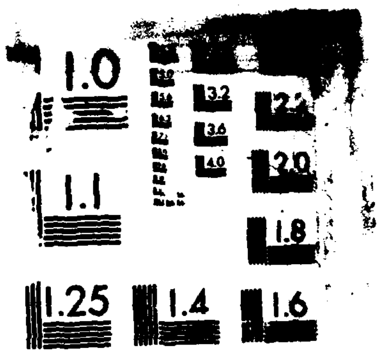
AD-A179 945 OPTIMAL LIKELIHOOD GENERATORS FOR EDGE DETECTION UNDER GAUSSIAN ADDITIUE NOISE(U) ROCHESTER UNIU NY DEPT OF COMPUTER SCIENCE D SHER AUG 86 TR-185 DACA76-85-C-0001 1/1

UNCLASSIFIED NL

END


1.0
1.1
1.25
1.4
1.6
1.8
2.0
3.2
3.6
4.0

Optimal Likelihood Generators for Edge Detection under Gaussian Additive Noise

David Sher
Computer Science Department
The University of Rochester
Rochester, New York 14627

TR 185

August 1986

Department of Computer Science
University of Rochester
Rochester, New York 14627

(2)

Optimal Likelihood Generators for Edge Detection under Gaussian Additive Noise

David Sher
Computer Science Department
The University of Rochester
Rochester, New York 14627

TR 185

August 1986

A technique is presented for determining the probability of an edge at a point in an image. The image is modeled as an ideal image that is convolved with a linear blurring function and also with uncorrelated Gaussian additive noise. The ideal image is modeled by a set of templates for local neighborhoods. Every neighborhood in the ideal image is assumed to fit one of the templates with high probability. A computationally feasible scheme to compute the probability of edges is given. The output of several of the likelihood generators based on this model can be combined to form a more robust likelihood generator using the results described in *Developing and Analyzing Boundary Detection Operators Using Probabilistic Models* presented in the first Workshop on Probability and Uncertainty in Artificial Intelligence by the author [13].

This work would have been impossible without the advice and encouragement of Chris Brown my thesis advisor. This work was supported in part by the Defense Advanced Research Projects Agency U. S. Army Engineering Topographics Lab. grant number DACA76-85-C-0001 and The National Science Foundation, grant number DCR8320136..

$NSF

A179945

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: TR-185	2. GOVT ACCESSION NO.
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Optimal Likeilhood Generators for Edge Detection under Gaussian Additive Noise	5. TYPE OF REPORT & PERIOD COVERED: Technical Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): David Sher	8. CONTRACT OR GRANT NUMBER(s): DACA76-85-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Computer Science Department, The Unviersity of Rochester, Rochester, New York 14627	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: DARPA/1400 Wilson Blvd. Arlington, VA 22209	12. REPORT DATE
	13. NUMBER OF PAGES: 10
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Office of Naval Research, Information Systems, Arlington, VA 22217	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this document is unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

from 20 → edge detection, template, likelihood, Bayesian reasoning.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A technique is presented for determining the probability of an edge at a point in an image that is convolve with a linear blurring function and also with uncorrelated Gaussian additive noise. The ideal image is modeled by a set of templates for local neighborhoods. Every neighborhood in the ideal image is assumed to fit one of the templates with hight probability. A computaionally feasible scheme to compute the probability of edges is given. The output of several of the likelihood generators based on this model can be →

next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (Continued)

Combined to form a more robust likelihood generator, using the results decribed in Developing and Analyzing Boundary Detection Operators Using Probabilistic Models presented in the first Workshop in Probability and Uncertainty in Artificial Intelligence by the author(13).

Keywords:

1. Edge Detection: The Problem and Previous Approaches

The major problem of low-level vision is that images are ambiguous: two different scenes can result in the same image. The major source of ambiguity that I am concerned with is noise. Noise is generally the result of imperfections of the sensors used to produce the image. Because of noise the same scene can result in any observed image whatsoever. It is much more likely however to result in some images than others. My work is about techniques for combating noise and the resulting ambiguity and thus is applicable to vision tasks where noise presents a significant problem.

My approach to low level vision is unusual for such research. Consider the problem of segmentation, in particular, consider the problem of finding regions of uniform reflectance. The image is modeled as a set of regions of constant reflectance with occlusion boundaries between them. Most approaches to this problem try to return an answer that is best, in the sense that the probability of the given answer differing from the correct answer in a significant way is minimized. Such an algorithm applies *estimation theory* to the problem of low level vision.

Instead, this paper derives algorithms that attempt to calculate the probability of a boundary passing between two points. In low-level vision usually one can acquire a sufficiently specific model for the probability to be uniquely defined, even through the image is ambiguous. One advantage of this approach is that a variety of different estimates of the segmentation can be derived from these probabilities by simple operations.

This paper concentrates on the problem of deriving the probability of a boundary from a window on the image. Classically this task has been called edge detection. I am using a template based model for this work: It is assumed that if the image was viewed through a noiseless sensor then every window on the image would match one element of a set of templates. Since the image wasn't produced by noiseless sensors its windows look like some template followed by noise according to the model.

Recently two works have been published that take an approach similar to mine. One that is similar is by Art Owen [12] on pixel classification for Landsat images. The operator he derives returns likelihoods for neighborhoods instead of pixels. Owen's work uses a somewhat more sophisticated model to derive his priors (a Poisson model of boundaries). The work has no noise model and does not consider combination rules. Likelihoods are derived by training on test cases. Owen can use training to get his likelihoods because of the small number of categories he uses and because he uses binary (thresholded) images. This reduces the number of cases he had to deal with so the operator can be conveniently trained.

Another work that takes an approach similar to mine is that of Li and Dubes [9] on matching small templates in binary images. They use Neyman-Pierson statistics. Neyman-Pierson statistics are used because there is a well defined null hypothesis (the object is not in the scene). Li and Dubes derive a likelihood ratio test. Such a test has maximal power if it is based on a complete and sufficient statistic. The way they derive the likelihood ratio is to derive likelihood generators. They approximate the likelihoods deriving operators much in the same spirit that I derive mine in section 3.

There has been some work on using Bayesian techniques (techniques using likelihoods and prior probabilities) to estimate edge positions. In particular the work described in [3] and [6] use Bayesian techniques for image reconstruction and [8] uses Bayesian technique for reconstruction and edge detection (as a side effect). These techniques have the weakness that they look for the maximum *a posteriori* likelihood (the *MAP* assumption). The MAP assumption only holds when a small set of answers are the only ones acceptable as correct with 0 loss and all other segmentations have the same loss (1 loss). I believe that a 0-1 loss function is unrealistic for most applications. A 0-1 loss function is realistic if getting a boundary wrong at a single point is as bad as getting it wrong everywhere, because both possibilities result in 1 loss according to the 0-1 loss function. In low-level vision the usefulness of an estimate drops off gradually as errors accumulate. Some good results have been gained using these techniques.

Much work has been done using signal detection theory for deriving operators. However most work based on signal detection theory is limited to operators that compute linear functions on the image. Because of this limitation the operators generated are the optimal linear operators given a figure of merit. In particular the Wiener filter is optimal for reconstructing images given a least squares cost function and a correct noise model and image model. [1].

Canny [5] has developed an operator that is optimal according to a figure of merit that contains detection and localization. He limited himself to linear shift invariant operators. His operators looked a great deal like difference of gaussian operators. He modeled edges as a template and developed a technique for generating an operator for an arbitrary template. I intend to generate optimal detectors under my system with the same models.

Canny [5] Lunscher and Beddoes [10] and Torre and Poggio [15] limit the class of functions that they consider for edge detection to linear shift independent operators. Thus their operators are convolutions. When they indicate that their operators are optimal they mean that they do the best job for functions in the class of linear shift independent operators. The class of functions I use is the class of functions of a window on the image. Such operators are shift independent but they are not necessarily linear. The optimal operator from this class theoretically is the best possible edge detector for a specified window size.

Much of the work done in computer vision has been developed with idiosyncratic objectives. Because of the their objectives differed from mine the algorithms some people developed have serious shortcomings from my viewpoint. One alternate set of objectives is those held by researchers inspired by biological modeling. An excellent work in biological modeling is that of Fleet [7]. His work is on the temporal and spatial characteristics of center-surround operators. Torre and Poggio's work [15] also is of this form.

When working on modeling one tries to develop algorithms whose behavior closely approximates that of a human vision system. An example of such approximation is to have only band limited operators because the cells on the mammalian optic nerve have been shown to be band limited. I only band limit operators if it is shown that the phenomena being detected are band limited or that a band limited operator is sufficient to detect the phenomena without loss of accuracy.

Much work has been done on segmentation without considering optimality or probability. A summary of work on edge detection and relaxation occurs in [4] Recently some good work on edge detection has been done by Canny [5] and Nalwa [11].

2. The Image Model

In the image restoration literature much work has been done on a particular form of noise. The noise introduced by the sensor is modeled by a linear blurring function followed by gaussian additive mean 0 noise [2]. The log image from a photograph has gaussian additive noise in its linear region from the randomness inherent in film grain. Gaussian additive noise occurs in any system whose noise is a result of many small perturbations added together (by the central limit theorem). Blur can result from vibrations in the camera, motion in the scene and the physics of light. I make a standard simplification in that I assume the blur is linear and shift invariant. Blur from vibrations in the camera and the physics of light has this property. Blur from motion in the scene tends to be linear and shift invariant within a rigid object. Thus I model the noise as convolving the image with a blur function and then adding a gaussian additive mean 0 random factor.

I also need a model of an image to derive a likelihood generator. A *likelihood generator* is an intermediate stage in an algorithm that calculates the probability of a boundary at a point. More details on likelihood generators are in the next section.

Here, I derive the optimal likelihood generator that looks at a window in the image. Thus I need only model windows in the ideal image. I model the ideal image as consisting of windows that each match an element of a set in a set of sets of templates. Thus if I can derive the likelihood of the observed window given that its ideal counterpart matches each template in a set and the a priori probability of each template then I can derive the likelihood of the window belonging to the set of templates. As an example consider the set of templates that consist of a uniform intensity (figure 1).

Figure 1: A template of uniform intensity.

100	100	100	100

This set of templates models the interior of a region of uniform intensity. Consider what an occlusion edge between two such regions looks like. Such an event can be modeled by a template of the form in figure 2.

Figure 2: A template of a step edge.

100	100	200	200

This template is often called a step edge in the edge detection literature. I also need to model the event that there is an off center edge in the window. I call this event a *near edge* event. The near edge events are modeled by templates like those of figure 3.

Figure 3: Templates for a near edge.

100	200	200	200

100	100	100	200

So 3 useful sets of templates are templates like those in figure 1, 2, and 3 with all possible intensities substituted for 100 and 200. These templates model all possible configurations of a 1 by 4 window in an ideal image where all regions are at least 3 pixels wide. If I can derive the likelihood of an observed window having a counterpart in each of these sets then I can derive the probability of a boundary in the middle of the window using Bayes' law (see next section).

3. Likelihood Generators

Often it is easier to state and solve the inverse vision problem (which is why computer graphics can generate realistic images that current image understanding systems can't analyze). For low level vision it is easier to describe the probable structure of an observed intensity image in the presence of a boundary than to describe the probability distribution on the boundary given an observed image. In particular the models described in the previous section have this property.

The probability that the observed window's pixels are assigned a set of values a when a feature f takes on value v is the *likelihood* of v for a. I use $L_f(a \mid v)$ as shorthand notation for the likelihood. A *likelihood generator* is an algorithm that uses a model D to estimate the likelihood of v for a. Thus I use $L_f(a \mid v \& D)$ as notation for the output of a likelihood generator. Given a likelihood generator for D and a prior estimate of the distribution of t's values then one can make a feature detector for t using Bayes'

Rule:

$$P_f(v \mid a\&D) \equiv \frac{L_f(a \mid v\&D)prior_f(v)}{\sum_{v \in V} L_f(a \mid v'\&D)prior_f(v')} \tag{1}$$

I call the feature detector thus derived a *Bayesian feature detector* for model D.

The set of likelihoods for a feature f given an observation a contains more information than (1) uses. The denominator in (1)

$$\sum_{v \in V} L_f(a \mid v'\&D)prior_f(v') \tag{2}$$

is the probability that a would occur given the prior estimate of the distribution on f's feature space. If the probability is too low then the model being used probably is not correct. I use this information combined with *a priori* information about the reliability of the model to derive an evidence theory in [14].

4. Likelihoods for a Single Template

The problem I address in this paper is to find the likelihood of an observed window given a template and a model for the noise. Let $O=\left\{o_i\right\}$ represent the window that was observed. Let $T=\left\{t_j\right\}$ represent the template. Then I need $P(O \mid T\&D(\sigma,B))$ where σ is the standard deviation of the gaussian mean 0 additive noise and B represents the blurring function. Assume that B is negligible outside a window of size (w_B,l_B) pixels and the template is of size (w_T,l_T). Then the effect of the blurring function $B \otimes T$ ($\otimes$ is correlation where the template never falls beyond the window's edge, $X \otimes X$ is a single number that is the sum of squares of X's elements) is completely determined in a region of size $(w_T - w_B + 1, l_T - l_B + 1)$ pixels (see figure 4).

Figure 4: Effect of a Blur Function on a Template.

T:

100	100	100	200	200	200

$\otimes$

B:

.25	.5	.25

$=$

$T \otimes B$:

100	125	175	200

I assume in the rest of this paper that the observation window O lies completely within the determined region. So the only remaining probabilistic element is the gaussian additive noise. If I is the identity function then I need to determine $P(O \mid T \otimes B\&D(\sigma,I))$. I refer to the elements of $T \otimes B$ as the set $\left\{t'_k\right\}$.

Since the only noise left in the problem is the uncorrelated gaussian additive noise (since blur has been handled) the likelihood is the product of the likelihoods at each pixel.

$$P(O \mid T \otimes B\&D(\sigma,I)) = \prod_i P(o_i \mid t'_i\&D(\sigma,I)) \tag{3}$$

Since the noise is gaussian the likelihood at a point has this form:

$$P(o_i \mid t'_i \& D(\sigma,I)) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[\left[-\left[o_i - t'_i\right]^2\right] / \left[2\sigma^2\right]\right] \tag{4}$$

Thus the equation for the likelihood of the window can be stated as:

$$P(O \mid T \otimes B \& D(\sigma,I)) = \frac{1}{\left[\sqrt{2\pi}\sigma\right]^n} \exp\left[\left[-\sum_i \left[o_i - t'_i\right]^2\right] / \left[2\sigma^2\right]\right] \tag{5}$$

The likelihood can be restated mathematically as:

$$\frac{1}{\left[\sqrt{2\pi}\sigma\right]^n} \frac{\exp\left[\left[O \otimes (T \otimes B)\right] / \left[\sigma^2\right]\right]}{\mathrm{En}(O,\sigma)\mathrm{En}(T \otimes B,\sigma)} \tag{6}$$

Where $\mathrm{En}(X,\sigma)$ is $e^{(X \otimes X)/(2\sigma^2)}$ which I refer to as the energy of X relative to σ. Note that $\mathrm{En}(O,\sigma)$ is independent of the template while $\mathrm{En}(T \otimes B,\sigma)$ is independent of the observed window. These results mean that $\mathrm{En}(T \otimes B,\sigma)$ can be precomputed while the cost of computing $\mathrm{En}(O,\sigma)$ can be amortized over the entire set of templates.

5. Likelihoods for Sets of Templates

Here, I examine efficiently calculating the likelihood of a set of templates given an observed image. In particular I examine the set of templates whose elements are all linear functions of a characteristic template, T_0. Thus I describe such a set as aT_0+b. I call such a set a *linear set* of templates. The set of step edges with a fixed step point can be described as a linear set. The set of symmetric peak edges are linear functions of a prototypical peak edges hence are a linear set. The linear slopes are linear functions of the function $f(x)=x$ hence are a linear set too.

I limit my blur functions to blurs that leave uniform intensity images unchanged. Then my set of $B \otimes T$ is of the form $aB \otimes T_0+b$. The likelihood of the observed image given a member of a linear set is:

$$\frac{1}{\left[\sqrt{2\pi}\sigma\right]^n} \frac{\exp\left[\left[2(aO \otimes (T_0 \otimes B) + bO \otimes I)\right] / \left[2\sigma^2\right]\right]}{\left[\mathrm{En}(O,\sigma)\right]\left[\mathrm{En}(aT_0 \otimes B + b,\sigma)\right]} \tag{7}$$

The triplet $(\mathrm{En}(O,\sigma), O \otimes T_0 \otimes B, O \otimes I)$ is sufficient for determining the likelihood of this set of templates. The class of templates is indexed by a and b. To find the likelihood I need *a priori* probabilities for the different templates. I describe these probabilities with $P_{T_0}(a,b)$.

The likelihood of a linear set is:

$$\frac{1}{\left[\sqrt{2\pi}\sigma\right]^n \mathrm{En}(O,\sigma)} \sum_{a,b} \frac{\exp\left[\left[(aO \otimes (T_0 \otimes B) + bO \otimes I)\right] / \left[\sigma^2\right]\right]}{\mathrm{En}(aT_0 \otimes B + b,\sigma)} P_{T_0}(a,b) \tag{8}$$

Let F_{T_0} be defined in equation (9).

$$F_{T_0}(C,S) = \sum_{a,b} \frac{\exp\left[\left[(aC+bS)\right]/\left[\sigma^2\right]\right]}{\mathrm{En}(aT_0 \otimes B + b,\sigma)} P_{T_0}(a,b) \tag{9}$$

Then equation (7) can be rewritten as equation (10).

$$\frac{1}{\left[\sqrt{2\pi}\sigma\right]^n \mathrm{En}(O,\sigma)} F_{T_0}(O \otimes (T_0 \otimes B), O \otimes I) \tag{10}$$

This implies an algorithm for deriving the likelihood of a linear set of templates.

```
Let V be the variance of the noise σ²
Let K be [√2πσ]ⁿ
For each window W in the image do
      {
S:    Let S be the sum of the pixels in W
SS:   Let SS be the sum of the squared pixels in W
C:    Let C be the correlation of W with the T₀⊗B
F:    Let F be F_T₀(C,S)
E:    Let E be exp( SS / (2*V) )
O:    Output F / ( K * E )
      }
```

If there are N pixels in the image steps S and SS require $O(N)$ operations counting adds and multiplies. Step C requires $O(N \log N)$ operations. Steps E and O are also $O(N)$ operations steps. Thus the algorithm requires $O(N \log N)$ operations plus whatever is required to execute step F. I propose to calculate F_{T_0} by table-lookup on the values of S and C. Thus step F is just a table-lookup.

The size of the table that holds F is the product of the number of possible values of C and S. Both of these can be calculated given the number of gray-levels in the image, G, and the number of pixels in the window, n, and $T_0 \otimes B$. The number of possible values for S is nG and the number of values that C can be is $G(T_0 \otimes B)$. Thus the number of elements in the table is $n(T_0 \otimes B)G^2$.

For a central step edge with a 1 by 8 window $n=8$ and $T_0 \otimes B=4$. Thus the size of the table is $32G^2$. Table 1 is a table of G values and resulting table sizes.

Table 1: Table Sizes to calculate F_{T_0}

G	Number of Table Entries	Storage for Table in bytes (in double precision)
4	512	4K
16	8192	64K
64	131072	1M
256	2097152	16M

The more gray-levels the more difficult it becomes to store the table. It also becomes more work to calculate the entire table. Thus to handle 64 or more gray-levels I suggest that a smaller table be used with interpolation. If there are symmetries in F_{T_0} a smaller table is sufficient to store the function. As an example if $F_{T_0}(S,C) = F_{T_0}(S+16,C)$ then only F_{T_0} need only be calculated for S between 1 and 16. At this moment no such symmetries have been discovered.

6. Detecting 1-D Step Edges Optimally

For the model of regions of uniform intensity with step edges between them I need only calculate the likelihoods for two linear sets of templates. One template is the uniform intensity template. The likelihood of this template can be calculated from the standard deviation of the observed window. The other is the step edge template with the step in the middle. If I have a prior on the probability of a boundary then I have the tools necessary to build an optimal edge detector for my model.

The near edge templates can be approximated by the likelihoods calculated at the neighboring (overlapping) windows for the central step edge linear set. Since I am deriving a 1 dimensional edge detector, the likelihood of an edge in the center of an overlapping window is the likelihood of an edge directly to the right or the left of the center of the window. In the step edge model all regions are at least $w/2$ pixels wide given a template width of w. Thus the near edge events are exclusive of the central edge events.

I assume a cost function that simply counts the number of points mislabeled as boundaries or nonboundaries when the opposite is the case. The prior probability of a central edge and any near edge event is equal under models that do not have a prefered position for objects. Thus if the likelihood of a central edge is not maximal among all the overlapping windows then the optimal estimate does not have an edge at this point. Only local maxima among the likelihood of step edge function are reported. Thus multiple reporting of an edge is precluded. Also only edges that satisfy the inequality (11) are reported:

$$P(O \mid E)P_E > P(O \mid U)P_U \tag{11}$$

where E represents the event that there is an edge in the center of the window and P_E is the prior probability of that event while U represents the event that there is no edge anywhere in the window and P_U is the prior probability of that event.

I can also use my work on evidence combination to combine likelihood generators that make different assumptions about the noise and blur. Many of the operations I use to evaluate the likelihood of a linear set of templates under one kind of noise can be used for many different kinds of noise. As an example $En(O,\sigma)$ is used by all likelihood generators based on linear sets of templates. Also all templates that have the same value for $En(T_0 \otimes B, \sigma)$ and $(T_0 \otimes B) \otimes I$ and have the same values for P_{T_0} can share the same table to calculate F_{T_0} since it depends only on these parameters. Thus if all the differently oriented edge templates have the same sum of pixel values and the same sum of squares of pixel values they can share the same table for F_{T_0}.

7. Conclusions

In this paper I demonstrated an algorithm for edge detection that is mathematically optimal for a popular model. Since F_{T_0} is increasing in $O \otimes (T_0 \otimes B)$ this algorithm thresholds using a function of the sum of the pixels in the window and the sum of the squares of the pixels in the window. The algorithm only reports an edge if there are no nearby edges with greater likelihood. That test is similar to edge thinning in standard work. Thus the algorithm is similar to algorithms that run a thresholded convolution and then thin. Currently this algorithm is being implemented and experimental results will soon be forthcoming.

References

[1] H. C. Andrews and B. R. Hunt, *Digital Image Restoration*, 126-147 , Prentice-Hall, INC., Englewood Cliffs, New Jersey 07632, 1977
.

[2] H. C. Andrews and B. R. Hunt, *Digital Image Restoration*, 8-26 , Prentice-Hall, INC., Englewood Cliffs, New Jersey 07632, 1977
.

[3] H. C. Andrews and B. R. Hunt, *Digital Image Restoration*, 187-211 , Prentice-Hall, INC., Englewood Cliffs, New Jersey 07632, 1977
.

[4] D. H. Ballard and C. M. Brown, in *Computer Vision*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1982, 14,63,85-86.

[5] J. F. Canny, Finding Edges and Lines in Images, 720, MIT Artificial Intelligence Laboratory, June 1983.

[6] H. Derin, H. Elliott, R. Cristi and D. Geman, Bayses Smoothing Algorithms for Segmentation of Binary Images Modeled by Markov Random Fields, *PAMI* 6,6 (Novermber 1984), 707-720, IEEE.

[7] D. J. Fleet, The Early Processing of Spatio - Temporal Visual Information, 84-7, University of Toronto, Research in Biological and COmputational Vision, September 1984.

[8] S. Geman and D. Geman, Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images, *PAMI* 6,6 (Novermber 1984), 721-741, IEEE.

[9] X. Li and R. C. Dubes, The FIrst Stage in Two-Stage Template Matching, *Pattern Analysis and Machine Intelligence* 7,6 (Nov 1985), 700-707, IEEE.

[10] W. H. H. J. Lunscher and M. P. Beddoes, Optimal Edge Detector Design I: Parameter Selection and Noise Effects, *Pattern Analysis and Machine Intelligence* 8,2 (March 1986), 164-177, IEEE.

[11] V. S. Nalwa, On Detecting Edges, *Proceedings: Image Understanding Workshop*, October 1984, 157-164.

[12] A. Owen, A neighbourhood-based classifier for LANDSAT data, *The Canadian Journal of Statistics 12*,3 (September 1984), 191-200, Statistical Society of Canada.

[13] D. Sher, Developing and Analyzing Boundary Detection Operators Using Probabilistic Models, *Workshop on Probability and Uncertainty in Artificial Intelligence*, August 1985.

[14] D. B. Sher, Evidence Combination for Vision using Likelihood Generators, *Proceedings: Image Understanding Workshop (DARPA)*, December 1985, 255-270. Sponsored by: Information Processing Techniques Office Defence Advanced Research Projects Agency.

[15] V. Torre and T. A. Poggio. On Edge Detection, *Pattern Analysis and MAchine Intelligence* 8,2 (March 1986), 147-163. IEEE.