median filters and generating optimal structuring elements by the non-linear filtering and mathematical morphology research communities, two prob-

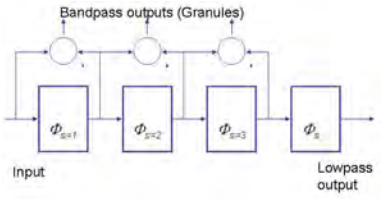


Figure 1.3: *Sieves are a series of increasing scale rank order filters (ϕ_S, where S is the filter scale). Where $max_{S,E}$ is a locally maximum extremum of scale S, similarly $min_{S,E}$, the filters, ϕ, can be c where at each scale $max_{S,E}$ are merged with their nearest neighbours. o Where $min_{S,E}$ are merged. M Where first $max_{S,E}$ then $min_{S,E}$ are processed and N where it is $min_{S,E}$ then $max_{S,E}$, or m where extrema are processed data order. All preserve scale-space, robustly reject outliers and are idempotent.*

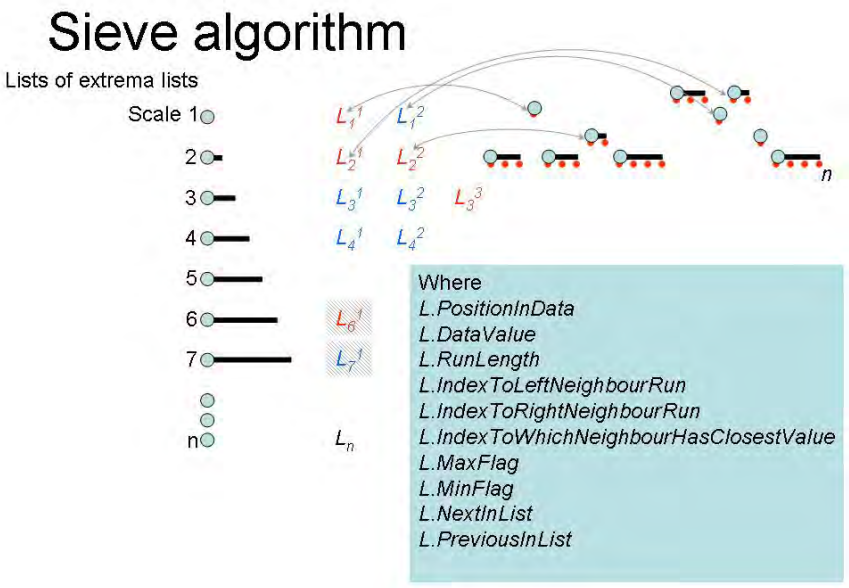


Figure 1.4: *Illustration of the 1D sieve algorithm. In one pass the n data values (red dots) are run-length coded (cyan circles) and local extrema, identified by 'looking' left and right, are mapped into a list of lists. One extrema list for each scale, 1 to n. The filtering process then starts by merging all elements in the scale 1 list with their nearest, in value, neighbouring runs. Each merge typically requires two pointers to be changed as two runs are linked and the new, longer, run is remapped into the list of lists. Filtering stops when the desired scale is reached and the output is rebuilt.*

lems remain. Firstly, experiments show that the claimed ability of median filters to preserve the shape of pulses (in one dimension) is, on average, no better than linear Gaussian filters [18, 73].

Likewise in two dimensions, some edges 'look' better after median filtering than after Gaussian filtering (blurring) but detailed analysis shows they are located no more accurately than after Gaussian filtering. Worse, in two dimensions the shape of the window (structuring element) is imprinted on the output of the filter and it becomes difficult to disentangle underlying properties of the image from systematic errors that have been added, Figure 1.1. A second consequence is that such independent filters do not preserve scale-space causality and this too means that median filtering obscures underlying, large scale, information in images.

Scale Space sieves solve the problem

Sieves are covered by Segmentis patents US 6,081,617, US 5,912,826, US 5,917,733 together with European equivalents.

Although developed for single channel analysis, the concept of sieves [2] was first exploited for analysing protein structures [3]. It shown to be the best way to distinguish hydrophobic from hydrophyllic regions. The algorithm was criticised because, at the time, it was not understood 'why' the algorithms worked so well.

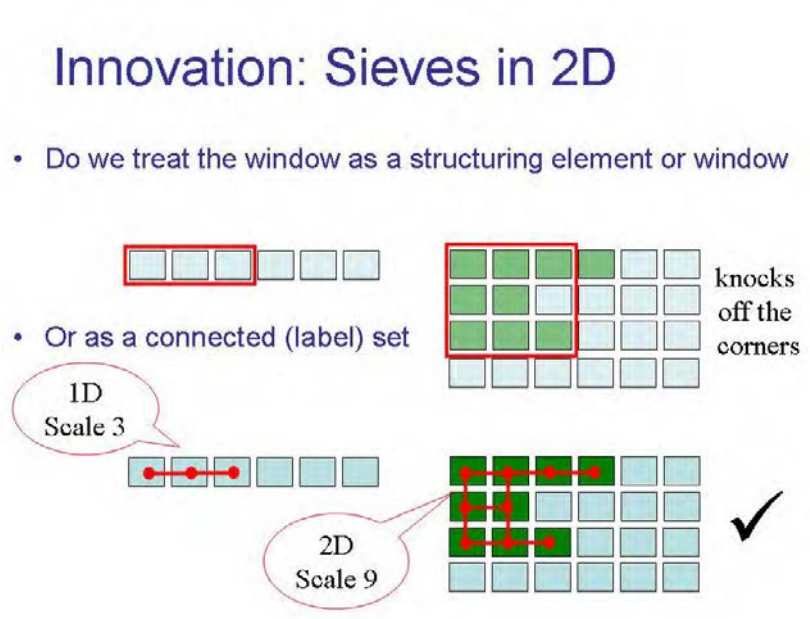


Figure 1.5: *Sieves do not use structuring elements (top). Instead, they operate on level connected sets of pixels, i.e. they follow edges in the data: they are shape free. Bottom right, the sieve is about to remove a set of nine pixels that form a local extremum. Unlike a filter based on a structuring element or window (top) the shape of the selected set follows the edges. In 1D increasing scale sieves remove increasingly long extrema, in 2D they remove extrema of increasingly large area and in 3D, volume.*

A criticism that precipitated more detailed work, in the 1990's, on the robustness of one-dimensional sieves [4, 6, 7, 5]. In one dimension there is no difference between an algorithm that operates on a window and one that operates on level connected sets, however, in higher dimensions the difference between the two is profound [91], Figure 1.5 and Figure 1.1 bottom.

The key properties of sieves arise because they use level connected sets. At this time there was also interest in 'reconstruction filters' that also appear to preserve scale space causality (Vincent, Salembier) that are something of a hybrid between sieves and classical mathematical morphology.

Initial results Simple experiments on segmentation [45, 43, 41, 42, 44] and pattern recognition [69, 6, 45, 68, 70] provided a better insight into their practical value and how sieves might be applied to segmentation [6, 67, 8]. Sieves proved very useful and the work was, therefore, followed up by establishing the mathematical properties of 1D sieves [22, 10, 9, 20], multidimensional sieves [14, 12, 11], their relation to weighted median filters [90, 89, 88] and mathematical morphology [15]. A short formal description of sieves is given in Chapter 2.

Complexity of sieve algorithms It should be emphasised that the obvious ways to implement sieves have a high order complexity,

$> O(n^2)$ (where n is the number of samples or pixels in the image) and this did not encourage other research laboratories to work with sieves. Fo2PiX (UEA, Segmentis) did not to draw attention to the patents that explain how it can be implemented.

A fast algorithm is essential. A 1D implementation is summarised in Figure 1.4. Sieving starts by passing though n data points once to map the signal into the list of lists and a second time to pass through each list of the list of lists to both filter, $< n$ elements, and rebuild the processed signal. Note that as the scale becomes larger so the number of lists reduces yielding $O(n)$ [13]. This is quite unlike classical morphological or linear filters where processing time per point usually increases with scale. The 2D case is similar where runs become patches. It is more complex because each patch usually has more than two neighbours requiring a search for the neighbour with the closest value to the current extremum. However, patches are either small and so have few neighbours, or are large in which case they are only analysed when the numerous small regions have been eliminated. The algorithms work out to be approximately $O(N^{1.1})$ in two or more dimensions [11]. Currently, there are two implementations of 2D sieves niether have been optimised to take advantage of modern pipelining

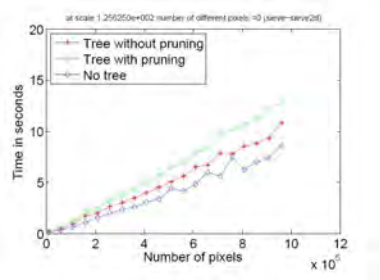


Figure 1.6: *Order complexity of different versions of the sieve, $O(N^{1.1})$*

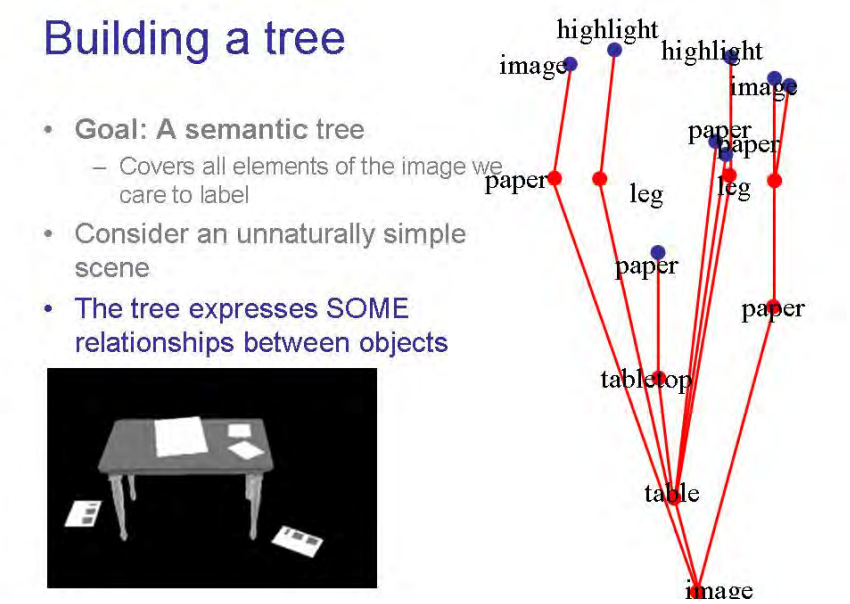


Figure 1.7: *Shows a simple image, left, and the corresponding semantic, object, tree. It is a goal of computer vision to do this automatically. It is not possible using Gaussian filters, wavelets or Fourier transforms.*

Merging at each scale can be done in various ways. If, at each scale, the maxima and minima are merged in data sequence, the result is equivalent to using a recursive median filter at that

scale, an m sieve. It is left-right assymetrical, i.e. parsing left to right is likely to yield a different result from right to left. At each scale an alternative is to merge first the maxima then the minima, or vic versa. These two alternatives are also different, i.e. they are up-down asymmetric. In practice, particularly in two or more dimensions, these four alternatives yield almost identical results and all four are have a robustness comparable to median filters.

A slew of papers on 2D images then followed. Once again their robustness was established [32, 33]. More importantly, a new representation of the sieve transform was developed: a new twist.

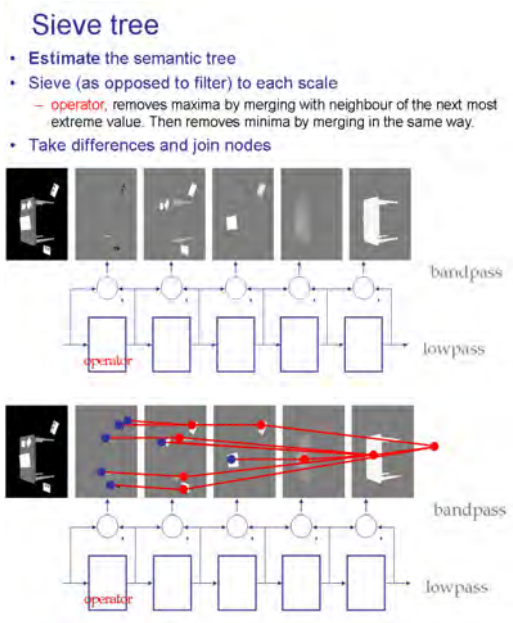


Figure 1.8: *Top, a sieve decomposition produces granules that are closely related to the objects. Bottom, shows that a sieve goes a long way to achieving the decomposition into objects. Thus the sieve looks a promising starting point for analysing real images.*

	1D-sieve	2D-sieve	DTCWT	LBP	Co-occurrance
Mean	0.718	0.943	0.556	0.509	0.692

Table 1.1: *Reliability identifying anisotropic textures,* 1 *is perfect. The 2D-sieve performs the best.*

The Sieve decomposition and generating tree structures

Simply filting an image is useful, about 60% of objects that volunteers identified (about 200 people each marking up 50 images) included large scale extrema and so removing detail by removing small scale extrema is likely to concentrate information. But it is crude. What is really needed is a hierarchy of objects such as that shown in Figure 1.7. Albeit a stylised image it makes a point because, unlike the Fourier transform or wavelet decomposition, the sieve will actually generate such a hierarchy, Figure 1.8 in one pass. This forms a conceptual framework for understanding how sieves might be used.

In reality, it is more difficult. A sieve decomposition of real images produces trees with far too many nodes [16, 17, 37]. It is only recently that methods for simplifying the tree and, as important, efficient code for implementing such simplifications, have become available. Figure 1.9 shows output from the implementation with

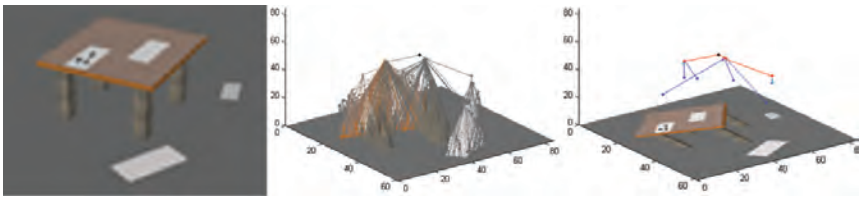


Figure 1.9: *Left panel shows a simple image. Middle panel shows the associated sieve tree. It is a mapping of the image, transformed by no loss of information. Right panel shows the result of a simple clustering algorithm to control the building of the tree during the transform. If a node (granule) is sufficiently similar to its parent the two are simply merged. The concentration of information is greatly increased.*

an order complexity shown in Figure 1.6. It is promising.

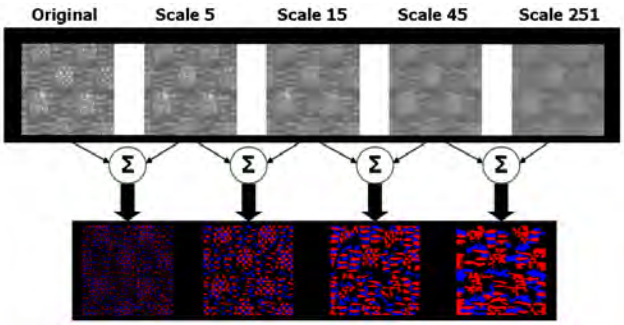


Figure 1.10: *A textured image sieved to five scales using a $\mathcal{M}$-sieve. Resulting Channel images are Bi-polar. Red is used to denote +ve granules and blue -ve granules.*

Applications of sieves

Applied to dynamic shape recognition Recently there have been thorough studies into the relative merits of sieves over the best competitor algorithms. For example, the 1D sieve can generate features for lip reading (hidden Markov models implemented in HTK) and the results remain the best so far, first outlined in [31] and culminating in [57].

Applied to texture recognition The identification of textures is a longstanding computer vision problem. Particularly, anisotropic textures such as cloth where what it looks like depends on the viewing angle. Table 1.1 shows result suggesting that the 2D-sieve is the best. Texture is characterised by sieving each texture image to scales, $[s_1 \dots s_N]$ where $\log_{10} s_n$ are equispaced between 0 and $\log_{10} S_{max}$, S_{max} is the maximum chosen scale and N the number of sieved images.

Tex-Mex features are formed from statistics derived from channel images. Noting that the setting $S_{max} = 30$ removes all the texture from the images and setting $N = 5$ results in five images sieved to scales $[1, 2, 5, 13, 30]$. Five channel images are formed from these sieved images at scales 0 to 1, 1 to 2, 2 to 5, 5 to 13 and 13 to 30. Figure 1.10 shows some example sieved images and resulting channel images. The intensity of the granule, or channel, images as a function of scale is an indicator of the scale-distribution of the texture features.

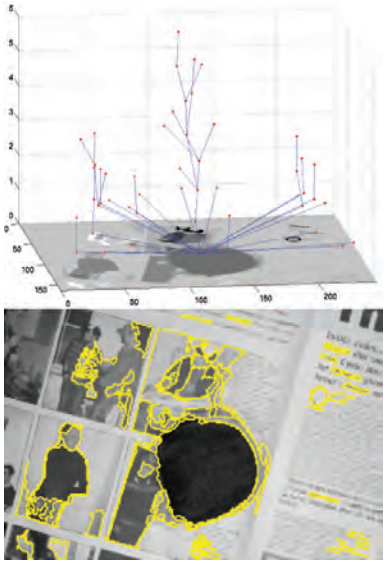


Figure 1.11: *Bottom: an image with its associated stable salient contours. These contours can be assembled into a saliency tree (top) which is an edited version of the full sieve tree.*

Applied to feature point detection Early explorations into using sieves to help match stereo pairs of images [34, 61, 60, 62, 16, 21] have been overtaken by a method that uses 'Maximal Stable Extremal Regions', MSER [56]. However, recent work shows that

the MSER appears to be a special case of the sieve (an opening sieve). Actually, alternating sequential and recursive sieves are found to be still more robust [R. Harvey and Y. Lan, in preparation]. Harvey also made a systematic comparison of algorithms [c.f. K. Mikolajczyk, C. Schmid, *A performance evaluation of local descriptors* Technical Report Oxford, to appear PAMI] that show that sieve algorithm outperforms competitor methods for finding salient regions.

Image retrieval Current interest is in finding regions in the image that are likely to remain unaffected by noise, projective transformations, compression and lighting change. In a comprehensive set of trials [58] a type of region known as Maximally Stable Extremal Regions (MSERs) were found to be the best performing. It turns out that MSERs are generated by a variant of the sieve algorithm known as open/close-sieves. It is therefore possible to parse a sieve tree and to generate *Stable Salient Contours* (SSCs) which are carefully selected nodes from the sieve tree that have all the stability and robustness properties associated with MSERs. Thus, as in Figure 1.11 the sieve tree generates stable regions "for free".

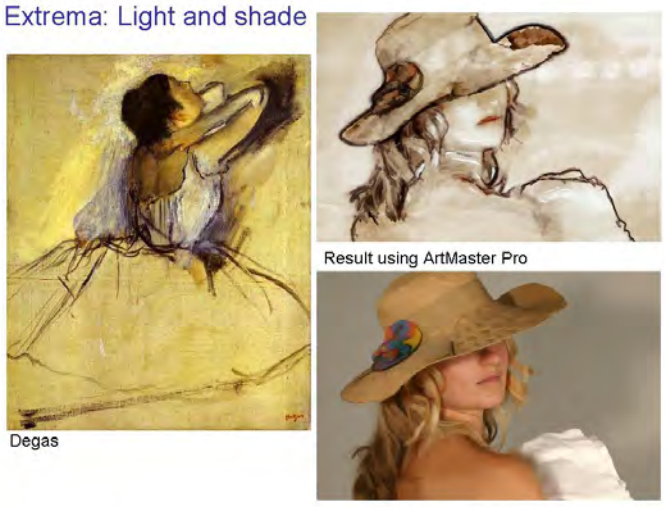


Figure 1.12: *Highlights and lowlights (extrema) are important in pictures. So too are edges, but not too many and not too slavishly accurate.*

Related Recently, the sieve algorithm has started to appear in the technical literature under different names apparently reinvented independently. In addition to the MSER (above) a Dutch team reported an algorithm that partially achieves the sieve [84]. A

French team describes the algorithm, they refer to finding image components [63].

Applied to finding components for artistic pictures It appears, therefore, that sieves form a promising starting point for a number of computer vision solutions. For most of these, however, the sieve is just one component of a number of algorithms that are required for the overall functionality.

However, there is one application area where the algorithm makes a particularly large contribution: forming artistic pictures. It is reported in [19] that a sieve based scale-space decomposition replicates a basic aspect of painting. Namely, it identifies light and dark regions of a photograph and does so at multiple scales. It parallels the way artists represent light and dark at multiple scales, Figure 1.12.

The sieve finds extrema of light and dark, preserves the edges of such areas and does so at multiple scales. Experience in the laboratory and commercial market shows that it contributes greatly to producing good digital art. It is this application of sieves that is pursued in Fo2PiX.

See full technical details on Page 2-2.

Initially, the algorithm was released as a simple Photoshop plugin. However, it quickly became clear that the properties of the sieve lead people to use it not as an 'effect' but as a tool.

For the first time it was possible to extract artistic components from the original photograph that could be used to create pictures. Photoshop actions were constructed to make the process easier but it became clear that a better environment for creating pictures, one that used 'steps', would be extremely helpful, see Page **??**.

The wide variety of applications in which the sieve appears to excel suggests that the algorithm will find wide commercial use as the fast sieve algorithm becomes better known.

Colour sieves Usually the sieve works on the luminance colour channel (grayscale). There is a conceptual problem extending it to colours since it depends on find an *order* (maxima and minima) and one can only order in one dimension, i.e. one colour channel. However, there is a pragmatic work around. The sieve has recently been extended into the color domain [28, 29] via the use convex hulls to define color extrema which are then merged to their nearest neighbours, found using a Euclidean distance measure. Figure 1.13 shows an example color sieve decomposition of a sample image.

Arguably, it is the best 'posterising' (colour segmentation) result yet seen. However, at present the algorithm itself is too slow to be practical. It needs more engineering.

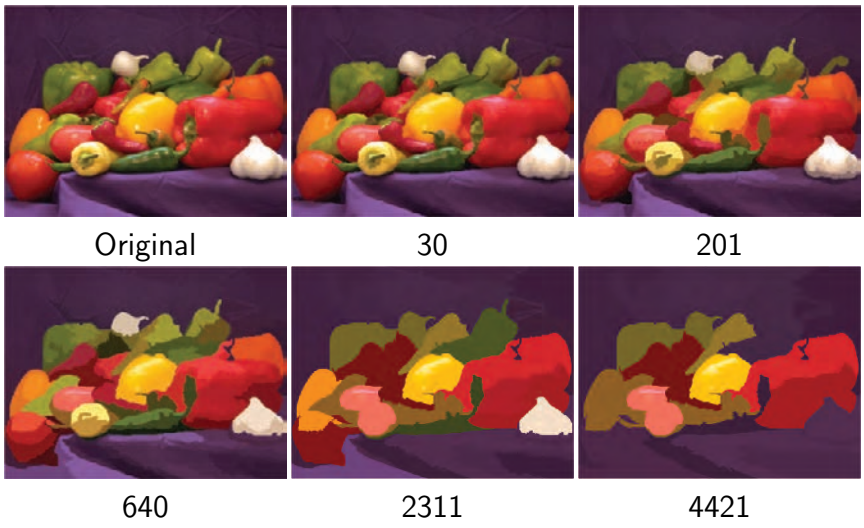


Figure 1.13: *Sample image and RGB space color sieve decomposition to labelled scales.*

Algorithms for Art

2 Algorithms for Art

This chapter is extracted from the paper underpinning a prizewinning 'best poster' [19].

Abstract

Artists pictures rarely have photo-realistic detail. Tools to create pictures from digital photographs might, therefore, include methods for removing detail. These tools such as Gaussian and anisotropic diffusion filters and connected-set morphological filters (sieves) remove detail whilst maintaining scale-space causality, in other words new detail is not created using these operators. Non-photorealistic rendering is, therefore, a potential application of these vision techniques. It is shown that certain scale-space filters preserve the appropriate edges of retained segments of interest. The resulting images have fewer extrema and are perceptually simpler than the original. A second artistic goal is to accentuate the centre of attention by reducing detail away from the centre. The process also removes the detail providing perceptual cues about photographic texture. This allows the 'eye' to readily accept alternative, artistic, textures introduced to further create an artistic impression. Moreover, the edges bounding segments accurately represent shapes in the original image and so provide a starting point for sketches.

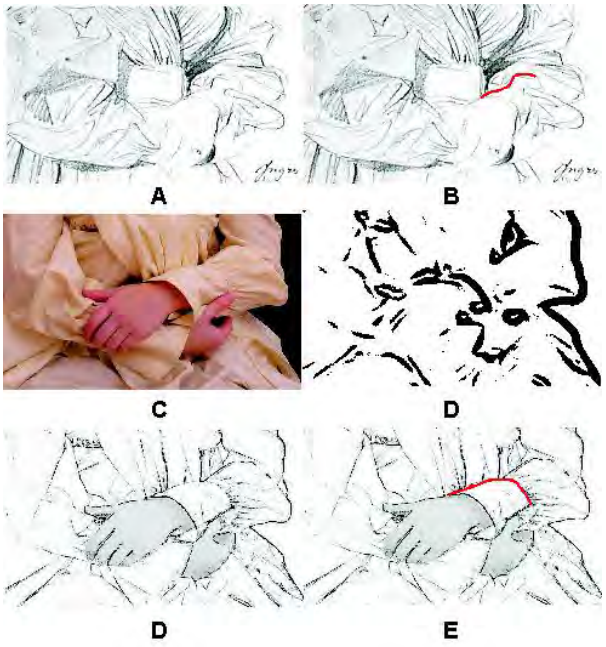


Figure 2.1: *Hockney [38] draws attention to how, in this Ingres drawing (A), "the cuff of the left sleeve is not followed 'round the form' as you would expect, but carries on into the folds". (B) Red overlay indicates the relevant lines. (C) Photograph of a similar subject. A Sobel edge filter (D) does not reveal the artistic line. (E) Shows in red the line line from (D) that follows the cuff rather than the light. There is too much detail. Blurring does reveal the large scale cuff-to-sleeve highlight however it yields edges (D) that are incomplete.*

Introduction

A photographer tends to choose uncluttered backgrounds and make careful use of focus to direct attention. Of course, lens blurring is both easy and effective for it exploits the natural and powerful way in which the brain rejects non-foveated regions of a scene (they are simply out-of-focus). The technique finds its way into rendering, digital art, advertising, and video through the Gaussian blur filter widely used to de-focus background material. But the method is rarely used by painters. Rather, they direct attention by selecting detail and manipulating textures and geometry.

By contrast, a painter starts with a blank canvas, adds paint and the more skilled knows when to stop. It is the progressive addition of detail that characterizes the process of producing representational art in which only some detail directly represents that in the original scene. It difficult to capture representational detail manually from three-dimensional (3D) scenes onto two-dimensional (2D) canvases, but this does not satisfactorily explain why trained artists limit the amount of detail they use. After all two dimensional, photographic quality, images have been traced for over five centuries by those projecting images onto surfaces using concave mirrors and lens [38]). But the evidence from the resulting pictures suggests that artists pick only those details that resonate with their artistic

interpretation. They *choose* to ignore some objects and lots of detail. Painting is not photography.

The importance of controlling detail

Selectively removing detail simplifies a photograph and is implicit in existing methods for producing painterly pictures. Systems for creating pen-and-ink drawings from existing images clearly remove both color, L_c, and spatial detail, L_s, (these two related ideas are lumped together in $L_r = \lambda(L_c, L_s)$) and simultaneously add techniqe and artistic detail in the form of pen strokes [74]. In the case of painting Haeberli *samples* the image and then modulates and spreads the color over a region using a brush, an action that also adds technique detail by simulating the medium and stochastic brush strokes that simulate the artistic interpretation [30]. By modelling the flow of water dragging pigments over paper Curtis [25] removes detail from the original photograph and simultaneously substitutes texture detail (L_t) replicating a watercolor painting. Hertzmann starts by removing detail with large brushes and then uses finer brushes to selectively refine the picture where the sketch differs from blurred photograph [35]. These methods sub-sample the source image either before or after smoothing: the standard way to remove detail and prevent aliasing. In this paper, however, we concentrate on another way to control the level of detail in a digital image. We do not address the separate problems of adding technique and artistic detail.

Of course, form is extremely important and should be exploited in representational art where available, as to an artist working from life or to a digital artist working with a three dimensional graphics system [36]. This, however, is not enough and, anyway, is less easy to come by when starting with a photograph alone. Here, the play between light extrema, light and shade is key. The notion receives some quantitative support from observations on the process of painting, Figure **??**. The artist started with gray paper and subsequent analysis of the fifty images taken at two minute intervals during painting shows that the mean intensity of 80% of paint marks, added by the artist, are more extreme than the mean of their immediate surroundings. In other words, the artist built the portrait by adding ever lighter and darker strokes (indeed it is difficult to see how else it could be done!). Moreover, a quantitative association between light and dark extrema and objects has been reported: when asked to outline objects within photographs, 60% of regions that people demarcated manually were associated

with light or dark extrema [32]. These observations support the contention that control of scale and extrema are important in the creation of pictures from photographs.

The new algorithm decomposes an image by ordering level-connected sets in two ways: first by scale (detail) and secondly by value (extrema of light and shade). It provides the digital artist with access to a choice of differently scaled detail. Unlike blurring, the system simplifies without distorting edges and it can, in principle, be used 'underneath' other painterly algorithms both to simplify the image prior to sampling and to provide, perhaps better, lines for clipping [53].

Selectively removing detail simplifies a photograph and is implicit in existing methods for producing painterly pictures. Systems for creating pen-and-ink drawings from existing images clearly remove both color and spatial detail and simultaneously add artistic detail in the form of pen strokes [74]. In the case of painting Haeberli *samples* the image, effecting a simplification. He then modulates and spreads the color over a larger region using a brush, an action that also adds technique detail by simulating the medium and stochastic brush strokes that are modulated by edge gradients to imply artistic interpretation [30]. By modelling the flow of water dragging pigments over paper Curtis [25] removes detail from the original photograph by a form of blur and simultaneously substitutes texture detail that replicates watercolor. Hertzmann starts by removing detail with large brushes and then uses finer brushes to selectively refine the picture where the sketch differs from blurred photograph, a form of multiscale removal of detail [35]. These methods sub-sample the source image either before or after smoothing: the standard way to remove detail and prevent aliasing. In this paper, however, we concentrate on another way to control the level of detail in a digital image. Scale-space filtering to both remove detail and uncover large scale image maxima (highlight) and minima (lowlights).

Chiarascuro (bright highlights and dark shadows) and its manipulation characterizes the work of many artist's, since the renaissance. Hockney [38] draws attention to the way the line used by Ingres follows the light rather than the form (as evidence of optical assistance of which he gives many other examples). Figure 2.1(A) shows an extract from the original drawing of Madame Godinot 1829. (B) Shows the artist's lines that, Hockney argues, follows the light. We illustrate the problem by analysing the photograph shown in (C).

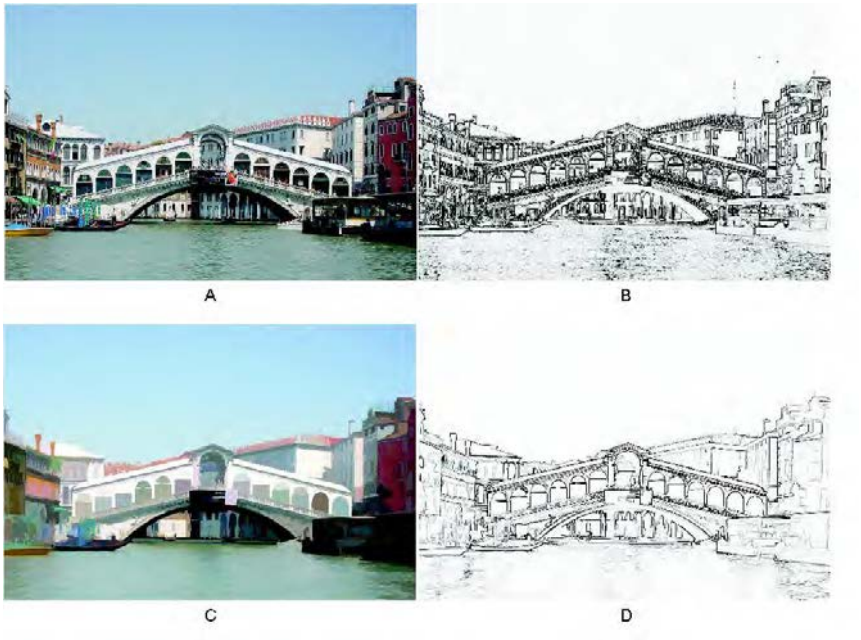


Figure 2.2: *(A) Photograph and (B) associated edges. (C) Sieved to remove detail and (D) fewer edges make a more sketch-like picture.*

Conventional edge detectors (D and E) produce a prominent line along the back edge of the cuff (F): a boundary that was ignored by the artist. The problem lies with the local edge filter. Typically they have a small region-of-support that responds to the strong edges around the form and so cannot 'see' the larger picture (the Canny filter (D) is more complex but has related problems). Simplifying the image by blurring, Figure 2.1(G), increases the region-of-support and does both reveal the expected large scale highlight running from the cuff *into* the sleeve but it removes detail.

Thus Gaussian scale-space filters meet two requirements of a pre-processor for non-photorealistic rendering. As such it is used to segment images and create pictures where the artist's eye-gaze governs the level of detail rendered at different positions in the image [26]. Whilst pleasing, the results are limited in the range of styles they can support because such filters introduce significant geometric distortion reflected in the edges (H) that, whilst graphically interesting, do not form the basis of a sketch: important to the artist. Here we pursue alternative scale-space filters.

Simplification maintaining scale-space causality

In image processing the process of removing detail from a digital image emerged from studies on finding *salient*, edges [55]. The work with Gaussian filters lead to the, theoretically tidy, representation of images known as scale-space [40, 87, 50]. This

important concept is seen as a requirement of image simplification systems since it guarantees that extrema in the simplified image are not artifacts of the simplification process itself. Computation systems that preserve scale-space causality are usually associated with Gaussian filters [1] and diffusion [65] in which the image forms the initial conditions for a discretization of the continuous diffusion equation: $\nabla \cdot (c \nabla f) = f_s$. If the diffusivity is a constant this becomes the linear diffusion equation, $\nabla^2 f = f_s$ which may be implemented by convolving the image with the Green's function of diffusion equation: a Gaussian convolution filter. Of course, care is needed when this equation is discretized [51] but, if it is done correctly, a scale-space with discrete space and continuous scale may be formed[1].

Approximations to this Gaussian blur filter are common in image-editors and graphical rendering systems. The problem with blurring, when finding salient edges at large scales, is that edges wander away from the true edge and objects become rounded: a consequence of convolution, Figure 2.1H. It is better if diffusivity depends upon contrast, as in anisotropic diffusion, but computation then becomes lengthy and unwanted small scale detail with a high enough contrast may nevertheless be preserved. In other words, as with linear diffusion, there is an interaction between the intensity and scale of an object.

More recently the multiscale analysis of images has been explored in the field of mathematical morphology. Two rather different approaches to constructing a morphological scale-space have been suggested. In the first [79, 48] the image is either eroded or dilated using an elliptic paraboloid. As is often the case in morphology (and convolution filters) the shape of the structuring element (window) dominates over structure in the image. That said however, the brush like 'texture' introduced by the structuring element can be useful in digital art and is used in photo-editor plug-ins (Adobe Photoshop Gallery Effects).

The second approach uses those connected-set alternating sequential filters sometimes termed sieves [11]. Sieves [15] appear in a variety of guises but they have their starting point in connected-set graph morphology [72, 81, 82] and watersheds [75]. At small scale they filter out maximally stable extremal points [47] or detail and at larger scale, entire objects. Figure 2.2(C) confirms that fine detail is removed and that edges (D) of remaining features are well preserved. These edges are more sketch-like than those derived

[1] Or with a discrete scale parameter if preferred.

directly from the image (B) or from a Gaussian smoothed image Figure 2.1(H). This, and the more poster-like simplified image provides a reason to explore further how these scale-space filters can be used in non-photorealistic rendering.

Methods

We implement the sieve described in [11]. The algorithm first creates a list of all maxima and minima. These extrema are level 8- (or 4-) connected-sets of pixels that are then sorted by area. A scale decomposition progresses by merging all extrema of area 1 to the next most extreme neighbouring pixel(s), i.e. all extreme values are replaced by the value of the next most extreme adjacent pixel. If the segment remains an extremum it is added to the appropriate scale extremum list. The decomposition continues by merging all extrema of scale 2, 3 and so on. Thus, for example, by scale 100 there are no maxima (white) or minima (black) areas of less than 100 pixels. We use low-pass, band-pass and high-pass filters created by combinations of sieving operations.

The image is represented as a graph [80] $G = (V, E)$. The set of edges E describes the adjacency of the pixels (which are the vertices V). A pixel, x, is connected to its eight neighbours. A region, $C_r(G, x)$, is defined over the graph that encloses the pixel (vertex) x, $C_r(G, x) = \{\xi \in C_r(G) | x \in \xi\}$ where $C_r(G)$ is the set of connected subsets of G with r elements. Thus $C_r(G, x)$ is the set of connected subsets of r elements that contain x. For each integer $r \geq 1$ the operators ψ_r, γ_r, $\mathcal{M}^r$, $\mathcal{N}^r : \mathbf{Z}^V \rightarrow \mathbf{Z}^V$ are defined as $\psi_r f(x) = \max_{\xi \in C_r(G,x)} \min_{u \in \xi} f(u)$,, $\gamma_r f(x) = \min_{\xi \in C_r(G,x)} \max_{u \in \xi} f(u)$, $\mathcal{M}^r = \gamma_r \psi_r$, $\mathcal{N}^r = \psi_r \gamma_r$. $\mathcal{M}^r$ is a connected-set grayscale opening followed by a closing defined over a region of size r.

The types of sieve known as M- or N-sieve are formed by repeated operation of the $\mathcal{M}$ or $\mathcal{N}$ operators that are also known as connected alternating sequential filters. An M-sieve of f is the sequence $(f^{(r)})_{r=1}^{\infty}$ given by

$$f^{(1)} = \mathcal{M}^1 f, \quad f^{(r+1)} = \mathcal{M}^{r+1} f^{(r)}, \quad r \geq 1 \tag{2.1}$$

The N-sieve is defined similarly. It has been shown how connected set openings can be performed in approximately linear time [85] using a modification to Tarjan's disjoint set algorithm and a similar implementation is used here for the alternating sequence of openings and closings that forms the sieve.

$f^{(r)}$ is a low-pass filter removing all extrema up to scale r. $f^{(1)} - f^{(r)}$ is a high-pass filter keeping all extrema from scale 1

to r and $f^{(s)} - f^{(r)}$ is a band-pass filter for extrema (granules) of scales between s and r.

The sieve requires two orderings. Level connected-sets are ordered by value and extrema are removed in order of scale. Where a pixel represents a triple, red, blue and green (RGB), there is no clear way of jointly ordering by value. This is addressed in two ways. We note that all three channels have a high correlation with brightness (unlike hue, saturation, value) and so the three colour planes are sieved independently, the RGB-sieve. The effect of removing detail can be seen by comparing Figure 2.2(A) and (C). It is evident that (B) has less detail yet, in contrast to alternative scale-spaces, the edges of large scale objects are preserved. This is, perhaps, more obvious in the edge images compare Figure 2.2B and D. The resulting image is both grayer than the original (extrema are removed) and the colours change slightly because they arise from the signals obtained from colour channels sieved independently. There is no link between a pixel and its colour.

A new alternative is the convex 'colour sieve' which follows from a geometric interpretation of the colours of a region and its neighbours. A convex hull is fitted to points in the region projected into colour space. All points that lie on the convex hull itself are extreme (ref. to anonymous paper here) and those enclosed are not extreme. This provides an ordering - the distance from the convex hull. This definition is tidy because many typical colour transformations such as gamma correction and linear transformations affect the geometry but not the topology of the convex hull and the extrema inherit the invariance properties.

To simplify the image we merge smaller regions into larger ones without introducing additional extrema by merging to the neighbour with the closest Euclidean distance. Neighbouring regions with identical colour distances are further ordered by computing the difference of their luminance $L = (r + g + b)/3$ and further tiebreaks are achieved by ordering by their G,R and B values. The merging is repeated iteratively until idempotence.

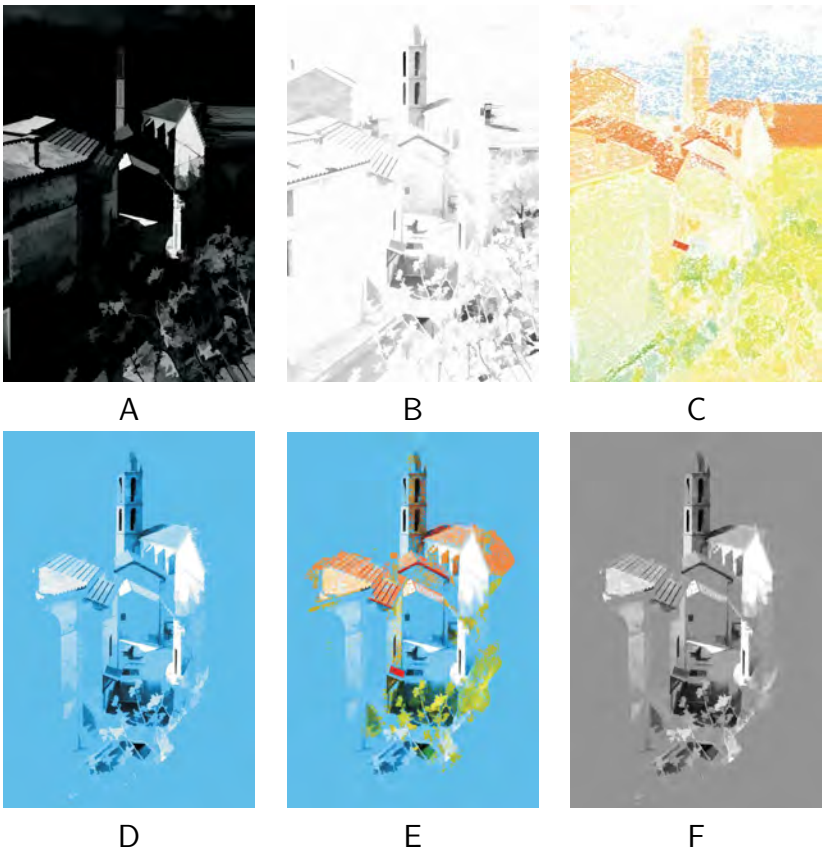


Figure 2.3: *'(A) Band-pass highlights displayed against a black backround. (B) Band-pass lowlights displayed against a white backround. (C) Band-pass HSV saturation channel used to select colours, such as the red roofs, that stand out from backround. (D) Highlights and lowlights incorporated into a picture. (E) Replacing the chroma of (D) with colours from (C) adds colour highlights (visible in colour prints, pdf only) that are not visible in the luminance image (F).*

Results

We consider how high- and low- lights extracted from the image using a high-pass sieve can be incorporated into non-photorealistic image renderings. The grayscale image is band-pass sieved, $q = f^{(s)} - f^{(r)}$, to find the associated scale highlights, where $q > 0$, $y = q$, else $y = 0$. Figure 2.3A shows the result. Likewise, lowlights where $q < 0$, $y = 1 + q$, else $y = 1$, Figure 2.3B. Combined by painting them onto a mid-tone background, Figure 2.3D, the effect is similar to chalk and charcoal. Colour highlights are located at a particular range of scales by sieving the HSV saturation channel and using this to control the chroma, $q_{hue} = f^{(s)}_{hue} - f^{(r)}_{hue}$, where $q_{hue} > t$, $hue = q_{sat}$, $sat = q_{sat}$, $val = 1$, else $hue = 1$, $sat = 1$,

$val = 1$, where t is a threshold that can be adjusted by the user[2]. The colour highlights have been painted by *replacing* the NTSC chroma values on the canvas with those from the colour highlights, Figure 2.3E[3]

Interestingly, as Livingstone [54] points out, by colouring the canvas with the complement (the two colours sum to gray) of, for example, the red roofs an optical illusion is created. The effect is to make the colour appear more interesting that it otherwise might be for two reasons. Firstly, it challenges the viewer's vision system (and monochrome display devices) because the NTSC grayscale (perceptual luminance) does not change even when the chrominance does: all trace of the colour change vanishes in an NTSC grayscale print, Figure 2.3F. Exactly how Figure 2.3E appears on the printed page depends on the printer software. For readers able to see colour this in colour, Figure 2.3E plays to another colour illusion. The sharp boundary between the complementary colours enhances perceived brightness [54].

The brushwork in Figure 2.3A-F places the centre of attention in the centre of the picture by leaving the periphery free of detail. This is typical of many paintings. We, therefore, devise an algorithm that automatically selects a central region to be rendered in more detail than a middleground which, in turn is set against a backround with low detail. In other words, an algorithm that creates foreground, M_f, and middleground, M_m masks.

The process is outlined in Figure 2.4B. The idea is to create masks that exactly follow the boundaries of objects in the image and which place M_f in the centre and M_m around it. Each mask is created separately. The image is sieved to a scale, s, quantised by an amount q and the flat zones labelled. Those zones that intersect the innermost darker-cross and the pale-cross are then marked as shown by the white segments, Figure 2.4B. It white segment has an area A. The part of the pale-cross not covered by the marked zones has an area $\overline{A}$. We then search for a scale, s, and quantisation q that minimises difference between the areas, $A - \overline{A}$. An exhaustive search of only a few s and q suffices. Typical masks for M_f and M_m are shown in Figure 2.4B bottom panel and they have been used to combine images created by RGB-sieving to three scales, Figure 2.4C. The result is more detail towards the centre of the image helps draw the viewers attention. The changes

[2]For digital artists the convention, that user-adjustable thresholds should be avoided, is not relevant.

[3]PDF version of the paper.

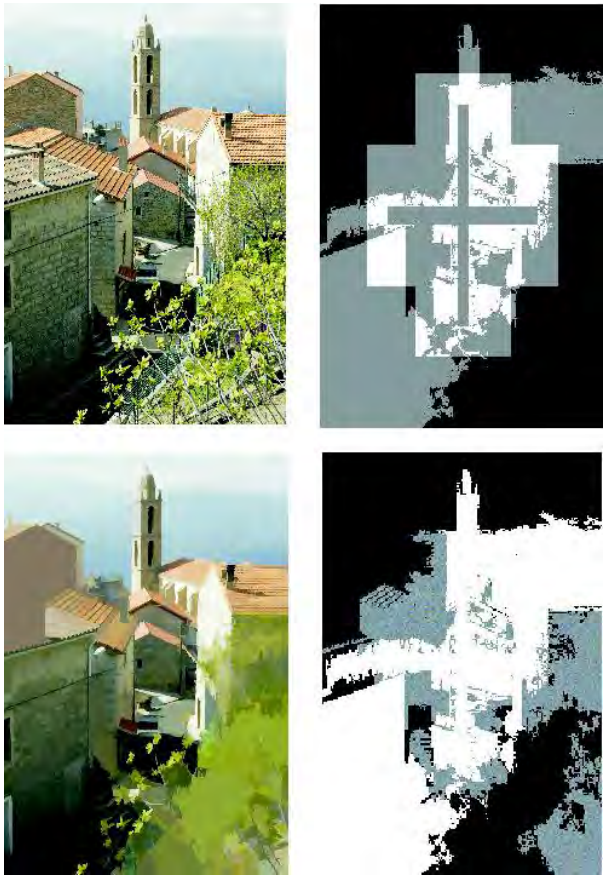


Figure 2.4: *Photograph. (B) Top, the central cross designating the region of primary interest together with its border. Bottom, white segment indicates the region automatically selected to be foreground resolution, gray segment at middleground resolution, black at background resolution. (C) The union of foreground, middleground and background resolution image segments. Controlling the level of detail helps direct attention to the interest points in the centre.*

A B

Figure 2.5: *'(A) Figure 2.4 textured using a photograph of a water-colour wash and pencil cross-hatching (B).*

of scale are subtle since they follow the boundaries of objects in the image rather than some externally imposed mask.

Notice that many of the areas in Figure 2.4C are flat because texture, fine detail, has been removed by the sieve. This creates an opportunity to replace the original texture with another as an artist might do by using paint or pencil. The image was mixed with a photograph of a simple watercolour wash (not shown) (multiplication rather than addition), Figure 2.5A, produces a distinctly watercolour like result. Mixing the same texture with the original is much less effective (easily achieved in Photoshop) because the underlying original detail leaves old texture cues intact. A more extreme example is shown in Figure 2.5B. Here, a photograph of an area of pencil cross-hatching is mixed with Figure 2.4C. Unlike Figure 2.5A however, each labelled level set in Figure 2.4C is filled with a segment of the cross-hatched picked from a random position in the texture image. In other words each of the objects is hatched separately. This is most clearly seen in the large flat areas top left and bottom right. Superimposing the edges completes the effect.

Conclusion

The sieve, particulary the convex-hull colour, algorithm is a useful starting point for non-photorealist rendering of photographs. It provides the digital artist with access to a choice of images with differently scaled detail. Unlike blurring, the system simplifies without distorting edges thus the edges provide a useful starting

point for creating sketches. The large scale level sets it creates provide a mechanism for segmenting the image into regions that, by having different amounts of detail, create a centre of attention. It is data-driven rather than dependent on a pre-defined geometry. Band-pass sieves also allow artistically important high-, low- and bright coloured highlights to be found. Thus in Figure 2.6A the sieve removes small scale detail and the highlights are now treated in a way that is redolent of Figure 2.1A with edges that follow the light Figure 2.6B. We do not attempt to map photographs directly into art: the artist is still essential. Rather the aim is to provide the digital artist with tools. Further automatation might include object recognition to create ways of improving composition and tools to balance colour composition.

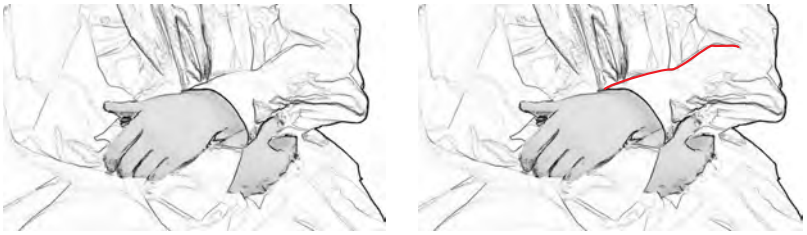
A  B

Figure 2.6: *(A) Sobel edges after sieving RGB Figure 2.1C to scale 2000: the line carries into the folds. (B) Red line indicates the line.*

So what about shape and appearance

Figure 3.1: *Distortions of shape and appearance using active appearance models to make the portrait more like a: teenager, young adult (equivalent to the original), older adult, more feminine, more masculine, Modigliani, Botticelli, Mucha. (Try David Perret's website demonstration* http://www.dcs.st-and.ac.uk/%7Emorph/Transformer/ *)*

Applying Research into Soft Models

Introducing statistical models of images. The ideas applied to the attractiveness of flowers (it is the perception by bees that counts) can be explored using the Matlab toolbox has just been released in association with a paper in Science [83](*http://www.cmp.uea.ac.uk/Research/cbg/Documents/Bangham-Coen-Group/AAMToolbox/AAMToolbox.htm*).

Shape models Two dimensional sieves used for art produce a 'shape free' decompositions of images. This is its great power, the decomposition is affine independent. However, shape is important. The most direct way to analyse and work with shapes is use statistical shape models. These were introduced in botany for

analysing plant parts [39, 71] and recently refined for understanding the evolution of flower colour [83]. Separately, it was developed for analysing medical images [24, 23, 24], speechreading [57], talking heads [78, 77], and tracking people (Matthews). Perret used them to study the psychophysical basis of beauty [66].

Figure 3.1 shows a statistical model developed by Perret distorting a portrait of Emma. Technically, a statistical analysis of shape and shape-free appearance is used to create a model. The images are then generated by the model from different positions in the shape-appearance space.

Figure 3.2 shows the output from a similar statistical model. This time created by analysing a set of photographs. To reshape the original the model was fitted to the original then the result synthesised by shifting closer to the mean shape, keeping the appearance the same.

Figure 3.2: *Using the model to rotate a photograph.*

Applications to photography and digital art It would be desirable to apply the technique to photography and creating artistic pictures. One can imagine 'retouching' photographs by slightly enlarging the eyes, increasing the smile or changing the pose and one can imagine producing pictures with the 'flavour' of old masters (it will not work well, if only because an inkjet print of an oil painting is lamentable). The problem is size. There is a practical constraint on the size of image that can be modelled using the current methods. There are, however, at least two ways in which this limitation can be overcome and these form the subject of current research at UEA.

Figure 3.3: *Shape model definition.*

How the models are created in Matlab Our toolbox for generating statistical shape and appearance models is available together with sample botanical data (http://www.cmp.uea.ac.uk/Research/cbg/Documents/Bangham-Coen-Group/AAMToolbox/AAMToolbox.htm). A set of portraits were centred in a 400x400 pixels image. A point model was created by dotting 199 points around the face, hair, shoulders and hat, (Figure 3.3). Primary points (black) were at easily recognisable features, the corners of the mouth, nose, etc. Secondary points (circles) were equally spaced between the primary points (the pmplace routine helps by automatically sliding the points long a cubic spline fitted to the points).

The positions of n points for each leaf ($[X_j, Y_j], j = 1, , n$) were manually selected using "pmplace" function of the 'Shape

model toolbox' that automates the rest of following process. The points for each portrait are saved in separate files each containing $2n$ data values. The mean shape is calculated from M portraits, $[\bar{X}_j, \bar{Y}_j]$ where $j = 1, \cdots, N$, and the mean $\bar{X}_j = \sum_{i=1}^{i=M} X_{i,j}/N$ and likewise for Y (ignoring the distinction between primary and secondary points). Differences between shapes associated with differing species are reflected in the way portraits shapes differ from the mean. This is captured by subtracting the mean from each point $D_{i,j} = X_{i,j} - \bar{X}_j, D_{i,n+j} = Y_{i,j} - \bar{Y}_j$, notice that the X and Y differences are concatenated into a single data vector forming a column of $2N$ values. D is a $2n$ column by M row data matrix where each row represents a leaf and each column a set of measurements.

The measurements are correlated, i.e. if one compares a wide portrait with a narrow one, adjacent points tend to differ in similar ways. In other words, the measurements do not provide a compact description of shape. To find a compact linear description of shape we can construct the smallest set of linearly independent vectors that span the space of interest. To find independent (orthogonal) measures of shape, the differences for each image, D_i, are represented as a linear combination of orthogonal principal components $D_i = b_{i,0}\mathbf{p}_0 + b_{i,1}\mathbf{p}_1 + b_{i,2}\mathbf{p}_2, + \cdots, b_{i,2n-1}\mathbf{p}_{2n-1}$ where p_l is the first principal component and $b_{i,l}$ is a weight. Thus each portrait shape, i, has a vector of weights b_i. To the extent that D can be represented linearly in this way (there may be underlying non-linearities) the weights, $b_{i,j}$, associated with portraits i are j independent measures of shape that can substitute for $D_{i,j}$.

Principal component analysis Principal component analysis (PCA) is used to find P where $P = [p_0 p_1 \cdots p_{2n-1}]$, such that $\mathbf{b}_i = \mathbf{P}' D_i$ where the superscript tick, $(.)'$, denotes the transpose (a capital T is an alternative). The components are ordered to account for decreasing variance and it is found that a good representation of the shape of the portrait, i, can be made using the weights of just the first three components $[b_{i,0} b_{i,1} b_{i,2}]$. These account for most of the variance of shape about the mean shape. The estimated shape, $\hat{D}_i$, corresponding to just these components, $\hat{b}_i = [b_{i,0}, b_{i,1}, b_{i,2}, \cdots, b_{i,2n-1}]$, can be found from $D_i = \mathbf{P}\mathbf{b}_i$, Figure 3. $\mathbf{P}$ is called the Point Distribution Model (PDM) and it is obtained from $\mathbf{D}$ in Matlab by finding the covariance matrix, $C = cov(D)$, the eigenvectors $(\mathbf{E})$, where $[\mathbf{E}, \mathbf{V}] = eig(\mathbf{C})$, and by sorting $\mathbf{E}$ by decreasing importance according to the eigenvalues,

$\mathbf{V}$ (the covariance of $\mathbf{E}$). Thus, [vals, I]=sort(diag(V), 'descend') and the PDM is the sorted eigenvalues, $\mathbf{P} = \mathbf{E}(:, \mathbf{I})$. To find $\mathbf{b}$ from $\mathbf{D}$ in Matlab use $\mathbf{b}(i, :) = \mathbf{P}' * \mathbf{b}(i, :)$.

Bibliography

[1] 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*, Vol. 8:pp 26–33, 1986.

[2] J. A. Bangham. *Improved filtering techniques: median filter and datasieve*. Patent, 1987.

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

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

[5] J. A. Bangham and R. V. Aldridge. Multiscale decomposition using median and morphological filters. In *IEEE Workshop on Nonlinear Signal Processing*, pages 6.1–1.1 – 6.1–1.4, Tampere, Finland, 1993.

[6] J. A. Bangham and T. G. Campbell. Sieves and wavelets: multiscale transforms for pattern recognition. In *Proceedings IEEE Workshop on Nonlinear Signal Processing*, pages 1.1–4.1 –1.1–4.6, 1993.

[7] J. A. Bangham and T. G. Campbell. Sieves and wavelets: multiscale transforms for pattern recognition. In *IEEE Workshop on Nonlinear Signal Processing*, pages 1.1–41 – 1.1–4.6, Tampere, Finland, 1993.

[8] 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.

[9] J. A. Bangham, P. Chardaire, P. Ling, and C. J. Pye. Multiscale nonlinear decomposition: the sieve decomposition theorem. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 18:518–527, 1996.

[10] J. A. Bangham, P. Chardaire, and P. D. Ling. Multiscale nonlinear decomposition. In *International Conference on Mathematical Morphology*. Kluwer, 1994.

[11] J. A. Bangham, R. Harvey, P. Ling, and R. V. Aldridge. Nonlinear area and volume scale-space preserving filters'. *Journal of Electronic Imaging*, 5(3):283–299, July 1996.

[12] J. A. Bangham, R. Harvey, P. Ling, and R. V. Aldridge. Nonlinear scale-space from *n*-dimensional sieves. In *Proc. European Conf. on Computer Vision*, volume 1, pages 189–198. Kluwer, 1996.

[13] J. A. Bangham, S. J. Impey, and F. W. D. Woodhams. A fast 1d sieve transform for multiscale signal decomposition. In *EUSIPCO*, pages 1621–1624, 1994.

[14] J. A. Bangham, P. Ling, and R. Harvey. Scale-space from nonlinear filters. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 18(5):520–528, 1996.

[15] J. A. Bangham and S. Marshall. Image and signal processing with mathematical morphology. *IEE Electronics and Communication Engineering Journal*, 10:117–128, 1998.

[16] J. A. Bangham, K. Moravec, R. W. Harvey, and M. Fisher. Scale-space trees and applications as filters, for stereo vision and image retrieval. In *British machine vision Conference*, pages 113–122, 1999.

[17] J. A. Bangham, J. Ruiz-Hidalgo, R. Harvey, and G. Cawley. The segmentation of images using scale-space trees. In *BMVC*, pages 1–8, 1998.

[18] J.A. Bangham, T. G. Campbell, and M. Gabbouj. The quality of edge preservation by non-linear filters. In *IEEE Workshop on Visual Signal Processing and Communications*, pages 37–39, 1992.

[19] J.A. Bangham, S.E. Gibson, and R.W. Harvey. The art of scale-space (supporting conference 'best poster' prize). In *British Machine Vision Conference*, volume 1, page 10 pages, 2003.

[20] 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.

[21] J.A. Bangham, K. Moravec, R.W. Harvey, and M.H. Fisher. Scale-space trees and applications as filters, for stereo vision and image retrieval. In *10th British Machine Vision Conference (BMVC99)*, Nottingham, UK, 1999.

[22] P. Chardaire, J.A. Bangham, C.J. Pye, and D.Q. Wu. Properties of multiscale morphological filters namely the morphology decomposition theorem spie nonlinear signal processing. In *SPIE*, volume 2180, pages 148–154, 1994.

[23] T. Cootes and C. Taylor. Active appearance models. *Lecture notes in Computer Science*, 1407:484–498, 1998.

[24] T. F. Cootes, C. Taylor, and A. Lanitis. Active shape models: evaluation - their training and application. *Computer vision and image understanding*, 61:38–59, 1995.

[25] C. J. Curtis, S. E. Anderson, J. E. Seims, K. W. Fleischer, and D. H. Salesin. Computer-generated watercolor. In *Proceedings of SIGGRAPH 97*, pages 421–430, Los Angeles, California, August 1997. ACM SIGGRAPH, ACM Press/Addison-Wesley. ISBN 0-89791-896-7.

[26] Doug DeCarlo and Anthony Santella. Stylization and abstraction of photographs. In *SIGGRAPH'2002 Proceedings*, pages 769–776. ACM, 2002.

[27] Ringo Doe. This is a test entry of type @ONLINE. `http://www.test.org/doe/`, June 2009.

[28] S. Gibson, R. Harvey, and G. Finlayson. Convex colour sieves. In *in Proceedings 4th International Conference, Scale-Space 2003*, volume 2695 of Lecture Notes in Computer Science, pages 537–549, 2003.

[29] D. Gimenez and A.N. Evans. Colour morphological scale-spaces for image segmentation. In *Proceedings of the British Machine Vision Conference 2005 (BMVC '05), Oxford, England*, volume 2, pages 909–918, 2005.

[30] P. E. Haeberli. Paint by numbers: Abstract image representations. In *Proceedings of SIGGRAPH 90*, pages 207–214, August 1990.

[31] R. Harvey, I. Matthews, J. A. Bangham, and S. J. Cox. Lip-reading from scale-space parameters. In *IEEE Computer vision and pattern recognition conference*, pages 582–587. IEEE, 1997.

[32] R. W. Harvey, A. Bosson, and J. A. Bangham. The robustness of some scale-spaces. In *Proceedings of British machine vision Conference*, pages 11 – 20, 1997.

[33] R. W. Harvey and J. A. Bosson, A. Bangham. Scale-spaces filters and their robustness. In *Proc. of First Int. Scale-space Conf.: Scale-space theory in Comp. vision*, pages 341 – 344, 1997.

[34] R.W. Harvey, K. Moravec, and J.A. Bangham. Stereo vision via connected-set operators. In *Proc. European Signal Processing Conference*, Rhodes, 1998.

[35] Aaron Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In M. E. Cohen, editor, *Proceedings of SIGGRAPH 98*, pages 453–460, Orlando, Florida, July 19-24 1998. ACM SIGGRAPH, ACM Press/Addison-Wesley. ISBN 0-89791-999-8.

[36] Aaron Hertzmann and Denis Zorin. Illustrating smooth surfaces. In Kurt Akeley, editor, *Siggraph 2000, Computer Graphics Proceedings*, pages 517–526. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.

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

[38] D. Hockney. *Secret Knowledge*. Routledge and Kegan Paul, 2001.

[39] G. W. Horgan. The statistical analysis of plant part appearance - a review. *Computers and Electronics in Agriculture*, 31:169–190, 2001.

[40] 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.

[41] J. Impey, Stephen, J.A. Bangham, and J. Pye. Multiscale 1 and 2d signal decomposition using median and morphological filters. In *Workshop on Mathematical Morphology and its Applications*, pages 17–21, 1993.

[42] S. J. Impey and J. A. Bangham. Pattern recognition by area decomposition of hsv components'. In *Signal Processing VIII*, volume 1, pages 169–171, 1996.

[43] S.J. Impey and J.A. Bangham. Improvements to the 'top hat' transform used for analysing pigmented patches on flower petals. In *Signal Processing VII: Theories and applications*, pages 852–855, 1994. Holt, Cowan, Grant and Sandham.

[44] Stephen Impey. *Nonlinear scale-space*. PhD thesis, University of East Anglia, 1999.

[45] Pye J., Bangham J A., Impey S., and Aldridge R V. 2d pattern recognition using multiscale median and morphological filters. In Venatsonopoulos Vernazza and Braccini, editors, *Image Processing: Theory and applications*, pages 309–312. Elsevier, 1993.

[46] M. Babaud J. Babaud, A. P. Witkin and R. O. Duda. Uniqueness of the Gaussian kernel for scale-space filtering. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 8:532 – 540, 1986.

[47] M. Urban J. Matas, O. Chum and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, volume 1, pages 384–393, Cardiff, 2002.

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

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

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

[51] Tony Lindeberg. Scale-space from discrete signals. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12:234–254, 1990.

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

[53] Peter Litwinowiez. Processing images and video for an impressionist effect. In *SIGGRAPH'97 Proceedings*, pages 407–414. ACM, 1997.

[54] Margaret Livingstone. *Vision and Art: the biology of seeing*. Harry N. Abrams, New York, 2002.

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

[56] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proc. British Machine vision Conference*, pages 384–393, 2002.

[57] I.A. Matthews, T. Cootes, J.A. Bangham, S.J. Cox, and R.W. Harvey. Extraction of visual features for lipreading. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(2):198–213, 2002.

[58] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65(1/2):43–72, 2005.

[59] 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.

[60] K. Moravec, R.W. Harvey, and J.A. Bangham. Connected-set filters to improve and obtain stereo disparity maps. In *British Machine Vision Conference*, 1998.

[61] K. Moravec, R.W. Harvey, J.A. Bangham, and M.H. Fisher. Using an image tree to assist stereo matching. In *IEEE ICIP'99*, Kobe, Japan, 1999.

[62] Kimberly Moravec. *Stereo*. PhD thesis, University of East Anglia, 2000.

[63] L. Najman and M. Couprie. Quasi-linear algorithm for the component tree. In *IS&T/SPIE Symposium on Electronic Imaging: Vision Geometry XII*, 2004.

[64] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 12(7):629 – 639, 1990.

[65] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12:629–639, 1990.

[66] D.I. Perrett, K. May, and S. Yoshikawa. Attractive characteristics of female faces: preference for non-average shape. *Nature*, 368:239–242, 1994.

[67] C. J. Pye and Bangham J A. 2d pattern recognition using 1d multiscale sieve decomposition. In *Signal Processing VII, 'Theories and applications*, pages 856–859. EUSIPCO94, 1994.

[68] C. J. Pye, Bangham J. A., and R. W. Harvey. Using a genetic algorithm to adapt 1d nonlinear matched sieves for pattern classification in images. In *SPIE*, volume 2424, pages 48–55. SPIE, 1995.

[69] J. Pye, J.A. Bangham, S. Impey, and R.V. Aldridge. 2d pattern recognition using multiscale median and morphological filters. In Venatsonopoulos Vernazza and Braccini, editors, *Image Processing: Theory and applications*, pages 309–312. Elsevier, 1993.

[70] Jeremy Pye. *Sieves*. PhD thesis, University of East Anglia, 2000.

[71] T. A. Ray. Landmark eigenshape analysis-homologous contours-leaf shape in syngonium (aracaeae). *Am. J. Botany*, 79:69–76, 2002.

[72] P. Salembier and J. Serra. Morphological multiscale image segmentation. In *SPIE Visual Communication and Image Processing*, pages 620–631, 1992.

[73] P. Salembier, J. Serra, and J. A. Bangham. Edge versus contrast estimation of morphological filters. In *IEEE ICASSP-93*, volume vol. V, pages pp 45–48, 1993.

[74] M. P. Salisbury, S. E. Anderson, R. Barzel, and D. H. Salesin. Interactive pen-and-ink illustration. In A.S. Glassner, editor, *Proceedings of SIGGRAPH 94*, pages 101–108, Orlando, Florida, July 1994. ACM SIGGRAPH, ACM Press/Addison-Wesley.

[75] 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.

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

[77] B. Theobald, J.A. Bangham, I. Matthews, and G.C. Cawley. Near-videorealistic synthetic talking faces: Implementation and evaluation (submitted on invitation). *Speech Communication Journal*, 2004.

[78] B. Theobald, G.C. Cawley, J.R.W. Glauert, J.A. Bangham, and I. Matthews. 2.5d visual speech synthesis using appearance models. In *British Machine Vision Conference*, volume 1, pages 43–52, Norwich, UK, 2003.

[79] 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.

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

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

[82] L. Vincent. Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms. *IEEE Trans Image Processing*, vol. 2:pp 176–201, 1993.

[83] A. C. Whibley, N. Langlade, C. Andalo, A. I. Hanna, A. Bangham, C. Thebaud, and Coen E. Evolutionary paths underlying flower color variation in antirrhinum. *Science*, September, 2006.

[84] M. H. F. Wilkinson and J. B. T. M. Roerdink. Fast morphological attribute operations using tarjan's union-find algorithm. In L. Vincent J. Goutsias and D.S. Bloomberg, editors, *Int Symposium on Mathematical Morphology*, pages 311–32. Kluwer Academic Publishers, 2000.

[85] M. H. F. Wilkinson and J. B. T. M. Roerdink. Fast morphological attribute operations using tarjan's union-find algorithm. In L. Vincent J. Goutsias and D.S. Bloomberg, editors, *Int Symposium on Mathematical Morphology*, pages 311–320. Kluwer, 2000.

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

[87] A. P. Witkin. Scale-space filtering. In *8th Int. Joint Conf. Artificial Intelligence*, pages 1019–1022. IEEE, 1983.

[88] O. Yli-Harja, J. A. Bangham, and R.W. Harvey. State reduction of recursive median filters. In *IS and T/SPIE 11th Symposium on Electronic Imaging Science and Technology*, San Jose, California, USA, 1999.

[89] O. Yli-Harja, P. Koivisto, J.A. Bangham, G.C. Cawley, R.W. Harvey, and I. Shmulevich. Simplified implementation of the recursive median sieve. *Signal Processing*, 81(7):1565–1570, 2001. 446BW SIGNAL PROCESS.

[90] O. Yli-Harja, I. Schmulevich, J.A. Bangham, R.W. Harvey, S. Dasmahapatra, and S.J. Cox. Run-length distributions of recursive median filters using probabilistic automata. In *Scandinavian Conference on Image Analysis, SCIA'99*, Kangerlussuaq, Greenland, 1999.

[91] J. A. Young, R. L.; Bangham. *Improved data processing method and apparatus: datasieve using a recursive ordinal filter*. Patent, 1994.