Chapter 3

Isolation of intensity discontinuities.

David Vernon and Giulio Sandini

There is but one art, to omit

Robert Louis Stephenson

3.1 An Overview of Edge Detection Techniques.

We will begin our presentation of the visual capabilities of *VIS ā VIS* with what is, perhaps, one of the most fundamental of all processes: edge detection. Edge detection techniques belong to a generic class of image processing and analysis operations, a class normally referred to as segmentation, which effect the isolation of specific image regions corresponding, unambiguously, to given objects. Such object isolation is often effected quite simply by edge detection alone but this is the case only if the scene is extremely simple. For more realistic natural scenes, the segmentation process requires more sophisticated grouping techniques, such as those discussed in chapter 7.

Edge detection itself is a complex procedure for it requires not only the isolation of the image features we call edges, but it is also concerned with the inference of the physical cause of those features. We will not concern ourselves here with the latter problem and we will confine our attention to the detection of edge features. First, we should attempt to define exactly what is an edge. There are many definitions but we will adopt the following: an edge exists at a given point in an image if some feature,

normally the image intensity or grey-level, changes its val function of image position. That is, we will characterize ed intensity. As might be expected when dealing with a proce to image processing, in general, and image segmentation, in concerning edge detection is large. We will not attempt to in detail, but instead we will have a brief look at the differe have a good feeling for the type of edge detection mechan in *VIS ā VIS*.

There are four distinct approaches to the problem of edge detection: template matching, edge fitting, statistical edge detection, and gradient and difference based filtering. We will look at each in turn

3.1.1 Template matching.

Since an ideal edge is essentially a step-like pattern, one straight-forward approach to edge detection is to try to match templates of these ideal step edges with regions of the same size at every point in the image. Several edge templates must be used, each template representing an ideal step at a different orientation. The degree of match can, for example, be determined by evaluating the cross-correlation between the template and the image. The template producing the highest correlation determines the edge magnitude at that point and the edge orientation is assumed to be that of the corresponding template. Detection is accomplished by thresholding the edge response.

3.1.2 Edge fitting.

If an ideal edge can be modelled as a step discontinuity in intensity at a particular location in the image, it can also be described by a small number of well-defined parameters. For example, in a formulation due to Hueckel, a step function can be defined in a circular region by a function $S(x, y)$ defined as follows (refer to figure 3.1):

$$S(x,y) = \begin{cases} b & (x\cos\theta + y\sin\theta) < r \\ b+h & (x\cos\theta + y\sin\theta) \leq r \end{cases}$$

where h is the step height (intensity difference), b is the base intensity, r and θ define the position and orientation of the edge line with respect to the origin of this circular region.

The approach to edge detection in this case is to determine how closely this model fits a given image neighbourhood and to identify the values of b, h, r, and θ that minimise some measure of the distance between this circular step function and a corresponding area in the image.

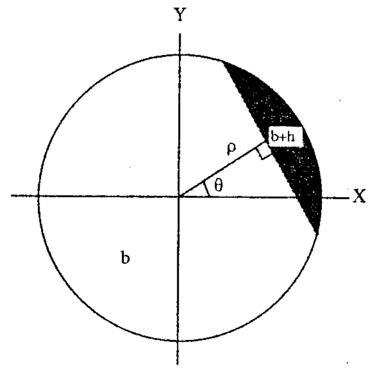

Figure 3.1: Parameters defining a step function in a circular region

3.1.3 Statistical approaches to edge detection.

The third class involves statistical techniques. If you consider a small region or window in an image (e.g. a 7x7 pixel area), you can view an edge as a boundary between two inhomogeneous sub-regions. On the other hand, if the entire region is homogeneous, then no edge exists. This is the basis of a statistical edge detection technique due to Yakimovsky who treated edge detection as an exercise in hypothesis testing, choosing between the two hypotheses that:

H0: The image values on two sides of a line through the window are taken from the same region.
H1: The image values on one side are from one region and on the other side from a different region.

Assuming that each region comprises pixels having normally distributed grey-levels and each pixel in the region is mutually independent, Yakimovsky argued that that the choice can be made by considering the ratio of (a) the grey-level standard deviation of the combined region (raised to the power of the number of pixel in the region) to (b) the product of the grey-level standard deviations of both individual regions each with standard deviation raised to the power of the number of pixels in the respective region. To decide whether an edge of a given orientation exists in a window, one can bisect the window at that orientation, thus creating two sub-regions, calculate the appropriate standard deviations and decide as to the presence of an edge based on some selected threshold. This must be done for several orientations, for example, 0°, 30°, 60°, 90°, 120°, and 150°. There are several other statistical edge detectors but

this one serves to illustrate the approach.

3.1.4 Gradient and difference-based filtering.

Finally we come to the approach which is most relevant to *VIS à VIS*. If we define a local edge in an image to be a transition between two regions of significantly different intensities then the gradient function of the image, which measures the rate of change, will have large values in these transitional boundary areas. Thus gradient-based, or first-derivative based, edge detectors enhance the image by estimating its gradient function and then signal that an edge is present if the gradient value is greater than some defined threshold.

In more detail, if $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ represent the rates of change of a 2-D function $f(x, y)$ in the x and y directions respectively, then the rate of change in a direction θ (measured in the positive sense from the X-axis) is given by:

$$\frac{\partial f}{\partial x}\cos\theta + \frac{\partial f}{\partial y}\sin\theta$$

The direction θ, at which this rate of change has the greatest magnitude is given by

$$\arctan\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

with magnitude

$$\sqrt{\frac{\partial f}{\partial x}^2 + \frac{\partial f}{\partial y}^2}$$

The gradient of $f(x, y)$ is a vector at (x, y) with this magnitude and direction. Thus the gradient may be estimated if the directional derivatives of the function are known along two orthogonal directions. Once the gradient magnitude has been estimated, a decision as to whether or not an edge exists is made by comparing it to some pre-defined value; an edge is deemed to be present if the magnitude is greater than this threshold. Obviously the choice of threshold is important and, in noisy images, threshold selection involves a trade-off between missing valid edges and including noise-induced false edges. Indeed, the most successful gradient-based edge detectors incorporate an adaptive thresholding strategy which dynamically alters the threshold on the basis of the local structure of the gradient image.

So far, difference-based edge detection has been discussed on the basis of first derivative directional operators. However, an alternative method uses an approximation to the Laplacian:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

i.e. the sum of second-order, unmixed partial derivatives. The digital Laplacian has zero-response to linear ramps (and thus gradual changes in intensity) but it does respond on either side of the edge, once with a positive sign and once with a negative sign. Thus in order to detect edges, the image is enhanced by evaluating the digital Laplacian and isolating the points at which the resultant image goes from positive to negative, i.e., at which it crosses zero. The Laplacian has one significant disadvantage: it responds very strongly to noise.

3.2 The Marr-Hildreth theory of edge detection.

A different and much more successful application of the Laplacian to edge detection was proposed by Marr and Hildreth in 1980. This approach first smooths the image by convolving it with a two-dimensional Gaussian function, and subsequently isolating the zero-crossings of the Laplacian of this image:

$$\nabla^2(I(x,y) * G(x,y))$$

where $I(x,y)$ represents the image intensity at a point (x,y) and $G(x,y)$ is the 2-D Gaussian function, of a given standard deviation σ, defined by:

$$G(x,y) = \frac{1}{2\pi\sigma^2} \exp^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

The operator possesses a number of useful properties: for example: the evaluation of the Laplacian and the convolution commute so that, for a Gaussian with a given standard deviation, we can derive a single filter: the Laplacian of Gaussian:

$$\nabla^2 (I(x,y) * G(x,y)) = \nabla^2 G(x,y) * I(x,y)$$

Furthermore, this 2-D convolution is separable into four 1-D convolutions (see Appendix for a derivation):

$$\nabla^2 (I(x,y) * G(x,y)) = G(x) * \left(I(x,y) * \frac{\partial^2}{\partial y^2} G(y) \right) + G(y) * \left(I(x,y) * \frac{\partial^2}{\partial x^2} G(x) \right)$$

Bearing in mind that an implementation of the operator requires an extensive support, e.g. 63x63 pixels for a Gaussian with standard deviation of 9.0, this separability facilitates significant computational savings, reducing the required number of multiplications from n^2 to $4n$ for a filter kernel size of n pixels. The Laplacian of Gaussian operator also yields thin continuous closed contours of zero-crossing points. This property is useful in subsequent processing, such as when characterising the intensity discontinuities or edges as object boundaries. The location of the zero-crossings in space is not the only information that can be extracted from the convolved image: the amplitude (which is related to the slope) and orientation of the gradient of the convolved image at the zero-crossing point provide important information about edge contrast and orientation. The slope of a zero-crossing is the rate at which the convolution output changes as it crosses zero and is related to the contrast and width of the intensity change.

Since the Gaussian is used to smooth the image and since different standard deviations yield edges detected at different scales within the image (successively smoothing out image detail), Marr's theory also requires the correlation of edge segments derived using Gaussians of different standard deviation. The basic idea underlying this correlation is that edge segments which have corresponding intensity discontinuities at several spatial scales are more likely to correspond to physical edges in the scene being imaged. We will return to this topic again in chapter 6 when we discuss the raw primal sketch.

3.3 Generating Edges in *VIS à VIS*.

3.3.1 Filtering with Laplacian of Gaussian (∇^2G) masks and extracting zero-crossing contours.

From the above, we can see that the isolation of intensity discontinuities can be viewed equally as the isolation of zero-crossings in the Laplacian of a Gaussian-filtered image. This is accomplished in *VIS à VIS* by transferring an intensity image to a convolution image, which effects the convolution: $\nabla^2 * I$, and subsequently from the convolution image to a zero-crossing image, which effects the localization and characterization of the zero-crossing point. When using *VIS à VIS* interactively, the user is asked to enter the standard deviation of the Gaussian function, expressed in pixels. Figure 3.2(b) shows the result of this convolution of the image shown in figure 3.2(a) with a Laplacian of Gaussian mask, where the Gaussian has a standard deviation of three pixels. When transferring the convolution image (figure 3.2(b)) to a zero-crossing image (figure 3.2(c)), which is an array image representing the zero-crossing points of the convolution image, the user can optionally cause the associated contour, slope, and orientation images to be generated. You will recall from chapter 2 that these contour images represent the contours explicitly on a point by point basis: the contour image represents the BCC of the zero-crossing contour, while the slope and orientation images have as their values estimates of the strength of the zero-crossing and the local orientation (tangential angle) of the zero-crossing contour. Note that the zero-crossing image represents the zero-crossing value either with the value 255 (white) or the value of the slope measure, or the value of the orientation at that zero-crossing point. The choice is specified by the user. Figures 3.2 to 3.4 show the zero-crossings of the Laplacian of Gaussian of the original intensity image for standard deviations of 3, 6, and 9 pixels, respectively.

3.3.2 Computing contour statistics.

Whenever the contour, slope, and orientation images are generated, several statistics of these features are computed. Specifically, these statistics are the mean and standard deviation of *all* slope, orientation, and curvature values in the image, together with the mean and standard deviation of contour length. Additionally, the mean and

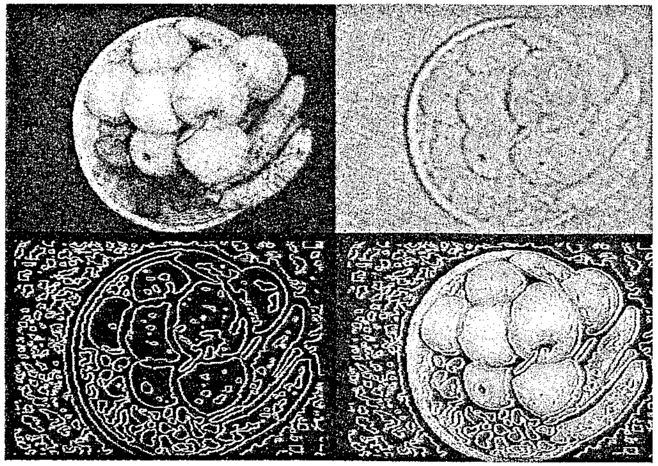

Figure 3.2: (a) Top left: an intensity image. (b) Top right: Laplacian of Gaussian $\nabla^2 G$; $\sigma = 3$ pixels. (c) Bottom left: zero-crossings. (d) Bottom right: zero-crossings overlaid on the intensity image.

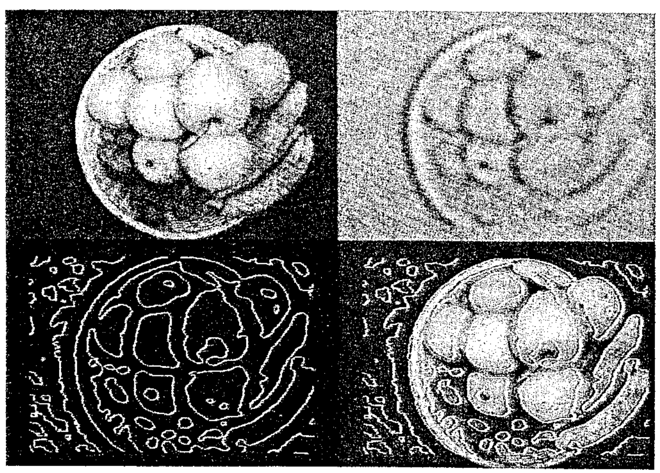

Figure 3.3: (a) Top left: an intensity image. (b) Top right: Laplacian of Gaussian $\nabla^2 G$; $\sigma = 6$ pixels. (c) Bottom left: zero-crossings. (d) Bottom right: zero-crossings overlaid on the intensity image.

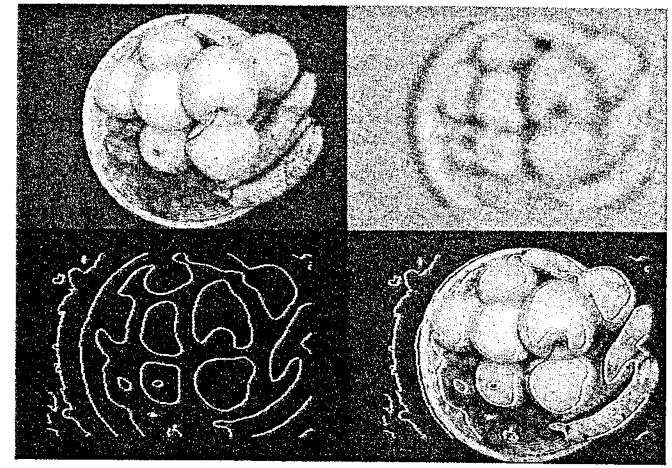

Figure 3.4: (a) Top left: an intensity image. (b) Top right: Laplacian of Gaussian $\nabla^2 G$; $\sigma = 9$ pixels. (c) Bottom left: zero-crossings. (d) Bottom right: zero-crossings overlaid on the intensity image.

standard deviation of slope, orientation, and curvature are computed for each individual contour. Using *VIS à VIS* interactively, you can step through these statistics on a contour by contour basis, beginning with the longest contour.

3.3.3 Conditional selection of contours.

New zero-crossing images can be generated on the basis of the global and individual contour properties. In particular, entire contours can be selected for inclusion in the new image by providing threshold ranges L_i and H_i for some sub-set of gross contour features f_i (i.e. for each feature stored in the contour descriptor). All contours are selected which satisfy the Boolean condition:

$$(L_1 \leq f_1 \leq H_1) AND (L_i \leq f_i \leq H_i) \ldots AND (L_n \leq f_n \leq H_n)$$

where i is the feature number, L_i and H_i are the low and high limits on the threshold, and f_i is the contour feature. All contours which satisfy the selection condition are effectively added to the current list of selected contours (in the fashion of a logical OR operation) and, in this way, it is possible to construct artibrarily complex selection criteria. For example, a user might first specify the condition that contours are to be selected whose lengths are in the range L_1 = 10 pixels to H_1 = 150 pixels. Subsequently, (s)he might specify that contours whose mean slope value is in the range $L_2 = 10$ to $H_2 = 20$ and whose standard deviation in curvature is less than 20 (i.e. $L_3 = 0$, $H_3 \leq 20$. Now, the selected contours are those which satisfy both the first (length condition and the second compound (slope and curvature) condition. Obviously, the user would have had to de-select all contours before commencing this selection procedure for this to be useful; *VIS ā VIS* provides just such a utility for selection and de-selection of all contours.

Selection criteria of partial contours based on local contour analysis is also provided. When transferring from a contour pseudo image to a zero-crossing image, only those contours (and partial contours) which are currently selected are actually referenced. In transferring to contour pseudo images, only contours selected by global features (contour descriptor information) are transferred, although the partial selection information is retained. This mechanism provides a powerful tool for interactively investigating the inter-relationship between intrinsic image types based on both global and local properties.

3.3.4 A case study in adaptive selection of zero-crossing contours.

As will be evident from the examples in figures 3.2 – 3.4, one of the problems associated with the Laplacian of Gaussian operator is that it frequently gives rise to spurious, or 'noisy', zero-crossing contours, especially when the standard deviation of the Gaussian is small. These contours are not relevant to image understanding, they often convey little or no information regarding the physical structure of the scene, and they present an unnecessary computational load for subsequent vision processes. This is due in part to its sensitivity to noise (or, from another perspective, to weak edges) and in part to the aforementioned property of yielding closed contours, some sections of which are not necessarily caused by physical edges. In order to remove unwanted zero-crossing contours, one must nominate a threshold on some contour property (or set of properties) which distinguishes the desirable zero-crossings. Properties which have been used in the past include the local slope of the zero-crossing, the local orientation of the zero-crossing contour, and a measure of the energy of the region enclosed by the contour. Methods for identifying the thresholds include manual selection, the use of a priori knowledge regarding the content of the scene, and adaptive techniques based on some characteristic of the image at hand.

We used *VIS ā VIS* to determine whether or not there are any invariant properties of the zero-crossing contours, in the sense that they are independent of the scene, which may be used to isolate the relevant contours from those which are irrelevant. It was decided that, since complete continuous contours are more useful than highly segmented ones, the decision process would be based on the statistics of features of individual contours with reference to global image-wide statistics.

Seven features were used in this investigation: the mean of the contour length, and the mean and standard deviation of the zero-crossing slope (i.e. strength), orientation and curvature. These features were chosen for reasons of computational expediency: they are easy to compute and they are obvious intrinsic parameters of contours. Other parameters such as the degee of spatial coincidence of contours at several scales (i.e. derived from Laplacian of Gaussian filters of varying standard deviation) are, without doubt, of great importance. However, the computation complexity of matching n contours (at just two different scales) is $O(n^2)$ and the ramifications for interactive assessment of the pairing is significant: a subject would have to anwer in the order of 10,000 questions pertaining to the correlation between contours pairs in a scene containing just 100 contours. This was not practicable.

Ten images of real scenes of varying complexity were collected. The rationale underlying the choice of scene is that the objects should vary in size, exhibit natural and varying texture, colour, and shape, and there should not be a preponderance of perfect straight edges typical of block world scenes. The zero-crossing contours derived from each image were presented to a subject, one contour at a time, and (s)he was asked to classify it as relevant or irrelevant. The subject was told that, for the purposes of the experiment, a relevant contour is one which (s)he considers meaningful and useful for the interpretation of the structure of the scene. To allow the subject to assess easily the relevance of each contour, four sub-images were simultaneously displayed on the monitor. These depicted the original grey-scale image, the complete set of zero-crossings, the single zero-crossing contour currently being classified, and the single zero-crossing contour superimposed on the grey-scale image.

Three standard deviations (3.0, 6.0 and 9.0) were used to derive three Laplacian of Gaussian masks and each were used in turn to generate a set of zero-crossing contours which were classified in this manner. Furthermore, a total of five subjects were asked to participate in this experiment, resulting in one hundred and fifty distinct data-sets. For each of these one hundred and fifty data-sets, the mean and standard deviation of the slope, orientation, and curvature of each contour was computed. Including length, this provides seven features. The mean and standard deviation of these features were then computed for all contours (μ_a, σ_a), for the relevant contours (μ_r, σ_r), and for the irrelevant contours (μ_i, σ_i). These values are summarised in Table 3.1 and their distributions are graphed in figures 3.5 to 3.10. Note that the mean and standard deviation of feature 6 (the mean contour curvature) is zero and is not included in these figures: Laplacian of Gaussian filtering yields recursively nested regions of alternating sign and thus the curvature of a given contour will be either 2π or -2π (normalised by the contour length) depending on whether the contour encloses a positive or a negative region. Thus, the mean of this feature for all contours over all data-sets will be zero.

It is clear from Table 3.1 and figures 3.8, 3.9, and 3.10 that the distributions of features 4, 5, and 6 (mean contour orientation, standard deviation of contour orientation, and standard deviation of contour curvature) for the three classes (all contours, relevant contours, and irrelevant contours) are approximately the same and do not offer any basis for discriminating between relevant and irrelevant contours. On the other hand, the distribution of features 1, 2, and 3 (contour length, mean contour slope, and standard deviation of contour slope) for relevant contours are quite different from those for irrelevant contours. Unfortunately, it transpires that the criterion of length, while nominally useful for discrimination, is not of any practical value since some images yield very large, but spurious, zero-crossing contours. The remaining two features should be useful for classification of contours. In order to

Feature	Feature number	All Contours		Relevant Contours		Irrelevant Contours	
		μ_a	σ_a	μ_r	σ_r	μ_i	σ_i
Length	1	59.9	185.6	256.8	395.3	34.9	116.1
Mean Slope	2	7.2	10.7	24.3	23.1	5.1	4.5
Std. Dev. Slope	3	2.7	4.1	9.6	7.1	1.9	2.4
Mean Orientation	4	122.5	18.3	125.0	15.5	122.2	18.6
Std. Dev. Orientation	5	56.0	16.6	70.0	8.6	54.2	16.5
Mean Curvature	6	0.0	0.0	0.0	0.0	0.0	0.0
Std. Dev. Curvature	7	69.3	24.3	55.5	22.8	71.1	23.9

Table 3.1: Distribution of Features by Class

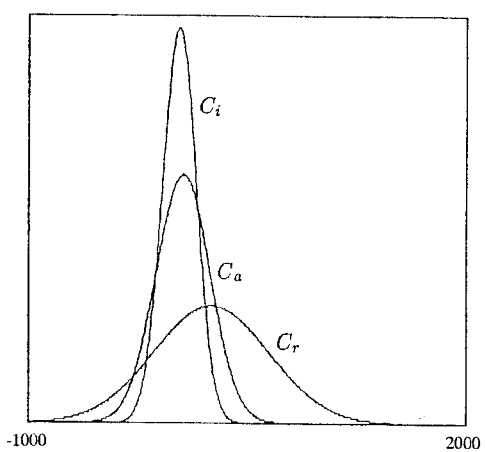

Figure 3.5: Distribution of contour length for all contours C_a, relevant contours C_r, and irrelevant contours C_i

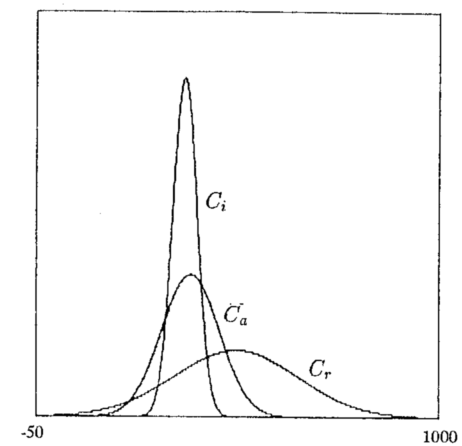

Figure 3.6: Distribution of mean slope of contour for all contours C_a, relevant contours C_r, and irrelevant contours C_i

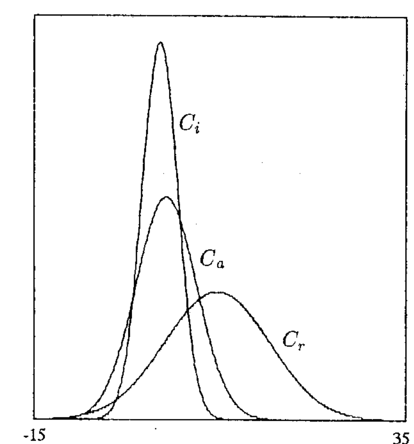

Figure 3.7: Distribution of standard deviation slope of contour for all contours C_a, relevant contours C_r, and irrelevant contours C_i

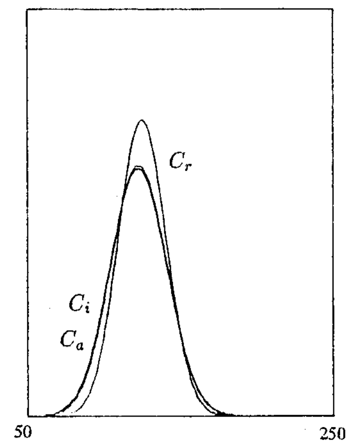

Figure 3.8: Distribution of mean orientation of contour for all contours C_a, relevant contours C_r, and irrelevant contours C_i

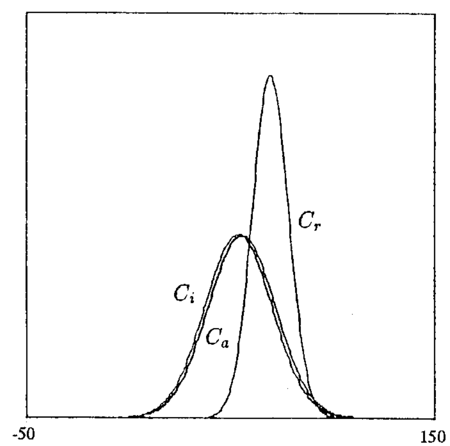

Figure 3.9: Distribution of standard deviation of orientation of contour for all contours C_a, relevant contours C_r, and irrelevant contours C_i

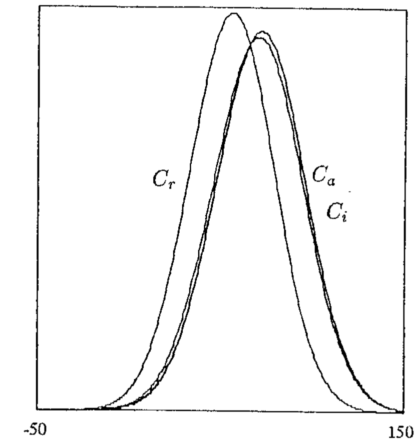

Figure 3.10: Distribution of standard deviation of curvature of contour for all contours C_a, relevant contours C_r, and irrelevant contours C_i

minimise the likelihood of misclassification, we can interpret the two normal curves corresponding to the distributions for relevant and irrelevant contours as probability density functions and select as the threshold criterion the value of the contour feature (either mean contour slope or standard deviation of contour slope) for which the contour is equally likely to be relevant or irrelevant. This value is given by the intersection of the two curves.

To make this threshold adaptive and not scene dependent, it is formulated in terms of the mean and standard deviation of the distribution of all contours (μ_a, σ_a): for feature 2, mean contour slope, the threshold value is given by:

$$t_1 = \mu_a + b\sigma_a \tag{3.1}$$

Similarly, for feature 2:

$$t_2 = \mu_a + c\sigma_a \tag{3.2}$$

Deriving coefficients b and c from Table 3.1, we have:

$$b = 0.583 \tag{3.3}$$

and

$$c = 0.711 \tag{3.4}$$

These classification criteria were used to remove irrelevant contours from each of the ten test images at three spatial scales (i.e. with three standard deviations of the

Gaussian in the Laplacian of Gaussian filter); for examples of the ten scenes used in this study, see figures 3.11 to 3.25 which show the union of these two sets of relevant contours, i.e. the contours which satisfy either classification criterion.

The objective of this study was to identify some means to remove automatically and adaptively the extraneous zero-crossing contours of Laplacian of Gaussian filtered images. This is done in order to simplify the computational complexity of subsequent image processing such as estimation of stereo disparity, computation of optical flow, or the generation of the Raw Primal Sketch. The zero-crossing selection criteria described work well, removing for the most part only irrelevant contours and retaining those which are useful for the interpretation of the structure of the scene. The most obvious deficiency of the technique is that it utilises gross selection of complete contours. This can occassionally result in the removal of valid intensity discontinuities which account for only a part of an entire contour. Obviously, this selection process should be modified somewhat so that contours are segmented (for example, at points of high curvature) and to select/deselect these segments rather than the entire contour on the basis of the classification criteria described above.

3.3.5 Summary.

VIS ā VIS uses the Marr-Hildreth Laplacian of Gaussian filter to isolate intensity discontinuities in images by identifying the zero-crossings in the filtered image. These zero-crossings are characterized by their position, strength or slope of crossing, orientation, and curvature. Each of these image types have appropriate representations in *VIS ā VIS*; iconic arrays for intensity, convolution ($\nabla^2 G * I$), and zero-crossing images, while the zero-crossing attributes are represented by sequences of augmented Boundary Chain Codes (BCCs). The extraction of this zero-crossing information is effected by transferring from an intensity image to a convolution image, and from there to a zero-crossing image. The contour, slope, and orientation images can be optionally generated, together with statistics of these zero-crossing attributes. Subsequently, the zero-crossing contour can be 'selected' or 'de-selected' from further processing on the basis of these attributes. A simple analysis of these zero-crossing features reveals that the most important zero-crossing contours, from the point of view of image interpretation, can be automatically retained (selected) on the basis of their attributes in a manner which is reasonably independent of the scene. We are not suggesting that this selection criterion is some fundamental 'truth' about the nature of zero-crossings; it is, rather, a robust heuristic derived by empirical analysis using *VIS ā VIS*. But is it none the less useful for that.

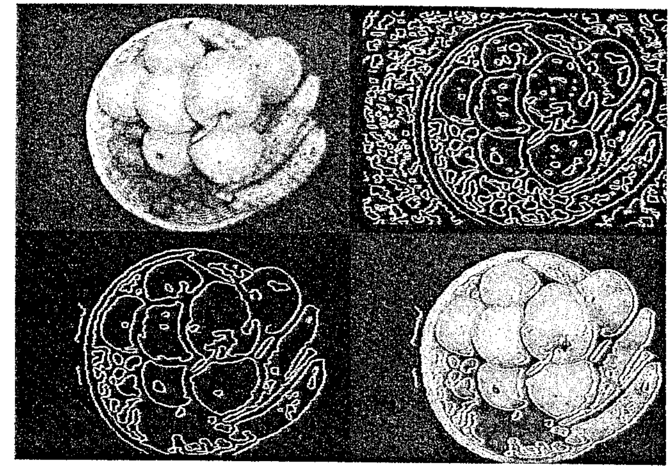

Figure 3.11: Standard deviation of Gaussian = 3.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

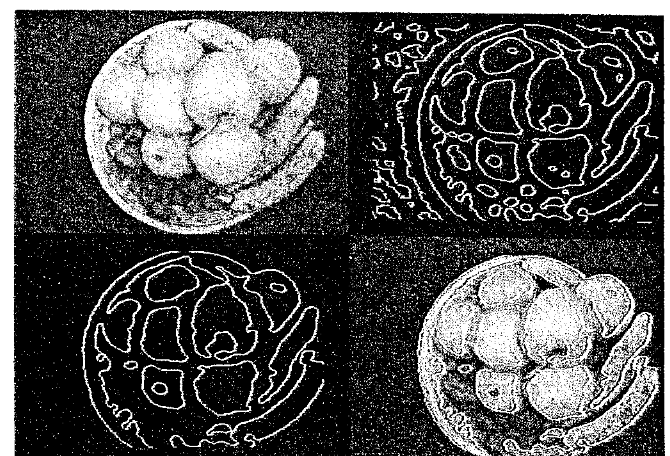

Figure 3.12: Standard deviation of Gaussian = 6.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

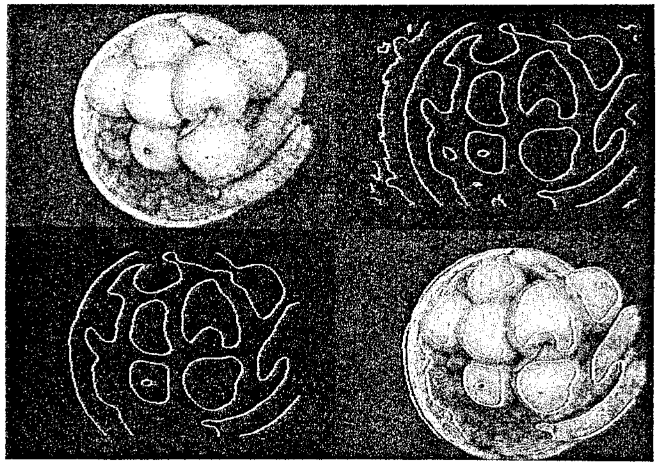

Figure 3.13: Standard deviation of Gaussian = 9.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

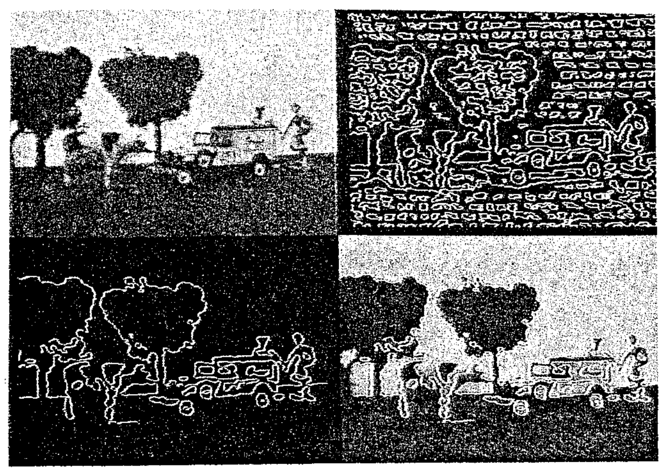

Figure 3.14: Standard deviation of Gaussian = 3.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

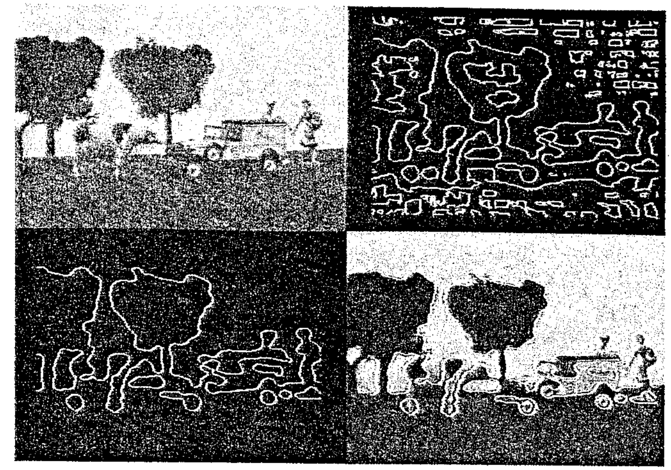

Figure 3.15: Standard deviation of Gaussian = 6.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

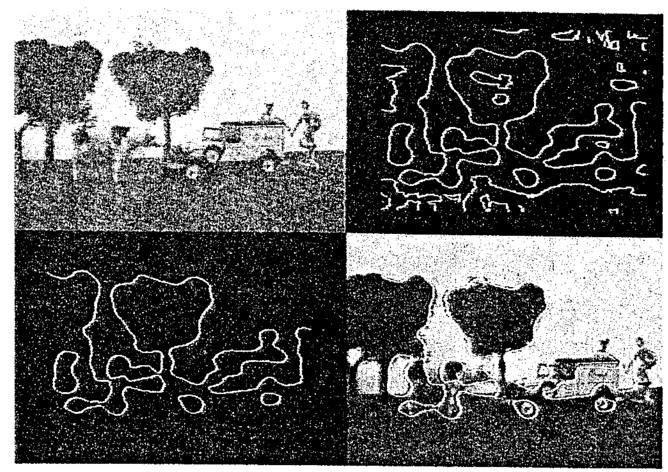

Figure 3.16: Standard deviation of Gaussian = 9.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

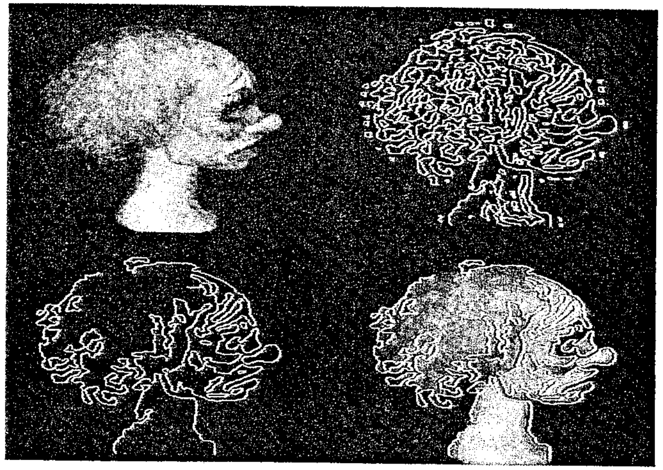

Figure 3.17: Standard deviation of Gaussian = 3.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

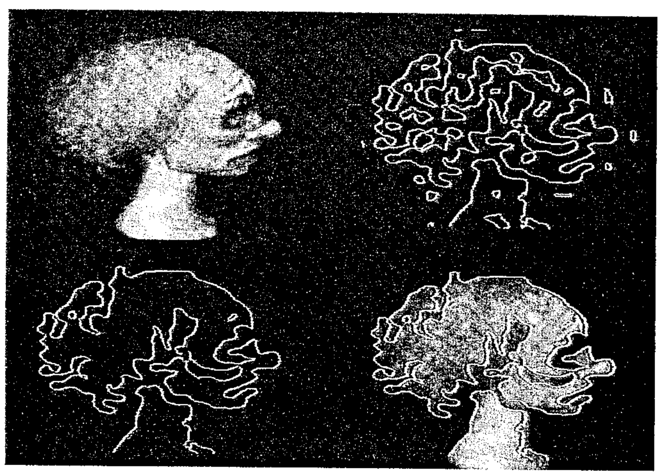

Figure 3.18: Standard deviation of Gaussian = 6.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

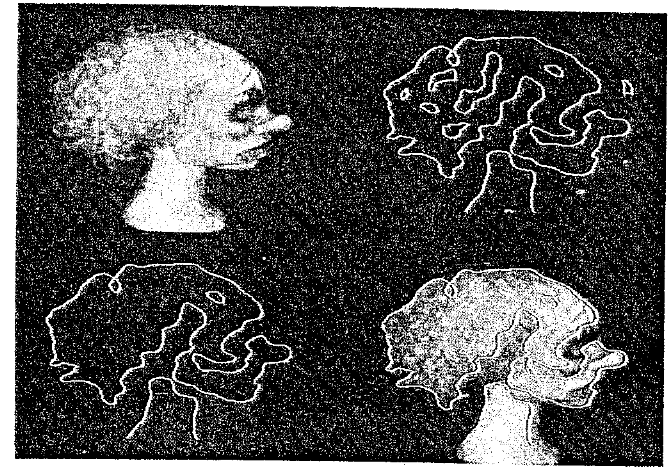

Figure 3.19: Standard deviation of Gaussian = 9.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

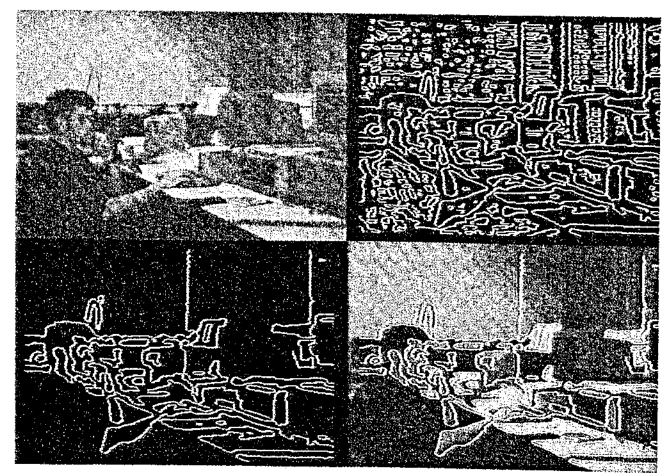

Figure 3.20: Standard deviation of Gaussian = 3.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

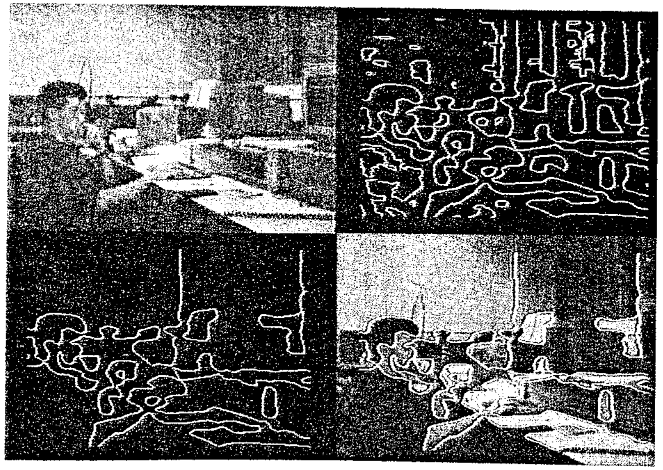

Figure 3.21: Standard deviation of Gaussian = 6.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

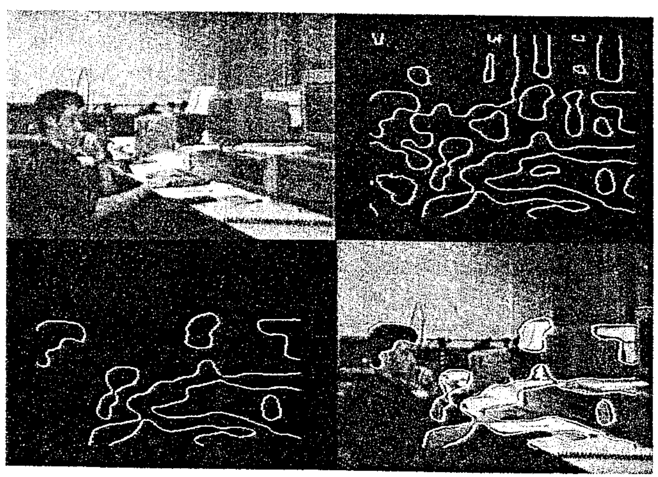

Figure 3.22: Standard deviation of Gaussian = 9.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

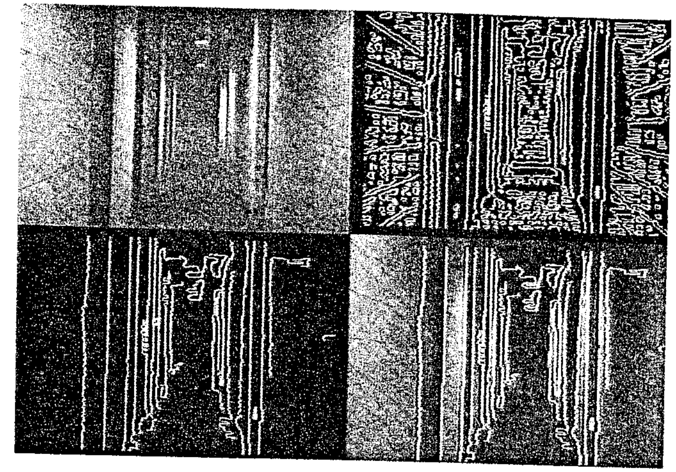

Figure 3.23: Standard deviation of Gaussian = 3.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

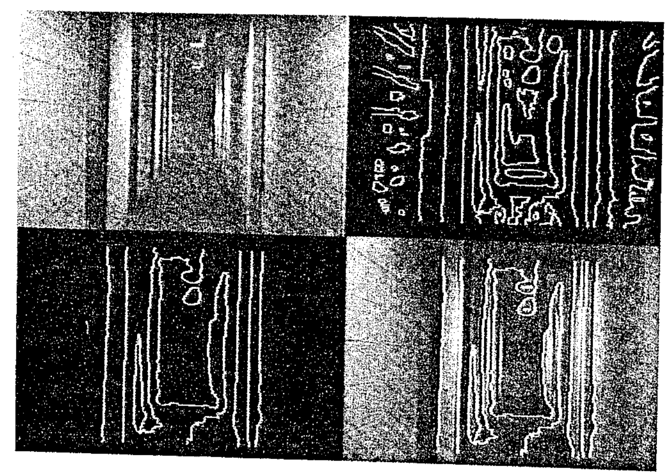

Figure 3.24: Standard deviation of Gaussian = 6.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

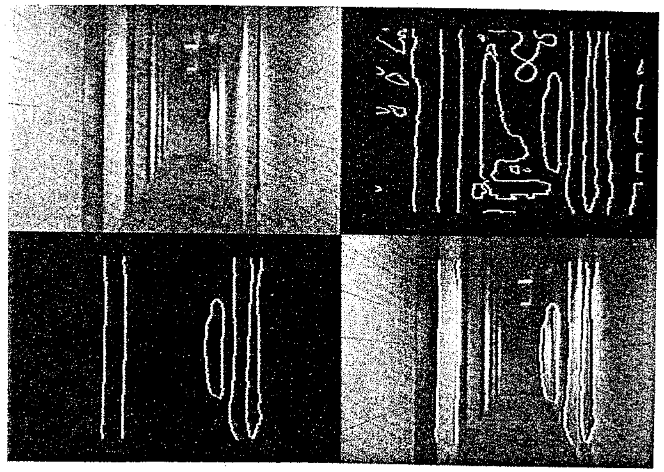

Figure 3.25: Standard deviation of Gaussian = 9.0. Top left: original intensity image. Top right: all zero-crossing contours. Bottom left: adaptively selected relevant contours. Bottom right: relevant contours superimposed on intensity image.

Bibliography

[1] Abdou, I.E. and Pratt, W.K. 1979. 'Quantitative Design and Evaluation of Enhancement/Thresholding Edge Detectors', Proceedings of the IEEE, Vol. 67, No. 5, (May), pp.753-763.

[2] Argyle, E. 1971. 'Techniques for Edge Detection', Proc. IEEE, Vol. 59, pp.285-287.

[3] Barrett, W.A. 1981. 'An Iterative Algorithm for Multiple Threshold Detection', Proc. IEEE Computer Society Conference on Pattern Recognition and Image Processing, Dallas, pp.273-278.

[4] Canny, J. 1986. 'A Computational Approach to Edge Detection', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, pp.679-698.

[5] Cooper, D. and Sung, F. 1983. 'Multiple-Window Parallel Adaptive Boundary Finding in Computer Vision', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-5, No. 3, pp.299-316.

[6] Davis, L.S. 1975. 'A Survey of Edge Detection Techniques', Computer Graphics and Image Processing, Vol. 4, No. 3, pp.248-270.

[7] Fischler, M.A. and Bolles, R.C. 1986. 'Perceptual Organisation and Curve Partitioning', IEEE Transaction of Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 1, pp.00-05.

[8] Fram, J.R. and Deutsch, E.S. 1975. 'On the Quantitative Evaluation of Edge Detection Schemes and their Comparison with Human Performance', IEEE Transaction of Computers, Vol. C-24, No. 6, pp.616-628.

[9] Frei, W. and Chen, C-C. 1977. 'Fast Boundary Detection: A Generalisation and a New Algorithm', IEEE Transactions on Computers, pp.988-998.

[10] Giordano, A., Maresca, M, Sandini, G., Vernazza, T., and Ferrari, D. 1985. 'A Systolic Convolver for Parallel Multiresolution Edge Detection', Internal Report, DIST - University of Genoa.

[11] Giordano, A., Maresca, M, Sandini, G., Vernazza, T., and Ferrari, D. 1987. 'VLSI-based Systeolic Architecture for fast Gaussian Convolution', Optical Engineering, Vol. 26, No. 1, pp.63-68.

[12] Griffith, A.K. 1973. 'Edge Detection in Simple Scenes using A Priori Information', IEEE Transactions on Computers, Vol. 22, No. 4, pp.371-380.

[13] Grimson, W.F.L. and Hildreth. E.C. 1985. 'Comments on Digital Step Edges from Zero Crossings of Second Directional Derivatives', IEEE Trans.on Pattern Analysis and Machine Intelligence, Vol. PAMI-7, No. 1, pp.121-127.

[14] Guari, E., and Wechsler, H. 1982. 'On the Difficulties involved in the Segmentation of Pictures', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-4, No. 3.

[15] Haralick, R.M. 1984. 'Digital Step Edges from Zero-Crossing of Second Directional Derivatives', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, No. 1, pp.58-68.

[16] Hildreth, E.C. 1981. 'Edge Detection in Man and Machine', Robotics Age, Sept/Oct, pp.8-14.

[17] Hildreth, E.C. 1985. 'Edge Detection', A.I. Memo No. 858, M.I.T. A.I. Lab.

[18] Huertas, A. and Medioni, G. 1986. 'Detection of Intensity Changes with Subpixel Accuracy Using Laplacian-Gaussian Masks', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 5, pp.651-664.

[19] Kirsch, R. 1971. 'Computer Determination of the Constituent Structure of Biological Images', Computers and Biomedical Research, Vol. 4, No. 3, pp.315-328.

[20] Lee, C.C. 1983. 'Elimination of Redundant Operations for a Fast Sobel Operator', IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-13, No. 3, pp.242-245.

[21] Lunscher, W.F. and Beddoes, M.P. 1986. 'Optimal Edge Detector Design I: Parameter Selection and Noise Effects', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 2, pp. 164-177.

[22] Lunscher, W.F. and Beddoes, M.P. 1986. 'Optimal Edge Detector Design II: Coefficient Quantisation', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 2, pp. 178-187.

[23] Marr, D. 1976. 'Early Processing of Visual Information', Philosophical Transactions of the Royal Society of London, B275, pp.483-524.

[24] Marr, D. and Hildreth, E. 1980. 'Theory of Edge Detection', Proceedings of the Royal Society of London, B207, pp.187-217.

[25] Nalwa, V.S. and Binford T.O. 1986. 'On Detecting Edges', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, pp.699-714.

[26] Sandini, G. and Torre, V. 1985. 'Thresholding Techniques for Zero-Crossings', Proceedings of 'Winter '85 Topical Meeting on Machine Vision', Incline Village, Nevada.

[27] Suciu, E. and Reeves, A.P. 1982. 'A Comparison of Differential and Moment-Based Edge Detectors', Proc. IEEE Computer Society Conference on Pattern Recognition and Image Processing, pp.97-102.

[28] Torre, V. and Poggio, T.A. 1986. 'On Edge Detection', IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 2, pp.147-163.

LLIS HORWOOD SERIES IN
RTIFICIAL INTELLIGENCE
eries Editor: Professor JOHN CAMPBELL, Department of Computer Science, University
ollege London

ARALLEL COMPUTER VISION
he VIS à VIS System

r D. VERNON, Department of Computer Science, Trinity College Dublin, Ireland, and
rofessor G. SANDINI, DIST, University of Genoa, Italy

escribing a computer vision environment which supports coarse granularity parallelism on
ansputer-based systems, this book discusses a system which treats the organization of the
nage data and its representation and interdependencies equally with the visual processes
hich operate on that data. As an environment, the system – VIS à VIS – provides dynamic
eation of images and attendant data structures, each of which are linked to provide an
plicit representation of the interdependence of the visual cues. As a vision system, VIS à
S offers all the required functionality of 3-D image understanding, from image acquisition
rough detection and analysis of intensity discontinuities, computation of depth from stereo
d camera motion, segmentation using raw and full primal sketches, through to 3-D model
nstruction and object recognition. Both geometric and functional parallelism are achieved
exploiting both the in-built vision control language – with its facility for remote execution
procedures – and the integrated system data-structures.

of these issues are dealt with in considerable detail, and the book's emphasis on system
ganization integration of information and computer architectures, rather than purely
eoretical and algorithmic issues, will make it a very welcome addition to the literature.

adership: Computer vision scientists. Industrial research and development engineers.
mputer scientists. Information technologists.

vid Vernon's interests stretch from robot vision to the philosophy of science. His recent research
ivities have been concerned with the development of robot systems which display true autonomy. His
rk encompasses computational theories of perception, self-organization, and intelligence. Dr Vernon
Fellow of Trinity College Dublin, Ireland.
lio Sandini has been working on research in computer and biological vision for more than twelve
rs, his primary interests at present being the integration of many low-level visual cues in a robust
nputational framework. He has also been very active in exploiting the neurobiological principles of
ly vision to postulate and implement artificial visual sensing at the 'retinal' level of computer vision.
fessor Sandini is currently at the University of Genoa in Italy.

es of related interest

RMAL SPECIFICATION OF ADVANCED AI ARCHITECTURES
N D. CRAIG, Department of Computer Science, University of Warwick

TIFICIAL INTELLIGENCE AND INTELLIGENT SYSTEMS:
e Implications
VID ANDERSON, Division of Computer Science, Teesside Polytechnic

TIFICIAL INTELLIGENCE TERMINOLOGY:
eference Guide
neral Editor: COLIN BEARDON, Department of Computer Science, University of Exeter

ISBN 0-13-932716-9
9 780139 327162

LIS HORWOOD

Vernon and Sandini **PARALLEL COMPUTER VISION** the VIS $\vec{a}$ VIS system

ELLIS HORWOOD

ELLIS HORWOOD SERIES IN ARTIFICIAL INTELLIGENCE

PARALLEL COMPUTER VISION
the VIS $\vec{a}$ VIS system

D. Vernon and G. Sandini