MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1320 March, 1992

The Multi-Scale Veto Model: A Two-Stage Analog Network for Edge Detection and Image Reconstruction

Lisa Dron

Abstract

This paper presents the theory behind a model for a two-stage analog network for edge detection and image reconstruction to be implemented in VLSI. Edges are detected in the first stage using the multi-scale veto rule, which states that an edge is significant if and only if it passes a threshold test at each of a set of different spatial scales. The image is reconstructed in the second stage from the brightness values adjacent to the edge locations. Among the key features of this model are that edges are localized at the resolution of the smallest spatial scale without having to identify maxima in brightness gradients, while noise is removed with the efficiency of the largest scale. There are no problems of local minima, and for any given set of parameters there is a unique solution. Images reconstructed from the brightnesses adjacent to the marked edges are very similar visually to the originals. Significant bandwidth compression can thus be achieved without noticeably compromising image quality.

Copyright © Massachusetts Institute of Technology, 1992

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-85-K-0124. L. Dron is supported by a graduate fellowship from AT&T Bell Laboratories. Support for the Analog VLSI for Machine Vision research project is provided by NSF Grant MIP-91-17724 and NSF and DARPA contracts MIP-88-14612.

1. Introduction

In a real-time system, it is desirable to find edges, or sharp changes in the image brightness function, *quickly* and *accurately*. Speed is necessary to save time which can be better spent on more computationally intensive processes, such as feature matching, which use the edges. Accuracy is needed to supply these processes with reliable input. Accurate edge detection means being able to selectively ignore gradients in the brightness function caused by high spatial-frequency features attributable to noise, while marking those caused by high frequency features such as corners and junctions. It also requires that the edges be well localized to the contours of features in the image which generate them. Noise can be removed by applying a linear lowpass smoothing filter. However, this has the effect of attenuating all high frequency components indiscriminately and introducing uncertainty in edge locations. Non-linear methods, such as median filtering, which preserve important edges and remove noise have been in existence for some time. These methods generally require more computation than linear filtering, however, and cannot be implemented by convolution. Of particular interest to designers of real-time systems are methods which can be built in silicon. One recently developed technique designed in analog VLSI is the resistive fuse network invented by Harris [8] based on the weak membrane model of Blake and Zisserman [3]. In this paper we propose another computational model which can also be implemented in analog VLSI and which overcomes some of the disadvantages of the weak membrane model.

The *multi-scale veto*, or MSV, model is similar to the weak membrane in that it assumes an image can be approximated by a collection of piecewise smooth functions. Edges are 'break points', *i.e.,* locations where the brightness function is not required to be smooth. The MSV model differs from the weak membrane, however, in two respects. It does not reconstruct the image from all of the data, but only from the brightness values of pixels on either side of the edges. Second, the networks used for edge detection and image reconstruction are physically distinct. As a result, problems associated with the non-convexity of the weak membrane are avoided.

The MSV model derives its name from the method it uses for detecting edges. Edges are defined as loci of sharp changes in the image brightness function which are significant over a range of spatial scales. An important aspect of the MSV model is that edges do not necessarily correspond to local maxima in the magnitude of the gradient. It therefore responds not only to step changes in brightness, but also to

strongly shaded surfaces which do not always give rise to well defined maxima in the gradient. On a discrete two-dimensional array edges occur between two pixels (nodes). The spatial scale is determined by the space constant of the smoothing network to which voltage sources proportional to the sampled brightness values at each pixel are connected. Differences are computed between the smoothed voltages at neighboring nodes of the network and compared to a threshold which is also a function of scale. In applying the multi-scale veto rule, two or more scales are used, and all must agree on the presence of a significant difference between two nodes before an edge is marked. If at any scale the difference between the smoothed brightnesses is below threshold, the edge is vetoed. As will be discussed in Section 4.1, this method allows edges to be localized at the resolution of the smallest scale, while noise is removed with the efficiency of the largest scale.

Two points, which are discussed later in detail, are significant to note about the MSV edge detection network:

- It does not require computation of second differences, and
- All of the difference operations and threshold tests at different scales can be performed on the same physical network.

Both points represent a considerable savings in circuitry, a crucial consideration if the network is to be designed to work with large image arrays.

The second piece of the MSV model is the reconstruction network. While this circuit performs nothing more complicated than interpolation from the brightness values next to the marked edges, it is significant that the images reconstructed in this manner are very similar visually to the originals. Since only a fraction of the original data points are needed for reconstruction—typically from 15–45% of the image, depending on the amount of detail in the scene—this means that we can save storage and transmission bandwidth by only encoding these values. Combined with existing compression methods such as run-length and Huffman coding, the total savings may be significant.

This paper is organized as follows: In the next section we review related work in edge detection, multi-scale methods and image reconstruction. In Section 3 we describe the circuit models of the edge detection and reconstruction networks, and in Section 4 we discuss performance issues and show results from computer simulations

on some test images. In the last section we compare the MSV model to the weak membrane and characterize the differences between the computations they perform.

2. Related Work

2.1. Edge detection and the use of multiple scales

As explained by Torre and Poggio [22], the numerical differentiation of images is an ill-posed problem that must be regularized in order to obtain a stable solution. The regularization function in this case takes the form of a smoothing filter which must be applied before differentiation. In most work in computer vision, edges are defined to be the loci of maxima in the magnitude of the smoothed brightness gradient and can be detected from zero-crossings in the second derivative. This is the basis on which many edge and line detectors, such as the Marr-Hildreth Laplacian-of-Gaussian (LOG) filter [17], the Canny edge detector [4], and the Binford-Horn line finder [9], have been designed.

As stated in the introduction, isotropic smoothing filters such as the Gaussian have the disadvantage that they smooth away important features as well as noise. Smoothing can displace points of maximum gradient, such as around the cusp of a brightness 'corner', or remove them altogether. Many efforts have therefore focused on developing more selective, edge-preserving smoothing methods. One possibility is non-linear filtering. The median filter [7], for example, has often been used in image processing because it is particularly effective in removing impulse, or 'salt-and-pepper', noise.

Another approach put forward in recent years is the idea of edge detection, or more precisely image segmentation, as a problem in minimizing energy functionals. The first proposal of this nature was the Markov Random Field (MRF) model of Geman and Geman [6]. In an MRF the minimum energy state is the maximum a posteriori (MAP) estimate of the energies at each node of a discrete lattice. The MAP estimate corresponds to a given configuration of neighborhoods of interaction. 'Line processes' are introduced on the lattice to inhibit interaction between nodes which have significantly different prior energies, thereby maintaining these differences in the final solution. Mumford and Shah [18] studied the energy minimization problem

reformulated in terms of deterministic functionals to be minimized by a variational approach. Specifically, they proposed finding optimal approximations of a general function $d(x,y)$, representing the data, by differentiable functions $u(x,y)$ that are minimizers of

$$E(u,\Gamma) = \mu^2 \int\int_R (u-d)^2 dxdy + \int\int_{R-\Gamma} |\nabla u|^2 dxdy + \nu|\Gamma| \qquad (1)$$

where Γ is a closed set of singular points, in effect the edges, at which u is allowed to be discontinuous. Blake and Zisserman [3] referred to (1) as the 'weak membrane' model, since $E(u,\Gamma)$ resembles the potential energy function of an elastic membrane which is allowed to break in some places in order to achieve a lower energy state. They derived a continuation method, which they referred to as the Graduated Non-Convexity (GNC) algorithm, to minimize (1) iteratively.

The weak membrane model was one of the first methods to be implemented in analog VLSI. Digital circuits for performing Gaussian convolution and edge detection began appearing in the early 80's [1,11]. The possibility of performing segmentation and smoothing with analog circuitry, however, did not seem practical until the problem had been posed in terms of a physical model. Harris [8] invented the first CMOS *resistive fuse* circuit for minimizing (1) on a discrete grid. A resistive fuse is a two-terminal non-linear element which behaves as a linear resistor over a certain voltage range, but transforms into an open circuit if the voltage across its terminals becomes too large.

The issue of scale arises in edge detection because of the tradeoff between accurate localization of features and sensitivity to noise. Since important features generally occur over a range of spatial scales, many methods have been based on the use of information at multiple scales. Marr and Hildreth first proposed finding edges from the coincident zero-crossings of different sized LOG filters. Witken [23] introduced the notion of scale-space filtering, in which the zero-crossings of the LOG are tracked as they move with scale changes. In the weak membrane model, there are two parameters to specify which, in a sense, determine the scale: μ, which controls the smoothness of the fitted solution $u(x,y)$, and ν, which determines the penalty assigned to the discontinuities. Richardson [21] developed a scale-independent iterative algorithm for minimizing an energy formulation similar to (1). In each iteration, the variational problem is solved for some input image, $d(x,y)$, and some value of μ and ν. The result is that feature boundaries apparent at the coarsest scale defined

by the initial values of μ and ν are localized with the resolution of the finest scale used in the last iteration. Small features, however, are not detected because they do not generate discontinuities at the coarse scale and hence are smoothed away. The principle applied in Richardson's algorithm is very similar to that of the multi-scale veto rule. The MSV model, however, does not involve solving a variational problem.

The MSV model differs from other edge detection methods in that it does *not* define edges as points of maximum gradient, and hence does not require second derivative operators. By defining edges as the loci of significant abrupt changes in the image brightness function, it detects edges generated by features which generate step changes in brightness, as well as those generated by features such as shaded surfaces that do not necessarily give rise to maxima in the gradient. The MSV model is similar to the weak membrane in that it assumes the image can be well-approximated by a set of piecewise smooth functions whose boundaries are the edges. Multiple scales are used in order to ensure that the differences measured between neighboring pixels are due to spatially significant features and not to noise. As will be discussed further in Section 4.1, the method allows good localization of features because, unlike the points of maximum gradient, points where the brightness differences are significant will not move with smoothing.

The edges produced by the MSV model are not as 'refined' as those produced by more complex methods such as Canny's edge detector [4] or Richard's CARTOON algorithm [20]. This is in part due to the way edges are defined, and in part due to the need to make the circuitry as simple as possible in order to minimize silicon area. There is no room for contour filling-in or texture edge removal. Our contention is that the edges produced by the MSV network are nonetheless functionally useful. We will demonstrate their usefulness in conjunction with the reconstruction network, and we believe that they will prove to be sufficient as well for other early vision tasks such as primitive feature matching.

2.2. Image Reconstruction

In the weak membrane, the functions $u(x, y)$ which minimize (1) given the discontinuity set, Γ, result from smoothing all the data with a filter of scale $1/\mu$, with the restriction that smoothing is inhibited across edges. In the MSV model the reconstructed image is generated by interpolation from the brightness values adjacent

to the marked edges, and hence uses only a fraction of the original data. Before continuing, we mention briefly some other methods for reconstructing images from sparse data points.

A significant amount of work in communications theory has been devoted to the problem of reconstructing signals from their zero-crossings. An often cited theorem by Logan [15] is that almost all bandpass one-dimensional signals of bandwidth less than one octave are uniquely specified by their zero-crossings. Curtis and Oppenheim [5] extended Logan's theorem to two dimensions and showed that any real two-dimensional doubly-periodic bandlimited function $f(x, y)$ is uniquely specified to within a constant scale factor by its zero-crossings, or its crossings of an arbitrary threshold. The number of zero-crossings needed to specify $f(x, y)$ may be large, however, and their method is not likely to be practical for reconstructing large images with significant high frequency components, since it requires precise knowledge of the zero- or threshold-crossing locations.

One well-known example of an instance where an image can be reconstructed from sparse data is the case of Mondrian patches, first used by Land and McCann [13] to demonstrate their theory of the computation of lightness. The human visual system is very good at determining the reflectance of an object, under a variety of illuminating conditions. Land and McCann showed that one could recover, to an arbitrary scale factor, the reflectances of Mondrian patches by measuring the ratio of brightnesses at each step change on a closed path around the image. Horn [10] later showed how the same computation could be performed on a parallel network by first computing and then thresholding the Laplacian of the logarithm of brightness. More recently, Blake [2] suggested a modification to Horn's algorithm by having the threshold operation depend on the magnitude of the gradient rather than the Laplacian of the logarithm of brightness. In a sense, the MSV model can be considered as an extension of these algorithms; although it is the original brightness function, and not surface reflectance which is being recovered. Algorithms for the computation of lightness first showed that under certain circumstances it is possible to regenerate an image from the differences in (log) brightness across patch boundaries, where there is a step change in brightness. In the MSV model, we show that an image can be recovered from the brightnesses adjacent to edges under more general conditions.

3. Circuit Models

In this section we describe the circuit models for the edge detection and reconstruction networks. As the actual circuit is currently in the design phase, implementation issues will not be discussed in this paper.

3.1. Edge Detection

The fundamental principle of this network is the multi-scale veto rule for detecting significant changes in the image brightness function. This rule states that an edge exists between neighboring pixels if and only if the change in brightness between them is significant over a range of spatial scales. The scales are determined by the space constants of isotropic smoothing filters applied to the entire image. Differences are computed between the smoothed values at neighboring pixels and compared to a threshold which is a function of the scale. If the magnitude of the difference is greater than threshold at each scale, an edge is marked. If at any of the scales the difference is below threshold, however, the edge is vetoed.

It is not necessary to build a multi-dimensional network in order to implement the multi-scale veto rule. By including time as a dimension, a single smoothing network with controllable spatial scale, such as the resistive grid with variable vertical resistances shown in Figure 1, can be used. The combined result of the threshold tests at each scale is encoded by a capacitor whose charge represents the AND of the different tests. The network shown in Figure 1(a) is one-dimensional; however, the extension to two dimensions is straightforward. By equating the current through the vertical resistors connected to the node voltage sources d_i, which are proportional to the sampled brightnesses, to the sum of the currents leaving the node through the horizontal resistors, one easily arrives at the resistive grid equation:

$$u_i - \frac{R_v}{R_h} \sum_k (u_k - u_i) = d_i \tag{2}$$

where the subscript k is an index over the nearest neighbors of node i.

The continuous 2-d approximation to this circuit is the diffusion equation

$$u - \lambda^2 \nabla^2 u = d \tag{3}$$

with

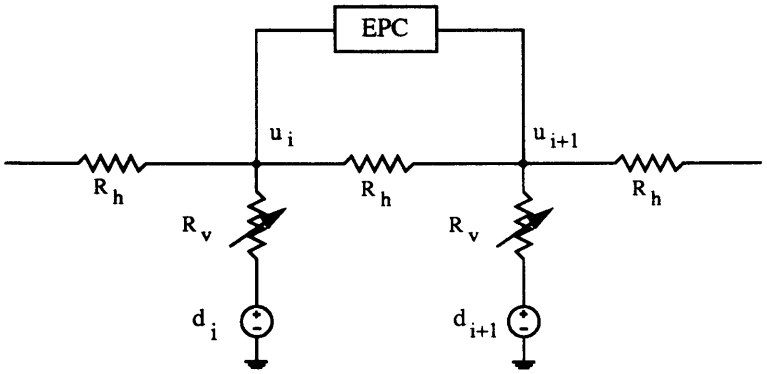
$$\lambda = \sqrt{\frac{R_v}{R_h}} \tag{4}$$

which is the characteristic length over which an point source input will be smoothed.

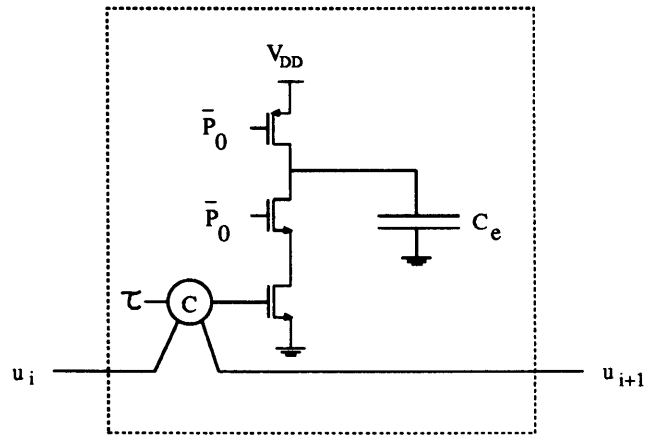
One operational cycle of the MSV network corresponds to sensing an image, performing the threshold tests at each scale, and offloading the results. The cycle is divided into time intervals with operations controlled by external circuitry. It is assumed that the number of threshold tests is small (~5–10) and that the length of time they require is short compared to the image acquisition time so that operation can proceed at frame rate. In the first interval, corresponding to image acquisition, during which the voltage sources d_i are generated, a control signal, P_0, connected to the edge precharge circuit goes high, pre-charging all of the capacitors, C_e. At the end of the sampling period, P_0 goes low and stays low for the remainder of the cycle. In the following intervals, R_v is changed to set the value of the space constant. The absolute value of the differences between neighboring node voltages are compared to a threshold, and the edge capacitors at sites where the tests fail are discharged. The final phase of the cycle corresponds to moving the edge charges and brightness values neighboring the edge locations onto another circuit where further processing takes place. The smallest scale used in the computation may correspond to $R_v = 0$, *i.e.*, no smoothing at all, and the largest one may correspond to $\lambda \gg 1$. The values used are externally set parameters.

3.2. The Reconstruction Network

The reconstruction network, as shown in 1-D in Figure 2, regenerates the image by interpolation from the brightness values on either side of the marked edges. Voltage sources proportional to the original brightnesses d_i are switched to the resistive grid according to whether or not the node is adjacent to an edge. The control signal which closes the switch is logically equivalent to the OR of the states of all the edge capacitors adjacent to the node. As seen from equations (2) and (3) with $R_h = \infty$ ($\lambda = \infty$), the distribution of voltages on the resistive network at non-edge nodes solve a discrete form of Laplace's equation. Along the outer border we impose the



(a) 1-d multi-scale veto edge detection network. d_i and d_{i+1} are voltage sources proportional to the sampled brightnesses. The box labeled EPC is the edge precharge circuit shown below.



(b) Edge precharge circuit. P_0 is a pulsed clock signal which goes high during the image acquisition period. The comparator output is high if $|u_i - u_{i+1}| < \tau$, where τ is a globally specified threshold. The capacitor C_e encodes the edge location.

Figure 1: Components of the edge detection network

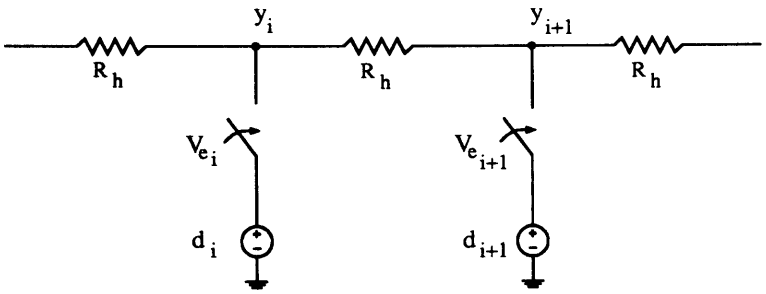


Figure 2: The 1-D reconstruction network. d_i and d_{i+1} are voltage sources proportional to the original brightnesses; y_i, y_{i+1} are the reconstructed brightnesses. Ve_i and Ve_{i+1} control switches connecting the sources to the grid. Each is logically equivalent to the OR of the capacitor states between the node and its neighbors.

condition that the current flowing out of the grid, the normal derivative of the voltage, is zero. It is easy to see that the solution to the reconstruction network is therefore unique and well-defined since there are exactly as many equations as unknown node voltages.

It should be emphasized that several implementational issues are left open in presenting this conceptual picture of the reconstruction network. Clearly, the manner of setting the switches and charging the voltage sources in the reconstruction network is a major design problem whose solution will depend on the application in which the network is used. In this paper, however, we would like to focus on what the network does, rather than on how it should be built, and demonstrate that the results it produces are in fact worth the design effort.

The idea that the image can be reconstructed by solving Laplace's equation on a resistive grid subject to the given boundary conditions is based on the assumption that we can model an image as a collection of piecewise harmonic functions. If this assumption held exactly, only the brightness values bordering edges, where the functions are not required to be harmonic, would need to be specified in order to recover the image completely. A real image is of course always corrupted by noise and will never be exactly harmonic except coincidentally. What we seek to reconstruct is a visually acceptable approximation. For the method to work well, the edges, which determine where the switches are closed in the reconstruction network, must accurately represent locations where the image brightness function deviates significantly from harmonicity. This is another reason for not defining edges as local maxima in the

magnitude of the gradient, since the brightness function may deviate from harmonicity without exhibiting a maximum in its gradient. This happens often at junctions between the projections of different objects in the scene, as well as in many other instances. Marking only the points of maximum gradient would miss these locations, with the result that the network would force an interpolated solution between nodes which should not otherwise interact. The reconstructed image in this case will not be a visually acceptable approximation to the original.

4. Performance issues: Theory and Results

4.1. Effect of the multi-scale veto rule

One way to understand the effect of the veto operation is to consider how it relates to the Fourier spectrum of energies contained in an edge. Since the operations are performed on a discrete network, it is appropriate, and simpler, to use discrete Fourier transforms. Let $x[n]$, with Fourier transform $X(e^{j\omega})$, denote a one-dimensional sequence of sampled brightnesses which has an abrupt change in value between $n = 0$ and $n = -1$. We will assume that the dimension of the network is $\gg 1$ so that we can approximate frequency, ω, by a continuous variable from $[-\pi, \pi]$. Let $y[n] = x[n] - x[n-1]$ denote the difference sequence, and $h_k[n]$ denote the convolution kernel of a lowpass filter $H_k(e^{j\omega})$ of support size k. From [19] the value of $y[0]$ is equal to

$$y[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} (1 - e^{-j\omega}) X(e^{j\omega}) d\omega \tag{5}$$

and the value of the smoothed difference $y_k[n]$ at $n = 0$ is

$$y_k[0] \equiv h_k[0] * y[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_k(e^{j\omega})(1 - e^{-j\omega}) X(e^{j\omega}) d\omega \tag{6}$$

Equations (5) and (6) are valid, even though we are working with two-dimensional images, since we are taking differences in only one direction. We can integrate the 2-D Fourier transform over the orthogonal frequency and redefine variables accordingly.

We are interested in determining under what conditions an edge will be detected at $y[0]$, given the input sequence $x[n]$ and the smoothing filter $h_k[n]$, when the veto rule is applied to the difference sequences $y[n]$ and $y_k[n]$. We will examine two special cases: one where the input is a step, and one where it is an impulse of the same height as the step. These cases correspond to ideal 1-D profiles of a step edge and of an isolated noise spike. We want to show how the multi-scale veto rule can discriminate between these cases by marking the step edge at the point where the input changes abruptly and rejecting the impulse as noise.

Let τ_0 be the threshold used for the unsmoothed differences $y[n]$, and let τ_k be the threshold used for the smoothed difference sequence $y_k[n]$. Suppose $x[n] = Au[n]$ where A is a positive constant and $u[n]$ is the unit step. Then $y[n] = A\delta[n]$, where $\delta[n]$ is the unit impulse; $y[0] = A$ and $y_k[0] = Ah_k[0]$. If $\tau_0 < A$ and $\tau_k < Ah_k[0]$, the edge will be marked at $n = 0$. At other values of $n \neq 0$, $y_k[n] = Ah_k[n]$, which is not 0 in general. It is even possible that for some n, $|Ah_k[n]| > \tau_k$, but since $y[n] = 0$ for all $n \neq 0$, the unsmoothed differences will veto the marking of an edge everywhere except at $n = 0$. Clearly, this is the desired result.

Now suppose that $x[n] = A\delta[n]$ and $y[n] = A(\delta[n] - \delta[n-1])$ so that $y[0] = A$ and $y_k[0] = A(h_k[0] - h_k[1])$. The difference at $n = 0$ will pass the threshold test for the unsmoothed differences if $\tau_0 < A$, but will only be marked as an edge if

$$A > \frac{\tau_k}{h_k[0] - h_k[1]} \tag{7}$$

For a discrete smoothing filter, $1 > h_k[0] - h_k[1] \geq 0$ always, and the value of $h_k[0] - h_k[1]$ will be smaller as k gets larger. Hence, more contrast is needed to mark an impulse than a step. This also is a desired result.

For more general inputs, equations (5) and (6) can be interpreted as meaning that an edge will be marked by the multi-scale veto rule if and only if the total energy within the passbands of each of the applied filters is significant. Isolated impulse noise, whose difference signal does not have significant energy in the low frequency end of the spectrum, can be easily removed. If, instead of an impulse, the input signal is an extended pulse—as would be the case for the ideal 1-D profile of a line—the amount of contrast needed to mark the rising and falling edges of the pulse will also depend on the the scale and threshold of the largest filter, but it will rapidly decrease as the width of the pulse increases. We use this fact, as discussed below, to adjust

the selectivity of the edge detection network for small scale features.

It is important to note that while the scale and threshold of the largest filter determines the effectiveness with which noise and small features are removed, the smallest filter determines the accuracy with which edges are localized because it determines the extent over which a change in brightness will be smeared by smoothing. Beyond this extent, the small scale differences will be insignificant and will veto any differences at larger scales.

4.2. Choosing thresholds and scales

It might seem that the number of free parameters—the different thresholds and scale sizes—that need to be specified in order to apply the multi-scale veto rule would make the method impractical or even arbitrary. However, there are simple ways to choose thresholds and scales based on the types of features which one wants to retain. From the resistive grid and diffusion equations, (2) and (3), it can be seen that the impulse response functions of the smoothing filters which can be implemented on the network are approximately decaying exponentials or Bessel functions. For certain values of R_v and R_h these can be well approximated by even-ordered binomial filters. The 1-D binomial filter of order k is given by

$$b_k[n] = \frac{1}{2^k} \sum_{i=0}^{k} \binom{k}{i} \delta\left[i - \frac{k}{2} - n\right] \tag{8}$$

For the sake of simplicity we will use $b_k[n]$, with k even, to approximate the impulse response of the grid since the coefficients of the binomial filter are easily computed. Suppose we only want to retain step edges and remove thin lines or ridges. This can be arranged using only two scales: unsmoothed, $k = 0$, so that the step will be well-localized, and a second scale with large k so that lines will be strongly attenuated.

As a specific example, suppose $\tau_0 = 10$ and $k = 16$. A step of height 10 and extent $\gg 16$ will, after smoothing, have a height of $10 \times b_{16}[0] = 1.964$. Let this be the value of τ_{16}. From (7), a 1-pixel line (an impulse) will pass the veto only if it has magnitude

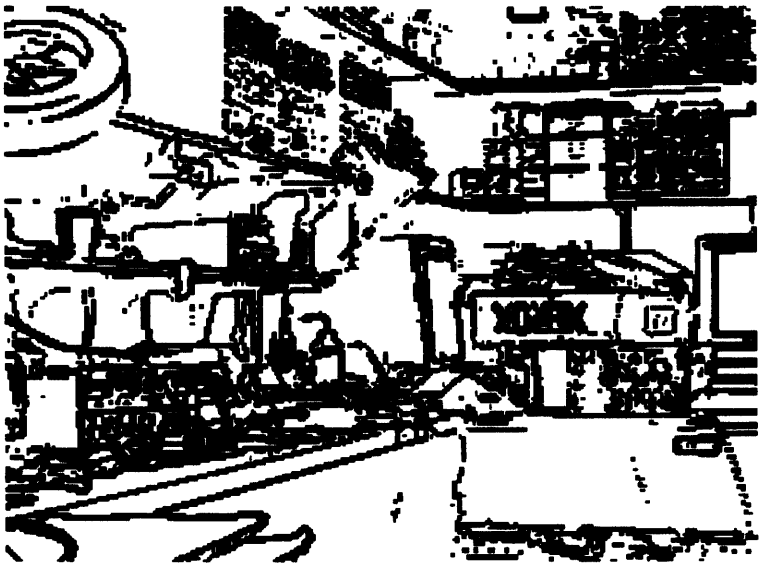
Figure 3: Lab scene—original image.

$$A > \frac{1.964}{b_{16}[0] - b_{16}[1]} = 90. \tag{9}$$

For wider lines, it can easily be checked that a 2-pixel line, $x[n] = A(u[n]-u[n-2])$ will need $A > 26$. A 3-pixel line, $x[n] = A(u[n] - u[n-3])$ will need $A > 15$, and so on. We can increase the selectivity of the veto operation by increasing τ_0 or k. Conversely, we can make the veto less selective for narrow lines by decreasing τ_{16}. For instance, if $\tau_{16} = 1.4$ a 1-pixel line would still need a large magnitude (> 64) to pass, but a 2-pixel line would pass with $A = 19$.

4.3. Simulation results

Simulated results of the edge detection and reconstruction networks are shown for two test images in Figures 3–10. The first set of results is for the 240×320 picture of a cluttered lab shown in Figure 3. The second set is for the 256×256 picture of David shown in Figure 7. Brightness values in the images are quantized from 0–255. In these simulations we approximated the smoothing function of the edge detection



(a) Binary edge map

(b) Reconstructed image

Figure 4: Binary edge map and reconstruction of lab scene. Thresholds and scale set for attenuating thin lines, $\tau_0 = 20$, $k = 14$. Number of data points = 30156 (39% of image). RMS difference between original and reconstruction = 10.1 gray levels.

(a) Binary edge map

(b) Reconstructed image

Figure 5: Binary edge map and reconstruction of lab scene. Smaller second scale used to preserve some thin lines, $\tau_0 = 20$, $k = 10$, Number of data points = 32700 (42.6% of image). RMS difference between original and reconstruction = 7.5 gray levels.

Figure 6: Top: original image. Middle: reconstruction with $k = 14$. Bottom: reconstruction with $k = 10$.

network by even ordered binomial filters since these are good approximations to the network point spread function and are easy to generate. We will use the notation b_k to refer to the 2-D filter generated by the convolution of a horizontally- and a vertically-oriented 1-D filter of order k as given by (8).

In the first test we show the effect of changing the scales on detecting small features such as thin lines. Figures 4(a) and (b) are respectively the binary edge map and reconstruction from using a relatively high threshold, $\tau_0 = 20$, and a large second scale, $k = 14$. The dark points in the binary map indicate where switches are closed in the reconstruction network. They are the locations of image pixels which are adjacent to an edge and thus always occur in pairs. The image contains a large amount of detail, resulting in many edges being marked. Notice, however, some of the smaller scale features such as some of the cables hanging from the scope and the workbench. Those with relatively low contrast are not picked up by the edge detector, and hence, except for a few points which hint at their existence, do not show up in the reconstructed image. In the second test, Figures 5(a) and (b), the same threshold τ_0 was used for the unsmoothed data, but a smaller filter, $k = 10$, was used as the second scale. In Figure 6 the two reconstructed images are shown together with the original in order to facilitate comparison. Note how some, though not all, of the cables reappear in the reconstructed image.

In the lab scene there is a lot of clutter, but most of the objects in the image—boxes, tables, workstations—are close to having planar or approximately harmonic surfaces. It is not too surprising that the reconstructed images are very similar to the original. An example of a different type of image is Figure 7 which has little clutter and only one major object in the scene, namely a face, which is a very non-planar surface. It is interesting to examine how such an image can be reconstructed from piecewise harmonic functions and, more importantly, how many data points are needed to give a recognizable result. In generating the images in Figures 8–10 the same scales, $k = 0$ and $k = 10$ were used, while the threshold τ_0 was varied. In the face, most of the information on shape is contained in the variation of the brightness gradient. By changing τ_0, we change the number of edges which are marked, and therefore change the amount of variation in the brightness gradient of the reconstructed image.

The results are shown in Figures 8–10 where thresholds τ_0 of 9, 12, and 15 were used. The three reconstructed images are shown together with the original in Figure 11. As τ_0 increases fewer edges are marked and the reconstructed image appears

Figure 7: David—original image.

correspondingly flatter. Even in the last example, however, with only 12% of the original brightness values used for interpolation, the face is still recognizable. Figure 10 could be an acceptable reconstruction if we are willing to trade the loss in apparent facial shape with the savings in the number of data points that need to be specified.

Although we have only demonstrated it here for the face image, it is true in general that, even though the subjective visual quality of the reconstructed image degrades as τ_0 increases, the result remains recognizable over a wide range of thresholds. For the lab scene, which contains more contrast than the face, the range is larger, but the same phenomenon is observed. This is an important practical observation since it implies that the choice of a specific threshold value is not crucial to the outcome.

(a) Binary edge map

(b) Reconstructed image

Figure 8: Binary edge map and reconstruction of David with low threshold, $\tau_0 = 9$, to pick up more detail. Number of data points = 13368 (20.4% of image). RMS difference between original and reconstruction = 7.1 gray levels.

(a) Binary edge map

(b) Reconstructed image

Figure 9: Binary edge map and reconstruction of David with intermediate threshold, $\tau_0 = 12$, to eliminate some edges. Number of data points = 10190 (15.5% of image). RMS difference between original and reconstruction = 9.1 gray levels.

(a) Binary edge map

(b) Reconstructed image

Figure 10: Binary edge map and reconstruction of David with $\tau_0 = 15$. The high threshold eliminates much of the detail on face. Number of data points = 8031 (12% of image). RMS difference between original and reconstruction = 11.5 gray levels.

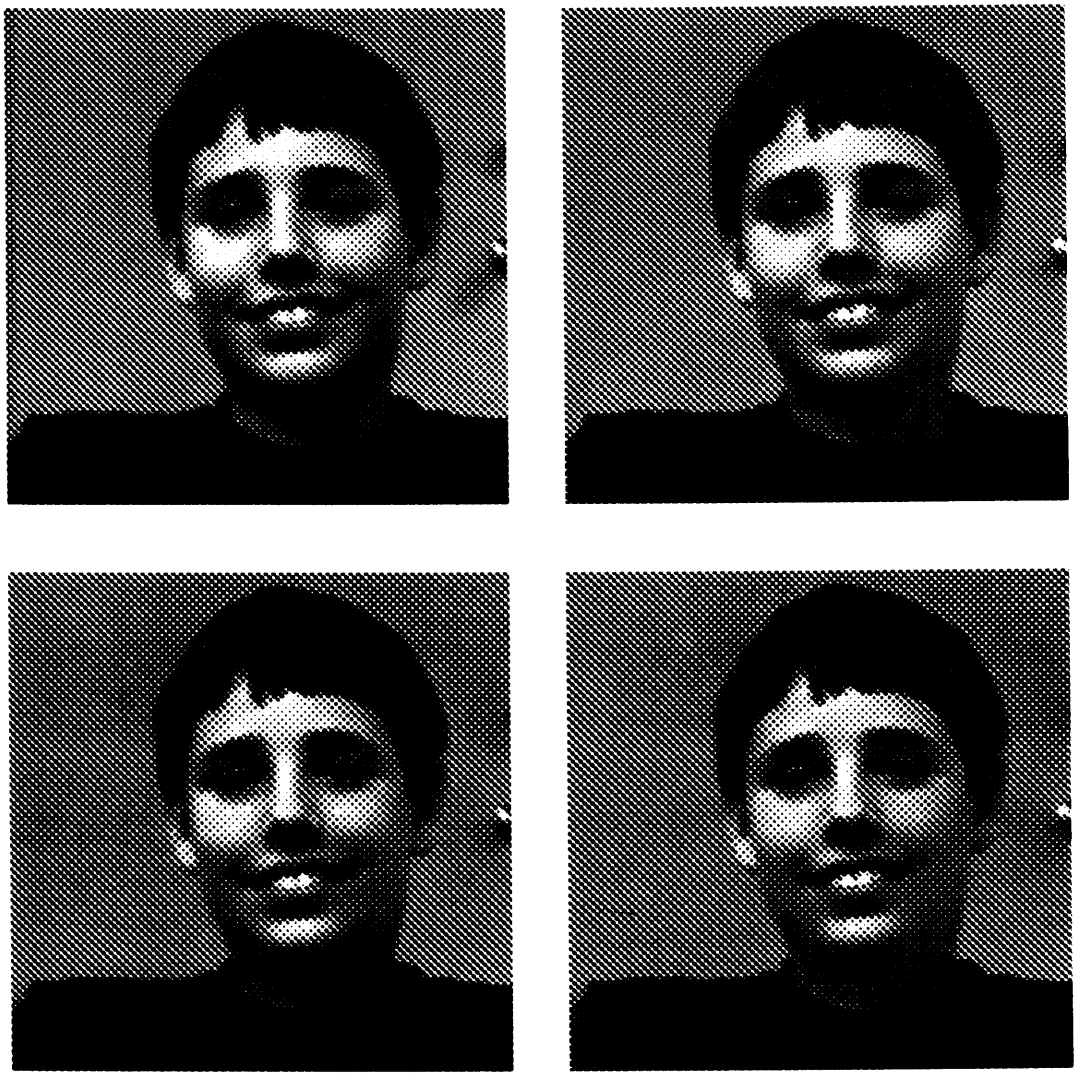


Figure 11: Top left: original image. Top right: reconstruction with $\tau_0 = 9$. Bottom left to right: reconstructions with $\tau_0 = 12$ and $\tau_0 = 15$.

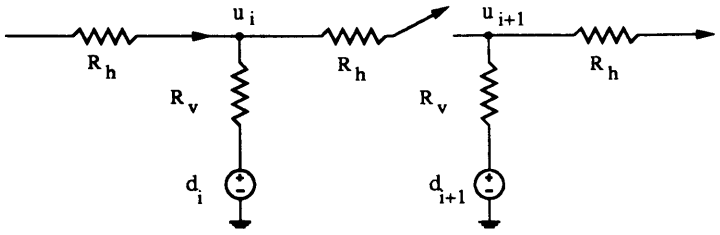


Figure 12: Resistive fuse network for solving discrete variational problem of equation (11). Horizontal elements behave as linear resistors for small voltages across their terminals, but are open-circuits if the voltage difference is too large.

5. Comparison of the MSV Model to the Weak Membrane

Like the weak membrane, and other variational models, the MSV model segments an image into a set of piecewise smooth functions by determining the points in the image where the brightness function departs significantly from smoothness. Clearly, it is desirable to find the minimum number of such points which will result in a good approximation of the image. The weak membrane model formulates these goals as an optimization problem whose solution yields both the points on the discontinuity set and the piecewise smooth functions which approximate the image.

The weak membrane has some problems associated with its formulation, however, which the MSV model is able to avoid. One is that the energy function, equation (1) which is repeated below

$$E(u,\Gamma) = \mu^2 \int\int_R (u-d)^2 dxdy + \int\int_{R-\Gamma} |\nabla u|^2 dxdy + \nu|\Gamma| \qquad (10)$$

is non-convex and cannot be solved by gradient descent methods. This problem, which is well explained by Blake and Zisserman in [3], is intrinsic and arises because of the penalty which must be paid for creating a discontinuity before the system can reach a lower energy state. This problem does not occur in the MSV model because there is no feedback between the reconstruction and edge detection networks.

Equation (10) can be discretized and modeled by a resistive network. In one-dimension the discrete equation is

$$J(u,l) = \frac{1}{R_v}\sum_{i=1}^{N}(u_i - d_i)^2 + \frac{1}{R_h}\sum_{i=1}^{N-1}(u_i - u_{i+1})^2(1-l_i) + \nu\sum_{i=1}^{N-1} l_{i,j} \qquad (11)$$

where the {0–1}-valued variables, l_i, model the discontinuity set, Γ, of equation (10). The equivalent circuit for (11) is shown in Figure 12 [16]. The horizontal elements in this network are resistive fuses, which break if the voltage across their terminals rises above a critical value, but otherwise behave as linear resistors. Several implementations of the 2-D version of the network in Figure 12, which differ principally in their design of the resistive fuse elements, have been built in VLSI [8,14,24]. Circuit implementations of the weak membrane cannot escape the non-convexity problem, however, and some effort is required to nudge them to the optimal solution [16].

A second problem with the weak membrane is that the optimal piecewise smooth functions u are determined from all of the data and not just the values adjacent to a discontinuity. They are also strongly determined by the scale parameter μ. The resistive fuse network of Figure 12 and the multi-scale veto edge detection network of Figure 1 appear similar, because both perform smoothing by a resistive grid. Both the MSV model and the weak membrane reconstruct an brightness function from the data, but with different boundary conditions. Returning to the continuous formulation, if Γ is given in (10) then the functions $u(x,y)$ which minimize E satisfy the Euler equation

$$u - \frac{1}{\mu^2}\nabla^2 u = d \qquad (12)$$

subject to the condition

$$n \cdot \nabla u = 0 \text{ on } \Gamma \qquad (13)$$

which, comparing (12) with the resistive grid equation (3), is the same as saying that u is a smoothed version of d. In the MSV network we generate the piecewise smooth reconstruction of the data by solving Laplace's equation subject to the boundary condition $u = d$ on Γ. This is equivalent to minimizing only the second term in (10), or setting $\mu = 0$.

Both methods can therefore be viewed as alternative ways of regularizing the

brightness data with interpolating splines. The solution obtained by solving (12) and (13), however, must tradeoff how effectively noise can be removed by smoothing against how natural the resulting image will be. This problem can be understood by considering the limiting cases: $\mu \to 0$ and $\mu \to \infty$.

As $\mu \to 0$ equation (12) becomes

$$\nabla^2 u \approx 0 \tag{14}$$

The solution in this case is approximately harmonic, but due to the boundary condition (13), it must approach a constant, since that is the only harmonic function which has zero normal derivative everywhere on its boundary. In this case noise within the regions between the discontinuities will be completely smoothed away, but the resulting image will be a collection of patches of constant brightness and will appear very cartoon-like.

At the other extreme, $\mu \to \infty$, we have

$$u - d \approx 0 \tag{15}$$

In this case, the output will appear more natural because it is approximately the same as the input, but there is also very little smoothing.

The images reconstructed in the MSV model look more natural and are very similar visually to the originals because there are fewer constraints to satisfy. The functions only have to match the data where it is given and satisfy Laplace's equation everywhere else. Furthermore, noise can be more effectively removed since any feature which does not generate an edge is erased entirely from the reconstructed image and not just smoothed into the background.

It should be noted that the weak membrane does have some features which are not shared by the MSV model, for instance the hysteresis property, which gives an existing edge the tendency to extend itself, just as a tear does in a real membrane. Also, the weak membrane model can be formulated as a well-defined minimization problem, so that one *can* speak of an optimal solution. We do not know of a way to formulate the problem that the MSV model attempts to solve, namely finding the minimal discontinuity set bounding piecewise harmonic functions which are good

approximations, in some sense, to the original image, as a variational problem. The method used in the MSV model for finding edges is a heuristic, and is based on the idea that the magnitude of the gradient for a harmonic surface which extends over any significant area can be bounded over most of its extent by a small number, such as the threshold used in the tests. This is seen from the fact that the functions f which minimize

$$\int\int_D |\nabla f|^2 dxdy \tag{16}$$

over some domain D are solutions to $\nabla^2 f = 0$, within the domain [12]. By marking the points where the change in brightness is above some threshold and is significant over a range of spatial scales, we determine the locations where the underlying brightness function is most likely to depart from harmonicity, and where interpolation from neighboring values is least likely to be a good approximation to the data. In terms of finding the minimal discontinuity set, it is easy to show that this heuristic is not optimal. For instance, a steeply inclined plane will give rise to a discontinuity at every point on its slope, even though a plane is a harmonic function for which it would suffice to specify its boundary points. In practice, however, such features cannot occur very often because the spatial extent of a steeply sloped surface is limited by the dynamic range of the image. It can only rise for a few pixels before it has to level off. The philosophy of the MSV method is that it is better to accept a less than optimal heuristic than to complicate the circuit design to deal with these cases.

6. Summary and Discussion

We have presented a model of a two-stage analog network for edge detection and image reconstruction. Edges are detected in the first stage using the multi-scale veto rule, which states that an edge is significant only if it passes a threshold test at each of a set of different spatial scales. The image is reconstructed in the second stage from the brightness values adjacent to the edges. The two-stage design offers several advantages for both performance and applications. Because there is no feedback between the stages, there is also no problem of stability or local stationarity. Also since the networks are physically distinct, they do not have to be physically close to operate properly. This increases the flexibility with which the system may be

designed, as well as the types of applications for which it may be used.

The multi-scale veto rule allows edges to be localized at the resolution of the smallest spatial scale without having to identify maxima in brightness gradients, so that second differences do not need to be computed. At the same time noise is removed with the efficiency of the largest scale used. The computations can be performed on a single network with relatively little circuitry per pixel. The simplicity of the circuit is an important feature of the model since it directly impacts on the size of the image arrays with which it can work.

Images are reconstructed in the second stage from the brightness values adjacent to edges. The reconstructed images are very similar visually to the originals and could serve, for some applications, as acceptable replacements. Since the number of data points which need to be specified for the reconstruction network ranges typically from 15–45% of the number of pixels in the original image, depending on the amount of detail in the scene, and since the edge detection and reconstruction networks are physically distinct, this method offers possibilities for data compression. Combined with existing methods such as run-length and Huffman coding, the total savings in bandwidth may be significant.

This paper has presented the theory behind the MSV model, which is a piece of ongoing research. Work is currently in progress on the design and fabrication of circuits for the edge detection and reconstruction networks; the design of larger systems for solving early vision tasks that incorporate the edge detection network; and on the theoretical issues concerned with applying the model to image compression.

Acknowledgements

I am grateful to Eric Grimson, John Harris, Berthold Horn, and John Wyatt for having read an initial draft of this paper and for their helpful comments and suggestions.

References

[1] J. Batali. *A Vision Chip.* A.I. Memo 869, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, May 1981.

[2] A. Blake. Boundary conditions for lightness computation in mondrian world. *Computer Vision, Graphics, and Image Processing*, 32:314–327, 1985.

[3] A. Blake and A. Zisserman. *Visual Reconstruction.* MIT Press, Cambridge, MA, 1987.

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

[5] S. R. Curtis and A. V. Oppenheim. Reconstruction of multidimensional signals from zero crossings. *Journal of the Optical Society of America*, 4(1):221–231, 1987.

[6] S. Geman and D. Geman. Stochastic relaxation, gibbs distribution, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[7] R. C. Gonzalez and P. Wintz. *Digital Image Processing.* Addison-Wesley, Reading, MA, 1987.

[8] J. Harris, C. Koch, J. Luo, and J. L. Wyatt, Jr. Resistive fuses: analog hardware for detecting discontinuities in early vision. In C. Mead and M. Ismail, editors, *Analog VLSI Implementation of Neural Systems*, pages 27–55, Kluwer Academic Publishers, 1989.

[9] B. K. P. Horn. *The Binford-Horn LINE-FINDER.* A.I. Memo 285, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, July 1971.

[10] B. K. P. Horn. Determining lightness from an image. *Computer Graphics and Image Processing*, 3:277–299, 1974.

[11] T. F. Knight Jr. *Design of an Integrated Optical Sensor with On-Chip Prepocessing.* PhD thesis, Massachusetts Institute of Technology, 1983.

[12] G. A. Korn and T. M. Korn. *Mathematical Handbook for Scientists and Engineers.* McGraw-Hill, 1968.

[13] E. H. Land and J. J. McCann. Lightness and retinex theory. *Journal of the Optical Society of America*, 61(1):1–11, 1971.

[14] H-S. Lee and P. Yu. Cmos resistive fuse circuits. In *Digest of Technical Papers*, pages 109–110, 1991 Symposium on VLSI Circuits, Oiso, Japan, 1991.