Displacement and Disparity Representations in Early Vision

by

Steven James White

B.S., University of Washington (1985)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 1992

Signature of Author .....
Department of Electrical Engineering and Computer Science
September 9, 1992

Certified by .....
W. E. L. Grimson
Associate Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Certified by .....
B. K. P. Horn
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Campbell Searle
Chairman, Departmental Committee on Graduate Students

Displacement and Disparity Representations in Early Vision

by

Steven James White

Submitted to the Department of Electrical Engineering and Computer Science
on September 9, 1992, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

As the first stages of image processing, early vision algorithms generally fall into two categories: symbolic or continuous domain. The continuous domain approaches are rich in scene related information, but they lack any means to incorporate knowledge of the physics and optics of imaging to constrain the interpretation of image data into 3D scene properties. The symbolic approaches typified by edge-finding algorithms result in representations which have the opposite effect. The data is readily incorporated into "top-down" high-level vision algorithms, yet much of the useful information content relating to the scene is sacrificed in the continuous to symbolic transformation of marking edges. Both of these approaches have strengths which would be useful if they could be incorporated into a single model.

The Displacement and Disparity Models of early vision are continuous domain models based on edge features. Displacement representations measure edge position, orientation and contrast at all points in the image domain. The computations involved in the Displacement calculation are nonlinear and based on the Laplacian and gradient of Gaussian image representations. Disparity models are linear transforms of the Displacement representations. Some of these that are discussed are optical flow, cyclopean fused vision, feature focus, template alignment and stereo. The stereo model is developed in detail and tested on real images. These models allow incorporation of constraints based on real-world knowledge and incorporation into symbolic schemes, unlike prior continuous domain models. They also retain the full spatial precision and representational richness of the linear image transforms, unlike the symbolic models. The Displacement and Disparity models also provide many possible insights about the processes of biological early vision.

Thesis Supervisor: W. E. L. Grimson
Title: Associate Professor of Electrical Engineering and Computer Science

Thesis Supervisor: B. K. P. Horn
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I want to thank my co-advisors Eric Grimson and Berthold Horn for their advise and support throughout the development of this thesis as well Ellen Hildreth, Tomás Lozano-Pérez and Shimon Ullman who served on the committee and provided invaluable advise. A number of other individuals were extremely helpful in the development of the theory. Professor James J. Anderson, Rajan Ramaswamy, Sandy Wells, Professor Peter Schiller, Todd Cass, and Tao Alter are just a few. Frankly, the entire A.I. Lab deserves credit for providing such a unique environment where all kinds of interesting research can take place.

This research was supported in part by the Defense Advanced Research Projects Agency and was monitored by the Office of Naval Research under contract numbers N00014-85-K-0124

Contents

List of Figures

List of Tables

Chapter 1

Introduction

When an illuminated 3D scene is viewed, two-dimensional intensity images are sent to the brain. Vision is the process of extracting useful information about the world around the viewer from these images. This information can take many forms, such as the motion of the viewer, or of objects in the scene. It can be the 3D structure of the scene or it can be the identities and organization of objects in it.

Visible scene points in the 3D world will have corresponding image points in the 2D image. The shape, texture, illumination and motion of the scene objects will dictate the form and flow of the image features. It is not very hard to predict the image properties from the scene. It is, however, a difficult task to infer the scene properties from images of the scene. To arrive at such high-level knowledge from the limited information in the image intensity values is certainly a non-trivial task. These scene inference problems are the focus of vision research.

To arrive at scene properties, some knowledge of physical properties must be used to constrain the interpretation. Images contain too little information to unambiguously infer knowledge of the 3D world that generated them. Without reasonable assumptions about illumination, geometric constraints, optical properties of materials, dynamics and other common-sense criteria, reducing the one-to-many mapping of 2D images to 3D scenes to one-to-one is arguably impossible. It is certainly extremely

difficult.

Some information, however, can be extracted from scenes based almost entirely on the information content of the images. This process of extracting useful information from scenes is called "early vision". It has three objectives:

1. to reduce the substantial information content in intensity images unrelated to scene interpretation,

2. to preserve the information in the images relevant to the task of scene interpretation, and

3. to format that information in such a way that subsequent scene interpretation tasks are tractable.

The last step requires that the information encoded from image data relate to the properties of objects in the scene. Examples of such properties are color, shading, optical flow and feature position and orientation. They may still not directly correspond to the actual 3D world properties of the scene, but there should be some clear relationship between the two.

In computer vision, early vision often means "edge finding". Edges are image points where an intensity discontinuity, or step, occurs. They are usually associated with texture, occlusions, shadows, corners and other features in a scene. A typical approach is that once an edge is found, it is marked, usually with a single bit. All the non-edge points are left unmarked. A scene with somewhere between six and eighteen bits per pixel can be reduced to one or fewer in the process. Figure 1-1 shows a typical example of an image before and after edges are extracted. Since symbolic high level algorithms usually utilize these low level pre-processors to mark edges, these discrete edge marking routines are called "symbolic" in this thesis.

While this is most impressive as a data compression scheme, it leaves a lot to be desired when it comes to information content. Many marked edges are inevitably spurious and, if any significant compression is to take place, many "real" edges will

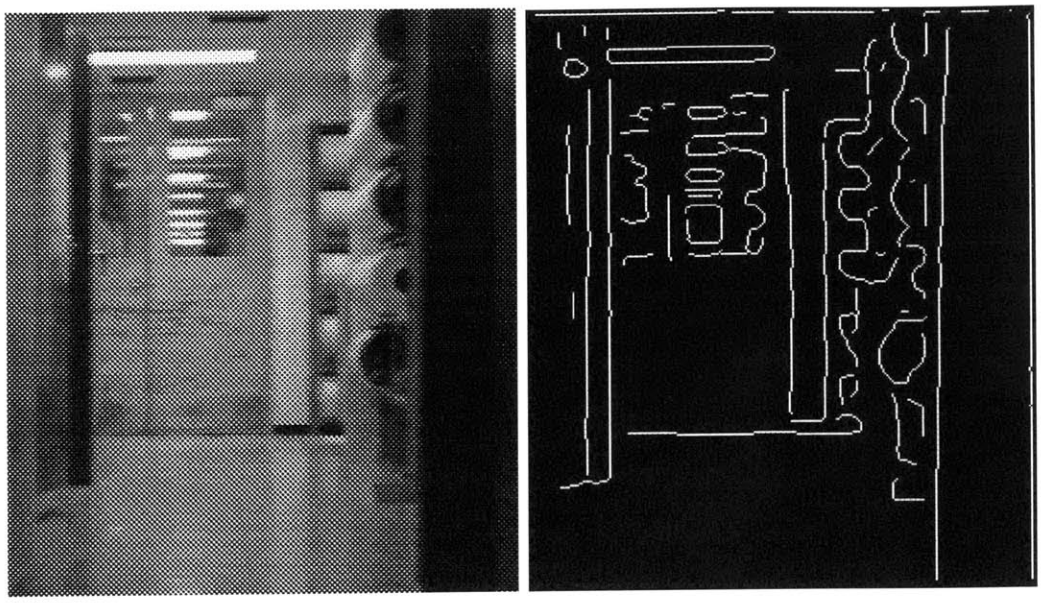

Figure 1-1: Hall image, before and after edge finding

be lost. On top of that, marking pixels is guaranteed to reduce acuity — the ability to locate edges precisely — to pixel level. Humans manage to sample images and process them without such stark sacrifices in information content at the early stages. Human acuity is a tiny fraction of the smallest retinal cell diameter.

This approach of converting images to bit-maps — either directly from the grey-level image data or some linear transformation of it — is almost certainly unlike any process involved in biological vision. Acuity is very high in almost all vision tasks tested in humans. This is sometimes explained by suggesting post-processing schemes to interpolate the pixel acuity results of the discrete edge feature models. Rather than being an afterthought or a functional appendage to an otherwise spatially impoverished representation, acuity instead appears to be an intrinsic property of almost all early vision representations.

Therefore, while edge finding reduces unneeded information content and produces generally useful spatial primitives for analysis, it tends to sacrifice substantial information that would be useful in the scene interpretation process — information the human system takes pains to preserve.

Some researchers have developed continuous domain models that preserve most, if not all, of the image information content. These models are usually based on linear methods such as correlation, convolution, or Fourier transforms, so they should lend themselves to well-understood analytical methods.

One such model uses "Gabor" functions [41, 80]. A typical 1D Gabor model transforms the image with two convolving basis functions:

- $G_S = g(x) \sin \omega x * I(x)$
- $G_C = g(x) \cos \omega x * I(x)$.

The sine and cosine terms borrow from the Fourier transform approaches. The Gaussian mask $g(x)$ imposes locality.

With some simple image models, such as the random dot stereograms often used to test for stereopsis, the images are made up of discrete bright points. Figure 1-2 shows one such point in the intensity image (I). The Gabor sine (G_S) and cosine (G_C) transforms shown are like local Fourier transform pairs. Indeed, when the phase is measured by taking the inverse tangent of the ratio of the sine and cosine Gabor signals, a very reliable phase measure results as shown in the figure.

From any point in the domain in the vicinity of the point feature, the phase measure accurately measures the distance (in radians) to the location of the bar feature. The energy measure (defined as the sum of the squared sine and cosine measures) is also a useful indicator of the presence of a feature as well as a measure of its contrast.

When two such images are compared, the relative phase difference will indicate lateral misalignments. Since stereo depth is determined by measurements of the lateral shifts of feature positions between two views, differences between left and right Gabor phase measures can be used to measure stereo disparity. Optical flow can also

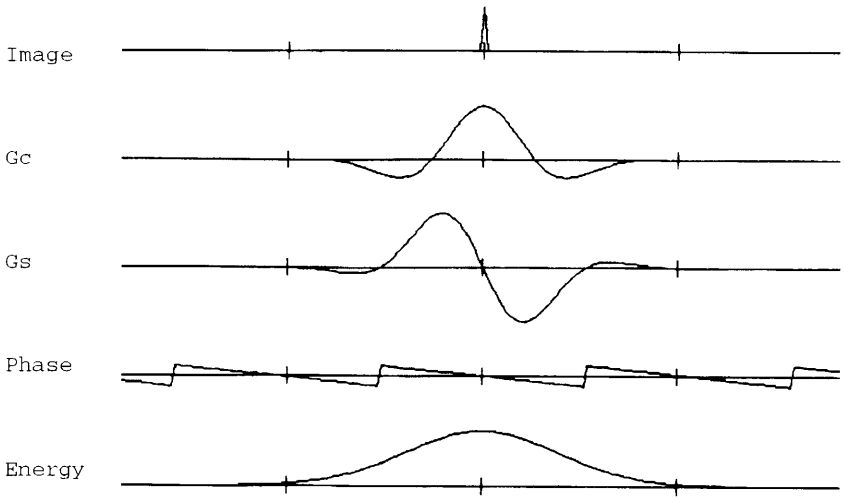

Figure 1-2: Gabor Response to a Bright Bar

be measured using such techniques [22]. With simple images such as the one shown here, these methods work well. With more realistic images, however, where features can no longer be modeled as simple isolated bright points, the "phase" measure of feature position is less straightforward. The linear theoretical underpinnings are also of little help in such situations.

In fact, it is fairly direct to render the phase measure insensitive to feature position. For instance, if the bright point image of Figure 1-2 is replaced with a dark point on a light background, the phase measure becomes insensitive to feature position. The top plot in Figure 1-3 shows such a dark point image input. When convolved with the Gabor basis functions, the two resulting representations G_C and G_S result as shown in the figure. Taking the ratio of these produces a phase measure which is effectively constant. No feature location information is available in this instance. The "energy" measure is equally insensitive to this stimulus. On the other hand, the human vision system is supremely sensitive to dark bars on bright backgrounds.

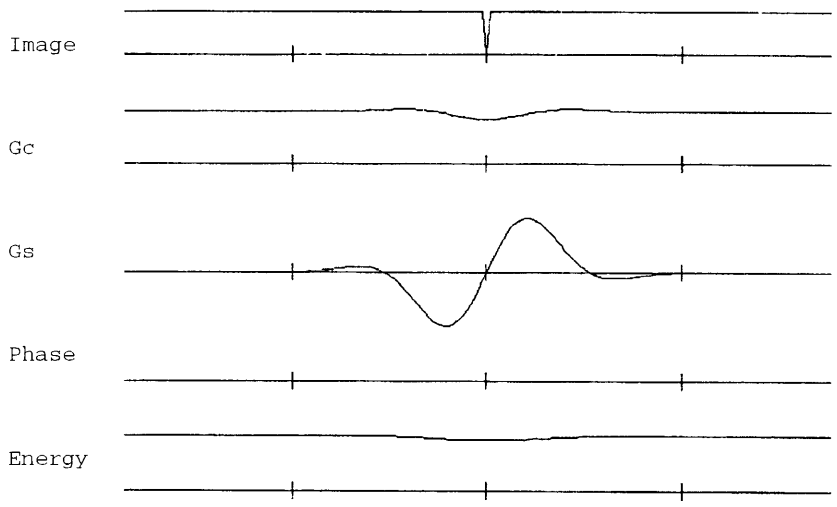

Figure 1-3: Gabor Response to a Dark Bar

Another problem with the Gabor model is found where the width of the Gaussian shrinks to the point where the operators are both local and consistent with biological models. As the support shrinks, the phase measure rapidly degrades in spatial accuracy [80].

A common thread found in these continuous domain approaches is that images are usually treated as signals. The scene content and the optics of imaging are largely not considered in these models. The goal is to match phases or measure energy peaks or find objective function minima. This makes any scene analysis, such as the task of distinguishing a table from a chair, a daunting challenge. What is the effect of illumination on the phase spectra? How does one distinguish a texture edge from a shadow in the energy spectra? This might explain the common preference in the computer vision community for marking edges and being done with the continuous world.

Since there is no obvious interface between the continuous and high-level symbolic

algorithms, they are usually considered incompatible. As such, while these early vision models often demonstrate useful low level properties such as stereo disparity extraction from image pairs, they seldom find application in more complex vision research.

Therefore, while these continuous domain approaches preserve the information content in the images, it is not clear that they are either selective for relevant scene information, or are suited for extension into the symbolic domain where necessary "top-down" knowledge can be introduced into the scene interpretation problem. The edge-finding algorithms result in representations that have the opposite effect. The data is readily incorporated into "top-down" high-level vision algorithms yet much of the useful information content relating to the scene is sacrificed in the continuous to symbolic transformation of marking edges. Both of these approaches have strengths that would be useful if they could be incorporated into a single model.

The principal proposal of this thesis is that a continuous domain model can be devised that is, at its heart, feature based. The approach proposed here is based on contrast edges and measures useful properties of the edges in an image at all points in the image domain. It measures location to precisions limited only by the quality of the imager and the noise in the system. It also measures contrast and edge orientation. From this representation, measures of edge focus and optical flow — the perceived motion of edges in images when objects move in scenes — are easily computed. Most importantly, the model can be integrated into a symbolic image analysis environment since it is, at its heart, an edge based concept. This model is called the Displacement model and the distance, orientation, and contrast measures comprise the Displacement representation.

The Displacement model is based on discrete intensity edges, as shown in the top plot of Figure 1-4. As with the Gabor model, the image is processed using a pair of linear transforms — the Laplacian (I'') and the Gradient ($\mathbf{I}'$) of a Gaussian:

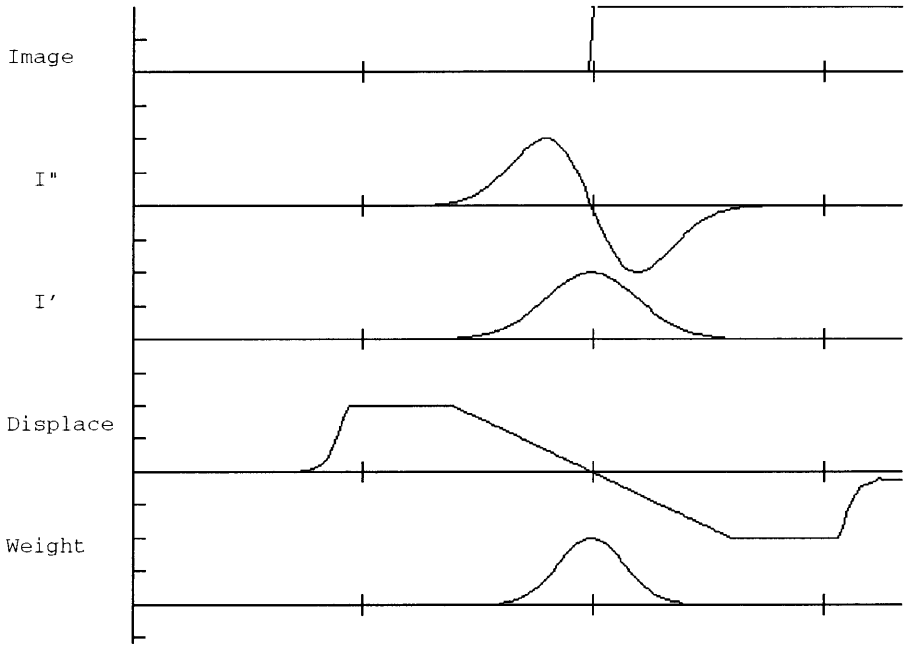

Figure 1-4: Displacement Response to Bright Edge

- $I''(x,y) = \nabla^2 g * I(x,y)$

- $\mathbf{I}'(x,y) = \nabla g * I(x,y)$

One dimensional cross-sections of these representations with a step intensity image input are shown in the second and third plots (I'' and I'). When the ratio of I'' to the magnitude of $\mathbf{I}'$ is taken, the result is a measure of the distance from any point in the image domain to the edge position (shown as "displace" in the figure). The model also provides a contrast measure (labeled "weight" in the figure). This contrast measure is also a measure of the inverse variance of the displacement function. The third component of the full Displacement representation — edge orientation — is not shown. Orientation is a very useful edge property in matching problems [7].

The Displacement model is not linear. It does not lend itself to much of the wealth of analytical methodology available for the statistical and the transform methods favored in other continuous domain schemes. For this reason, the bulk of this thesis is devoted to developing the theory and analytical methods to bridge this gap. On the other hand, once developed, much can be derived from the nonlinear Displacement calculation and its resulting representation.

Since the Displacement measures distance from scene points to intensity edges, subtracting two such image representations would result in the *difference* between respective edges in the two scenes. Figure 1-5 shows how stereo (D) can be calculated by subtraction of the left and right Displacement distance functions (d_e). The image here is of a simple pair of edges. The right image is somewhat brighter — larger I valued — than the left image and is shifted to the left. Note that the Displacement d_e functions associated with the leading and trailing edges of the image box I_m in the left image are shifted to the right of those of the right image. It is this shift that determines stereo disparity. When these are subtracted, the Disparity measure D results at the bottom of the plot.

Optical flow can be derived by temporal differentiation of the Displacement dis-

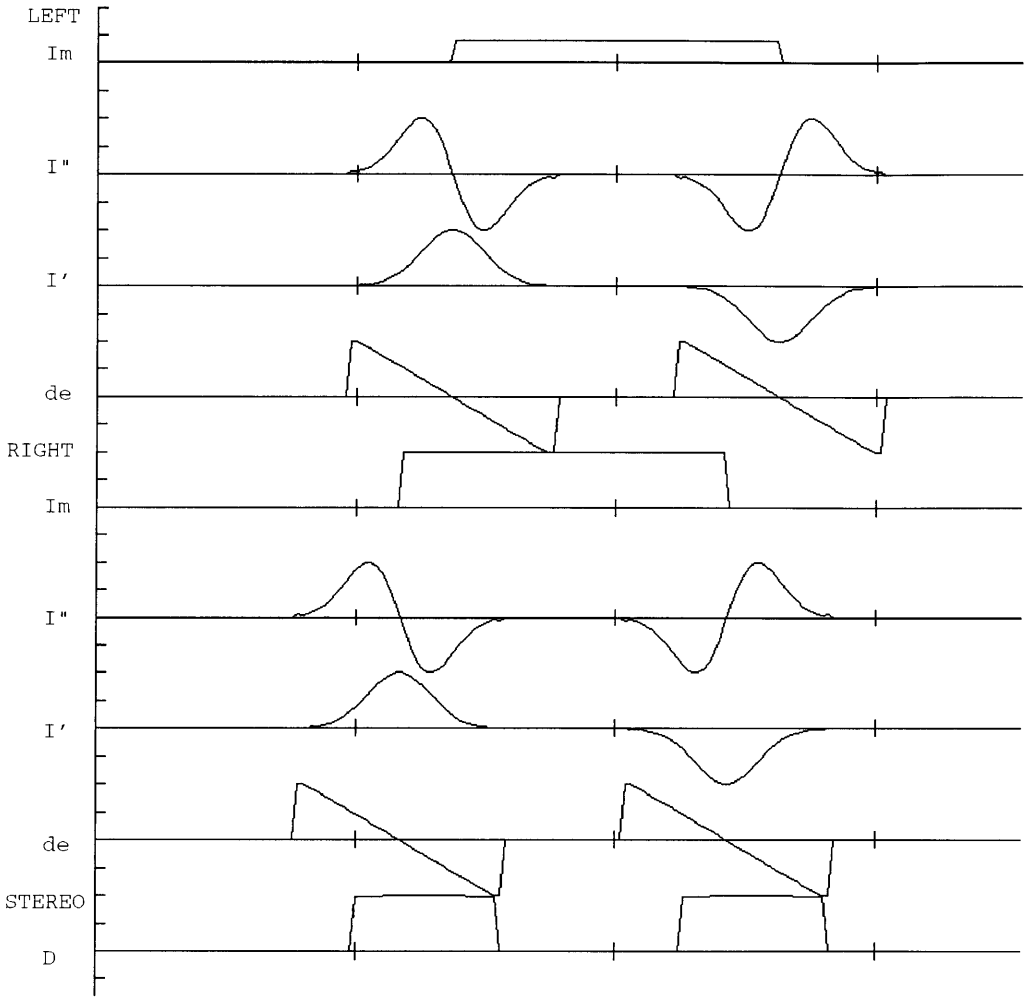

Figure 1-5: Disparity Response to Simple Stimuli

tance measure. Edge focus, ego-motion, alignment and matching can be posed as linear transformations of Displacement representations. Such linear transforms of Displacement representations are called Disparity representations, and all of the above are examples of Disparity representations that will be addressed in this thesis. In addition to the theoretical treatment of a range of application domains, a stereo Disparity algorithm is developed and tested. Since there are successful stereo algorithms in both the symbolic and continuous-domain camps, some comparisons of performance are possible.

It is important to note that the continuous domain, edge finding, and Displacement models are all basically "bottom-up". This means that they are data driven. The models have no knowledge of optics or physics or common-sense. Limited knowledge, of course, can be built-in. For example, Canny exhaustively built knowledge of real scene edges and how they behave into his program [10]. The continuous models, as noted, usually are not amenable to incorporation of such constraints.

A Disparity model can be and, in the case of the stereo design, is modified to reject edge matches known to be inappropriate. This is an example of how knowledge can be incorporated in Disparity algorithm designs, either in subsequent symbolic processing or in the form of internal constraints. At some point in the 3D scene interpretation of the image data,"top-down" processing is almost certainly required, so no purely "bottom-up" approach can hope to provide the complete answer. The Displacement model since it is based on discrete edge measures, can be incorporated into symbolic approaches at any level.

Finally, preserving the integrity of feature measures — preserving acuity — is at the heart of the Displacement/Disparity design. The representation not only provides precise scene measurements, it produces at each step a dense estimate of the variances of the Displacement and Disparity measures as well.

The Displacement/Disparity model provides the richness of image representation of the continuous domain models. As with the continuous models, the Displacement

model is a continuous domain representation at almost all stages. It is also feature based and is comparable to the symbolic methods in that it measures properties of isolated edges. It *does* have pathological properties, as does the human system, but these appear both tractable and consistent with the biological model.

The goal of this thesis is to provide a good computational model for researchers in computer vision. Much of the motivation stems, however, from the biological community and the pioneers of computational models of biological processing of images, particularly Marr [49]. Indeed, just as the Laplacian of Gaussian model of retinal processing proved to be of great utility in the computer vision community, the Displacement/Disparity model is designed keeping in mind the research findings of the natural vision community [50].

Studies of natural vision, whether psychophysical or neurophysiological, can not be used to "prove" that this or any model explains the early cortical processing of visual images. On the other hand, these studies can, and often do, lead theorists to abandon models that are clearly out of line with the experimental evidence. The Displacement/Disparity model does appear to be remarkably consistent with the wealth of early vision research. The model also addresses many open questions about the nature of visual processing — at least at the very earliest cortical stages.

At first glance, the Displacement/Disparity model fits neatly in neither the continuous domain nor symbolic feature camps. However, in spite of the fact that the Displacement model theory is based on discrete intensity step edges, it bears a striking resemblance in both form and function to the continuous domain Gabor models that have received much attention recently. Readers familiar with these models should be struck with the common theme that exists throughout the development — especially the parallel between phase measurement and Displacement distance measures. On the other hand, this model is also based on the Marr-Hildreth Laplacian of Gaussian retinal model. In spite of the very unique nature of this model, Displacement representations measure the distance to the same edges that were marked in the

Marr-Hildreth edge-finder.

The Displacement/Disparity model of early vision is indeed a hybrid. It borrows strengths from the continuous domain models and the symbolic designs. In spite of the amount of analysis needed to develop it, though, its best attribute is that it is so simple to use and understand.

The stereo model developed in this thesis does not provide a continuous domain solution to Marr's $2\frac{1}{2}$D sketch, since it provides disparity estimates only at edge feature points. Where such contrast edges exist, however, the simple computations involved in the stereo Disparity algorithm provide dense and accurate results with extremely few false correspondences.

Chapter 2 provides more background on the continuous domain and edge-based schools of early vision processing. This sets the stage for the approach proposed here that draws heavily from both traditions. Chapters 3 through 5 develop the complete theory. Chapter 3 develops the 1D model that provides the feature distance measure. Chapter 4 adds an edge contrast measure that also serves as an accurate variance model for the distance measure. Finally Chapter 5 generalizes the 1D model into a full 2D model and adds the feature orientation measure to complete the design. Throughout this development, examples are given of useful Disparity representation designs.

Chapter 6 describes the incorporation of the model into a working stereo disparity algorithm. Stereo, being a matching problem, can utilize constraints to narrow the large number of ambiguous matches to a unique match. Some of these are incorporated in the algorithm. This is then tested on a range of real images. Chapter 7 discusses some of the research in early natural vision, especially relating to primates.

Chapter 2

Approaches to Early Vision Algorithms

There is a significant body of research on "early vision" representations. Stereo disparity, optical flow, 2D matching, as well as such simple measures as focus, orientation and location can all be considered early vision since they all can contribute to computational approaches to 3D recognition. Some approaches are highly symbolic and usually transform the image into a discrete set of "edge" features. Others attempt to retain the continuous nature of the intensity image through a series of linear or nonlinear transforms to arrive at these representations.

This chapter reviews two fundamentally different approaches taken toward computational early vision. One converts image data into discrete symbolic form almost immediately after sensing and all subsequent processing uses this symbolic representation. The other approach is to maintain the image in a continuous domain format and process it much like a signal. Symbolic or logical analysis is usually incompatible with such approaches.

As discussed in the Introduction, these two approaches have distinct attributes which appeal to vision researchers. The continuous domain models are rich representations which preserve scene related information and can borrow from the rich

theoretical background of linear analysis. The symbolic methods permit ready incorporation into high-level vision algorithms. Both of these approaches have some shortcomings, however. The symbolic models tend to render extremely sparse representations which have lost much of the 3D scene related information of the 2D images.

The continuous domain models are limited by the difficulty in handling common image properties such as This thesis proposes a model which borrows attributes from both of these approaches in an attempt to overcome their respective shortcomings. This chapter examines the properties of some of these early vision approaches.

It would be impossible to review the breadth of the research in early vision algorithms. The two approaches above alone span at least a dozen or more unique approaches worthy of mention. Of these, this chapter discusses a few methods that have found acceptance in recent years. In the symbolic world, oriented filters such as proposed by Canny and Deriche [10, 15] and symmetric filters as proposed by Marr and Hildreth [50] are used to generate highly symbolic (bit-mapped) image representations. The continuous methods usually employ some form of linear process such as convolution, correlation, or Fourier transform. Some recent approaches, such as Gabor models, employ enhancements to render the filters local and spatially sensitive. These examples are discussed in this chapter.

2.1 Symbolic Approaches — Edge Finders

A feature can be defined as some collection of image events that correspond to scene properties in some invariant manner. For instance, a face could be called a feature in a scene, since the image of the face is often viewed front-on and upright, the individual facial features are regularly arranged, and the illumination effects are generally minor.

When faces are viewed upside down, or are distorted or the illumination is unusual (such as when a flashlight is held beneath the chin) the effects can be quite startling

and identification is extremely difficult [11, 79]. There is reason to believe that faces may have special "hardware" devoted to their recognition and classification in the brain [17, 19, 31].

With more mundane recognition tasks, however, such as recognizing tables or chairs, there are too many highly variant characteristics, such as object orientation, illumination direction and feature arrangement. Searching for some match to a fixed image template is a poor methodology for identifying tables. On the other hand, there are symbolic means to describe what a table is from its component parts; legs and top. From these descriptions and the physical properties of the component parts, predictions can be made about the appearance of tables in images.

To describe the component parts of objects, corners and occluding edges of the objects produce predictable 2D images when the illumination and the 3D pose are known. Even when the illumination is unknown, the locations of the edges and corners will be predictable and the can be detected by a sudden step discontinuity in the image intensity. Texture and shadows produce similar effects, although while texture is highly insensitive to illumination distribution, shadow edges are highly dependent on the direction of illumination.

Ultimately, the image feature most commonly found to correspond to salient object properties are these contrast edges. Points in the image where the intensity profile undergoes a stepwise discontinuity are called edges.

Much of the symbolic analysis of imagery is based on representations generated by algorithms that mark image edges. There exists a wealth of such methods, ranging from simple thresholding schemes to complex optimization heuristics. This section, while it is not a survey, discusses some of the more popular approaches used today [1].

[1] For a good survey of edge-finding methodologies, see Horn [36], Gonzalez and Wintz [25], and Barr and Feigenbaum [4].

2.1.1 Gradient Models

Where an edge exists in an image, the intensity can be modeled as a stepwise discontinuity. The simplest possible model of such an image is the single vertical edge of some contrast α and some position x_o:

$$I_m(x, y) = \alpha u(x - x_o)$$

where $u(x)$ is the unit step function.

As mentioned, this edge could be detected by thresholding the image, say by $\alpha/2$ and then marking the transition points where the image crosses the threshold.

Thresholding to create binary images is a very tricky business, however, since shading will introduce "edges" into the representation and other gradual image gradients will tend to defy simple threshold models [36]. When the gradient of the image is taken, however, these first-order effects are suppressed; smooth image gradients typically introduced by surface curvature and scene illumination are greatly attenuated compared with the edge features:

$$\mathbf{I}'_m(x, y) = \nabla I_m(x, y).$$

With the isolated edge image defined above, the gradient becomes

$$\mathbf{I}'_m(x, y) = \alpha \delta_k(x) \hat{\mathbf{i}}$$

where $\delta_k(x)$ is the Kronecker delta function. This gradient image, of course, is pathological and impossible to realize in physical imaging systems. Most images are not continuous domain, but instead sampled domain representations. Thus the gradient operator is implemented using local operators on the rectangular image tessellation. Examples of such operators are the Roberts operator, which uses the smallest possible

support of two pixels, and the Sobel operator which, to retain the image registration, uses a larger 3x3 support [4]. With these finite support operators, a reasonable approximation to the gradient image results.

One major problem, however, is that taking derivatives enhances the noise content of images, and sampling introduces artifacts known as "aliasing" where scene feature content of wavelength less than the twice the pixel spacing are transformed into false low frequency image events. As a consequence of these problems, a filtering process must take place prior to sampling [84]. One such model of filtering is the use of a Gaussian convolution. Thus, a gradient image representation is often produced by convolving an image with a gradient of Gaussian mask pair.

The final step of a gradient edge-finder usually involves locating the peaks in the gradient image corresponding to image contrast steps. Once detected, the edges are marked.

2.1.2 Laplacian Models

The gradient edge-finder operates by locating the peaks in the gradient (or gradient of Gaussian) image. Instead of taking the peaks, one could simply take the spatial derivative of the gradient representation and find the edges wherever this derivative is zero (which will occur at the maxima and minima of the gradient). The Laplacian operator serves this purpose [25].

$$I''_m(x,y) = \left(\frac{\partial}{\partial^2 x} + \frac{\partial}{\partial^2 y}\right) I_m(x,y)$$

A band-limited version of this, just as with the above gradient filter, is the Laplacian of Gaussian operator.

Marr and Hildreth [50] proposed the Laplacian of Gaussian as model for the center-surround retinal receptive field. They suggest that images might be processed by this

operator and edges be extracted at the zero-crossings. Hildreth also demonstrated that the tracking of zero-crossings across multiple scales can be used to disambiguate features [34, 35]. This concept has been successfully used in stereo algorithms [29]. Since such a procedure will also mark gradient minima, it has been suggested that the samples take place only where the gradient is large [30].

The Laplacian of Gaussian is a widely accepted retinal model in the biological research community as well, although some researchers prefer a Difference of Gaussians (DOG) model since precise receptive field fits using this parametric model more closely reproduce the ganglion cell responses [94].

This Laplacian of Gaussian has found much favor in the vision research community, but it does have its drawbacks. One already mentioned is that the zero-crossings tend to mark non edge events. Another criticism is that the zero-crossing contour will tend to deviate from the actual edge contour near sharp corners. Some researchers have proposed much more complex edge-finders to eliminate some of these problems. Two of these will be discussed in the next section.

2.1.3 Advanced Edge-finders

Canny cast the edge-finding problem as an optimal estimation problem where features are to be matched based on an objective measure that penalizes false edge markings, missing edges, and location errors. He developed an algorithm that uses optimized oriented filters [10].

This is one of the most widely used edge-finders in vision research. It is quite good at not marking edges falsely, and it does tend to reproduce sharp corners, since the filters are oriented to the edges.

In the final analysis, however, the edge operators have a surprising similarity to the gradient of Gaussian. Deriche [15] derived a generalization of Canny's edge detector,

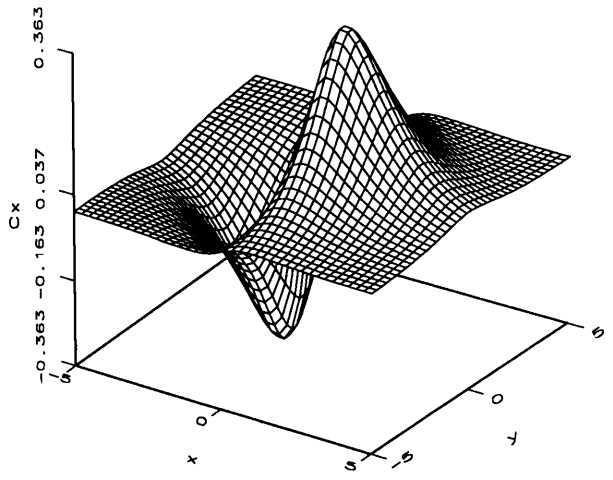

Figure 2-1: Deriche Optimized Canny Edge-finder (x component)

instead of being edge oriented, could be implemented using an orthogonal filter pair:

$$C(x,y) = -\beta \left[x(\sigma_b|y| + \sigma_b^2)\hat{\mathbf{i}} + y(\sigma_b|x| + \sigma_b^2)\hat{\mathbf{j}} \right] e^{-\frac{|x|+|y|}{\sigma}}$$

While this may not *look* like a gradient, its form is remarkably close to the simple gradient of Gaussian form of the simple edge-finders above (see Figures 2-1 and 2-2).

Although similarity in form does not connote similarity in function, the difference between these basic operators is very subtle. Both the DOG and Canny/Deriche models are exhaustively optimized alternatives to their extremely simple antecedents. The gradient of Gaussian representation and Laplacian of Gaussian may not, by themselves, render optimal edge maps or retinal (or cortical) receptive fields, but they come remarkably close. They are also much simpler to deal with than the heavily optimized versions in theory and practice.

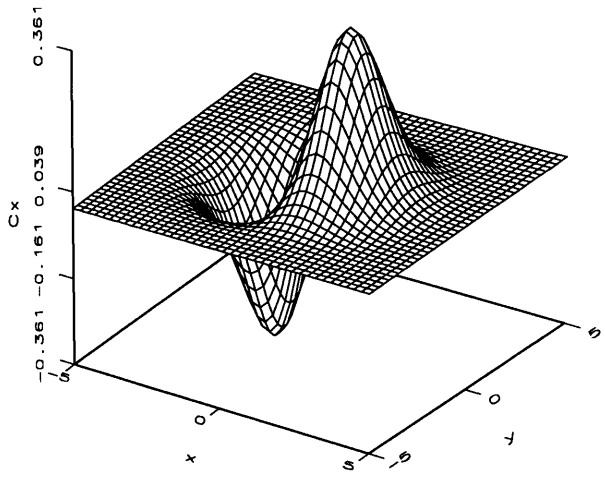

Figure 2-2: 2D Gradient (x component)

2.2 Continuous Models

The two basic categories of algorithms studied in this thesis are called symbolic and continuous domain. The symbolic models typically mark image features and operate on the sparse representation that this process generates. The term "continuous" domain is used — despite the fact that the domain is sampled — to distinguish the sparse symbolic representations from representations that are dense. Continuous domain representations provide measures at every image domain point.

As such, the image itself is a continuous domain representation. So too is the Laplacian and gradient of Gaussian convolutions of the image discussed in the symbolic models. The edge samples, however, transform the representations from continuous to symbolic. It is at this transformation that much is lost in representational richness.

The functions themselves involved in these continuous domain representations are often not continuous. Many involve nonlinear transformations. Most, however, are

strongly motivated by linear analysis. Some of the most prominent continuous domain models are discussed in this section.

2.2.1 Correlation, Convolution, and Matched Filters

One of the common threads in continuous domain analysis of images is that the images are not treated as representations of 3D scenes so much as they are considered two dimensional signals or random variables. When the signal metaphor is used, the models used tend to draw heavily from linear systems theory. When the image is treated as a random variable, often the models use probability theory as an inspiration. High level vision algorithms often can make use of such probabilistic models, especially when Bayesian estimation is used.

An example of the use of these metaphors is found in the processing of images using correlation or convolution. Correlation and convolution are essentially identical processes. The principal difference between them is that convolution of images is used when signal processing methods are being applied and correlation when statistical methods are used [16, 73]. Another approach is to apply least square methods for template matching. Once again, the resulting approach reduces to a convolution (or correlation) of the image with a template. This is also often treated as a matched filter design problem. All of these are variations of a common theme; the image is matched to a shifted prototype template. This matching involves a simple point-wise scalar product of the image $I_m(x,y)$ and the template $T(x,y)$:

$$C(x,y) = \sum_i \sum_j I_m(i,j)T(i-x, j-y).$$

Correlation can be cast in a normalized form whereby the above measure is scaled by the inverse variance of the image and the template. Although this may appear to be a simple scaling, when only the portion of the image under the (usually much smaller) template is used in the normalization step, the normalization can render

the correlation measure invariant to illumination level — a handy feature, in spite of the fact that this normalization has little or nothing to do with the statistical underpinnings of the approach.

Other variations on the theme can be found, such as when the gradient of a smoothed image is incorporated in the above normalized correlation model through using the scalar product of the image and template gradient vectors [7]. Others have correlated the Laplacian convolved images as well as DOG-like image models [8, 9]. Finally, most, if not all, "neural-net" pattern matching models can be cast as correlations with learned or programmed templates — embedded as weights in the model [7, 58].

All of these methods have demonstrated some substantial utility in many simple vision tasks such as face recognition, 2D alignment, and local stereo reconstruction. When the images undergo any transformation more complex than translation and amplitude scaling, however, the approach is of limited utility.

Simple image transforms such as axial rotations and scaling cannot easily be dealt with in correlation models. Normal changes in scene properties are even more problematic. Examples of these are occlusion effects, changes in scene illumination, distortions due to rotations in depth and minor object deformations.

Attempts have been made to render correlations local and thereby allow processing of at least some of these real image properties, but these modifications, result in a less robust estimator for complex features.

Reducing the support of the correlation — that is, making the template smaller — will allow the correlation to tolerate more local deformation. At the limit, however, the template will cover only a few pixels, and the algorithm takes on the character of the edge-finders discussed earlier. After all, the edge finders are matched filters using one of the smallest templates possible — the edge template.

As mentioned above, the correlation models basically operate by shifting the template $T(x, y)$ over the entire image and then looking for the offset (x_o, y_o) that results

in the maximum correlation measure. Just as with the gradient model of edge-finding, it would be preferable if a less brute-force peak search strategy could be employed. Fourier models hold such promise.

2.2.2 Fourier Models

The Fourier transform of an image produces an image mapping with unique characteristics, some of which are applicable to the processing of image data. The Fourier transform of the image $f(x, y)$ is:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\alpha, \beta) e^{-i(u\alpha + v\beta)} d\alpha d\beta$$

The correlation/convolution methods discussed in the preceding section require $O(N^2)$ computations per pixel for a N times N pixel template [2]. One method of making the matching more efficient for larger templates is to take the Fourier transform of the image and the template. In the Fourier domain, spatial convolution becomes multiplication. A Fast Fourier Transform (FFT) can run in $O(N \ln(N))$ per pixel, so the correlation can be achieved by taking the FFT, multiplying the transforms ($O(1)$ per pixel) and taking the inverse FFT (also $O(N \ln(N))$). This is an example of how the Fourier transform methods can be used to simplify image analysis problems. Fourier methods also can be solved readily using optical computing techniques.

Another approach is to use the transform directly to extract image properties, especially spatial relationships. The complex transform is often broken into two component parts. The real and imaginary components of the transform can be used to determine the amplitude and phase of the frequency components:

[2] There are methods to reduce this complexity using moments of the template intensity distribution.

$$A(u,v) = F(u,v)F^*(u,v) = (\Re F(u,v))^2 + (\Im F(u,v))^2$$
$$\Phi(u,v) = \arg(F(u,v)) = \arctan(\frac{\Im F(u,v)}{\Re F(u,v)})$$

When two scenes are reconstructed using scrambled phase and amplitude spectra, humans tend to easily recognize the images with the correct phase information whereas amplitude spectra appear to be less important [7]. Although the amplitude information is invariant with translation, the phase information can help align templates to images.

Problems arise, however, because the Fourier transform is a global measure of the entire image. Scene information of *all* the objects in a scene will affect the phase spectra in non-obvious ways. Unless the template and image are simple copies of each other, that would render the problem basically moot, then this is a major drawback of the Fourier phase alignment concept.

Other problems with the Fourier approach parallel the problems mentioned with the correlation models. Illumination, distortion, rotation, and scale are just a few difficult issues, although some clever approaches can and have been used to address these issues. The problem of locality is still arguably the biggest handicap of the Fourier approach. Gabor models are designed partly to address this issue.

2.2.3 Gabor Models

The Gabor model is often cited as a useful model of cortical RF's as well as being a practical computational device [41, 80, 94]. It can be represented in any number of ways, but the usual approach is that the image $I(x)$ is convolved with a complex Gabor function. An equivalent result is arrived at by convolving the image with even and odd Gabor functions to produce a representation pair ("cosine" and "sine") : [3]

[3]Note that this is a 1D version of the model, that simplifies the analysis without loss of generality.

$$g(a) = \frac{1}{\sqrt{2\pi}\sigma_b} e^{\frac{-a^2}{2\sigma_b^2}}$$

$$g_c(x) = g(x)\cos(\omega_c x)$$
$$g_s(x) = g(x)\sin(\omega_c x)$$

$$I_c(x) = g_c * I(x)$$
$$I_s(x) = g_s * I(x)$$

As discussed in the previous chapter, Gabor models are designed to be local operators, by virtue of the Gaussian mask, while preserving the phase detection properties of the Fourier model. One of the problems associated with Gabor models is that the phase sensitivity decreases with decreasing support [80].

Another interesting property of this filter pair is that, unlike the Fourier series, the cosine term $g_c(x)$, has a nonzero response to mean intensity, i.e. if $\overline{I(x)} \neq 0$ then $\overline{I_c(x)} \neq 0$ regardless of the spatial frequency ω_c or filter width σ_b. This agrees with the psychophysical and neurophysiological data, but is often overlooked in computational models.

This following sections examine the problems associated with this common oversight in both modeling biological systems using Gabors in the application of computational models, such as a model used by Sanger [80] and as exemplified by a recent model of Ohzawa, DeAngelis and Freeman [63] — although almost any Gabor model application could be cited.

2.2.4 Analysis of a Computational Model

Since Gabors are useful for both biological and computational models, it is also instructive to evaluate the impact of mean Gabor response with some commonly used model assumptions. For the sake of brevity, this section will examine one of the most common analysis methods now used, to determine if the mean intensity response of $g_c(x)$ undermines this analysis for realistic image features.

As with the Fourier model, Gabor representations can be reformulated into a phase/magnitude format that is particularly appealing for analysis:

$$\begin{aligned} ||I_g(x)|| &= (I_c^2(x) + I_s^2(x))^{0.5} \\ \phi_g(x) &= \arctan\left(\frac{I_c(x)}{I_s(x)}\right) \end{aligned}$$

Many, if not most, recent practitioners use the phase information in the Gabor representation for image analysis. This is because it is more sensitive to feature location, and thereby less prone to errors due to noise.

If one examines the phase information associated with a light bar on a dark background (Figure 2-3) it is clear that the phase information can be used to encode the feature location. When isolated positive "dot" features, similar to those found in random dot stereograms, are used in this algorithm using Gabors, the phase information produces correct stereo interpretation.

If, however, more complex inputs are used, such as real images, the approach seems to fail. One might appreciate why when a contrast edge feature (a sudden change in intensity) is used instead of isolated dots. Most important features in real scenes can be modeled as intensity steps, as discussed earlier. Figure 2-4 shows the response of one eye to a step intensity input. Note the ϕ_q response suddenly disappears past the location of the step edge.

Stereo can be determined by taking the difference between left and right phase

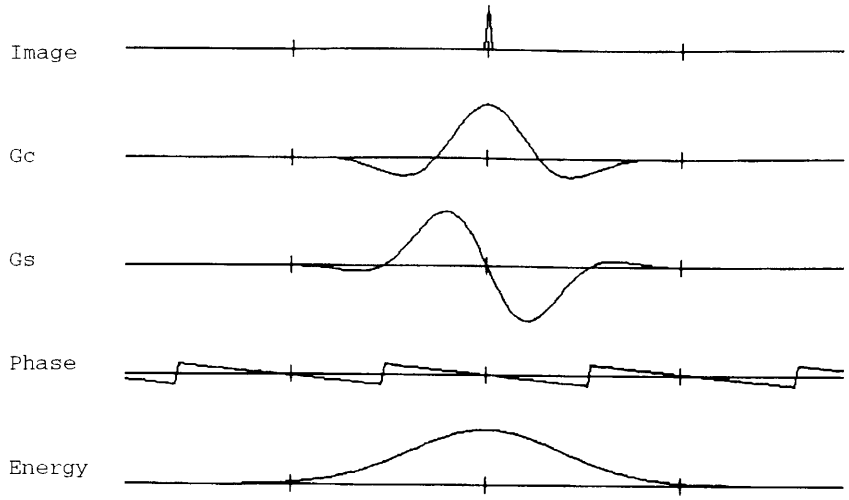

Figure 2-3: Gabor Response to a Bright Bar

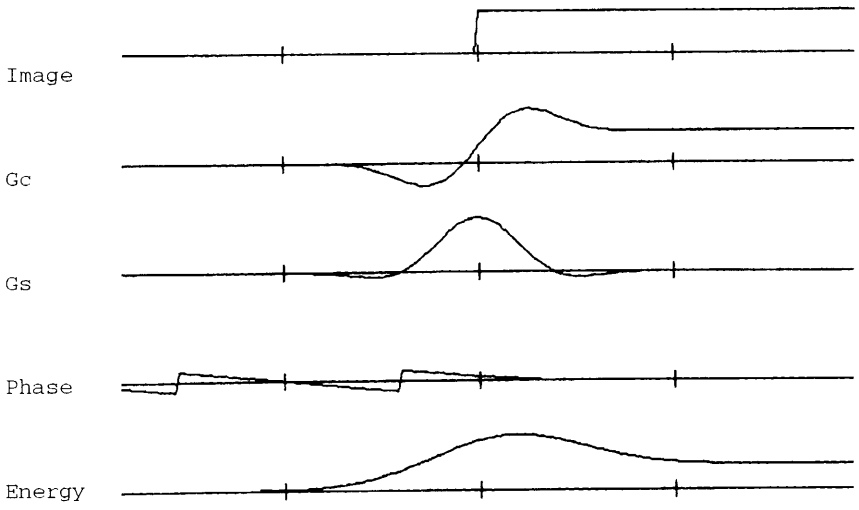

Figure 2-4: Gabor Response to Bright Edge

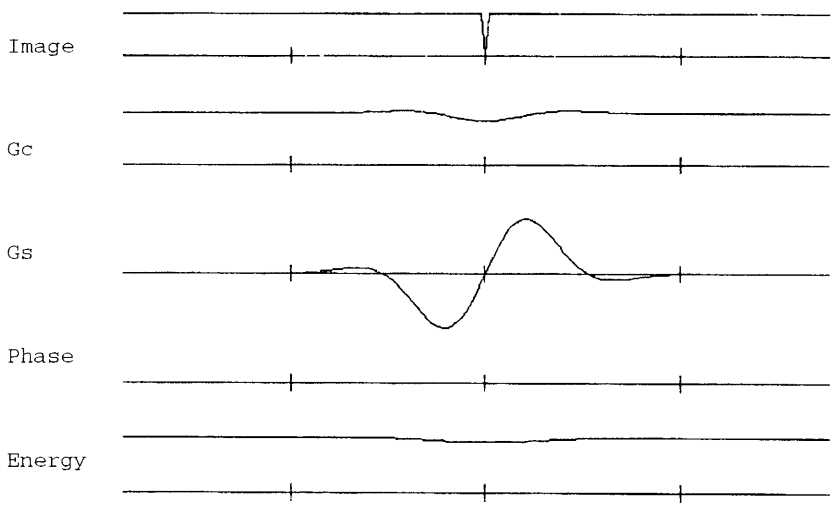

Figure 2-5: Gabor Response to Dark Bar

representations. On the right hand side of the step input, however, virtually all phase information is lost. The same results are obtained if one chooses to measure the phase of the cross-correlated left and right Gabors. The reason is simple — the I_c response to the mean intensity component of the step greatly diminishes the sensitivity of the phase measurement.

In spite of the fact that sine wave inputs are often used in formal analysis of these models, the fact that these inputs too must be offset by the peak amplitude of the grating, assures a considerable mean intensity influence. Experiments bear out that even with grating frequencies near ω_c the Gabor filters are influenced substantially by this factor.

Finally, Figure 2-5 shows the results for a negative bar input, the previously defined $I^-(x)$. Here virtually no phase information (*or* amplitude information) remains in the Gabor representation, regardless of calculation methods. Ironically, human subjects perform best with such features on acuity tests [88].

This demonstrates that the computational model used by Sanger and others not only loses resolution with diminishing support but also becomes sensitive to mean intensity. In a biological example a little more analysis is needed to determine the effect of this mean intensity sensitivity of the models.

2.2.5 Response of Gabors to Mean Intensity

Since the image function $I(x)$ is restricted to be positive, a nontrivial observation for all reasonable images is that $\overline{I(x)} \gg 0$. Since it is the intent of this section to explore whatever impact this might have on the analysis of the Gabor representation, a quick Fourier analysis of the cosine Gabor is needed.

$$\begin{aligned} G_c(\omega) = \mathcal{F}\{g_c(x)\} &= \mathcal{F}\{g(x)\} * \mathcal{F}\{cos(\omega_c x)\} \\ &= 0.5 \left(e^{\frac{-(\omega-\omega_c)^2\sigma_b^2}{2}} + e^{\frac{-(\omega+\omega_c)^2\sigma_b^2}{2}} \right) \end{aligned}$$

When evaluated at $\omega = 0$ the g_c response is:

$$G_c(0) = e^{\frac{-\omega_c^2\sigma_b^2}{2}}$$

Thus if $\overline{I(x)} = c$, the response of $I_c(x)$ to this will be:

$$\overline{I_c(x)} = ce^{\frac{-\omega_c^2\sigma_b^2}{2}}$$

2.2.6 Analysis of a Biological Model

Ohzawa DeAneglis and Freeman [63] used Gabor representations to model cell responses in the visual cortex. They proposed that this binocular cell could be modeled using a simple Gabor representations. A central conclusion in the paper is that the cell data supports the model. To this end, the authors plotted the responses of their

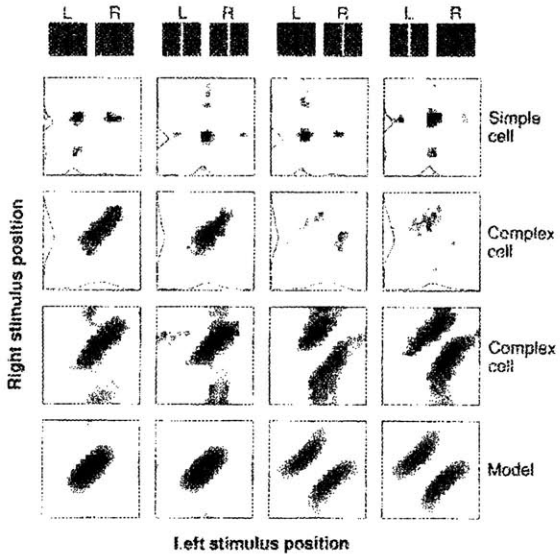

Figure 2-6: Figure From the Research Paper

model to various bar-like inputs. Figure 3 of their paper is reproduced in Figure 2-6. The intensity inputs to the left and right eyes are:[4]

$$\begin{aligned} I^{+}(x) &= 1.0 + \delta_k(x) \\ I^{-}(x) &= 1.0 - \delta_k(x) \end{aligned}$$

The $\pm$ choice depends on whether a dark bar or a light bar is being modeled. The mean intensity used here (1.0) is the *minimum* value possible to maintain $I(x) > 0$ for all x. When convolved by the Gabor filters, the authors' model (as described in the paper and the figure) is (using subscript l and r notations for the respective eye signals):

$$R(x_l, x_r) = \left[(g_c * I_l(x_l) + g_c * I_r(x_r))^2 + (g_s * I_l(x_l) + g_s * I_r(x_r))^2\right]^{0.5}$$

[4]The Kronecker delta function $\delta_k(x)$ is defined as equal to one if $x = 0$, and zero otherwise

Unfortunately, in the notes on the figure, it is clear that the authors modeled the intensity as simply $I(x) = \pm\delta_k(x)$ without the constant factor that is necessary to keep the intensity positive.

Using the above calculation for the cosine Gabor response to the constant component of the input intensity, the following corrected response results;

$$R'(x_l, x_r) = \left[(2e^{\frac{-\omega_c^2\sigma_b^2}{2}} + g_c(x_l) + g_c(x_r))^2 + (g_s(x_l) + g_s(x_r))^2\right]^{0.5}$$

As compared to that used in the paper; [5]

$$R(x_l, x_r) = \left[(g_c(x_l) + g_c(x_r))^2 + (g_s(x_l) + g_s(x_r))^2\right]^{0.5}$$

Is this important? It is fairly easy to check. Since the validity of their model rests on the visual comparison of cell responses to the graphic responses of the figure, it is possible to simply re-plot the revised response using the correct compensation for mean intensity $R'(x_l, x_r)$.

There are really only two important cases; matching bars and mismatching bars. Figure 2-7 shows the recalculation of their figure using their zero mean $R(x_l, x_r)$ assumption. With some minor differences it agrees with that of their figure. Figure 2-8 revises for the corrected model — $R'(x_l, x_r)$. Note that with the *minimum* possible background assumptions, the model no longer resembles the data.

Likewise, the mismatched feature model as originally calculated ($R(x_l, x_r)$) shown in Figure 2-9 is completely dissimilar to the revised, and presumably correct, model $R'(x_l, x_r)$ shown in Figure 2-10.

It would appear that this minimal mean input response of the cosine Gabor is of pivotal importance in evaluating the performance of the proposed model.

[5]The addition operations for the g_c and g_s terms assume I^+ form inputs (The second column in their figure). Subtraction is used for I^- left or right inputs.

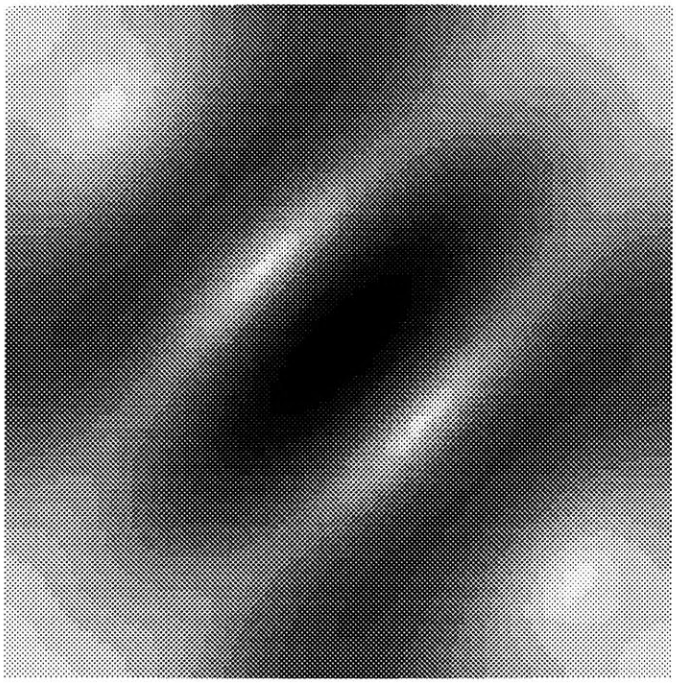

Figure 2-7: Freeman and Ohzawa's Matched Bar Response

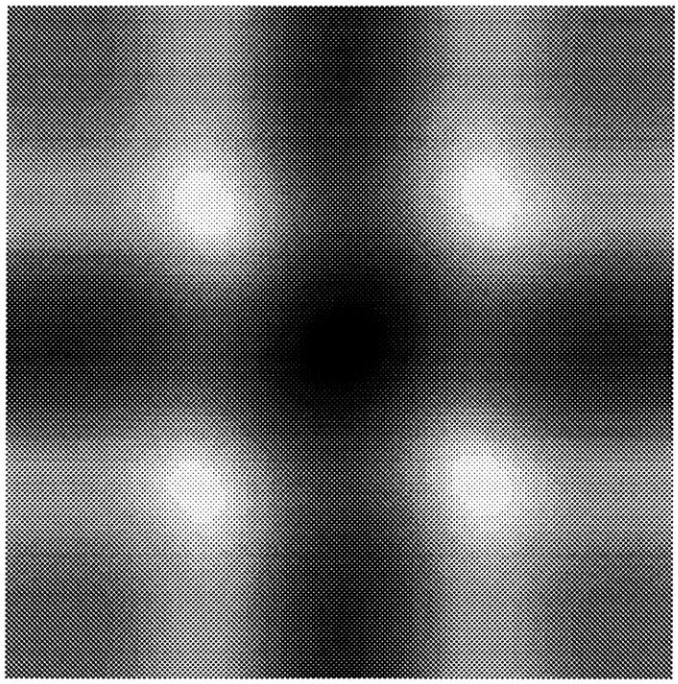

Figure 2-8: Actual Matched Bar Response

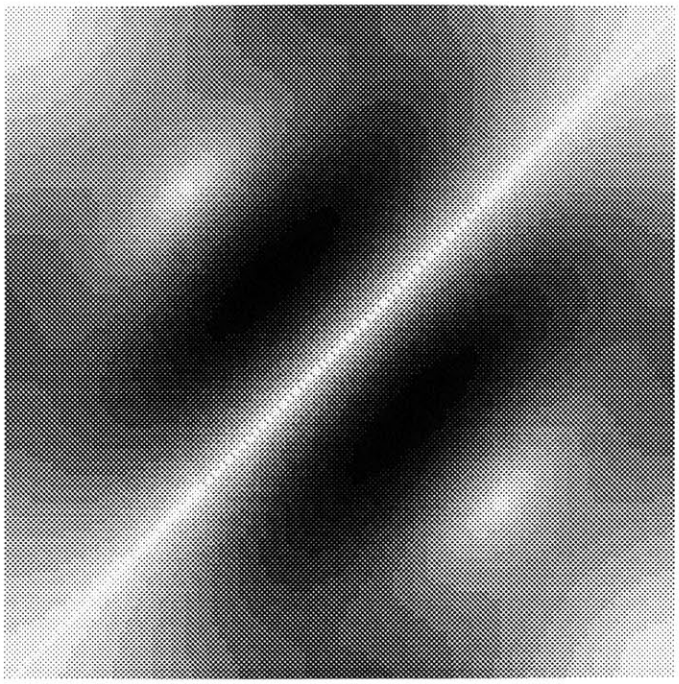

Figure 2-9: Freeman and Ohzawa's Mis-matched Bar Response

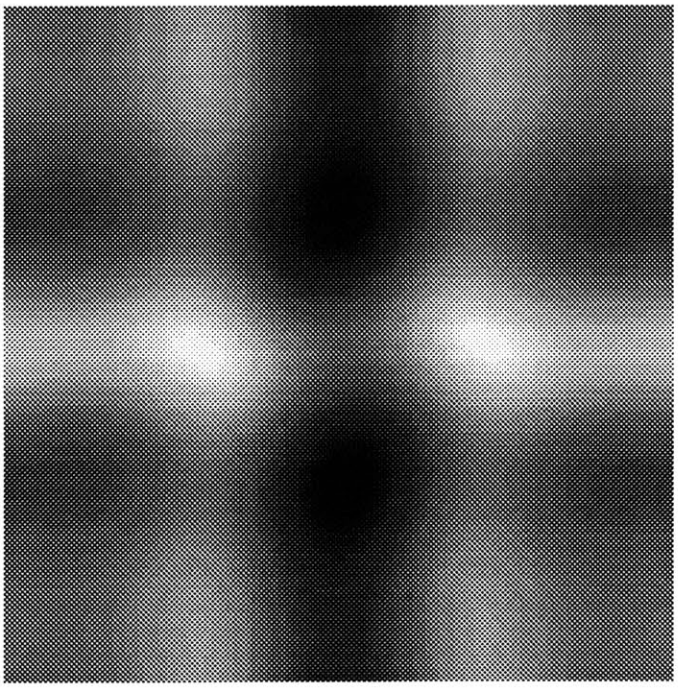

Figure 2-10: Actual Mis-matched Bar Response

2.3 Summary

This chapter has examined symbolic and continuous domain early vision algorithms. The symbolic models usually involve convolving the image with functions similar to the Laplacian or gradient of Gaussian and then searching in these transformed images for peaks or zero-crossings.

Of the continuous domain models, correlation is continuous, but it does not lend itself to problems such as affine transforms. Fourier methods have been devised for such transforms, but these are not suitable for local deformations and have unbounded support.

Gabor representations have been widely used both in the modeling of biological systems, and the synthesis of computational vision models. Gabor models, despite some acceptance, do not work well with local support and are very sensitive to average intensity effects.

Chapter 3

1D Displacement and Disparity Models

The previous two chapters discussed the distinction between early vision model domains which are continuous and those which are discrete. The discrete models treat contrast edges, or collections of edges, as features at specific points in the image. The sparseness of this representation is considered a strength by those who use information-theoretic data-reduction arguments [10]. Other researchers often seek a dense representation. An example is with stereo algorithms which seek to assign a depth estimate to every pixel in the $2\frac{1}{2}$D image proposed by Marr [49, 80].

Sometimes symbolic models are used as a first step toward generating these dense representations at subsequent stages of processing [27], but it appears unlikely that biological systems convert from continuous domain representations, such as are found at the retinal ganglion cell level, into sparse discrete symbolic representation schemes at the earliest stages of processing. Since stereo disparity selective cells are found in areas associated with the first stages of cortical processing (V1) in primates, it stands to reason that some non-symbolic process is involved. Proponents of continuous domain early vision models, such as Gabors, tend to focus on stereo vision as their problem domain.

Continuous domain approaches, such as the Gabor models discussed in the preceding chapter, have some serious drawbacks. The attempt to make the phase measurement capability of the Fourier model local through masking the sine/cosine series with a Gaussian works reasonably well — so long as the width of the Gaussian contains at least a few full cycles. Experiments show that this model has problems in areas of low contrast, high disparity gradient or with significant mean intensity. When the phase representations of the left and right eye are subtracted, a disparity measure can be derived. There are problems, however, with aliasing, and false correspondences between mismatched contrast sign edges. The main criticism, however, rests with the premise that any bottom-up stereo algorithm such as this can be used to provide dense depth information.

The stereo problem can be construed as a correspondence problem between pixels of two images. A central thesis of this work is that in image regions where there is little or no contrast, there is little or no basis for determining correspondences and thereby estimating 3D form. Conversely, where contrast events in an image are adequate to determine unique correspondences, these events can be modeled as discrete edge features. Virtually all images, except perhaps random-dot stereograms, have large regions of little or no texture. In those regions there exists no basis for a bottom-up algorithm to infer correspondences. Since correspondences determine 3D geometry in stereo, stereo is of no utility in such featureless regions. Importantly, an infinite number of 3D geometries can and will produce identical stereo images when such featureless regions exist. Stereo correspondences alone cannot disambiguate this one to many mapping. Some other approaches must be used that incorporate knowledge of scene optics, physics, geometry, and other "common-sense" knowledge.

It is, for instance, appropriate to restrict the interpretation to reasonable geometric constraints. Lakes and buildings in aerial imagery should not tilt. Shape from shading can help give form to areas with smooth image gradients. These are examples of (geometrically constrained) algorithms and heuristics.

Bottom-up algorithms *can* provide reasonably good correspondence estimates where sufficient texture exists. Humans can extract such correspondences with ease and without any geometric cues [42, 43]. So some *feature* driven *continuous* domain process does appear to be at work in providing stereo depth at a surprisingly early stage of cortical processing. This processing seems importantly unlike both the symbolic models and the continuous domain models in ways that suggest a new approach is needed — one that uses a non-symbolic continuous domain representation yet treats contrast features as discrete events.

This chapter introduces the concepts involved in one such model; the Displacement and Disparity model. This model of early vision computation is unlike the other methods in that the representations are valid over the whole image domain while it is based on an image model of isolated contrast edges. This model, therefore, borrows from both approaches.

One important aspect of this model that is also unlike its symbolic and continuous domain predecessors, is that the Displacement and Disparity computations retain all the information needed to measure feature position to the best precision possible given noisy signals. This feature is not appended on to the model as an afterthought, but is an essential property of the design. At all stages of the image transformations, not only are dense edge position, contrast, and orientation measures preserved, but variance estimates on the position measures are included in the representation at all points of the computation to insure the integrity of subsequent computations.

3.1 Displacement and Disparity Concepts

At any image point P, the Displacement representation is defined as a measure of:

- the distance from P to the nearest contrast edge,
- the contrast of the edge, and

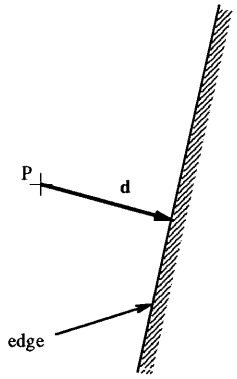

Figure 3-1: Displacement Representation **d**

- the orientation of the edge.

In the 2D Displacement model, the distance and orientation are encoded at any point P as a vector from the point to the edge contour and normal to that contour as illustrated in Figure 3-1.

A Disparity representation is defined simply as any representation derived by taking linear transforms of Displacement representations. When an object moves, its motion is usually detected by the motion of contrast edges in the image. This is called optical flow. Temporal differentiation of the Displacement measure **d** will provide an optical flow Disparity vector representation. Other examples of Disparity representations are possible to model:

- Stereo disparity is the difference between corresponding feature positions in two views of a scene. Subtraction of Displacement representations will generate Stereo Disparity measures to allow depth measurement.

- Edge focus will be shown to be proportional to the Displacement slope. Spatial differentiation of Displacements will yield this Disparity measure.

- 2D image alignment, whereby an image template is located in a scene, is useful in recognition schemes [85]. When all the feature Displacements are measured, and the optics of the edge imaging model included, it is possible to measure

the net misalignment Disparity as a least-squares measure on the full image Displacements.

- Fused Binocular Cyclopean vision is the process of combining the two stereo images into one [42]. Taking the average of the left and right Displacement images produces a fused Displacement image that combines the features of each image into a single one.

These are only a few Disparity representations that are easily calculated from Displacement inputs. These representations are also widely used in computational and biological vision research in modeling such high-level tasks as object recognition, 3D shape estimation and motion measurement.

Importantly, each representation preserves precise information about the underlying features, unlike the coarse quantization typical of the symbolic designs. Indeed, not only are geometric feature properties preserved, but a variance measure is assigned to the representations at each processing step. As will be shown in the next chapter, it is even possible to demonstrate that a well designed Displacement/Disparity model will render optimal measures of feature properties such as edge location, stereo disparity, and other spatial and temporal data. The best feature of the model, however, is its simplicity.

This chapter will first define the 1D Displacement and Disparity Representations by the properties these representations should possess. Instead of taking the entire 2D domain, individual 1D 'slices' of the image are used. This helps simplify the analysis of the model. The 1D model is also somewhat simpler than the general 2D model in that it encodes only feature contrast and distance, not orientation. A simple computational Displacement/Disparity model is then developed for the distance component only and using stereo Disparity as an example. Some examples of other Disparity representations are also given.

The next chapter analyzes the effects of noise on the 1D model and in the process introduces the contrast component of the model. It is at this point that a companion

variance measure is added to the Displacement representation. That chapter also contains a discussion of the effect of non-isolated features typical of real imagery, and how this can be expected to affect practical implementations. A full 2D model is developed in Chapter 5. The 2D model adds feature orientation to the distance and contrast measures of the 1D model resulting in the full Displacement vector representation defined earlier. It is at this point that all of the vision representations discussed above can be posed as Disparity models. Thus the following three chapters deal, in turn, with the three components of the Displacement representation; this chapter deals with the distance measure, the next the contrast measure, and the following chapter completes the model by adding the orientation component.

3.2 The 1D Displacement/Disparity Representations

The 1D Displacement model is intended to yield a representation that, at any point in the image, compactly encodes the distance to the nearest contrast edge and the contrast of the edge. For example, taking the 1D image $I_m(x)$ of Figure 3-2, assume that an image intensity step exists at some location x_o in the image domain. The Displacement function $d(x)$ would encode the distance of any point to x_o, i.e. it would be proportional to $x - x_o$.

The Displacement representation will also encode contrast, since contrast is an important measure of the Displacement signal quality when additive noise is present. Contrast also serves as a useful matching criteria. The remainder of this chapter will concentrate on the scalar 1D Displacement function $d(x)$ and Disparity representations computable from it. The contrast measure will be discussed in detail in the following chapter on noisy signals.

Disparity representations are linear transforms of Displacement representations. Temporal and spatial differentiation render optical flow and edge blur information.

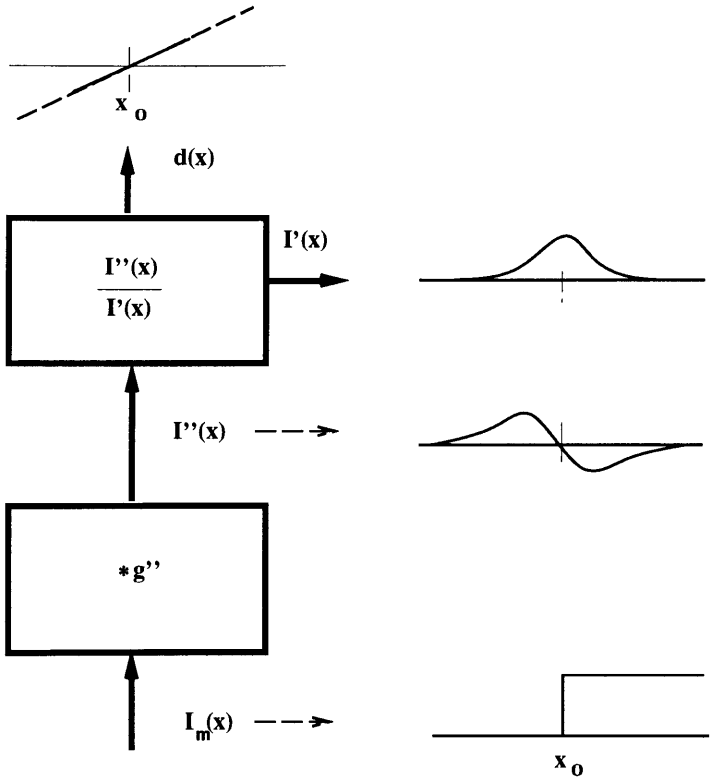

Figure 3-2: Basic 1D Displacement Function

Finite sums and differences yield stereo and Cyclopean fused images. Some of these will be discussed in this chapter, some will have to await the development of the other two components of the Displacement representation — feature contrast and orientation. First, however, the 1D Displacement model is developed.

3.3 The 1D Displacement Model

The Displacement model is predicated on image functions being composed of superposed "edges", or step intensity features. A simple image $I_m(x)$ has a single step at $x = x_o$:

$$d_o(x) = x - x_o$$
$$I_m(x) = \alpha u(d_o(x))$$

where $u(x)$ is the unit step function, $d_o(x)$ is the signed distance from any point x to the edge position x_o, and α establishes both the sign and contrast of the edge feature. Note that $d_o(x)$ is the desired distance component of the 1D Displacement representation since it measures the distance at any point in the image to the contrast edge. Therefore the goal is to develop a computational model to solve for $d_o(x)$. This section develops such a model.

The processing starts by convolving the image with a Gaussian $g(x)$ of some width σ_b, and then taking the second derivative of the resulting smoothed step. This is equivalent to convolving the image $I_m(x)$ with the second derivative of the Gaussian:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma_b} e^{\frac{-x^2}{2\sigma_b^2}}$$

$$I''(x) = g'' * I_m(x).$$

This is shown schematically in Figure 3-2. This convolution with $g''(x)$ is a 1D approximation to the Laplacian of Gaussian model of retinal operation [50]. This convolution is analogous to integration of the $g''(x)$ operator since the input is an image step, so the function $I_s''(x)$ for the single step input can be calculated directly[1]:

$$\begin{aligned} I_s''(x) &= \alpha g'(d_o(x)) \\ &= -\alpha \frac{d_o(x)}{\sigma_b^2} g(d_o(x)) \end{aligned} \tag{3.1}$$

A second representation, $I'(x)$, is calculated, either through numerical integration of $I''(x)$, convolution of the image with a $g'(x)$ operator, or simply taking the derivative of the Gaussian smoothed image. In any case, the resulting function for the isolated feature image $I_m(x)$ is:

$$\begin{aligned} I'(x) &= \int_{-\infty}^{x} I''(\gamma) d\gamma \\ I_s'(x) &= \alpha g(d_o(x)) \end{aligned} \tag{3.2}$$

The Displacement function is simply the ratio of these two representations scaled by a constant — the smoothing Gaussian σ_b^2:

$$\begin{aligned} d(x) &= -\sigma_b^2 \frac{I''(x)}{I'(x)} \\ d_s(x) &= d_o(x). \end{aligned}$$

Therefore, by taking the scaled ratio of these two image functions, we can arrive

[1]To avoid confusion between general representations, such as $I''(x)$ and the response to the special feature image $I_m(x) = \alpha u(d_o(x))$, the subscript "s" is used for the step image representations.

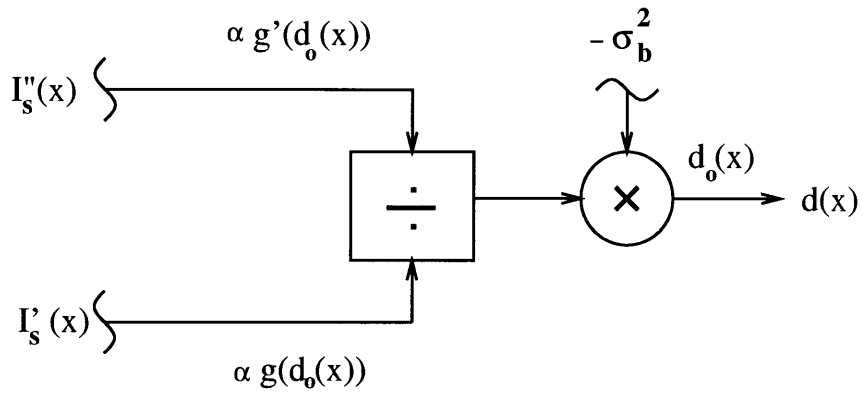

Figure 3-3: Basic Displacement Model

at a measure of the distance function d_o. This $d(x)$ function is the feature distance component of the 1D Displacement representation. The denominator, $I'(x)$, which encodes the feature contrast, α, is discussed in the next chapter. Figure 3-3 shows the 1D Displacement calculation schematically.

3.4 1D Disparity Models

As mentioned earlier, linear combinations of the Displacement function yield Disparity representations that prove useful for such high-level vision tasks as 3D structure, motion, and object recognition. These tasks, as discussed earlier, almost certainly cannot be cast in a purely image-driven (or "bottom-up") model such as the Displacement model, but Disparity representations, such as stereo disparity, optical flow, edge focus, 2D image alignment, and fused binocular Cyclopean vision are almost certainly useful in tackling such top-down tasks. This thesis focuses primarily on stereo as an application domain, and the bulk of the analysis will be directed to this problem.

Some of these other Disparity models will be discussed in this and the following chapters to illustrate the breadth of the Displacement/Disparity application domain. In fact, the stereo model developed and tested in this research utilizes other Disparity measures such as focus and Cyclopean fused vision and could have used others, such

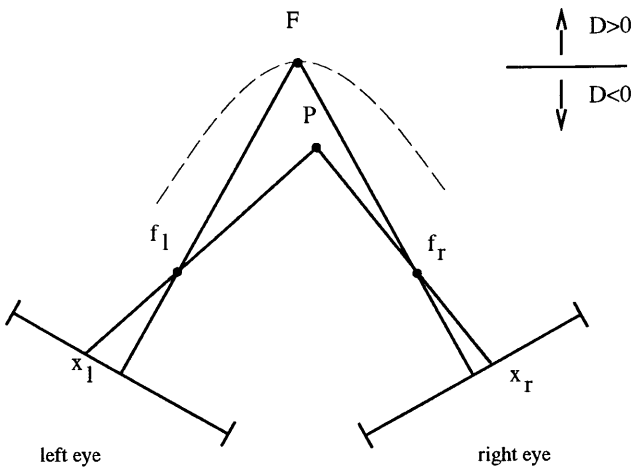

Figure 3-4: 1D Stereo

as motion, to fully qualify the stereo solution.

3.4.1 1D Stereo Disparity

Stereo vision, or stereopsis, is useful for extracting the depth (or distance) of 3D scene features based on the small differences between their left and right image feature positions. Figure 3-4 shows the 1D stereo problem where the left and right eye views are 1D projections of points on the plane through the two focal points f_l and f_r. Any scene point P — such as would be caused by a contrast edge, for example — will produce two image points x_l and x_r.

When $x_l = x_r$ this corresponds to a curve in the plane, called the horopter (shown as the dashed arc through point F). Stereo disparity is defined as $D = x_l - x_r$, so the disparity is zero along this curve. Any point closer to the eyes, or below the curve in the drawing, will result in $x_l < x_r$ and therefore $D < 0$. This is called "crossed" disparity, and indicates an object is closer to the observer than the fixation curve. When $D > 0$ this is "un-crossed" disparity, and indicates the object is further away. Thus, measuring the stereo disparity D provides information on object distance.

In the case of the Displacement model, stereo Disparity $D(x)$ is determined by subtracting the left and right Displacement functions $d_l(x)$ and $d_r(x)$ (see Figure

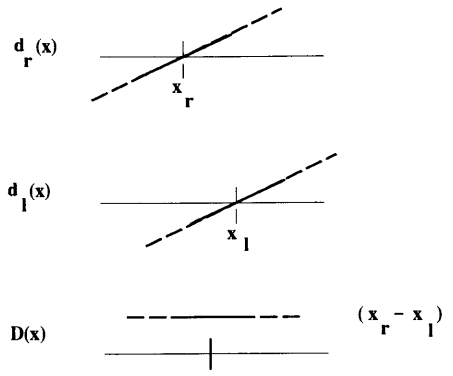

Figure 3-5: Basic 1D Stereo Disparity Function

3-5)[2]:

$$\begin{aligned} D(x) &= d_r(x) - d_l(x) \\ D_s(x) &= (x - x_r) - (x - x_l) \\ &= x_l - x_r. \end{aligned}$$

3.4.2 Cyclopean Vision

While subtracting left and right feature positions results in a stereo disparity measure, another representation can be produced to combine, or fuse, the image pairs into a single image. This has been referred to as Cyclopean fused vision [42]. It can be produced by combining the features of the two images at locations determined by the average of the left and right positions. For instance, the cyclopean position for P in Figure 3-5 would be

$$x_c = \frac{x_l + x_r}{2}.$$

The same approach can be used with Displacement representations; the Cyclopean Displacement is the average of the left and right Displacement representations

[2]This assumes that the 1D cross-sections of this simple model are aligned with the stereo epipolars. This assumption will not be needed in the 2D model discussed later.

$$\begin{aligned} d_c(x) &= \frac{d_l(x) + d_r(x)}{2} \\ d_{cs}(x) &= x - x_c. \end{aligned}$$

This is properly considered a Disparity representation since it is a linear transform of Displacement representations. Problems created by such phenomena as missing features, occlusions, focus errors, and contrast mismatches make this simple averaging model impractical, however. A better model will need the results of the next chapter on Displacement variance measures and will be presented there.

3.4.3 Optical Flow

Measuring 3D motion from 2D images is a complex and difficult task, but one way to start is to measure the motion of contrast features in the image. This is called optical flow. The motion of feature points in the image, $\dot{x}_o$, can be easily arrived at in this 1D model by taking the temporal derivative of the Displacement function $\dot{d}(x)$. The motion Disparity $M(x,t)$ is thus

$$M(x,t) = \frac{\partial}{\partial t} d(x,t)$$

where $d(x,t)$ is the displacement function;

$$\begin{aligned} d(x,t) &= \frac{-\sigma_b^2 I''(x,t)}{I'(x,t)} \\ d_s(x,y) &= x - x_o(t). \end{aligned}$$

The latter expression assumes an isolated contrast edge located at $x_o(t)$ at time t. We can solve for $M(x,t)$ (using the $\dot{x}$ notation for $\frac{\partial}{\partial t}x$):

$$\begin{aligned} M(x,t) &= -\sigma_b^2 \frac{\dot{I}''(x,t)I'(x,t) - \dot{I}'(x,t)I''(x,t)}{I'^2(x,t)} \\ &= \frac{1}{I'(x,t)} \left(-\sigma_b^2 \dot{I}''(x,t) - \dot{I}'(x,t)d(x,t)\right) \end{aligned}$$

The 2D image world is a bit more complex, due to the phenomenon known as the "aperture problem", so a more complete analysis will be deferred until Chapter 5.

3.4.4 Focus

When an edge is modeled as it was in $I_m(x)$ as a scaled unit step, the slope of the Displacement function $d(x)$ is unity — $d'(x) = 1.0$. On the other hand when edges are not so steep, as would happen with real imaging devices and optics as well as with diffuse edges in scenes, the effective width of the $I''(x)$ and $I'(x)$ functions increases. They are, in effect, convolved with the blurring function. Although most optical blurring has a "boxcar" (1D) or "pill-box" (2D) convolving function, many feature blurring effects such as diffusion or diffraction make a Gaussian model adequate.

Thus blurring can be modeled as an increase in the Gaussian convolving width σ_b due to scene and imaging properties. This would result in a decrease in the slope of $d(x)$, or $d'(x) < 1.0$.

The imager focus can usually be calibrated such that its effect on σ_b is known and thereby does not affect other calculations adversely. As a scene feature deviates in distance from the plane of focus of the camera, often referred to as "moving out of the depth of field" in photography, the feature will defocus in proportion to this distance. Thus focus is a 3D cue. Also, as mentioned, the 3D features themselves can be diffuse. In both cases, focus is a good measure of feature match and can be used to test correspondences in matching problems.

The stereo Disparity algorithm uses focus Disparity $d'(x)$ for precisely such ends. If two features are matched, then the stereo Disparity slope $D'(x) = 0$. Whenever a

mismatch occurs, either in distance or scene feature sharpness, then $D'(x)$ deviates from zero. This is a good test of feature match consistency.

The 2D model is only slightly more involved and will be discussed in Chapter 5.

3.4.5 Other Disparity Models

More complex linear analysis is also useful in generating Disparity measures. Least squares matching of images and templates can provide alignment vectors for fixation, attention, and recognition schemes. Least squares is also useful for egomotion measures. Most of these involve the 2D model and will be touched on in that chapter. A central observation, however, about Disparity representations is that they are capable of providing a rich set of image measures based on simple linear analysis of the representation arrived at by decidedly nonlinear means — the Displacement model.

3.5 Summary

This chapter defined the Displacement representation as comprised of the following measures:

- the distance from a point to the nearest contrast edge,
- the contrast of the edge, and
- the orientation of the edge.

Although in this chapter only the first, or distance, measure was developed, a substantial number of useful image properties emerged in the form of simple Disparity models of stereopsis, motion, and focus.

Chapter 4

Displacement Models With Noisy Signals

4.1 Variance Measures in a Nonlinear Model

The previous chapter described how the Displacement function $d(x)$ can be formulated to estimate the distance to an isolated edge in an image from any point in the image domain. This chapter focuses on just how sensitive the Displacement measure is to the effects of added noise. This is accomplished by developing a measure of the variance of the Displacement measure at every point in the domain. Developing this model is the central goal of this chapter. A good variance model will allow Displacement and Disparity calculations to be made as accurate as possible, despite the nonlinearity of the calculation. Indeed, this analysis will show that by using the variance model, it is possible to render Displacement/Disparity distance measurements as accurate as those based on best least squares analysis of any linear transform of the input signals.

The preceding discussion of the Displacement representation was based on a noise-free Gaussian smoothed 1D isolated step intensity image model — $I(x)$. The Displacement function is derived from this input by taking the ratio of the second spatial derivative of this signal, $I''(x)$, to the first, $I'(x)$. The nonlinearity of this calculation

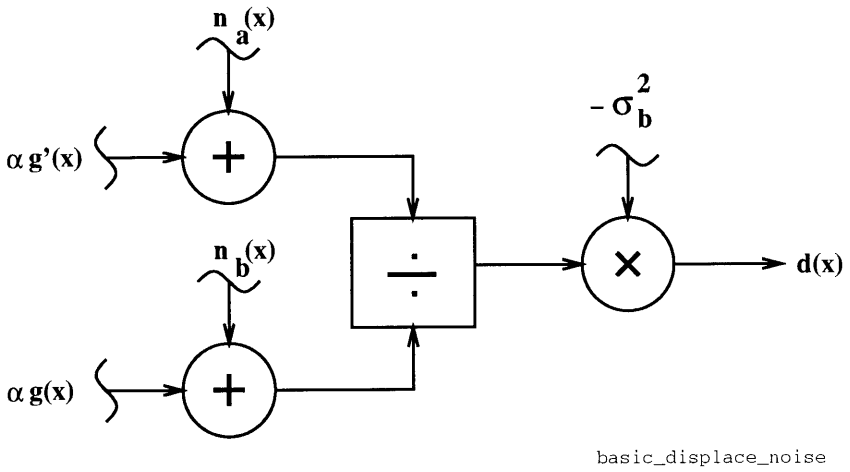

Figure 4-1: Basic Displacement Model With Added Noise

complicates the analysis involved in determining the variance of the Displacement function.

For example, Figure 4-1 shows a simplified block diagram of the displacement calculation $d(x)$ derived from the Gaussian input functions $g'(d_o(x))$ and $g(d_o(x))$ with added uncorrelated noise $n(x)$ [1]:

$$\begin{aligned} d(x) &= -\sigma_b^2 \frac{I''(x)}{I'(x)} \\ d_s(x) &= -\sigma_b^2 \frac{\alpha g'(d_o(x)) + n_a(x)}{\alpha g(d_o(x)) + n_b(x)} \end{aligned}$$

When the denominator function $\alpha g(d_o(x))$ is large, the effect of noise on the input signals will be quite different than when it is small. This changes the noise variance model from one that is spatially independent to one that is highly dependent on the feature location, x_o. This can be easily seen in Figure 4-2, that shows representative input signals with a small amount of added noise (5% of peak amplitude), and Figure 4-3 that shows the resulting Displacement signal and the variance distribution about

[1] The subscript s is used when the image input is $I_m(x) = \alpha u(d_o(x))$ as discussed in the last chapter.

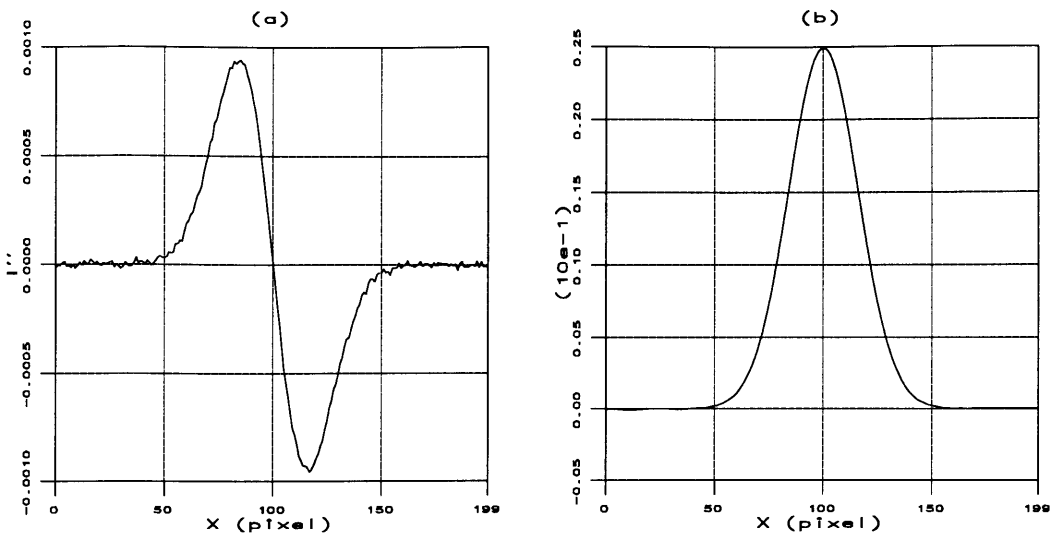

Figure 4-2: Input Signals: (a) $I''(x)$ with Noise, (b) $I'(x)$

the edge position ($x_0 = 100$) based on a large ensemble of trials. Clearly, the variance distribution is quite non-uniform.

It is important to have a good estimate of the variance of the Displacement signal so that subsequent analysis can best utilize the function where it is most reliable. As will be seen, a good model of the variance distribution allows calculation of a measure that is not only a best estimate of the edge position x_o in a maximum likelihood sense, but also an optimal measure of the position of the edge in a least squares sense based on the linear image input functions. More importantly, a good model of the Displacement variances is useful in estimating the variances of subsequent Disparity calculations. These calculations usually involve linear operations, and thereby lend themselves to linear analysis.

This chapter will also focus on how noise influences the design of the Displacement model. The addition of noise greatly influences the implementation, especially when analog or possible biological models are considered.

In order to develop the Displacement variance model a number of steps need to be taken:

- The first step is to develop a measure of feature position based on a least

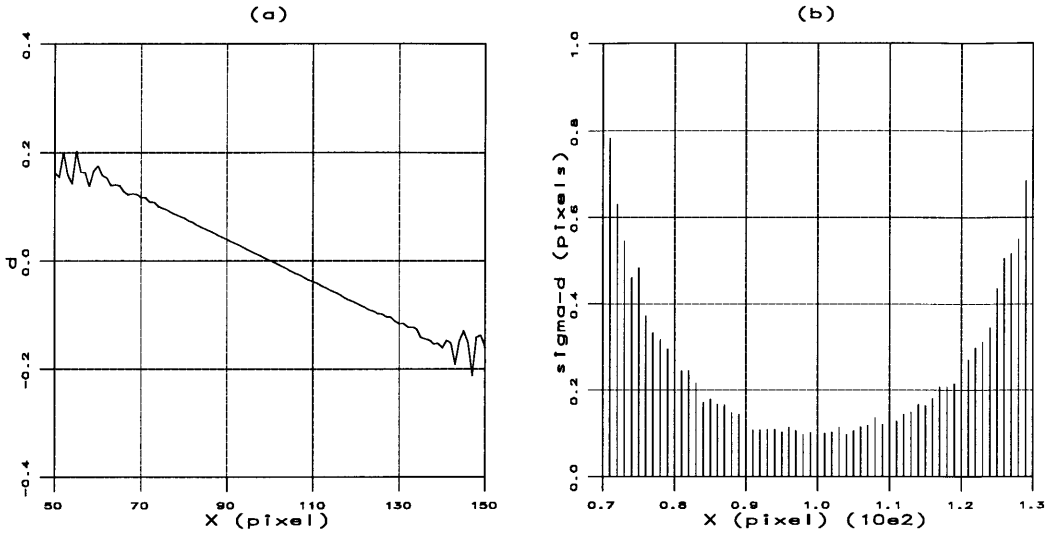

Figure 4-3: Displacement Signal ($\sigma_b = 16$): (a)$d(x)$ with Noise, (b) σ_d Distribution

squares measure of one of the linear transforms of the input step image used by the nonlinear Displacement calculation.

- Using this measure, the organization of the sensor/processor system is analyzed to determine which one of at least three possible designs makes sense from a noise immunity standpoint.

- Given such a design choice, the next issue is how to implement the design in a way that it is stable, computationally tractable, and yet is still accurate.

- Using that design, the Displacement variance distribution can be modeled using a maximum likelihood analysis. This is compared with the least squares analysis, thereby providing an analytical model of Displacement variances.

- Some minor adjustments need to be made when the continuous model is made discrete, and some algorithmic issues need to be considered in any practical implementation.

- The complete discrete Displacement model can be finally tested by using the analysis developed in the preceding sections and its performance compared with

the best least squares measure on the input signals.

4.2 Least Squares Estimation of x_o From A Gaussian Input

In order to evaluate the performance of the Displacement model and various algorithmic design options, some measure of feature position is needed by which comparisons can be made. This section discusses one such measure — the least squares estimate of x_o given a Gaussian signal with additive uncorrelated noise. This measure will prove useful in evaluating the Displacement noise model, algorithm designs, and the effects of sampling on discrete design models.

The Displacement model up to the point of the division step is linear. Since the Displacement function measures the edge position x_o, this section focuses on the problem of estimating x_o in the best least-squares sense from one of the two input signals in the presence of additive zero-mean noise. The denominator of the Displacement calculation, $I'(x)$, is Gaussian in form and serves this purpose well.

Similar analysis could be done on any other linear transformations of the image, such as the smoothed step input $I(x)$ or the second derivative signal, $I''(x)$. The choice of which linear transform will not affect the results. The Gaussian signal form of $I'(x)$ yields the simplest analysis, and is therefore used as the standard for locating edge position in this analysis.

The details of the least squares solution are shown in Appendix B. The best estimate for x_o in a least squares sense given the Gaussian input signal $I'(x)$ is found to be :

$$x_o = \frac{\int x I'(x) g(d_o(x)) dx}{\int I'(x) g(d_o(x)) dx}. \tag{4.1}$$

Note that the solution, as posed, is not in a simple closed form. The optimal position estimate x_o is the first moment of the function formed by the product of the noisy input function $I'(x)$ and the uncorrupted Gaussian $g(d_o(x))$. Of course, $g(d_o(x))$ depends on x_o. This recursive solution does not present a serious problem, however, since iterative solutions do converge rapidly in practice.

Note also that the derivation uses unbounded continuous intervals, as opposed to the finite domain summation operations usually appropriate for models of sampled images. Indeed, most of this analysis does not require continuous domain assumptions. Sometimes, however, these assumptions are important, as will be discussed later in the section on discrete models and again when experiments are run to test the model later in this chapter.

The findings of this least squares analysis will be first applied to the problem of deciding a basic model design issue; which representation should be sent from the sensor (or eye) to the processor (or brain). This issue should be sorted out before proceeding further down the noise model path since the maximum likelihood analysis will be tested using the model design developed in the next section.

4.3 Noise Considerations in Displacement Model Designs

In considering any practical analog or potential biological incarnation of the Displacement model, the addition of noise imposes some constraints on the design that would not exist in a noise-free environment. Noise can be introduced at any stage of the processing, and can take on many forms, but a first-order noise model can be posed that can be very helpful in addressing some of these design issues.

This section examines a simple noise model where the computation is broken into two stages; the sensor and the processor. The calculations involved in producing the Displacement representation can be split in any manner between these two sites.

Uncorrelated noise is added to the single representation transmitted from the sensor to the processor (see Figure 4-4a). The magnitude of that noise is proportional to the dynamic range (peak amplitude) of the transmitted signal. This is a typical communication system model [12].

The goal of the design is to minimize the variance of the best least squares x_o estimation of the Gaussian signal $I'(x)$ at the processor stage. This can be done by choosing wisely the order of the operations as well as optimizing some of the bandpass characteristics of the processing steps. Both of these issues will be discussed in this section.

4.3.1 Sensor and Processor Configurations

The Displacement model uses both the first and second spatial derivatives of a Gaussian smoothed intensity signal. The ordering of their computation is arbitrary when noise and sampling are not taken into account since these operations are linear. When the sensor produces discrete samples, however, the smoothing should precede sampling in order to minimize the effects of "aliasing", the phenomenon of high spatial frequency image events producing low frequency artifacts through the sampling process. This is important in both biological and electronic imaging systems.

It is less clear where to lump the stages of differentiation to minimize the influence of transmission noise. Three possible arrangements of the steps involved in the Displacement calculation are shown in Figure 4-4b-d. The issue at hand is how to allocate the differentiation operations — indicated as '$j\omega$' in the diagram — between the sensory step and the processing step. In Figure 4-4d, '$1/j\omega$' indicates an integration operation.

It is interesting to consider biological vision systems, especially those of the primates, in the context of this issue of noise sensitivity. Certain signal processing can take place at the sensor — the retina. Space, weight, and cell density limitations, however, obviously limit how much processing is possible at the sensor. It is reason-

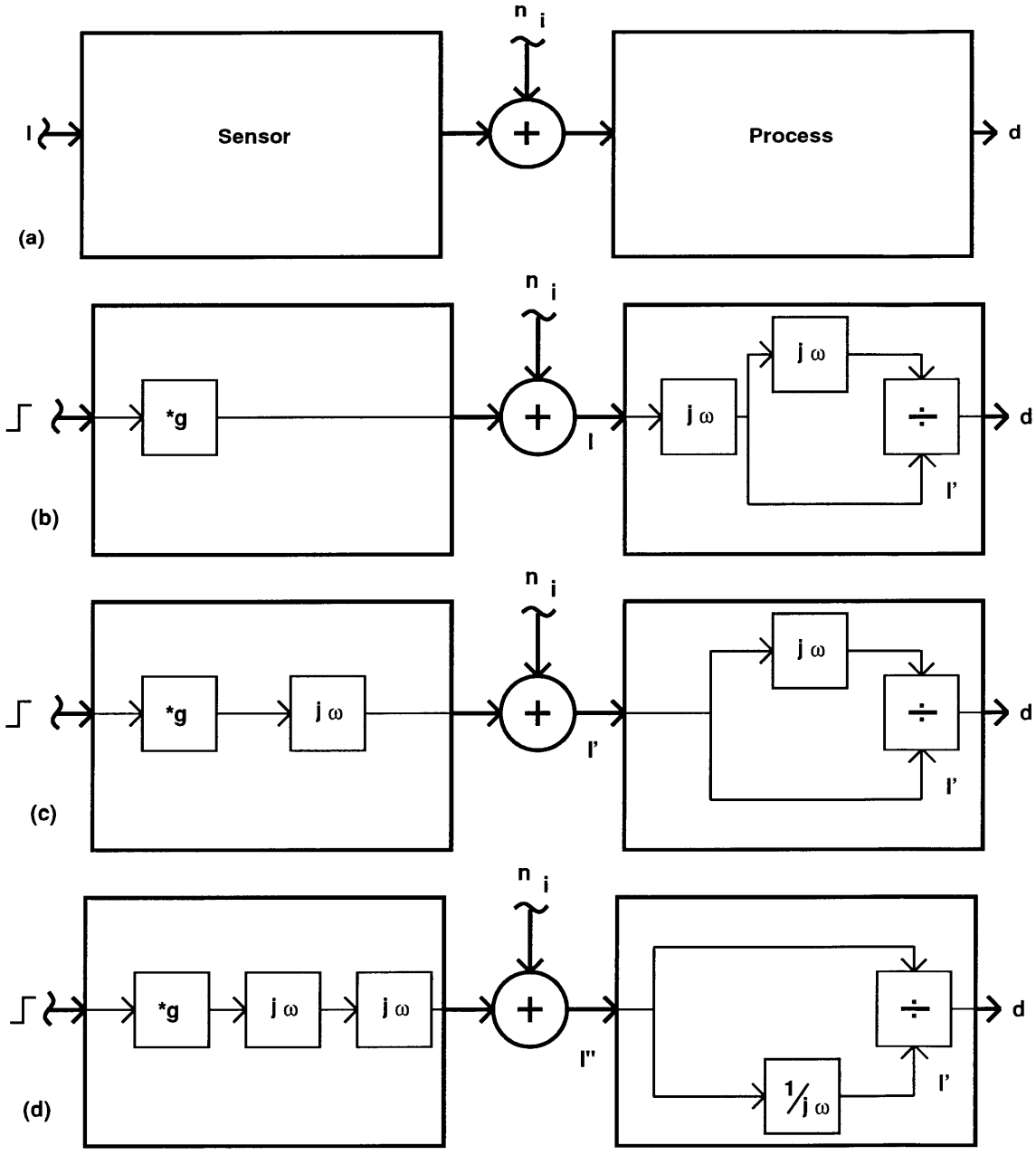

Figure 4-4: 1D Displacement Model With Added Noise

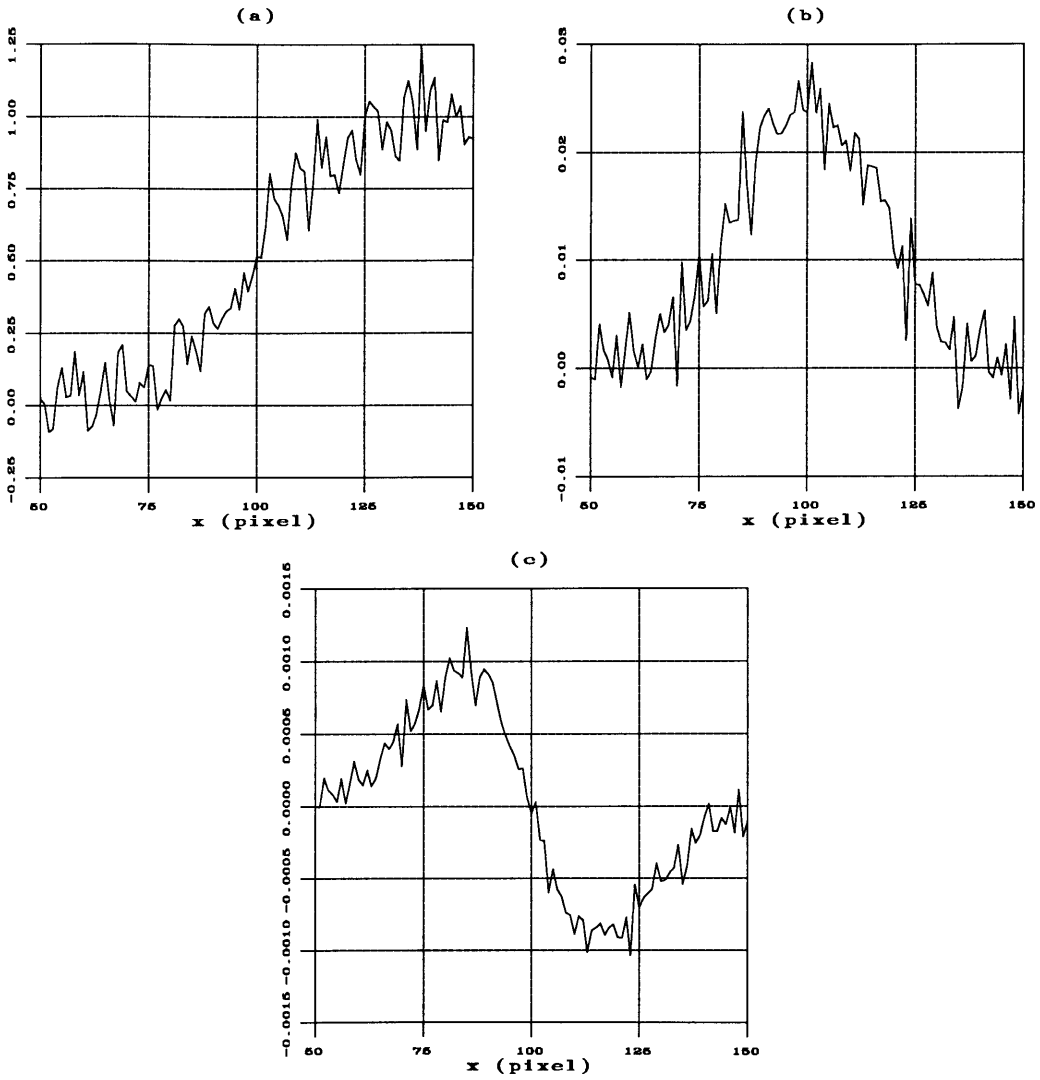

Figure 4-5: Noisy Input Signals: (a) Model b $I(x)$, (b) Model c $I'(x)$, (c) Model d $I''(x)$

able to presume that the retinal processing will be devoted to those functions that would suffer the most from noise contamination if attempted at the cortical level since the image signal must pass through the optic nerve, the Lateral Geniculate Nucleus and the optic radiation before reaching the visual cortex at the rear of the brain. This physically large distance almost certainly results in significant signal contamination. Thus, in a first-order noise model, the noise insertion stage of Figure 4-4a would most appropriately apply to this eye-to-brain pathway.

The central issue of this section is how noise can narrow design options in systems with finite dynamic range. Although with digital implementations, especially with readily available data formats such as high precision floating point, one can often disregard these constraints, with biological systems, one might expect these issues to result in "design" decisions that would occur naturally through evolutionary processes. It will be interesting to see if the noise model for the Displacement representation dictates design constraints consistent with those found in nature.

In Figures 4-4b, 4-4c, and 4-4d the signal on the transmission line is the smoothed image, its first derivative, and its second derivative, respectively. By injecting noise in the transmission line, the variance of the least squares x_o estimate of the $I'(x)$ denominator to the Displacement calculation can be compared. Figure 4-5 shows some sample functions for each of the three signal formats at the input to the processing step. The added noise has a normal distribution with its RMS level (σ_n) set to 10% of the peak amplitude of the signal. The filter width is 16 pixels and the edge is at $x_o = 100$.

Some tests were run on these models to see how they fared when comparable noise was added to the signal. These tests consisted of simulations using large ensembles of uncorrelated normal noise signals added to the sensor signal representations from Figure 4-4. Some band-limiting was necessary in the model of Figure 4-4d, as will be discussed further in the next section, but for the purposes of these tests, minimal band-limiting was used.

Model	$\sigma_b = 4$	$\sigma_b = 8$	$\sigma_b = 16$
b	0.653	1.036	1.205
c	0.214	0.292	0.434
d	0.172	0.189	0.343

Table 4.1: Noise $\sigma[x_o]$ Test Results

Table 4.1 shows how the three models of 4-4 (b) through (d) fared when the RMS noise level of 10% of the peak signal was added. This choice was arbitrarily made since varying the noise amplitude had no impact on the relative performance of the designs. The standard deviations — in pixels — of the least squares x_o measure (Equation 4.1) on the Gaussian denominator are shown in the table for the various filter widths tried. Note that the standard deviation of the x_o estimate is of the order of 0.2 pixels for the smaller filter (receptive field) sizes, despite the very large noise content of the experiments.

It is clear from the data that the last model (Figure 4-4d) is superior to the others. This experiment thus provides empirical support for the choice of transmitting the second derivative of the Gaussian smoothed signal over any other single representation from the sensor. This result also provides a nice indirect confirmation of the veracity of the Marr-Hildreth Laplacian of Gaussian retinal model from a signal integrity standpoint, even though it must be pointed out that this analysis is still strictly 1D [50]. Of course, it is a common design practice, especially in analog systems, to lump the differentiating (sometimes called "hi-pass") operations up at the "front end" to minimize noise content. This experiment simply confirms the wisdom of such design practices.

4.3.2 Filter Design

The preceding section demonstrated that a sensor design that produces as output the image convolved with the second derivative of a Gaussian is the most sensible from a signal integrity standpoint. The design criteria of the analysis constrained the

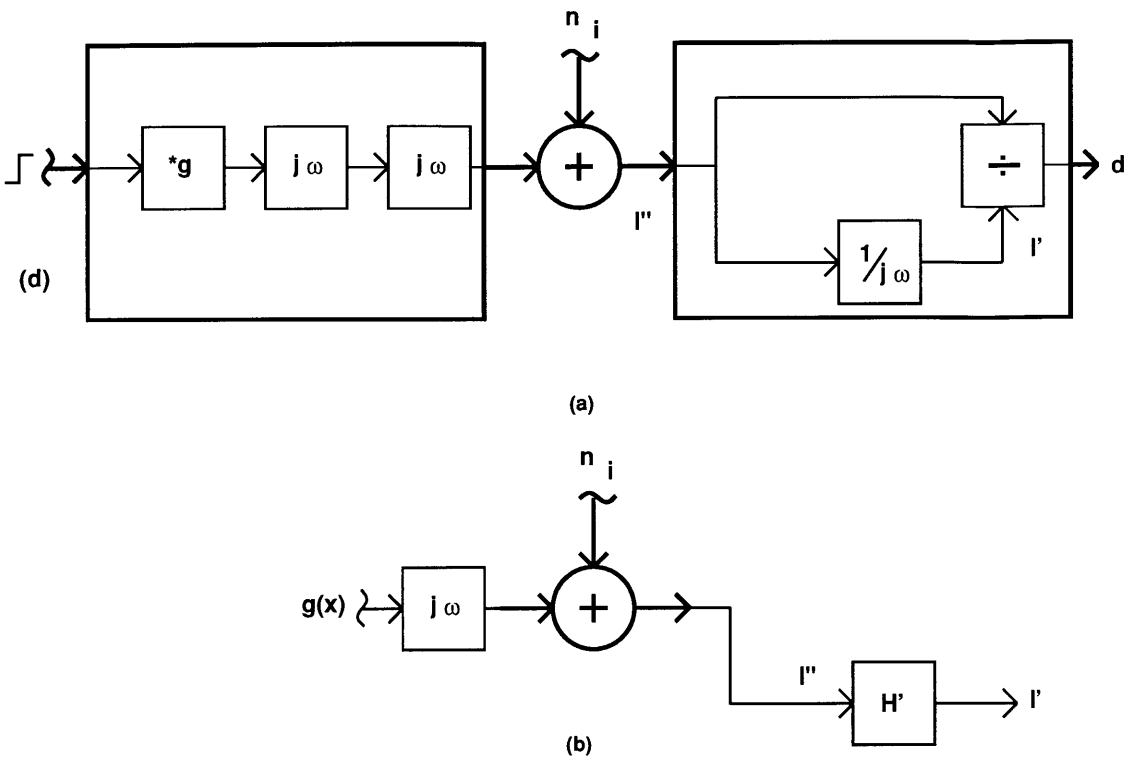

Figure 4-6: Optimal Filter Model

transmitted signal to be a single image representation, so the reconstruction of the first derivative signal $I'(x)$ requires an inverse filter — like an integrator — to reconstruct the Gaussian functional form used in the Displacement calculation. Figure 4-6(d) shows this stage in the processing step as integration $(1/j\omega)$.

Image reconstruction, the process of generating $I'(x)$ from $I''(x)$, is an ill-posed problem in the presence of noise [5, 6, 84]. The Brownian noise, resulting from the integration of white noise, renders the integral drift-prone. A "low-pass" approximation to integration, whereby the low spatial-frequency components are passed without being integrated, would be a practical and well-posed fix to the above problem. Indeed, this is the approach that will ultimately be favored in the algorithm designs used in the subsequent chapters.

In order to design such a filter, however, it is useful to examine the issue of what filter design would be optimal in some sense. For the purposes of this discussion, the

optimal filter is one that reproduces the Gaussian signal form of the noise-free sensor (retinal) stage when noise is added in the manner of the preferred design as shown in Figure 4-6(d), and where the reproduced signal $I'(x)$ is closest in form, in a least squares sense, to the noise-free input signal $\alpha g(d_o(x))$ when presented with an isolated step input image. Approximations such as low-pass designs are then based on this optimal filter design to insure maximum signal integrity while preserving stability.

The optimal filter is readily developed using matched filter methods. This design can then be used to generate filter designs that can be implemented in practical analog or digital implementations. The derivation of the optimal filter is shown in Appendix A. The optimal transform

$$H'(\omega) = \frac{-j\omega}{\omega^2 - (2\pi\sigma_b^2\rho^2)^{-1}e^{\omega^2\sigma_b^2}}$$

depends on some estimate of the signal to noise ratio of the $I''(x)$ signal, ρ. This is defined as the ratio of the peak signal amplitude to the RMS amplitude of the noise [36]. Figure 4-7 shows a plot of the filter response which clearly shows that this optimal design has three distinct response regions; a low frequency roll-off region, which renders the filter stable, an intermediate frequency integration region, similar to the naive design, and a high frequency cutoff point characteristic of the Gaussian signal form.

Appendix A also shows how this filter can be approximated by a second order filter with the impulse response:

$$h'(x) = (1 - \omega_c x)e^{-\omega_c x}$$

which can be implemented using convolution. The cutoff frequency ω_c is taken from the optimal filter response curve for the appropriate noise figure ρ and smoothing σ_b. This design preserves the low frequency roll-off region and intermediate frequency integration response. It does not, however, have the high-frequency cutoff characteristic

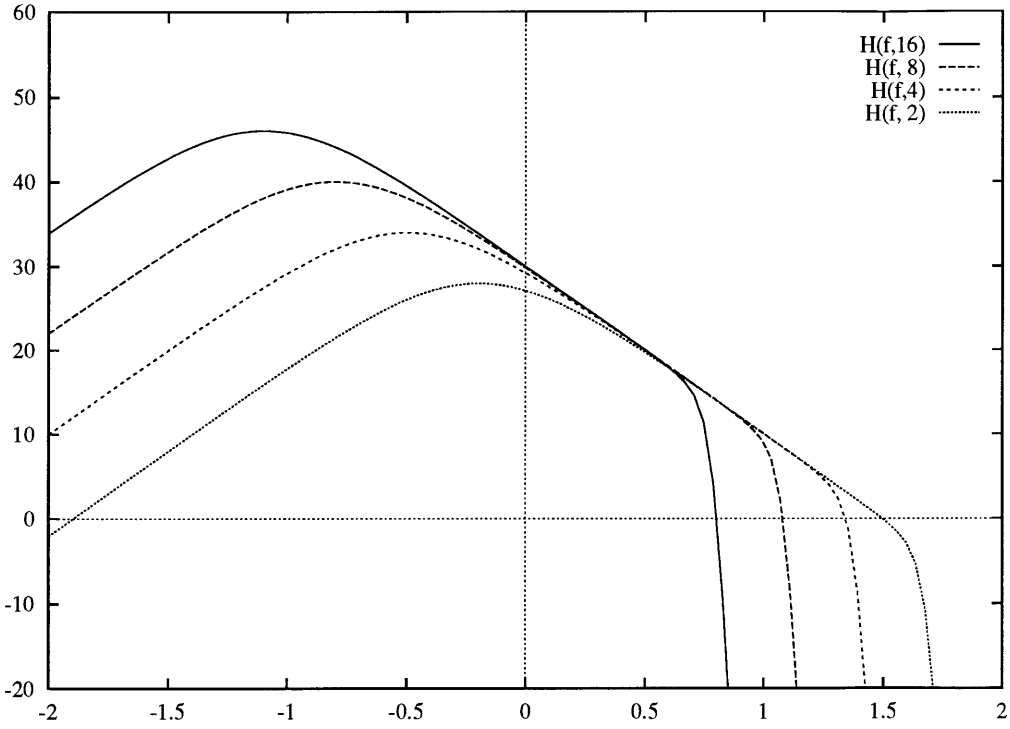

Figure 4-7: 1D Optimal Filter Response — Four σ_b sizes are shown (16, 8, 4 and 2) The vertical axis is db gain. The horizontal axis is in log cycles per image (0 = 1 cycle per image)

of the optimal model. It also requires large support.

Also developed in the appendix is an even simpler design — the band-limited integration, or recursive low-pass filter. This design allows more low-frequency signal component than the optimal design (and thus allows a bit more noise contamination) but can be implemented in a highly local algorithm. Given an input signal i_j for some pixel j and its immediate neighbor i_{j-1}, the filtered version o_j can be arrived at using *its* immediate neighbor o_{j-1}:

$$o_j = \gamma o_{j-1} + \frac{i_j + i_{j-1}}{2},$$

where γ is a constant coefficient. Based on the observed response of the optimal filter and the above first-order approximation, this coefficient can be approximated from the SNR figure ρ and the Gaussian width σ_b to be:

$$\begin{aligned} \gamma &= e^{-\frac{\omega_c}{2\pi}} \\ &\approx e^{-\frac{1}{(2\pi)^2 \rho \sigma_b}}. \end{aligned}$$

This filter design is what is used for the integration step in the analysis of the noise model.

4.4 Displacement Variance Models

The central goal of this chapter is to develop a mathematical model of the variance distribution of the nonlinear Displacement function. One approach toward this end is to examine the issue of arriving at a maximum likelihood estimate of the edge position x_o using the Displacement function. This will be shown to be equivalent to the optimal estimate of the edge position of the linear input signals in a best least squares sense.

4.4.1 Maximum Likelihood estimation of x_o from Displacement Functions

Each displacement estimate $d(x)$ over the domain of the input provides, to some precision, an estimate, $x_d(x)$, of the edge position x_o since

$$\begin{aligned} d_s(x) &= -\sigma_b^2 \frac{\alpha g'(d_o(x)) + n_a(x)}{\alpha g(d_o(x)) + n_b(x)} \\ &\approx x - x_o. \end{aligned}$$

Thus a local feature position estimate $x_d(x)$ can be defined as

$$\begin{aligned} x_d(x) &= -d(x) + x \\ x_{ds}(x) &\approx x_o \end{aligned}$$

It is reasonable to model the Displacement function itself as corrupted by spatially uncorrelated noise since the inputs to the Displacement calculation are derived from a series of linear operations on an image corrupted by uncorrelated additive zero mean noise.

As shown in Figure 4-3, however, not all points in the domain result in equally reliable estimators of x_o. In the vicinity of $x = x_o$, the variance is substantially lower than points further away. At each point in the domain, however, $x_d(x)$ provides some information about the edge position x_o. The best estimate d_o in a maximum likelihood sense of x_o can be derived when some knowledge of the measurement variance $\sigma_d^2(x)$ of $x_d(x)$ is available. This is arrived at as a weighted average based on the variance measures:

$$d_o = \frac{\int w_d(x) x_d(x)}{\int w_d(x)}$$

$$\text{where} \quad w_d(x) = \frac{1}{\sigma_d^2(x)} \tag{4.2}$$

The approach taken here will be to choose a weight function $w_d(x)$, attempt to prove that it renders the maximum likelihood d_o estimate equivalent to the best least squares estimate x_o in the linear image transform and then test this weight against the experimental data to test whether the $w_d(x)$ chosen is indeed inversely proportional to the data variance distribution.

For the first step, if one starts with the hypothesis that the inverse variance $w_d(x)$ is

$$w_d(x) = I'(x)g(d_o(x))$$

then it can be shown that d_o is equal to the best least squares solution for x_o using the linear inputs as developed in Section 4.2.

$$\begin{aligned} d_o &= \frac{\int w_d(x)[-d(x)+x]dx}{\int w_d(x)dx} \\ &= \frac{\int g(d_o(x))I'(x)\left(\sigma_b^2\frac{I''(x)}{I'(x)}+x\right)dx}{\int g(d_o(x))I'(x)dx} \\ &= \frac{\sigma_b^2\int g(d_o(x))I''(x)dx + \int xI'(x)g(d_o(x))dx}{\int g(d_o(x))I'(x)dx} \end{aligned} \tag{4.3}$$

By using the Equation 4.1:

$$d_o = \frac{\sigma_b^2\int g(d_o(x))I''(x)dx}{\int g(d_o(x))I'(x)dx} + x_o$$

Using integration by parts on the numerator, the above can be expressed as

$$d_o = \frac{\sigma_b^2\left[g(d_o(x))I'(x)|_{-\infty}^{\infty} - \int g'(d_o(x))I'(x)dx\right]}{\int g(d_o(x))I'(x)dx} + x_0. \tag{4.4}$$

The first numerator term can be dropped if $I'(x)$ is assumed to converge to zero at these unbounded limits. The second term is zero based on the results obtained in the least squares analysis of Appendix B Equation B.1. This leaves the equivalence of the least squares and maximum likelihood estimates:

$$d_o = x_o.$$

This result leads directly to the desired model of the variance of the Displacement function. The above proof that using $I'(x)g(d_o(x))$ as the weight function $w_d(x)$ in the maximum likelihood analysis results in a maximum likelihood Displacement estimate of x_o equivalent to the best least squares estimate of x_o *prior* to the nonlinear division operation, strongly suggests

$$\sigma_d^{-2}(x) \propto I'(x)g(d_o(x))$$

since this weight is a measure of the inverse variance of the Displacement function (equation 4.2).

In practice, *either* $I'(x)$ *or* $g(d_o(x))$ are good models for the reciprocal of the standard deviation of the Displacement function since the only difference between $I'(x)$ and $g(d_o(x))$ is the addition of uncorrelated noise:

$$\frac{1}{\sigma_d(x)} \propto g(d_o(x)) \text{ or } \frac{1}{\sigma_d(x)} \propto I'(x)$$

Figure 4-8 shows a comparison between the above model and the ensemble data with a σ_b of 32 pixels and $x_o = 100$. The signal to noise is 10:1. The plot shows the relationship between the domain position and σ_d (both in pixel units). When comparing the model to the ensemble data, the Variance distribution does indeed appear to be that predicted by the model, although the model needs some additional refinement at this point.

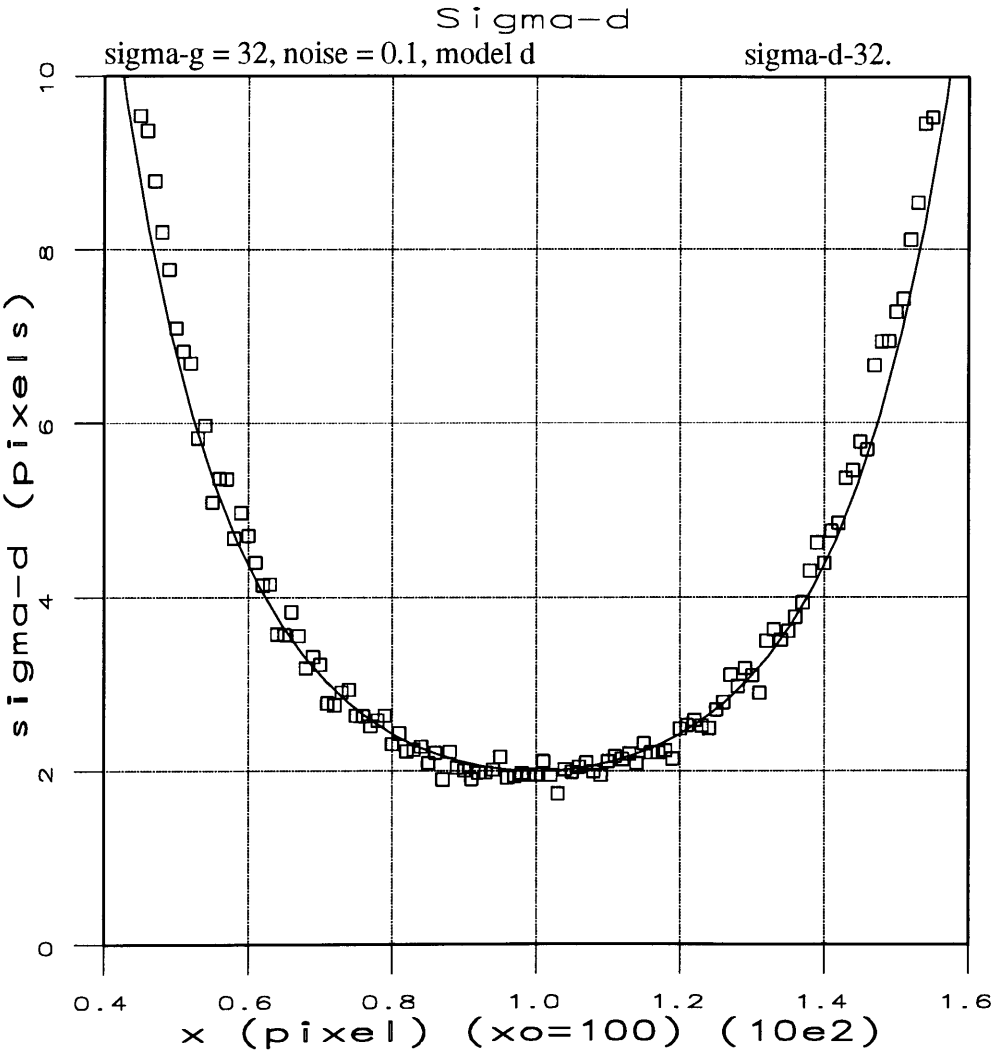

Figure 4-8: Noise Model vs. Ensemble Trials.

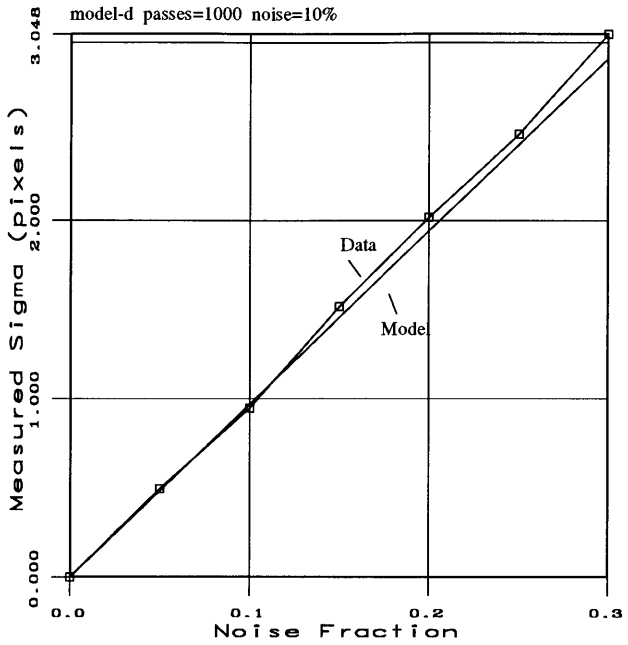

Figure 4-9: Effect of σ_n on $\sigma_d(x_o)$

4.4.2 Scaling the Variance Model

The reason for the proportionality, rather than an equivalence in the variance relation $\sigma_d^{-2}(x) \propto I'(x)g(d_o(x))$ is due to the fact that neither the least squares nor the maximum-likelihood estimation depend on constant scale factors such as the image contrast α. The variance of either measure certainly will depend on contrast in the presence of uniform noise, so some experiments are needed to determine this proportionality to complete the model. By sampling the Displacement function at x_o and observing the variance over a large ensemble of trials, some knowledge of the dependence of σ_d on contrast α can be gained.

Figure 4-9 shows the impact of increasing the added noise on the model. The noise fraction is defined as $\frac{\sigma_n}{P}$, where σ_n is the RMS level of the added noise and P is the peak amplitude of the input I'' signal. The plot is clearly linear, thus showing that

$$\sigma_d \propto \frac{\sigma_n}{\alpha}.$$

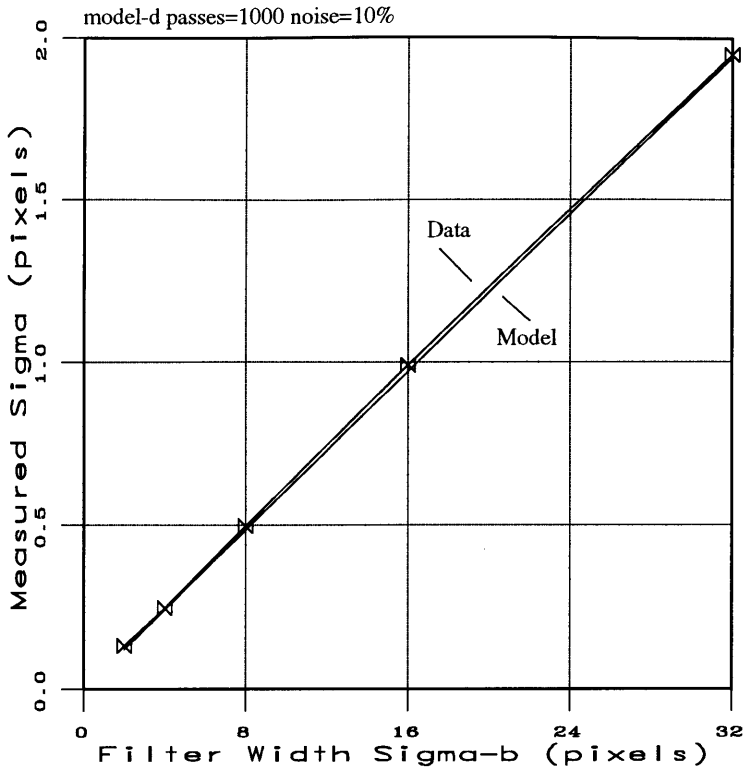

Figure 4-10: Displacement $\sigma[d(x_o)]$ ensemble trials vs. complete noise model $\sigma_d(x_o)$

Measure	$\sigma_b = 32$	$\sigma_b = 16$	$\sigma_b = 8$	$\sigma_b = 4$	$\sigma_b = 2$
Ensemble Trials	1.950	0.991	0.498	0.248	0.132
Model Prediction	1.941	0.971	0.485	0.242	0.121

Table 4.2: $d(x_o)$ model vs. ensemble trial test results

A complete variance model is at hand once the effect of the blurring function σ_b is included. Figure 4-10 and Table 4.2 show a clearly linear relationship between the width of the convolving Gaussian σ_b and $\sigma_d(x_o)$. In these experiments the amplitude of the additive noise σ_n is kept to 10% of the peak amplitude of the input signal $I''(x) = \alpha g'(d_o(x))$. This peak value is

$$I''_p = \frac{0.242\alpha}{\sigma_b^2}.$$

While the σ_n amplitude decreases quadratically with increasing σ_b in the exper-

iments, the standard deviation of the measured $\sigma[d(x_o)]$ increases linearly with its width. This relation, along with the fixed noise fraction constraint of the test leads to the following calibrated displacement variance model:

$$\begin{aligned} \sigma_d(x) &= \frac{\sigma_n \sigma_b^2}{\alpha g(do(x))} \\ &= \frac{\sigma_n \sigma_b^2}{I'(x)} \end{aligned} \tag{4.5}$$

The model predictions as shown in Figures 4-9 and 4-10 and Table 4.2 are supported by the test. Note that there is no need for arbitrary scale factor or any other modification to the above formula to arrive at pixel scaled measures of measurement errors based only on the added noise, scale size and contrast measure.

A useful observation at this point is that the Displacement measure is of little use past a distance of about $2\sigma_b$ of the edge position. When the standard deviation of the measurement exceeds the range over which the Displacement can be expected to provide reliable measurements, the Displacement measure should be disregarded. If a high confidence (say 2σ or 97%) is desired that the displacement is within these bounds then it is reasonable to limit $\sigma_d(x)$ to be less than σ_b. Using the above relation, then, the contrast cutoff is simply

$$I'(x) > \sigma_b \sigma_n. \tag{4.6}$$

For most purposes, these image invariant properties will not provide any additional information on relative *feature* variance, and thus can be left out of the computational variance model:

$$\frac{1}{\sigma_d(x)} = \alpha g(d_o(x)) \approx I'(x).$$

In most instances in this and future chapters, unless a precisely scaled model of

variance is needed, these invariant terms will be omitted.

The variance model developed in this section is important for reasons beyond simply testing the accuracy of the model, although this is a goal of this analysis. It also allows practical algorithms to incorporate all of the spatial information in the Displacement and Disparity representation in a maximum likelihood sense by use of inverse variance weights.

4.4.3 Discrete Models of Variance

When the filter width becomes small relative to the sample spacing, the continuous model needs some refinement. The previous section developed a model of the variance of the Displacement function using continuous domain assumptions and unbounded integrals. One important step for both the least-squares and the maximum likelihood analysis was:

$$\int I'(x)g'(d_o(x)) = 0$$

from equation B.1.

In a discrete imaging system these integrals become summations, and the domain boundaries are finite (say x_0 and x_N). The discrete form of the above integral becomes:

$$\sum_{i=0}^{N} I'(x_i)g'(d_o(x_i)) = \sum_{i=0}^{N} g(d_o(x_i))g'(d_o(x_i)) + \sum_{i=0}^{N} n_a(x_i)g'(d_o(x_i))$$

While it might be reasonable to argue the first summation on the right-hand side is diminishingly small in practice, the second (noise) term is a stochastic sum that in realistic cases — i.e. with small support σ_b — will be nonzero.

The example shown in Figure 4-8 uses a fairly large support — $\sigma_b = 32$ — so these effects are reasonably small. When a smaller Gaussian size is chosen, the results look somewhat different.

Figure 4-11 shows what happens when the size of σ_b is reduced to 8 pixels. Note

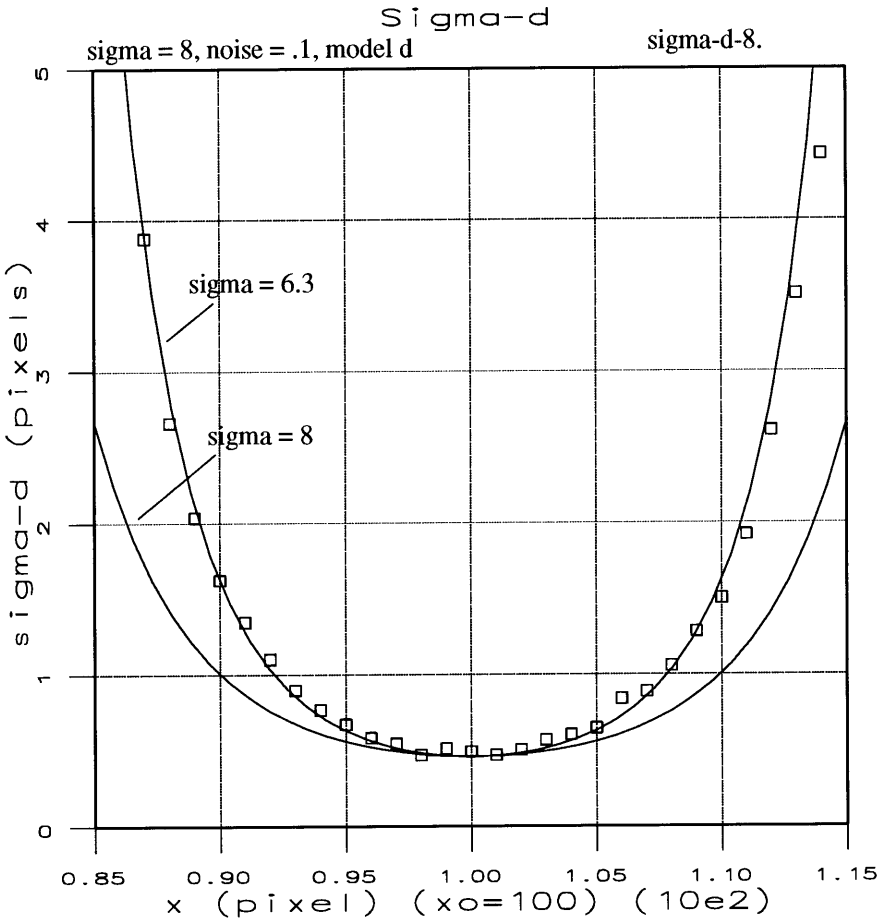

Figure 4-11: Noise Model vs. Small Support Trials

that the model curve (indicated by the "sigma=8" arrow) deviates significantly from the measured standard deviation of a large ensemble trial set — indicated by the squares in the figure.

It would appear that all that is needed to restore the model fit, perhaps, is to simply revise the width of the Gaussian of the noise model. Recall that the model derived was:

$$\frac{1}{\sigma_d(x)} = \alpha g(d_o(x)).$$

The function $g(d_o(x))$ has the same width — σ_b — as the underlying image smoothing function. Since the curvature of the variance distribution depends directly on this width parameter, perhaps all that is needed is a somewhat smaller Gaussian to match the ensemble data:

$$\frac{1}{\sigma_d(x)} \propto g_w(x).$$

where the w suffix indicates a width σ_w that is empirically derived from the ensemble trials.

Happily, least-squares analysis can also be used with a Gaussian sampled function to arrive at the width σ_w in the same manner that x_o was calculated. As derived in Appendix B, the best estimate for σ_w^2 given a Gaussian signal $y(x)$ corrupted by noise is (Equation B.5).

$$\sigma_w^2 = 2\left(\frac{\int x^2 y(x) g_w(x) dx}{\int y(x) g_w(x) dx} - x_o\right).$$

In these trials, x_o is a known constant (here it is 100).

When the reciprocal of the ensemble data shown in Figure 4-11 is solved in the best least squares sense using the above recursive algorithm, the result for the data of Figure 4-11 is $\sigma_w = 6.3$, instead of the $\sigma_b = 8$ of the continuous model. This is also plotted in Figure 4-11 and the fit, once again, appears quite good.

To test the fit, the original filter size of 32 pixels was tested using a large ensemble of trials (1000) and the signal to noise figure of 10:1. The resulting Gaussian fit to

the inverse variances resulted in a σ_w width of 30.9 pixels. This is plotted against the ensemble data in 4-12. When the residual errors — the differences between the ensemble data and the model — are examined, the maximum deviation found in this data is 0.7 pixels. The standard deviation of the residuals is 0.2 pixels.[2] These very small residual errors indicate that a simple Gaussian model for the Displacement variance distribution is extremely accurate, especially when the width σ_w is adjusted for small support sizes.

This process can be repeated for any size filter. In this model, integral powers of two are used for convenience. Repeating the experiments for filter sizes between 2.0 and 32.0, the data arrived at for σ_w is shown in Table 4.3. As a proportion of the width, the smaller filters appear to have a much larger divergence to the continuous model. In fact, to a surprising degree, the deviation of the discrete model width to the continuous model can be treated as a nearly *constant* deviation of about 1.5 pixels (see Figure 4-13),

$$\sigma_w \approx \sigma_b - 1.5.$$

One explanation for this phenomenon is that the second derivative operator in the discrete case is actually a close approximation to the second derivative of a narrow smoothing filter. When this is convolved with the image, the effective width of the σ_b Gaussian is increased slightly. This can not be eliminated since there is no way of making the derivative operator have a smaller support.

4.4.4 Algorithm Designs for d_o

In both the least-squares and the maximum likelihood solutions for estimating x_o, the function $g(d_o(x))$ (or $g_w(x)$) is needed. It has been noted that for the least-squares

[2]The Kolmogorov-Smirnov measure of the deviation of the residual CDF to a normal CDF is 0.14. The probability that the deviation to the distribution exceeds this is calculated to be less than 1 part in 10^5.

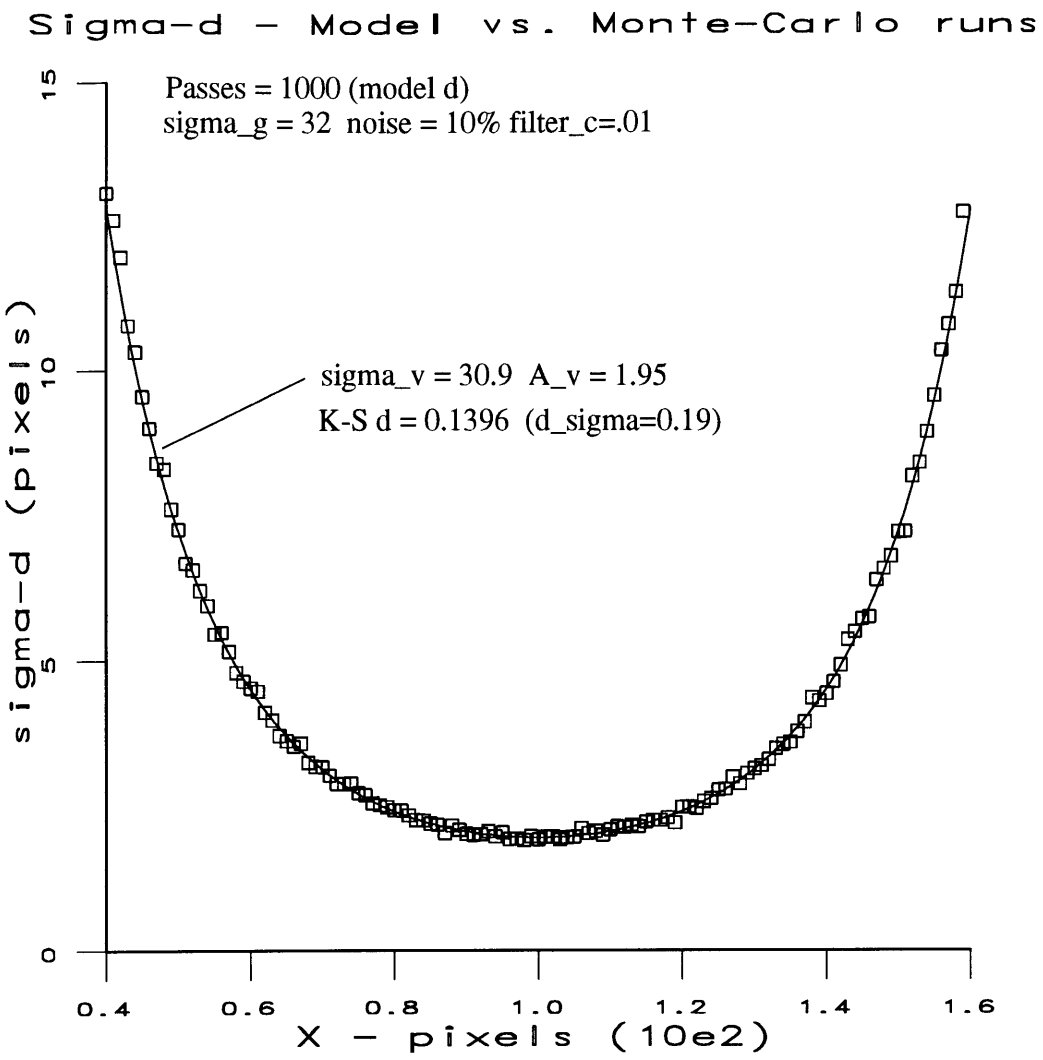

Figure 4-12: Noise Model vs. Large Support Trials

σ_b	σ_w
32.0	30.9
16.0	14.4
8.0	6.3
4.0	2.6
2.0	0.8

Table 4.3: Discrete Displacement Variance Widths

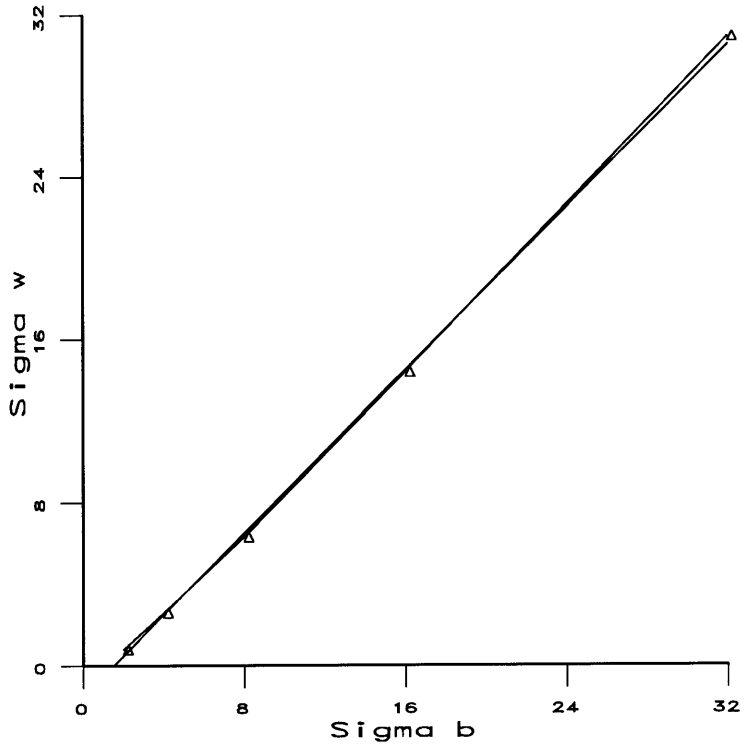

Figure 4-13: Discrete Displacement Variance Widths vs. $\sigma_w = \sigma_b - 1.5$

solution, recursion can be used. An approximate value for x_o is used in the first pass, and the result of each pass uses the x_o estimate of the previous iteration. There is a very rapid convergence, so this approach proves satisfactory in practice.

Another approach is possible in the case of estimating d_o that does not require recursion. The original approach for finding d_o was

$$d_o = \frac{\sum_{x=0}^{N} g_w(x) I'(x) x_d(x)}{\sum_{x=0}^{N} g_w(x) I'(x)}. \tag{4.7}$$

Another approach is to use convolution:

$$d_{co}(x) = \frac{g_w * [I'(x) x_d(x)]}{g_w * I'(x)} \tag{4.8}$$

and then sample $d_{co}(x)$ at $x = x_o$. This will be equivalent to *do*:

$$d_o = d_{co}(x)|_{x=x_o}$$

If x_o happens to be an integer, then sampling the ratio of the convolutions of equation 4.8 produces an equivalent measure of d_o as would be found in the implicit solution of equation 4.7. This is due to the fact that convolution $g_w * I'(x)$ effectively shifts the Gaussian by the independent variable x. It also is not difficult to determine where to sample since $I'(x)$ will be maximum at $x = x_o$, the convolution $g_w * I'(x)$ will be maximum at x_o as well. One simply samples the ratio at the peak of the denominator convolution. The denominator peak sample also produces a best least squares estimation of the contrast α as derived in Appendix B Equation B.3. Since both $d(x)$ and $I''(x)$ are zero at x_o, zero-crossings of these functions are also appropriate points to sample $d_{co}(x)$ to arrive at d_o.

Of course, x_o will not be an integer in practice, so it will not be possible to sample precisely at $x = x_o$. Given the smoothness of these functions, however, it is reasonable to assume that sampling one or even more pixels away from x_o will produce results at

very close to optimal accuracy. This presumption will be put to the test shortly, when this algorithm is compared to the least squares solution of the input $I'(x)$ signal.

In the previous section, it was observed that σ_w becomes exceedingly small in discrete models as σ_b approaches a width of approximately 1.5 pixels. An obvious question is raised about how to handle such small widths. When Gaussian widths approach zero, the Gaussian convolver takes on the form and function of the Dirac (or Kronecker) Delta sampling function. In such instances, the convolution leaves the signal unchanged and can be simply omitted from the calculation. Thus when σ_b is very small:

$$\begin{aligned} d_{co}(x) &= \frac{[I'(x)x_d(x)]}{I'(x)} \\ &= x_d(x) \end{aligned}$$

Thus the maximum-likelihood estimate of the Displacement for such small width filters becomes the Displacement sample nearest the edge position x_o.

In practice, especially when analog or biological systems are considered, both the above $d_{co}(x)$ function and the noisy $x_d(x)$ on which it is based are pathological because they increase monotonically across the domain of the image. If there are N pixels in the domain, when an edge is near pixel N these functions must have a dynamic range as large as N. The original Displacement function $d(x)$ was not so constrained since it is only valid over one or two times its width σ_b. This bounds the maximum dynamic range to be roughly $\pm 2\sigma_b$.

This means that for practical biological and analog electronic potential implementations, or even digital computation, $d_{co}(x)$ is not a very interesting function. Indeed, there is little or no obvious need for it beyond the main purpose of this chapter, which is to determine the variance $\sigma_d^2(x)$ of the Displacement function $d(x)$. The reason for developing the above algorithm is to test that $d_{co}(x)|_{x=x_o}$ is indeed equal (or nearly

Measure	$\sigma_b = 32$	$\sigma_b = 16$	$\sigma_b = 8$	$\sigma_b = 4$	$\sigma_b = 2$
$\sigma[x_o]$	0.468	0.311	0.193	0.118	0.047
$\sigma[d_o]$	0.477	0.295	0.163	0.083	0.027

Table 4.4: Displacement $\sigma[x_o]$ and $\sigma[d_o]$ vs. various σ_b sizes.

so) to the linear least squares solution of x_o.

The convolution approach developed in this chapter to arrive at the $g(d_o(x))I'(x)$ inverse variance weighting for the Displacement function can also be used in Disparity models. In those designs, however, the dynamic range is limited to the small multiple of σ_b of the Displacement and Disparity calculations. With limited precision biological or analog electronic models, this is where such processing would probably be implemented.

On the other hand, the reason for going through all this effort to create a good variance model is so that in subsequent stages of processing Disparity representations, the variance estimates can be preserved and, more importantly, used in the algorithms to insure accurate results.

4.4.5 Comparisons between Least Squares and Maximum Likelihood methods

Experiments were run to determine the performances of

- The Gaussian denominator $I'(x)$ based least squares estimate of x_o — $\sigma[x_o]$,
- The maximum likelihood weighted Displacement estimate $d_{co}(x)|_{x=x_o}$ — $\sigma[d_o]$.

These results are tabulated in Table 4.4 and plotted in Figure 4-14 for the filter sizes σ_b ranging between 2.0 and 32.0 and added noise of 10% ($\rho = 10$).

The central observation, especially with the larger filter sizes, is that the maximum likelihood estimate $\sigma[d_o]$ performs at least as well as the least squares solution based on $I'(x)$ — $\sigma[x_o]$. Over all the tests of all filter sizes, noise amplitudes, and subpixel offsets, there was never encountered any significant difference between these

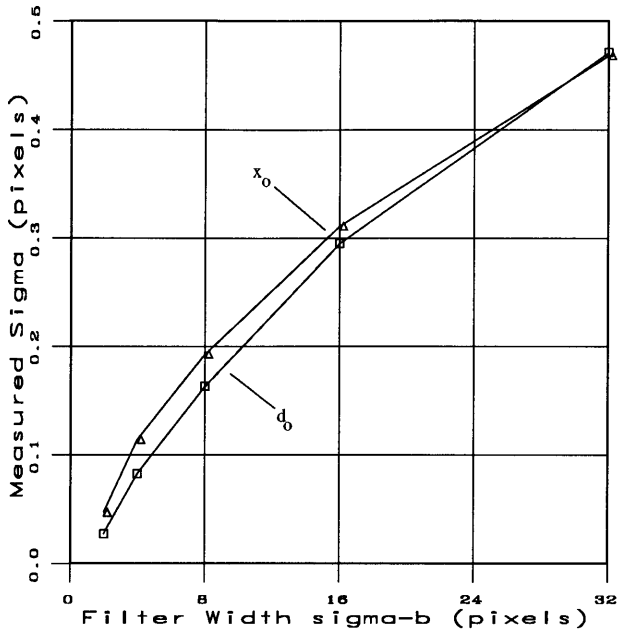

Figure 4-14: Displacement $\sigma[x_o]$ and $\sigma[d_o]$ vs. various σ_b sizes.

two variance measures, with the interesting exception that with *small* σ_b sizes, the Displacement estimate $\sigma[d_o]$ has a *lower* variance than the least squares measure $\sigma[x_o]$. This is probably due to the attention given the discrete Displacement model in this chapter, whereas the linear model was not modified in a similar manner. It should also be noted that there was no measurable mean error in measuring x_o in any experiment regardless of sub-pixel offsets of x_o.

4.5 Using Variance Measures in Disparity Models

When Displacements are subtracted or differentiated — both linear operations — to produce a Disparity function, $D(x)$, then the variance of the Disparity function is easily calculated. When Displacements are subtracted, their variances add. The "weight" — or inverse variance — associated with this Disparity measure is the inverse

of the sum of the associated Displacement variance measures:

$$\begin{aligned} \sigma_D^2 &= \sigma_{da}^2(x) + \sigma_{db}^2(x) \\ w_D(x) &= \left(\frac{1}{w_{da}(x)} + \frac{1}{w_{db}(x)} \right)^{-1} \end{aligned}$$

When differentiation is involved, such as with motion Disparity for optical flow, a simple scaling of the weight function can be used if errors are assumed to be correlated. Indeed, in all approaches to combining Displacement weights to get Disparity weights some assumptions must be made about whether the errors are correlated or not.

In stereo and motion, instead of convolving the biased Displacement function $x_d(x)$, with $g_w(x)$ as above, it is convenient to defer the convolution to *after* the Disparity calculation. Unlike the d_o calculation this has the advantage of greatly limiting the dynamic range, thereby making analog and biological models plausible. This weakens the arguments of optimality when the Disparity measure is large, so it is a good design practice to align the left and right images locally as best as possible. In the limit, when isolated edges are aligned, the convolved Disparity model yields optimal stereo disparity estimates.

To achieve this local alignment, the scale-space algorithm is used. It takes large σ_b filter Disparity measures to locally "shift" the alignment of the smaller σ_b stages. This algorithm will also be used in the full 2D model which is developed in following chapter and will be discussed in more detail there.

Another Disparity formulation is the Stereo Cyclopean Displacement model discussed in the previous chapter. This representation takes two images and creates a single image from them. In the stereo model, the variances of the features added since a missing or weak feature in *either* image should result in missing or weak Disparity

estimate. In fused vision, this is also a valid model.

$$\begin{aligned} d_{SC}(x) &= \frac{d_r(x) + d_l(x)}{2} \\ w_{SC}(x) &= w_D(x) \end{aligned}$$

Another approach is to accept a feature in *either* image. In this case the Fused Cyclopean Displacement is a weighted average of the left and right Displacement functions (assuming $w_l(x) + w_r(x) \neq 0$):

$$d_{FC}(x) = \frac{w_l(x)d_l(x) + w_r(x)d_r(x)}{w_l(x) + w_r(x)}$$

When weighted averages are based on inverse variance weights, as was done in the maximum likelihood analysis, the variance of the average is:

$$\frac{1}{\sigma_C^2(x)} = \frac{1}{\sigma_l^2(x)} + \frac{1}{\sigma_r^2(x)}. \tag{4.9}$$

This leads to a model for the weight of the Fused Cyclopean image. Since the inverse variance of a Displacement model is proportional to $I'(x)G(x)$, the candidate inverse variance model for the Fused Cyclopean Displacement must be

$$I'_C(x)g_C(x) = I'_l(x)g_l(x) + I'_r(x)g_r(x).$$

In other words, when the fused Displacement is created by a variance weighted averaging of the left and right images, the weighting Gaussian functions needed for the resulting fused form are simply the sum of the individual component functions.

4.6 Superposition of Features

With the variance models developed in this chapter, it is possible to examine a central issue to the Displacement/Disparity model — the effect of multiple features on a nonlinear simple edge model such as this. All interesting images are composed of a multitude of edges. Indeed, all images can be reconstructed trivially by allowing arbitrary contrast edges at each pixel boundary. Clearly some elaboration is needed on the effects of multiple edges on this single edge theory.

The simplest such image is a two edge image. When two edges are isolated, they interact very little, as might be expected, and the model as presented to this point proves quite adequate to explain the behavior of the Displacement function. At the midpoint between the two edges, some ambiguity exists as to which edge is "nearest" and thereby which distance is measured, but at that point the weight w_d is at a local minima, so no Disparity estimate is likely to be taken (recall that with Disparities, the edges are samples at the local maxima of w_D).

As the edges become proximate — when they come within 2 or 3 σ_b of each other, for example — they fall "under" the Gaussian. They begin to interact in the nonlinear Displacement calculation. Importantly, they interact in ways that depend very much on the signs of their respective contrasts, α. When the edges are of like sign, then the image looks like a staircase. This two step edge feature is called here a *Chevruel.* When the edges are of opposite sign and they approach under the Gaussian, the feature produced by the edges is called a *Delta.*

4.6.1 Chevruel Features

Characteristics of Chevruel features have been studied in human vision at some length [23, 75, 76, 77]. A Chevruel at the limit of only a pixel separation is little different in behavior than the simple isolated edge. It is strikingly unlike the simple edge when the spacing approaches about $2.5\sigma_b$. Measured Disparities are significantly

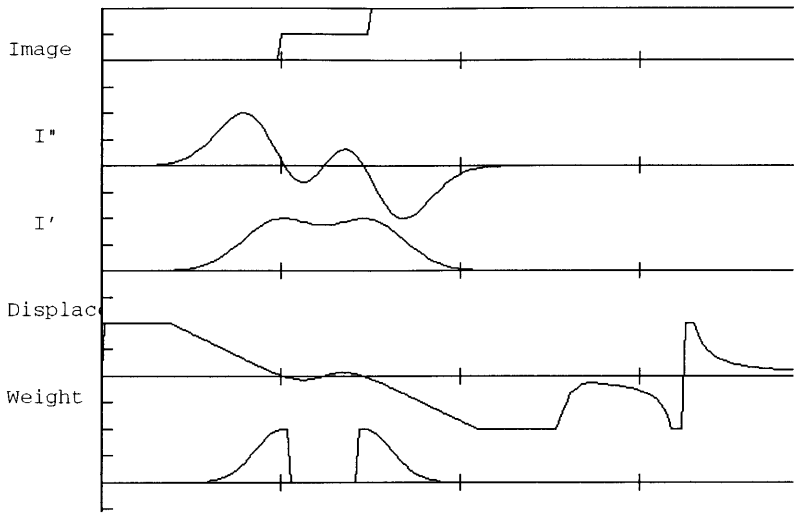

Figure 4-15: Chevruel Effect — Top to bottom, $I_m(x), I''(x), I'(x), d(x), w_d(x)$

underestimated when such stepwise images are used in the Displacement model. The qualitative effect is the same as a blurry edge, so it stands to reason that the Disparity estimates would be low as they would with any image blurring.

An important characteristic, however, of these features is that the underestimation is almost entirely due to the presence of a Displacement component between the edges near the location of the "false" edge cited by the researchers which has a slope of -1.0. When the Displacement slope, or focus Disparity, is tested to eliminate any negative components, the Disparity underestimation due to these illusory edges is completely eliminated.

Figure 4-15 shows how the computational model responds to a Chevruel feature where the edge spacing results in the largest false Displacement measure. The top plot shows the input image $I_m(x)$. The next plot of $I''(x)$ has three zero-crossings. Two correspond to the actual edges. The middle zero-crossing corresponds to the illusory edge. The next plot shows $I'(x)$ which, as an inverse variance measure, is nonzero throughout the region. The fourth plot shows the resulting displacement calculation $d(x)$. Note the slope reversal in the vicinity of the false edge. This is the culprit in the Disparity underestimation. The bottom plot shows where the weight function $w_d(x)$

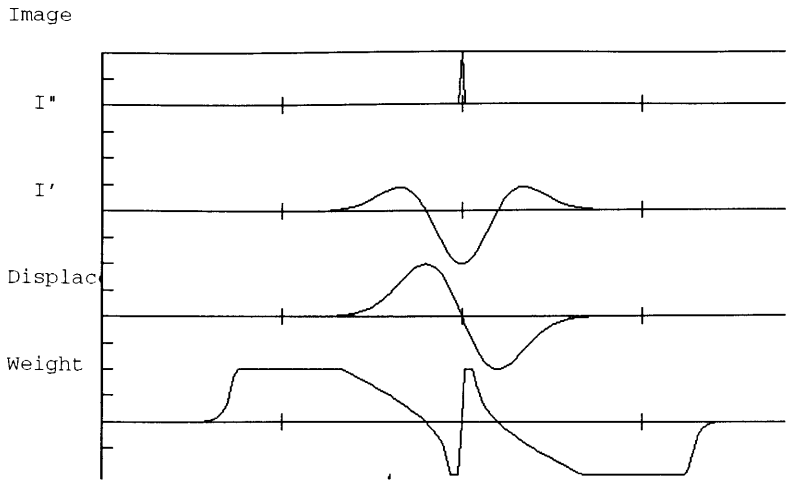

Figure 4-16: Delta Effect — Top to bottom, $I_m(x), I''(x), I'(x), d(x)$

is zeroed when the detected Displacement slope is less than zero. This adjustment to the model renders the Chevruel feature Disparity measurements accurate.

4.6.2 Delta Features

The companion to the Chevruel feature is the Delta feature which is defined as two opposite sign edges under the Gaussian. This results at the limit when the edges are a single pixel apart in the Dirac delta function as an image input. Figure 4-16 shows what happens to the Displacement model with such a feature as its input. As with the previous section, the first three plots show the image $I_m(x)$, second derivative $I''(x)$, and first derivative $I'(x)$ plots. The bottom plot of the Displacement function $d(x)$ shows a decidedly pathological trait at the delta input location x_o. While the Displacement slope is correct, it increases as the position approaches x_o from either direction. This will result in a local *overestimation* of Disparities due to the increased slope.

This can be understood better by looking at the continuous model with an input $I_m(x) = \delta_k(d_o(x))$. The Kronecker delta function, which has unit area, but unbounded amplitude at x_o, is used for the continuous analysis.

$$\begin{aligned}
I''_\delta(x) &= \alpha g''(d_o(x)) \\
I'_\delta(x) &= \alpha g'(d_o(x)) \\
d_\delta(x) &= -\sigma_b^2 \frac{\alpha g''(d_o(x))}{\alpha g'(d_o(x))} \\
&= -\sigma_b^2 \frac{\frac{d_o^2(x)-\sigma_b^2}{\sigma_b^4} g(d_o(x))}{-\frac{d_o(x)}{\sigma_b^2} g(d_o(x))} \\
&= \frac{-\sigma_b^2}{d_o(x)} + d_o(x)
\end{aligned}$$

This result shows that the delta input Displacement function $d_\delta(x)$ has two components. The second — $d_o(x)$ — is the desired displacement measure found with the step edge input. The other component — $\frac{-\sigma_b^2}{d_o(x)}$ is the pathological discontinuity found at x_o.

When two different filter sizes are used, say σ_a and σ_b, then the Displacement functions for each can be used to eliminate the singularity at the origin.

$$\begin{aligned}
\frac{d_a(x)}{\sigma_a^2} &= \frac{-1}{d_o(x)} + \frac{d_o(x)}{\sigma_a^2} \\
\frac{d_b(x)}{\sigma_b^2} &= \frac{-1}{d_o(x)} + \frac{d_o(x)}{\sigma_b^2} \\
\frac{d_a(x)}{\sigma_a^2} - \frac{d_b(x)}{\sigma_b^2} &= \frac{d_o(x)}{\sigma_a^2} - \frac{d_o(x)}{\sigma_b^2} \\
&= d_o(x)\left(\frac{1}{\sigma_a^2} - \frac{1}{\sigma_b^2}\right) \\
\frac{\sigma_a^2\sigma_b^2}{\sigma_b^2 - \sigma_a^2}\left(\frac{d_a(x)}{\sigma_a^2} - \frac{d_b(x)}{\sigma_b^2}\right) &= d_o(x)
\end{aligned}$$

This may appear laborious, but it really is only a simple linear combination of Displacement representations at different scales. Although it is often a dubious proposition to eliminate singularities by subtraction, and the solution is *still* undefined at

$x = x_o$, the inverse variance measure $I'(x)g(x)$ is *also* zero at $x = x_o$ and in practice this compensation step completely corrects for the delta effects without any need for careful crafting of the computation.

4.6.3 Scale Space Approaches to Complex Images

From these examples, it might appear that the problem of superposition — the property of a linear model whereby the behavior of the model to complex linear combinations of simple stimuli is the linear combination of responses to the simple stimuli — is clearly lost with the nonlinear Displacement design. Indeed, these two most simple of edge combinations resulted in quite different deviations to the simple Displacement edge response. These remedies needed to bring the model responses into close agreement with the edge response were nontrivial to devise, and were carefully crafted to the particular problems the interacting features created.

If this process were to repeat itself with *three* step features — there would presumably be something like three generic flavors of these features this time, not two — then quite soon the model would presumably be so burdened with machinery for such repairs, it would be impractical to consider for more complex imagery. Happily, this appears not to be the case.

When any arbitrary combination of edges — of any sign, contrast, spacing, or number — is assembled under the convolving Gaussian, it can be considered a "template" feature. This set, having many edges, should have many Displacement measures at any image point. But when taken as a group, the Displacement measure will assign a single measure of distance to the feature "location". When the sets are shifted and the Disparity is taken by subtracting the Displacement functions, it is found that the Disparity is correct to within a small fraction of the actual shift. In practice, the measured Disparity is always observed to be well within $\pm\frac{\sigma_b}{2}$ of the true shift.

If the image is shifted by the measured Disparity and the scale σ_b reduced by some fraction (say $\frac{1}{2}$), and the Displacement/Disparity measures recalculated, then

ultimately the individual edges can be aligned to arbitrary precision. Indeed, at the smallest scale where $\sigma_b < 1$ pixel , features do *not* interact, the Displacement function is a true and optimal measure of edge position and Disparity estimates will be equally accurate.

It is impossible to test every possible combination of edge contrasts, signs, locations, and number to test whether the above observation of an upper bound on Displacement errors is universally true. The nonlinearity of the model also makes a theoretical treatment of general superposition of edges problem very difficult to formulate. As of now, only the observation of a large variety of synthetic and real imagery leads to the claim that once the corrections for Chevruel and Delta features described in this section are taken into account, the Displacement model can use the scale space scheme just described to correctly correspond image features. At the smaller scales where edges become isolated, Displacements and Disparities will provide accurate measures for these discrete features.

4.7 Summary

The primary objective of this chapter has been to develop a good model of the variance distribution of a Displacement function. This is made difficult by the nonlinearity of the calculation, but it happens that there is a direct way of developing such a model.

A linear least squares analysis is possible on the linear processing of the image signal to provide a best estimate of the edge position x_o. It is also possible to pose a model of the Displacement variance that, when used in a maximum likelihood estimation of edge position d_o, is analytically and experimentally identical to x_o.

This variance model, $w_d(x) = \alpha g(d_o(x))I'(x)$ is the central result of the analysis. Other important observations were made along the way, however:

- The best least squares estimate of x_o using the Gaussian denominator function $I'(x)$ is the first moment of the product $I'(x)g(d_o(x))$.

- Using this x_o estimate, it is possible to test simple sensor/processor models to see which perform best with added noise. The 1D model that transmits the second derivative of a Gaussian convolved image (i.e. processed with a transfer function of $G\omega^2$) is clearly superior to lower order models.

- To reconstruct lower order derivatives at the processor stage, it is possible to make use of optimal filter theory to construct nearly optimal and very efficient and stable filters.

- Maximum likelihood analysis demonstrates that the $I'(x)g(d_o(x))$ function that has x_o as its first moment, is also proportional to the inverse variance weighting function in the maximum likelihood calculation. Experiments demonstrated that this weighting renders the continuous Displacement estimate d_o equivalent to the best least squares estimate x_o.

- Some modifications to the model can and should be made when the discrete nature of the implementation and the small size of the filters results in an imprecise match between the model and experimental results.

- A convolution model is also proposed for generating both optimal Displacement and Disparity estimations.

The result of these many steps is a noise model that renders excellent results at estimating edge position. By demonstrating this, a great deal of confidence can be justified in the model as a whole. Figure 4-17 shows the revised 1D model using the results developed in this chapter. The convolution of $w_d(x)$ and $d(x)$ with $g(x)$, as recommended, is not shown since this is deferred computationally until the Disparity calculation. This will be discussed more in the following chapter on the 2D model.

Finally, a side result of the foregoing analysis was a demonstration of the justification for the Marr-Hildreth Laplacian of Gaussian retinal model. Although this is a 1D model, this analysis provides additional support for the transmission of this single representation from retina to cortex based on simple signal communication criteria.

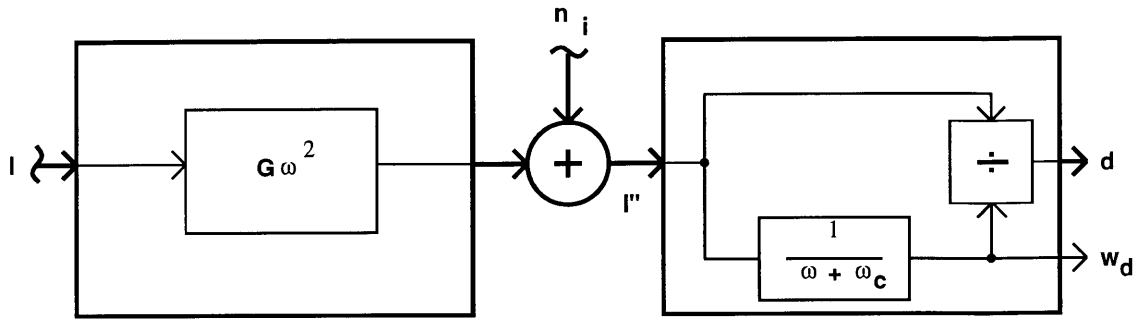

Figure 4-17: 1D Weighted Displacement Model

The variance model was also calibrated such that it was possible to determine Displacement uncertainties in pixel units using global image noise and Gaussian filter width information. Using this it is possible to set an acceptable maximum on the variance measures for features in real images

The Displacement model of the preceding chapter developed the distance measure $d(x)$. This chapter developed a model of the variance of this function, and a weighting function $w_d(x)$ that is inversely proportional to this variance. That weight is, in fact, the feature contrast measure; the second of the three components of the full Displacement model. The next chapter completes the model development by adding the second spatial dimension — orientation — to the Displacement representation.

Chapter 5

The 2D Model

In the previous two chapters, the 1D models of Displacement and Disparity representations have been developed. The Displacement representation has measures of feature distance and contrast. Orientation is the final attribute that needs to be added.

One possible approach to measuring orientation would be to treat it as a Disparity representation, i.e. to measure the change in the $d(x)$ estimates as a function of the cross-section position y. If a 2D displacement function $d(x, y)$ is defined as simply the 1D $d(x)$ function defined earlier plus the y sample position for the 1D cross-section, then orientation Disparity could be measured as:

$$\theta(x, y) = \frac{\partial}{\partial y} d(x, y).$$

This is not the approach taken here. Instead, the 1D Displacement model is generalized to a 2D model. The 1D analysis of the preceding two chapters still applies since the 2D model can be expressed as very much like the 1D model in almost all respects.

This chapter redefines the basic distance function d_o now to be a function of both distance and orientation and shows, as was done in Chapter 3, how a simple

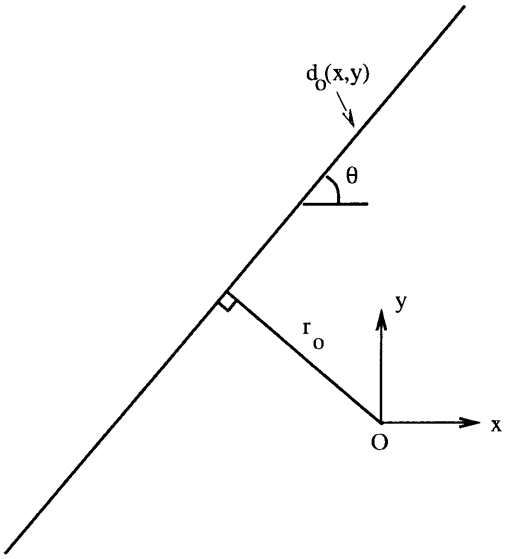

Figure 5-1: 2D Edge Function d_o

computational model can solve for it based on a Gaussian smoothed image. The Disparity models are revised for this new vector model, and the full stereo algorithm described in detail.

5.1 The General 2D Displacement Model

As with the 1D model, the first step is to define a function that at any point in the image domain measures the distance to an edge. In the original 1D model, the function $d_o(x)$ was defined as the distance to the point feature x_o. In the 2D world the edge is a line, not a point. As in the 1D model, there is a position parameter — r_o instead of x_o — but there is also an orientation parameter θ (see Figure 5-1):

$$d_o(x,y) = x\sin\theta + y\cos\theta - r_o.$$

Note that $d_o(x,y)$ is the signed distance to the edge from any point (x,y). The

image function with a single edge can be defined as it was with the 1D model:

$$I_m(x,y) = \alpha u(d_o(x,y)).$$

As before, α is the measure of image contrast. Although some redundancy exists in this representation, since more than one choice can be made for α, θ, and r_o to describe the same edge, no special restriction on the range of these parameters in imposed.

5.1.1 Gaussian Convolved Images

The image is convolved with a 2D Gaussian function. The 2D Gaussian can be expressed as the product of 1D Gaussians (subscripts are used to clarify the dimensions of the respective domains)

$$\begin{aligned} g_2(x,y) &= \frac{1}{2\pi\sigma_b^2} e^{-\frac{x^2+y^2}{2\sigma_b^2}} \\ &= \left(\frac{1}{\sqrt{2\pi}\sigma_b} e^{\frac{-x^2}{2\sigma_b^2}}\right)\left(\frac{1}{\sqrt{2\pi}\sigma_b} e^{\frac{-y^2}{2\sigma_b^2}}\right) \\ &= g_1(x) g_1(y). \end{aligned}$$

Convolutions of images with 2D Gaussians can be difficult to solve analytically in many cases, especially with the step edges used in the present model. For Laplacian and gradient of Gaussian convolved images, however, a simple convolution derivation is very useful in the analysis. The Gaussian convolution of the *delta* function located at the edge position $d_o(x,y)$ is:

$$g_2 * \delta(d_o(x,y)) = g_2 * \delta(x\sin\theta + y\cos\theta + r_o).$$

Since the Gaussian g_2 is rotationally symmetric, one can introduce a change of vari-

ables into the convolution, namely

$$a = x \sin\theta + y \cos\theta$$
$$b = x \cos\theta - y \sin\theta$$

thus producing an equivalent convolution

$$\begin{aligned} g_2 * \delta(d_o(x,y)) &= g_2(a,b) * \delta(a + r_o) \\ &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} g_1(b)db \right] g_1(\eta)\delta(a + r_o - \eta)d\eta \\ &= g_1(a + r_o) \\ &= g_1(d_o(x,y)). \end{aligned} \tag{5.1}$$

Note that the convolution of a 2D Gaussian with the edge delta function renders a 1D Gaussian cross-section normal to that edge function. Equation 5.1 will be used in the following analysis of gradient and Laplacian of Gaussian convolved images.

5.1.2 Gradient and Laplacian of Gaussian Convolved Images

The Gradient and Laplacian of Gaussian convolved images involve taking first and second partial spatial derivatives of the smoothed image

$$\begin{aligned} \mathbf{I}'(x,y) = \nabla g_2 * I_m(x,y) &= \left(\frac{\partial}{\partial x}\hat{\mathbf{i}} + \frac{\partial}{\partial y}\hat{\mathbf{j}} \right) g_2 * I_m(x,y) \\ I''(x,y) = \nabla^2 g_2 * I_m(x,y) &= \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) g_2 * I_m(x,y). \end{aligned}$$

The next step is to solve for $\mathbf{I}'(x,y)$ and $I''(x,y)$ given the isolated step image

$I_m(x,y) = \alpha u(d_o(x,y))$ [1]. Starting with the first partial derivatives, the associativity property of the convolution operation can be put to use:

$$\begin{aligned}\frac{\partial}{\partial x} g_2 * I_m(x,y) &= \frac{\partial}{\partial x} g_2 * \alpha u(d_o(x,y)) \\ &= \alpha g_2 * \frac{\partial}{\partial x} u(d_o(x,y))\end{aligned}$$

Since $\frac{\partial}{\partial a} f(g(a,b)) = f'(g(a,b)) \frac{\partial}{\partial a} g(a,b)$,

$$\begin{aligned}\frac{\partial}{\partial x} g_2 * I_m(x,y) &= \alpha g_2 * \delta(d_o(x,y)) \frac{\partial}{\partial x} d_o(x,y) \\ &= \alpha g_2 * \delta(d_o(x,y)) \sin\theta.\end{aligned}$$

This convolution has already been solved in Equation 5.1, so

$$\frac{\partial}{\partial x} g_2 * I_m(x,y) = \alpha g_1(d_o(x,y)) \sin\theta. \tag{5.2}$$

The same analysis will produce the solution to the first partial with respect to y

$$\frac{\partial}{\partial y} g_2 * I_m(x,y) = \alpha g_1(d_o(x,y)) \cos\theta. \tag{5.3}$$

Observing that $\hat{\mathbf{\Theta}} = \sin\theta\hat{\mathbf{i}} + \cos\hat{\theta}\mathbf{j}$ is a unit vector normal to the image edge, the gradient $\mathbf{I'}_s(x,y)$ can be expressed as

$$\mathbf{I'}_s(x,y) = \alpha g_1(d_o(x,y))\hat{\mathbf{\Theta}}. \tag{5.4}$$

The second partial derivatives can be derived from the first (Equations 5.2 and 5.3). This is the equivalent to computing the Laplacian by taking the divergence of

[1] As with previous chapters, functions based on this specific input will usually be indicated by a s subscript.

the gradient:

$$\begin{aligned}
\frac{\partial^2}{\partial x^2} g_2 * I_m(x,y) &= \frac{\partial}{\partial x} \alpha g_1(d_o(x,y)) \sin\theta \\
&= \alpha g_1'(d_o(x,y)) \sin^2\theta \\
\frac{\partial^2}{\partial y^2} g_2 * I_m(x,y) &= \frac{\partial}{\partial y} \alpha g_1(d_o(x,y)) \cos\theta \\
&= \alpha g_1'(d_o(x,y)) \cos^2\theta
\end{aligned}$$

so the Laplacian of Gaussian solution, being the sum of these, is:

$$\begin{aligned}
I_s''(x,y) &= \alpha g_1'(d_o(x,y)) \\
&= -\alpha \frac{d_o(x,y)}{\sigma_b^2} g_1(d_o(x,y)).
\end{aligned}$$

These two representations, $\mathbf{I}'(x,y)$ and $I''(x,y)$ form the linear representations on which the 2D Displacement model is based. Compare these with their respective 1D functions of Chapter 3 (Equations 3.2 and 3.1):

$$\begin{aligned}
I_s'(x) &= \alpha g_1(d_o(x)) \\
I_s''(x) &= -\alpha \frac{d_o(x)}{\sigma_b^2} g_1(d_o(x)).
\end{aligned}$$

What is new to the 2D model is that $d_o(x,y)$ measures the orthogonal distance from any 2D point to the edge and the 2D gradient encodes orientation as well as contrast. Note especially that the 2D model is essentially still 1D. The form of the gradient and Laplacian in the edge normal direction — $\hat{\mathbf{\Theta}}$ — is identical to the 1D model, and the functions are independent of position tangent to the edge. Figure 5-2 shows $I''(x,y)$ where θ is 30 degrees and $r_o = 3$.

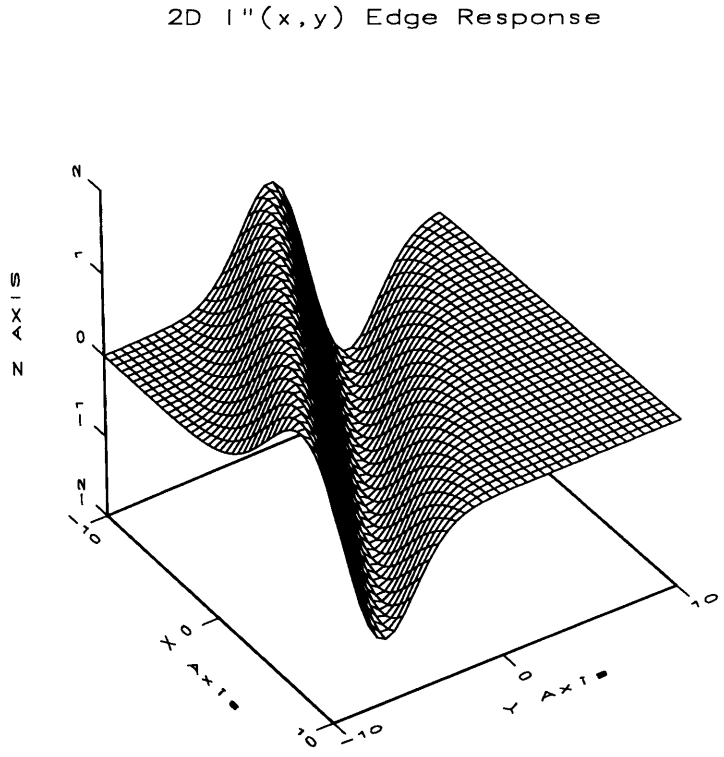

Figure 5-2: 2D $I''(x, y)$ Step Response

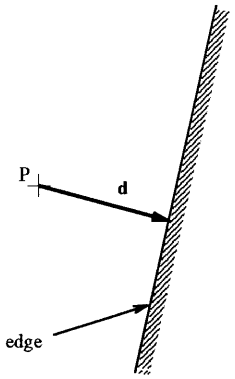

Figure 5-3: Displacement Representation **d**

5.1.3 2D Displacements

From these two representations, one can arrive at a Displacement estimate of $d_o(x,y)$ just as was done in the 1D case, except now it is a vector representation encoding not only the edge distance $d_o(x,y)$ but also its orientation $\hat{\mathbf{\Theta}}$:

$$\begin{aligned}
\mathbf{d}(x,y) &= -\sigma_b^2 \frac{I''(x,y)}{\|\mathbf{I}'(x,y)\|^2}\mathbf{I}'(x,y) \\
\mathbf{d}_s(x,y) &= -\sigma_b^2 \frac{I''(x,y)}{\|\mathbf{I}'(x,y)\|}\hat{\mathbf{\Theta}} \\
&= d_o(x,y)\hat{\mathbf{\Theta}}.
\end{aligned}$$

Figure 5-3 shows the 2D Displacement vector function as defined in Chapter 3. At any point P, the Displacement will point at the nearest isolated edge. It will be orthogonal to the edge and its magnitude will be the distance to the edge. The Displacement vector is not affected by the contrast or contrast sign of the edge, except for the fact that low contrast renders the measure sensitive to noise. This variance/contrast measure is developed in the next section.

5.2 2D Noise Models

The preceding section devised a vector field that provides the distance and orientation measures needed for the full Displacement model. The 1D weight model also needs to be updated for the 2D domain. The 1D noise analysis addressed the issue of I' reconstruction when a single signal format is transmitted. This section will provide the extension of this analysis into the 2D domain.

The 1D analysis developed a contrast/variance measure for the Displacement model. In short, the steps takes were:

1. Determine the best least squares estimate of edge position x_o based on the Gaussian form denominator signal $I'(x)$. This was found to be the first moment of the function $I'(x)g(d_o(x))$.

2. Show that in a maximum likelihood estimation of the nonlinear Displacement function $d(x)$ was equivalent to the best least squares estimate for x_o found above when the variance distribution was assumed to be inversely proportional to $I'(x)g(d_o(x))$.

3. Run experiments to confirm the above variance model, and examine how it needs to be modified for sampled domain implementations.

4. Test the model to see how the best edge estimate using displacements compares with the best least square estimates on the linear input representations. The ML displacement estimate d_o was shown to be identical in performance to the best least squares measure of x_o.

This section will not attempt to repeat the comprehensive analysis of the 1D Displacement model. The 2D model is in almost all respects identical to the 1D model, and those parallels will be used to draw conclusions for the 2D model based on the 1D analysis. As with the 1D model, however, a good measure of edge position r_o is needed based on one of the linear input representations — the gradient $\mathbf{I}'(x, y)$.

Chapter 4 also used the least squares x_o estimate to help determine which single signal format should be sent between the sensor and processor. This chapter will, based on that analysis and the model of Marr and Hildreth [50], assume that the transmitted signal is the noise corrupted Laplacian of Gaussian convolved image $I''(x,y)$.

5.2.1 Least Squares Estimation of 2D Edge Position

With the 1D model, one of the linear input forms, $I'(x)$, was chosen to develop a best least squares measure of feature position x_o. Any other linear transform of the noisy image could have been chosen, since the best estimate of x_o will not depend on which linear transform of the noisy image is selected. The analysis indicated that x_o was the first moment of the product of the $I'(x)$ signal and its Gaussian (noise-free) form $G(x) = g_1(d_o(x,y))$:

$$x_o = \frac{\int xI'(x)G(x)dx}{\int I'(x)G(x)}$$

Appendix B derives the best least squares estimate of r_o, the distance from the image origin to the edge function $d_o(x,y)$ using the Gradient of Gaussian convolved image, $\mathbf{I}'(x,y) = \alpha\mathbf{G}(x,y)+\mathbf{n}_i(x,y)$. The noise-free form of the function, as developed in the previous section, is $\mathbf{G}(x,y) = g_1(d_o(x,y))\hat{\mathbf{\Theta}}$. The best least squares measure of r_o is:

$$r_o = \frac{\int\int_R (x\sin\theta + y\cos\theta)\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR}{\int\int_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR}.$$

Once again, the best estimate of the edge position is based on a first moment of the Gaussian denominator of the 2D Displacement measure —$\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)$. Note, however, that r_o is the first moment of the Gaussian form in the direction *normal* to the edge.

5.2.2 Maximum Likelihood 2D Variance Models

In the 1D model, the maximum likelihood estimation of edge position d_o given a displacement function $d(x)$ and an inverse variance measure $w_d(x) = I'(x)G(x)$ was shown to be equivalent to the best least squares estimation of x_o based on the input signals. Interestingly, the best least squares measure used the first moment of the same $I'(x)G(x)$ function.

Since the forms of the 2D and 1D calculations are almost identical, there is good reason to expect that the maximum likelihood estimation of r_o using $\mathbf{d}(x,y)$ and an inverse variance measure $w_d(x,y)$ based on the above first moment result will have generally the same result. The last section demonstrated that the best least squares estimate of r_o is the first moment of the function $\mathbf{I}'(x,y) \cdot \mathbf{G}(x,y)$ in the direction of the edge normal $\hat{\mathbf{\Theta}}$. It would stand to reason that the inverse variance of the 2D Displacement function in that direction would be $w_d(x,y) = \mathbf{I}'(x,y) \cdot \mathbf{G}(x,y)$, and when used in in the maximum likelihood model, will produce the same result as the least squares analysis — r_o.

Defining the sample point $\mathbf{P}(x,y)$ as $x\hat{\mathbf{i}} + y\hat{\mathbf{j}}$, the maximum likelihood estimation of the edge position r_m using the 2D displacement $\mathbf{d}(x,y)$ is:

$$r_m = \frac{\sum_R w_{di}[\mathbf{d}(x,y) + \mathbf{P}(x,y)] \cdot \hat{\mathbf{\Theta}}}{\sum_R w_{di}}$$

where the inverse variance model is hypothesized to be proportional to

$$\frac{1}{\sigma_{di}^2} \propto w_{di} = \mathbf{I}'(x,y) \cdot \mathbf{G}(x,y).$$

The reason for the dot product is based on the geometry of the problem. Figure 5-4 shows a sample point $\mathbf{P}$ and its associated displacement $\mathbf{d}$. The scalar $[\mathbf{d}+\mathbf{P}] \cdot \hat{\mathbf{\Theta}}$ is the local estimate of r_o.

The reduction of this expression follows the same path as was performed in Section 4.4, so the reader is referred to that section for some of the details of the derivation.

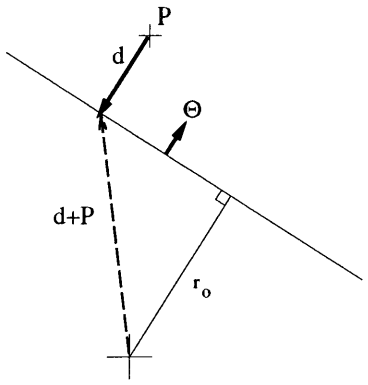

Figure 5-4: Maximum Likelihood Estimation of r_o

$$\begin{aligned} r_m &= \frac{\sum_R (\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y))[\mathbf{d}(x,y)+\mathbf{P}(x,y)]\cdot\hat{\mathbf{\Theta}}}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} \\ &= \frac{\sum_R (\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y))\mathbf{d}(x,y)\cdot\hat{\mathbf{\Theta}} + \sum_R (\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y))\mathbf{P}(x,y)\cdot\hat{\mathbf{\Theta}}}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} \end{aligned}$$

At this point, it is important to distinguish between local estimates of orientation $\hat{\mathbf{\Theta}}(x,y) = \mathbf{I}'(x,y)/\|\mathbf{I}'(x,y)\|$ and the actual orientation $\hat{\mathbf{\Theta}}$.

$$\begin{aligned} r_m &= \frac{\sum_R (\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y))\frac{I''(x,y)}{\|\mathbf{I}'\|}(\hat{\mathbf{\Theta}}(x,y)\cdot\hat{\mathbf{\Theta}})}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} \\ &\quad + \frac{\sum_R (x\sin\theta + y\cos\theta)(\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y))}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} \\ &= \frac{\sum_R (\hat{\mathbf{\Theta}}(x,y)\cdot\hat{\mathbf{\Theta}})I''(x,y)g(d_0(x,y))(\hat{\mathbf{\Theta}}(x,y)\cdot\hat{\mathbf{\Theta}})}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} + r_o \\ &= \frac{\sum_R (\hat{\mathbf{\Theta}}(x,y)\cdot\hat{\mathbf{\Theta}})^2 I''(x,y)g(d_0(x,y))}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} + r_o \end{aligned}$$

The fraction is very similar to that found in the 1D analysis with the exception of the $(\hat{\mathbf{\Theta}}(x,y)\cdot\hat{\mathbf{\Theta}})^2$ term. This is a stochastic error term that is difficult to deal with analytically. If one assumes that it, in effect, only introduces small random scaling

errors into the summation, then it is reasonable to continue the analysis:

$$r_m = \frac{\sum_R I''(x,y)g(d_0(x,y))}{\sum_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)} + r_o$$

This summation was shown in the previous chapter to be zero (Equation 4.4). Thus, assuming the effect of the $(\hat{\mathbf{\Theta}}(x,y)\cdot\hat{\mathbf{\Theta}})^2$ term in the summation can be neglected,

$$r_m = r_o.$$

5.2.3 2D Variance Models

The preceding maximum likelihood and least squares analyses lead to a variance model for the 2D Displacement measure:

$$\frac{1}{\sigma_{di}^2} \propto \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y).$$

Since the 1D model is identical in all respects to the 2D model so far, there is every reason to presume that the proportionality constant will be the same as well:

$$\sigma_{di}^2 = \frac{\sigma_n\sigma_b^2}{\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)}$$

but unless precisely scaled measures of variance are required, these constant terms can usually be dropped from variance models without affecting their performance. These terms will be dropped from the model in the following discussions.

As with the 1D model, since $\mathbf{I}'$ differs from $\mathbf{G}(x,y)$ only in the sense that it has additive uncorrelated noise, either $\mathbf{I}'$ or $\mathbf{G}$ can be used in the variance model, that is

$$\begin{aligned}\frac{1}{\sigma_{di}} &= \alpha\|\mathbf{G}(x,y)\| \approx \|\mathbf{I}'(x,y)\| \\ &= \alpha g(d_o(x,y)).\end{aligned}$$

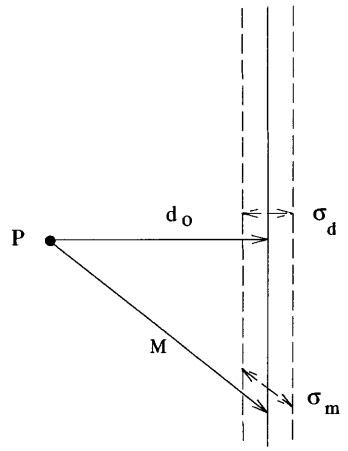

Figure 5-5: 2D Displacement Variance

One important aspect of the 2D model is that it can predict the distance to the nearest edge in *any* direction. For instance, in Figure 5-5, the distance to the edge along the normal to the edge from any point P is $d_o(x,y)$. But to estimate the distance to the edge in any other direction, the distance must be scaled by the angle between the edge normal $\hat{\mathbf{\Theta}}$ and the selected direction (say $\hat{\mathbf{m}}$):

$$m(x,y) = \frac{d_o(x,y)}{\hat{\mathbf{\Theta}} \cdot \hat{\mathbf{m}}}. \tag{5.5}$$

The variance model must also be adjusted, since the standard deviation of any scaled measure must also be scaled:

$$\begin{aligned} \sigma_m(x,y) &= \frac{\sigma_d(x,y)}{\hat{\mathbf{\Theta}} \cdot \hat{\mathbf{m}}} \\ \sigma_m(x,y)^{-1} &= \alpha g(d_o(x,y)) \hat{\mathbf{\Theta}} \cdot \hat{\mathbf{m}} \\ &= \alpha \mathbf{G}(x,y) \cdot \hat{\mathbf{m}} \end{aligned} \tag{5.6}$$

This demonstrates that $\alpha\mathbf{G}(x,y)$ (or $\mathbf{I}'(x,y)$) is a measure of the inverse standard deviation of the edge measure in any direction. Sometimes it is necessary to measure the distance to the edge in such oblique directions, so having the gradient

representation is a convenient way to predict the variances in any 2D direction.

It is also useful to pose the uncertainty of feature point location normal to the edge due to noise — $\sigma_d(x,y)$ — and the lack of any information for point positions tangential to the edge in a probabilistic model. Namely, given the variance model for location normal to the edge developed above ($\sigma_d^{-1} = \|\mathbf{I}'_i\|$), at some scene point $\mathbf{P}_i$ with associated edge Displacement measure $\mathbf{d}_i$, a model that measures the probability that any point $\mathbf{X}$ lies on the edge indicated by the Displacement function can be posed:

$$\begin{aligned} p(\mathbf{X} \text{ on the edge}|\mathbf{d}_i, \sigma_i) &= \frac{1}{\sqrt{2\pi}\sigma_d} e^{-\frac{[(\mathbf{X}-(\mathbf{d}_i+\mathbf{P}_i))\cdot\hat{\mathbf{\Theta}}]^2}{\sigma_d^2}} \\ &= \frac{\|\mathbf{I}_i\|}{\sqrt{2\pi}} e^{-[(\mathbf{X}-(\mathbf{d}_i+\mathbf{P}_i))\cdot\hat{\mathbf{I}}'_i]^2} \end{aligned}$$

In other words, the probability that $\mathbf{X}$ lies on the edge indicated by the local Displacement depends on the distance between $\mathbf{X}$ and the edge and is a normal distribution with the standard deviation σ_d. Note that this is a degenerate distribution, since the total cumulative probability is unbounded in this model due to the unbounded width of the 2D distribution in the direction tangential to the edge. Since no information exists with a single edge to bound this distribution, the problem at this point is under-constrained. In practice, however, any Displacement measure $\mathbf{d}_i$ will correspond to some feature within some relatively small radius proportional to the Gaussian width σ_b. There is very low probability that $\|\mathbf{d}_i\| > 2\sigma_b$ with noisy images. Thus it is possible to bound the probability distribution by using both the above $\sigma_d(x,y)$ width and the Gaussian filter width σ_b (see Figure 5-6).

The above distribution can be restated in another format:

$$\begin{aligned} p(\mathbf{X} \text{ on the edge}|\mathbf{d}_i) &= \frac{\|\mathbf{I}'_i\|}{\sqrt{2\pi}} e^{-(\mathbf{X}-(\mathbf{d}_i+\mathbf{P}_i))^T \Psi_i (\mathbf{X}-(\mathbf{d}_i+\mathbf{P}_i))} \\ \text{where} \quad \Psi_i &= \mathbf{I}'_i \mathbf{I}'^T_i \end{aligned}$$

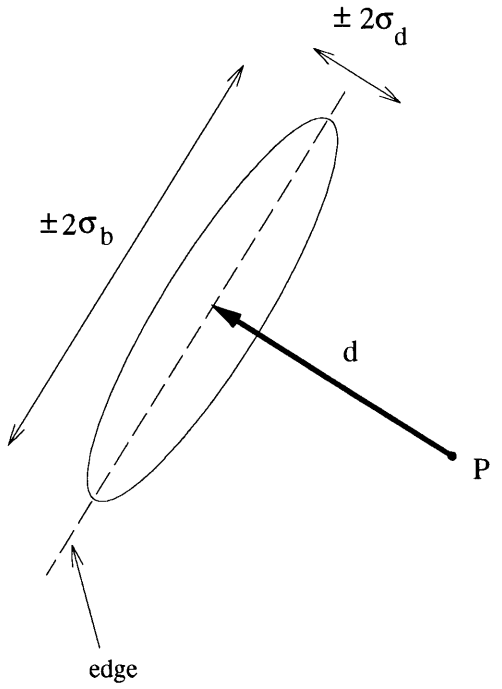

Figure 5-6: 2D Edge Point Distribution

Ψ_i is the covariance matrix of the 2D probability distribution and is, as expected, singular with the single edge model posed here. The addition of the σ_b distribution in the direction tangent to the edge is easily added. When $\mathbf{I'}_R$ is defined as the gradient vector $\mathbf{I'}$ rotated 90° ($\mathbf{I'}_R = I'_y\hat{\mathbf{i}} - I'_x\hat{\mathbf{j}}$), then

$$p(\mathbf{X} \text{ on the edge}|\mathbf{d}_i) = \frac{\|\mathbf{I'_i}\|}{\sqrt{2\pi}\sigma_b} e^{-(\mathbf{X}-(\mathbf{d}_i+\mathbf{P}_i))^T \Psi_i (\mathbf{X}-(\mathbf{d}_i+\mathbf{P}_i))} \tag{5.7}$$

$$\text{where} \quad \Psi_i = \mathbf{I'}_i\mathbf{I'}_i^T + \frac{1}{\|\mathbf{I'}\|^2\sigma_b^2}\mathbf{I'}_{Ri}\mathbf{I'}_{Ri}^T$$

This probability model will prove useful in many 2D Disparity models. 2D Disparity representation concepts will be discussed later in this chapter, and some of the issues relating to Disparity variance estimation will follow. As with the 1D model, however, it is useful to consider how an inverse filter for generating $\mathbf{I'}(x,y)$ from $I''(x,y)$ might be realized in practical implementations.

By itself, the above Displacement distribution is of little obvious value. Indeed, it may seem somewhat unusual to use measured quantities such as $\mathbf{d}_i$ and $\mathbf{I}'_i$ to build probability density distributions in such an obviously under-constrained situation. The Displacement measure $\mathbf{d}_i$, however, is the maximum likelihood estimate of edge position, given the observed data, so this probability model is the best estimate, given the data, of the relative probability of feature point locations. As will be seen, just as Displacements are really only useful in Disparity models, the above density function will begin to prove useful in those models.

5.2.4 2D Displacement Algorithm Implementations

In the previous chapter a sensor/processor model was posed that constrained the design to transmit a single image representation between the sensor and the processor. The 1D second derivative of a Gaussian convolved image, $I''(x)$, was shown to be the best choice based on a simple noise model.

In the 2D model, a Laplacian of Gaussian convolved image, $I''(x,y)$, would be the candidate of choice. As stated above, the 2D model has the same form and noise model as the 1D model in the direction normal to the edge and has infinite variance (zero weight) for the direction tangent to the edge. Any experiments to confirm the 2D variance model would presumably produce the same results as were found in the preceding chapter provided that an equivalent inverse filter could be devised to reconstruct $\mathbf{I}'(x,y)$ as was possible in the 1D model. Such an inverse filter was needed for the 1D testing to confirm the noise model.

Figure 5-7 shows the 2D system that would be required. The image is convolved with the Laplacian of Gaussian to produce the signal sent to the processing stage. Noise is added to this signal. The least squares analysis is based on the gradient representation $\mathbf{I}'(x,y)$. The edge position obtained is the best possible in this least squares sense for any linear transform of the noisy sensor signal $I''(x,y)$. The trick is to devise an inverse filter that can transform $I''(x,y)$ into the gradient representation

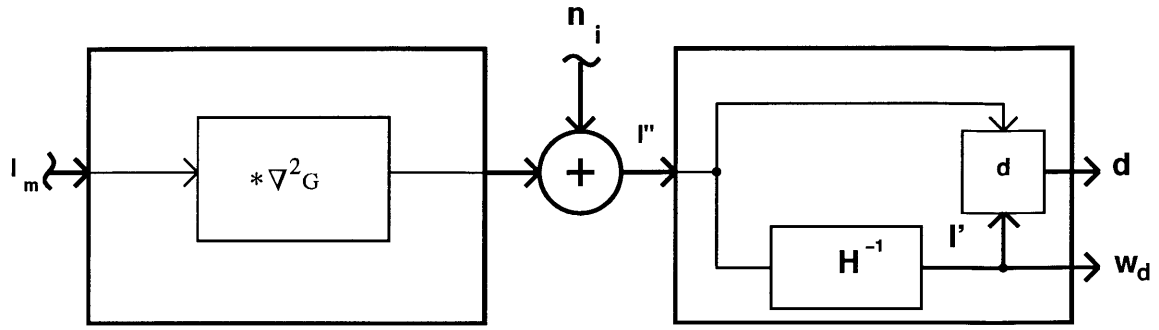

Figure 5-7: 2D Displacement Model

$\mathbf{I}'(x,y)$, indicated on the figure as the $\mathbf{H}^{-1}$ transform. This analysis will also be useful in considering the problem of reproducing the Gradient representation in analog or biological systems where only one representation can and should be transmitted between sensor and processor.

The need for such a reconstruction in algorithms implemented in digital computer systems can be avoided by taking the gradient of the Gaussian convolved image directly using Sobel or Roberts operators [4]. In analog or biological models, however, this approach would require fully three times as many signals be sent between the sensor and the processor — the single Laplacian and the two Gradient images. It would make sense to have the Gradient images be reconstructed from the single Laplacian representation as was done in the 1D model.

Unfortunately, the 1D filter analysis is not very helpful in attempting to design such an inverse 2D filter. Even the simplest — and least stable — 1D filter, the integrator, will not reconstruct the gradient from the Laplacian input. A new approach is needed.

One approach is to take the solution to the Poisson's equation for $I(x,y)$

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

and take the gradient of the restored smoothed image [36, 37]. When the boundary condition $I''(x_b, y_b) = 0$ for the boundary points (x_b, y_b) is satisfied, the solution is

found by finding the Green's function for the above equation. This function is then convolved with the $I''(x,y)$ function to restore $I(x,y)$. Thus it corresponds to the impulse response of the restoration filter. The Green's function is

$$g(r) = \frac{1}{2\pi}\log(r) + c$$

where $r^2 = x^2 + y^2$ and c is an arbitrary constant.

To arrive at the gradient, one approach would be to take the gradient of $g(r)$ and use the result as the convolving functions to arrive at $\mathbf{I}'(x,y)$ from $I''(x,y)$ directly. The gradient of the Green's function is

$$\nabla g(r) = \frac{1}{r}(\sin\theta\hat{\mathbf{i}} + \cos\theta\hat{\mathbf{j}}) = \frac{\hat{\mathbf{\Theta}}}{r}.$$

Note both $g(r)$ and $\nabla g(r)$ are undefined at the origin.

The y component of $\nabla g(r)$ is shown in Figure 5-8. This is very similar to the approximation to the 1D optimal filter developed in the previous chapter and Appendix A. In both cases, constraints imposed on the solution, either boundary conditions or band-limiting, result in an impulse response that can be convolved with the respective I'' input representation to produce the desired I' (or $\mathbf{I}'$) image. Although the forms of the solution are different, both require large support and are difficult to model in discrete sampled systems.

In the 1D model, a local recursive filter was designed that used both input I'' and neighboring output I' values to reconstruct a first order inverting filter with extremely small support – nearest neighbor — and the desired response characteristics. In the 2D model this can also be considered.

One such approach for the 2D model can be based on two constraining relations on the gradient field representation. The first is that the divergence of the gradient field is equal to the Laplacian image $I''(x,y)$. The second is that the curl of a gradient

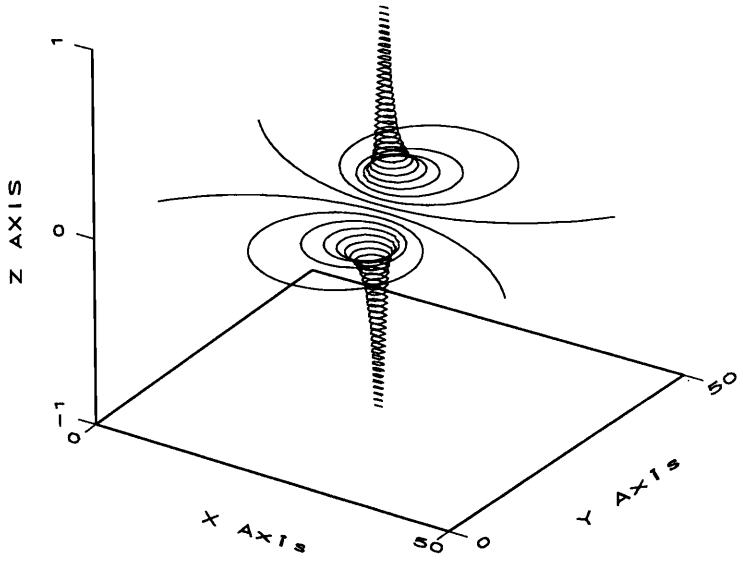

Figure 5-8: Green's Gradient Solution - $\hat{\mathbf{j}}$ Term

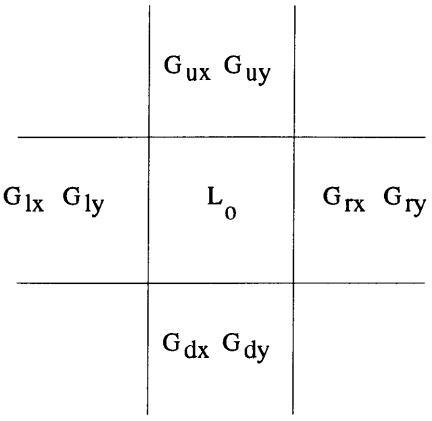

Figure 5-9: Discrete Sample Inverse Model

field is zero. If the gradient vector field is broken into its two scalar components:

$$\mathbf{I}'(x,y) = G_x(x,y)\hat{\mathbf{i}} + G_y(x,y)\hat{\mathbf{j}}$$

then these two constraints are:

$$\begin{aligned} \text{curl}_z\mathbf{I}'(x,y) &= \frac{\partial}{\partial y}G_x(x,y) - \frac{\partial}{\partial x}G_y(x,y) = 0 \\ \nabla \cdot \mathbf{I}'(x,y) &= \frac{\partial}{\partial x}G_x(x,y) + \frac{\partial}{\partial y}G_y(x,y) = I''(x,y). \end{aligned}$$

A recursive 2D filter can be designed that takes a local estimate of the curl and divergence of the gradient output and adjusts that output recursively to satisfy the above relations. One such design takes the adjacent gradient output estimates and computes the curl and divergence at some pixel. Figure 5-9 indicates the sampling of adjacent upper and lower (subscript u and b) and right and left (subscript r and l) gradient x and y components. These output samples can then be used to estimate the curl and divergence at the central pixel (subscript o):

$$c_o = G_{xu} - G_{yr} - G_{xd} + G_{yl}$$

$$d_o = G_{yu} + G_{xr} - G_{yd} - G_{xl}.$$

The input Laplacian $I''(x,y)$ representation is known at this central pixel, L_o, and can be used in the constraint equations to generate error measures:

$$\epsilon_{co} = c_o$$
$$\epsilon_{do} = d_o - L_o$$

These errors are used to revise the surrounding gradient estimates. For example, the curl error ϵ_{co} is scaled by some constant gain factor (nominally 1/4), and subtracted from G_{xu} and G_{yl} and added to G_{yr} and G_{xd}. Thus a subsequent recalculation of ϵ_{co} would be reduced proportional to the filter gain. This correction step is repeated for the divergence measures. Since any gradient sample will be adjusted by four neighbors directly, the process must be iterated until it converges.

When a Laplacian of Gaussian $\nabla^2 g(x,y)$ centered at $x_o = y_o = 25$ with a width $\sigma_b = 4$ is run through this recursive filter, the solution converges rapidly to the desired gradient of Gaussian solution shown in Figure 5-10.

Therefore it is possible to retain the basic 1D model design where the Laplacian of Gaussian retinal model of Marr and Hildreth is the sensor operator, and some local recursive filter such as the one described here reconstructs the gradient field representation at the processor stage.

5.2.5 Testing the 2D Noise Model

With the 1D noise model, the steps of developing the least squares estimator for I', the maximum likelihood Displacement estimation d_o and the 1D algorithm design, were followed by experiments to confirm the design and the equivalence of the ML d_o estimate to the best least squares solution x_o. It is desirable, wherever possible, to provide indirect support to an analysis of this kind.

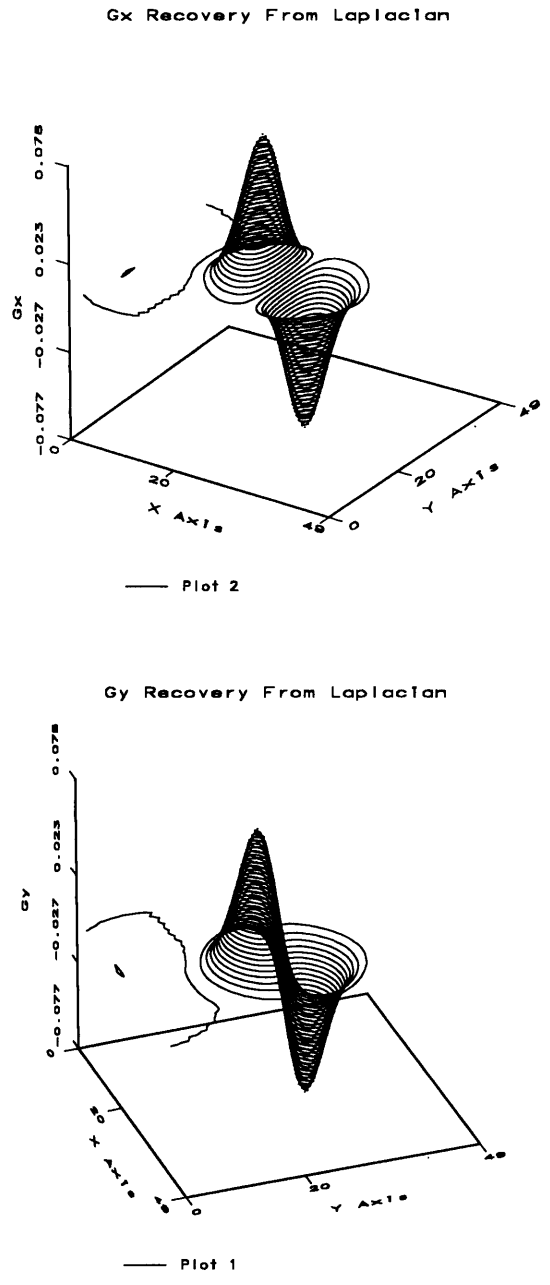

Figure 5-10: Gradient of Gaussian G_x and G_y Output

Implementations using this design on a digital computer to test the 2D noise model are, as yet, prohibitively slow to allow for the large ensembles needed for reliable results. With the 1D model of 200 pixels, the 1000 pass ensembles ran for 1-2 weeks for the range of tests required on various filter sizes, blurring widths, contrasts, sub-pixel offsets, and so on. Only a subjective analysis based on observations of the 2D model in operation so far provides experimental support that the 2D variance model is a simple extension of its 1D counterpart. As mentioned, the reduction of all aspects of the 2D model into its 1D predecessor in directions normal to the edge, strongly suggests that the 2D noise model should be, if not an exact solution, an excellent approximation.

The next step is to move on to general Disparity models, defined, as before, as linear transforms of Displacement representations. Along with a range of useful Disparity models, the noise model for each will be discussed.

5.3 2D Disparity Representations

Disparity representations are defined as linear transforms of Displacement representations. There are many such possible transforms. This section will briefly discuss just a few to give an overview of the possible scope of the model, as well as to touch on some of the new issues the 2D model raises. A number of simple transform approaches; subtraction, addition, temporal and spatial differentiation, and least squares analysis, will be discussed using practical specific vision problems as examples. First, however, an issue common to almost all Disparity models is discussed — the aperture problem.

5.3.1 Disparity Constraints and The Aperture Problem

One of the original motivations for the Displacement/Disparity model was the idea that since Displacements measure distances to features, differences between Displacement representations measure the distances between features in images. This is the

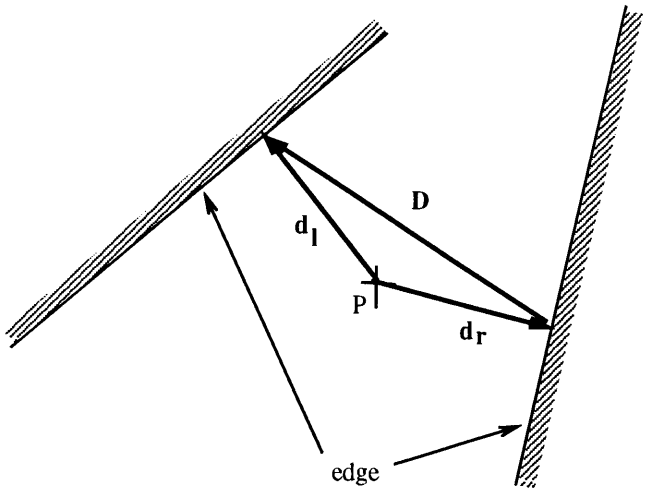

Figure 5-11: Basic 2D Disparity

principle behind the stereo Disparity, motion Disparity, and other models discussed later in this chapter. Other Disparity models such as the summation based Cyclopean Fused image model also rely on deriving Disparity representations based on pairs of Displacement representations.

All of these models are based on using "matched" features. This means that the Displacements used in the Disparity calculation at each point of the image either correspond to the same physical feature on the 3D object or are filtered out by some means. This is usually accomplished by using scale-space methods to insure that like features will be properly associated in the Disparity model — assuming such feature pairs exist. This was discussed at the end of the previous two chapters.

When features are correctly matched in Disparity calculations, however, the Displacement model accurately predicts only the normal distance from an image point to the edge and the edge orientation. With Disparities, this can provide information about the rotation and translation of the feature in the direction of its normal, but there is no information about translation in the direction tangential to the edge.

Regardless of the 3D imaging model, the 2D image features can only undergo rotation and translation between images taken at different times or locations. When

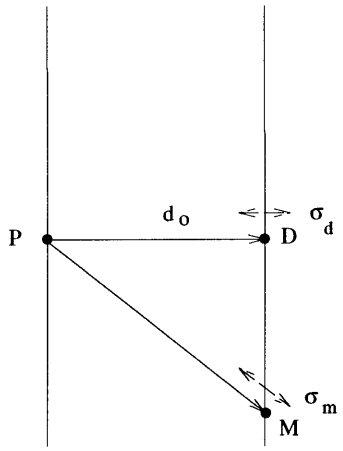

Figure 5-12: 2D Disparity Aperture Problem

the same feature is measured in two such successive images, taking the difference in the Displacement vectors will produce a Disparity vector that measures the distance the edge moved — the vector between the normal projection of P on each edge image as shown in Figure 5-11. Precisely which *point* in one Displacement feature corresponds to any *point* on the other is unclear.

This is important to keep in mind when Disparity measures are calculated. When features are translated relative to each other in an image, there is no information about the motion of any one feature in the direction tangential to the edge (normal to $\hat{\Theta}$). This is called "the aperture problem"[54, 61]. Figure 5-12 shows this. Suppose an edge is at point P at one time and subsequently moves, as shown, d_o units to the right in the image. Displacement measures would properly predict the motion. Suppose, however it moves in the direction and amount indicated by the vector M? In this case, precisely the same image would result, and consequently the same displacement function. Note that the gradient $\mathbf{I}'(x, y)$ is zero in the direction tangential to the edge, thus encoding the aperture problem in asserting there exists no knowledge of

tangential displacement given only the single edge of the $I_m(x, y)$ image.

For some Disparity models, such as stereo vision, this uncertainty is eliminated by the optics of the problem. In most other models, however, a single edge in such a subtractive Disparity model is under-constrained — information exists for rotation and the normal translation component, no information exists for the tangential translation component. The use of the gradient variance measures can help in designing algorithms to encode this important constraint. Locally, the solution may be ill-posed because of the problem is under-determined. Globally, however, combinations of features along with constraints on rigidity, 3D structure and motion, and so on can use the local feature constraints in an over-determined and well-posed heuristic.

One approach for solving the aperture problem is to use the probabilistic model developed in Equation 5.7. When two Displacement representations are, for example, subtracted ($\mathbf{M}_i = \mathbf{d}_i - \mathbf{d}_j$), at point $\mathbf{P}_i$, the probability distributions are convolved — assuming uncorrelated errors. The probability distribution of the true Disparity $\mathbf{M}$ — i.e. the actual pointwise correspondence vector between the two images — is

$$\begin{aligned} p(\mathbf{M}|\mathbf{M}_i) &= \frac{\|\mathbf{I'}_i\|\|\mathbf{I'}_j\|}{\sqrt{2\pi\sigma_b^2}} e^{-(\mathbf{M}-\mathbf{M}_i)^T \cdot \Psi_{Mi} (\mathbf{M}-\mathbf{M}_i)} \\ \text{where} \quad \Psi_{Mi} &= \Psi_i + \Psi_j \end{aligned} \tag{5.8}$$

The Disparity covariance matrix Ψ_{Mi} will now not be singular, like the Displacement matrices are, unless these matrices are linearly dependent, i.e. the two edges are parallel. In many cases the edges *are* parallel, or nearly so. In these cases, the Disparity probability function appropriately preserves the aperture problem ill-conditioning of the constituent Displacement distributions. Another observation is that when the edges are orthogonal, the covariance is unbounded in all directions, as it should be, since there is absolutely no positional information in the Disparity vector in such situations.

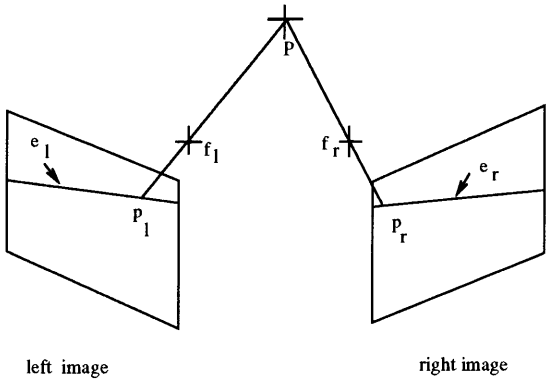

Figure 5-13: Epipolar Constraints in Stereo Vision

The next sections will touch briefly on many Disparity model concepts and their respective constraints. In some cases the models are developed in detail, such as with the 2D stereo algorithm discussed in the next chapter. In others the basic ideas are discussed in more general terms, such as with motion and matching. It is useful, however, to get a flavor of the breadth of possible approaches to Disparity models from the one simple 2D Displacement representation.

5.3.2 Spatial Subtraction — Stereo

The previous section discussed the dangers in simply subtracting Displacements to estimate pointwise feature Disparities. The optical reality of stereo imaging, however, imposes a constraint that makes the point correspondences unambiguous.

This special projective imaging model constraint is shown in Figure 5-13. A stereo image pair has the property that for any image point in one scene p_l, there exists a one dimensional slice on the other image e_r that must contain the corresponding point for the same scene feature, assuming it exists. This line is called an epipolar.

Conversely any point along the corresponding line in the second image p_r is constrained to match to a point along a line e_l in the first image through the point p_l. This is a constraint imposed by the geometry of the optical setup. These epipolar lines also lie on a common plane that passes through the imaged point as well as the

focal points of the cameras.

This constrains the search for correspondences between stereo image pairs to these 1D epipolars. A fixed camera configuration will have fixed epipolar arrangements, so a calibration step can be devised for any arbitrary sensor arrangement. Many stereo algorithms effectively restrict the sensor arrangement to, for instance, require horizontal epipolars.

Another issue concerns the alignment of edges to the epipolar lines. When a feature is normal to the epipolar, the variance of the Displacement measure along the epipolar is as stated before; inversely proportional to the gradient denominator $\|\mathbf{I}'(x,y)\|^2$. As features are rotated such that they are oblique to the epipolars, the variance on feature displacement estimation increases to the point where, when features are aligned with the epipolar, there is absolutely no way to measure the intersection point of the feature and the epipolar (See Figure 5-14).

The correct inverse variance weight w_e for an epipolar Displacement can be calculated from the Displacement weight, $\mathbf{I}'(x,y)$, and the epipolar direction vector at the image point, $\hat{\mathbf{e}}$ as was discussed earlier. From equation 5.6, the standard deviation of the displacement measure is inversely proportional to the dot product of the gradient and the epipolar direction:

$$\begin{aligned} I'_e(x,y) &= \mathbf{I}'(x,y) \cdot \hat{\mathbf{e}}(x,y) \\ &= \alpha g(d_o(x,y))(\hat{\mathbf{\Theta}} \cdot \hat{\mathbf{e}}(x,y)) \end{aligned} \tag{5.9}$$

To arrive at an inverse variance approximation, this is scaled by the convolving Gaussian

$$w_e(x,y) = I'_e(x,y) g_w(d_x(x,y))$$

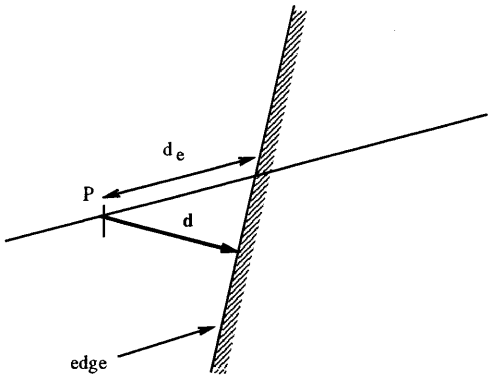

Figure 5-14: Epipolar Displacement d_e

The reason this is an approximation is that the width of the Gaussian should technically be scaled by the angle as well. The weight of obliquely aligned edges will be zero. The scaling by $g_w(d_o(x,y))$ is, in practice, implemented using convolution with the g_w Gaussian, as was discussed in Chapter 3.

A similar epipolar correction as was done in Equation 5.5 must be made to the 2D Displacement representation since a small normal displacement $\mathbf{d}(x,y)$ will result in large displacements along the epipolar when the feature is oblique to it.

$$\begin{aligned} d_e(x,y) &= -\sigma_b^2 \frac{I''(x,y)}{I'_e(x,y)} \\ &= \frac{d_o(x,y) g(d_o(x,y))}{I'_e(x,y)} \\ &= \frac{d_o(x,y)}{\hat{\mathbf{\Theta}}(x,y) \cdot \hat{\mathbf{e}}(x,y)}. \end{aligned} \tag{5.10}$$

The epipolar Displacement, therefore, is similar to the 1D Displacement in being a scalar function, but is based on the relationship between the 2D Displacement vector function $\mathbf{d}(x,y)$ and the epipolar field $\hat{\mathbf{e}}(x,y)$. It is, perhaps, useful to note that unlike many stereo algorithms, this approach imposes no restriction on the epipolar configuration or other imaging alignment assumptions.

To arrive at stereo Disparity, the difference between the left and right epipolar

Displacements is taken. The variance of the estimation is the sum of the two Displacement variance measures:

$$D_S(x,y) = \frac{(d_{er}(x,y) - d_{el}(x,y))}{2} \tag{5.11}$$

$$W_S(x,y) = \left(w_{er}^{-1}(x,y) + w_{el}^{-1}(x,y)\right)^{-1} \tag{5.12}$$

The reason for the scaling by one half is that the Disparity is defined as the distance from the left and right image features to the midpoint between them, or the Stereo Cyclopean edge that will be discussed in the next section. As discussed in Chapter 4, the features under the smoothing Gaussian in each image can be kept in correspondence by aligning the images locally based on the Disparity measured at the next larger scale $D^+(x,y)$. At the Largest scale, this prior Disparity alignment is zero, indicating that all the corresponding features in the image are assumed to be within some small multiple of its width, or approximately $\pm 2\sigma_b$, of each other. This sets the maximum Disparity range of the algorithm. In biological vision this is called Panum's fusional area [78].

The σ_b width of the Gaussian convolver can be reduced once the image features are locally adjusted to compensate for the local measure of Disparity of the previous filter:

$$D_S(x,y) = \frac{(d_{er}(x_r,y_r) - d_{el}(x_l,y_l))}{2} + D^+(x,y)$$

$$W_S(x,y) = \left(w_{er}^{-1}(x_r,y_r) + w_{el}^{-1}(x_l,y_l)\right)^{-1}$$

where (x_r,y_r) and (x_l,y_l) are positions along matched epipolars in the stereo image pair displaced by any prior Disparity estimate available — $D^+(x,y)$

$$(x_r,y_r) = D^+(x,y)\hat{\mathbf{e}}_r + (x,y)$$

$$(x_r,y_r) = -D^+(x,y)\hat{\mathbf{e}}_l + (x,y).$$

This forms the basis for the scale space stereo algorithm discussed in the next chapter. Similar approaches to aligning features across multiple scales are applicable to all the Disparity models discussed in this chapter.

5.3.3 Spatial Summation — Cyclopean Fused Images

Once stereo disparities are resolved from the image pairs, it is useful to combine the features of the two images into a single one for subsequent image processing. The previous chapter discussed two ways that a single image can be fashioned from a stereo pair. One would accept features from either either image by weighting the respective Displacement data by the measured inverse variance model. This has been referred to as Cyclopean Fused vision. The other representation would only preserve features that exist in both images, such as would be needed for the stereo model described above. These models have been discussed in the preceding two chapters in regards to the 1D model. This section will produce the 2D model.

Fused Cyclopean Displacements

When either image can contribute to the Cyclopean fused vision, the model of Chapter 4 can be updated to the 2D case:

$$\mathbf{d}_C(x,y) = \frac{w_l(x,y)\mathbf{d}_l(x,y) + w_r(x,y)\mathbf{d}_r(x,y)}{w_l(x,y) + w_r(x,y)}$$

where $w(x,y) = \alpha \mathbf{I}'(x,y) \cdot \mathbf{G}(x,y)$. Since this inverse variance based weighted average will produce a weight model where $\frac{1}{\sigma_C^2(x,y)} = \frac{1}{\sigma_l^2(x,y)} + \frac{1}{\sigma_r^2(x,y)}$ (see Equation 4.9), the following constraint must be true for the Cyclopean weights $\mathbf{I}'_C(x,y)$ and $\mathbf{G}_C(x,y)$:

$$\frac{1}{\sigma_C^2} = \alpha_C \mathbf{I}'_C(x,y) \cdot \mathbf{G}_C(x,y) = \alpha_l \mathbf{I}'_l(x,y) \cdot \mathbf{G}_l(x,y) + \alpha_r \mathbf{I}'_r(x,y) \cdot \mathbf{G}_r(x,y).$$

This leads to the conclusion that $\|\mathbf{I}'_C(x,y)\|^2 = \|\mathbf{I}'_r(x,y)\|^2 + \|\mathbf{I}'_l(x,y)\|^2$. Appropriate scaling of the left and right $\mathbf{I}'(x,y)$ inputs can produce a Cyclopean $\mathbf{I}'_c$

Displacement weight vector with the above scaling when features are aligned, zero amplitude when they aren't, and the direction of the average of the input Gradients.

Note that this variance model predicts that spatial discrimination tasks should improve by a factor of 1.4 when both left and right eyes are balanced (have equal acuity) and are open.

Stereo Cyclopean Displacements

The second Cyclopean representation is used for measuring the distance to *matched* features in the stereo pair of images. The above Fused Cyclopean representation accepts features from either sensory input, much the same way that one can still see with an eye closed. The Stereo Cyclopean representation accepts only features that the Stereo Disparity could match and assign a depth disparity measure to — namely features that exist in both images.

Since both features must be valid for such a measure and the Stereo Cyclopean x_c is defined simply as the midpoint of the left and right edge positions, there is no weighting involved in the computation.

$$C_S(x,y) = \frac{(d_{er}(x,y) + d_{el}(x,y))}{2}$$

The Stereo Cyclopean Displacement representation is simply the average of the epipolar Displacements. Compare this with the Stereo Disparity calculation of Equation 5.11. The variance model is identical with the stereo Disparity measure $W_S(x,y)$ (Equation 5.12).

This Cyclopean measure is primarily useful for labeling valid Stereo features. The Zero-crossings of the Stereo Cyclopean correspond to valid matched feature pairs in the Fused Cyclopean domain. Due to the different methods of computing these, the features will be shifted somewhat relative to each other. Sampling the Fused Displacement function at the zero-crossings of the Stereo Displacement will provide a measure of the relative shift. At these zero-crossings, the stereo Disparity and weight

measures can be sampled to extract the depth estimates for the image pair.

5.3.4 Spatial Derivatives — Focus

Image feature blurring was determined in the 1D model by taking the spatial derivative of the Displacement function. Sharp focus is indicated by a slope of 1.0. Any defocus caused by diffuse features or displacement from the focal distance is detected by a lower slope.

The same approach can be applied to the 2D model using the divergence of the 2D Displacement representation. If the feature is sharply focused:

$$\begin{aligned} \nabla \cdot \mathbf{d}(x,y) &= \nabla \cdot (d_o(x,y)\hat{\mathbf{\Theta}}) \\ &= \nabla d_o(x,y) \cdot \hat{\mathbf{\Theta}} \\ &= (\sin\theta\hat{\mathbf{i}} + \cos\theta\hat{\mathbf{j}}) \cdot \hat{\mathbf{\Theta}} \\ &= 1.0 \end{aligned}$$

The divergence measure will, like the 1D model, decrease when features are diffuse. This could be used, for instance, in checking candidate correspondences in stereo and matching algorithms.

5.3.5 Maximum Likelihood — Motion and Matching

Optical flow is defined as the apparent motion of image features based on a time series of images. Optical motion, in the 2D sense, usually refers to the *actual* motion of corresponding points in the 2D scenes. These are not the same, since the aperture problem discussed earlier guarantees that without some further constraints, an infinite number of 2D motions can be associated with any optical flow field. Temporal Disparities measure optical flow. Actual motion requires some additional constraints. Examples of constraints that are often used are rigidity of objects, smoothness of flow,

ego-motion constraints, and so on. This section will show how the probabilistic model of in Equation 5.8 can be applied to a very simple motion problem — extracting the net translation motion field.

Translation

The net translation vector $\mathbf{M}$ is the average feature motion vector that is the most consistent with the optical flow data — local Disparity measures — when the aperture problem is taken into account. Other similar problems that could be easily posed are measures of net rotation or scale. The main purpose of this section, however, is to illustrate the use of maximum likelihood analysis techniques for Disparity estimations.

As before, the local optical flow Disparity estimates are based on image sequences ($\mathbf{M}_i = \mathbf{d}_i - \mathbf{d}_{i'}$), where the i' notation indicates the same spatial sample taken a small time interval earlier. $\mathbf{M}_i$ will only measure the normal motion of any local feature. The probability density of the actual 2D translation motion $\mathbf{M}$ based on the measurement $\mathbf{M}_i$ is, assuming uncorrelated errors, the convolution of the two component distributions:

$$\begin{aligned} p(\mathbf{M}|\mathbf{M}_i) &= \frac{\|\mathbf{I}'_i\|\|\mathbf{I}'_i\|}{\sqrt{2\pi\sigma_b^2}} e^{-(\mathbf{M}-\mathbf{M}_i)^T \Psi_{Mi} (\mathbf{M}-\mathbf{M}_i)} \quad \text{where} \\ \Psi_{Mi} &= \Psi_i + \Psi_{i'} \end{aligned}$$

When all the scene features are treated as having uncorrelated errors — a common, but possibly bold assumption in many such vision models — then the net probability that the motion measure $\mathbf{M}$ is consistent with the Disparity data is

$$p(\mathbf{M}|\text{Image}) = \prod_i \left(\frac{\|\mathbf{I}'_i\|\|\mathbf{I}'_i\|}{\sqrt{2\pi\sigma_b^2}} \right) e^{-\sum_i (\mathbf{M}-\mathbf{M}_i)^T \Psi_{Mi} (\mathbf{M}-\mathbf{M}_i)}$$

The goal is to determine what $\mathbf{M}$ maximizes the probability function. Since the log

of the positive distribution will also have the same maxima the log can be maximized.

$$\ln p(\mathbf{M}|\text{Image}) = \sum_i \ln\left(\frac{\|\mathbf{I}'_i\|\|\mathbf{I}'_i\|}{\sqrt{2\pi\sigma_b^2}}\right) - \sum_i (\mathbf{M}-\mathbf{M}_i)^T \Psi_{Mi} (\mathbf{M}-\mathbf{M}_i)$$

Also, since the left summation is constant for all $\mathbf{M}$, the function to be minimized (since the sign is switched) is

$$O(\mathbf{M}) = \sum_i (\mathbf{M}-\mathbf{M}_i)^T \Psi_{Mi} (\mathbf{M}-\mathbf{M}_i)$$

The above expression can be solved by taking partials with respect to $\mathbf{M}_x$ and $\mathbf{M}_y$ and setting these to zero. A closed form solution can be derived that solves for $\mathbf{M}$ in terms of the gradient based covariance matrix Ψ components and the Disparity measures $\mathbf{M}_i$.

This equation is called the Disparity Constraint Equation. Note that the above relation is valid for *any* Disparity model such as motion, matching, or stereo. It basically states that the true match must be consistent with both the aperture problem and the Disparity measures. The 2D translation solution derived from treating $\mathbf{M}$ as a constant and taking partial derivatives is the first step in the derivation that is specific to the translation motion Disparity problem. If, instead of using temporal image sequences, template image models are compared to real scenes, the above solution could be used to align the image to the template by estimating the alignment saccade $\mathbf{M}$.

Numerous other Disparity models, including scale-space models much like the stereo algorithm uses, can be devised to solve for arbitrary affine transform mappings. One model would use the above approach to extract scale, rotation, and translation components from the scene. Once this is done, the residual Disparities $\mathbf{M}_i - \mathbf{M}(x,y)$ can be used for feature-based tasks such as face recognition.

These higher-level algorithms are beyond the scope of this thesis, however. The following chapter will concentrate on the stereo Disparity model $D_S(x, y)$ and examine it's performance on real imagery.

5.4 Summary

This chapter completed the last step in developing the full 2D Displacement model by adding the orientation measure in the 2D domain. Combined with the distance measure of the 1D model and the variance model of the last chapter, this 2D analysis completes the Displacement/Disparity theoretical model.

The model is surprisingly simple and retains the same form as its 1D predecessor. The variance model is also consistent with the results of the previous chapter, except that it is also possible to encode the well-known "aperture problem". As with the past chapter, precise measures of Displacement and Disparity variances are readily calculated from the image Gradient signal.

Stereo matching is shown to have no such matching ambiguity, and an accurate epipolar model is developed. For other Disparity algorithms, a constraint equation has been developed that can allow the Disparity model to be used in a number of useful higher level vision problem domains such as motion and matching.

The next chapter puts the Stereo Disparity model to the test by having real images run through the algorithm.

Chapter 6

The Stereo Algorithm with Real Images

The previous chapter developed an epipolar model of stereo matching of features. This chapter takes those concepts and develops a working algorithm for extracting stereo Disparity from real scene pairs. It has been run on a variety of real stereo image pairs. Some details of the algorithm and the results of the tests are presented here that demonstrate the range of the model's capability.

Although most, if not all, of the essential components of the stereo algorithm are contained in the previous chapters — especially in section 5.3.2 — many of the important features are partially obscured by the need to develop the general model in those chapters. This chapter recapitulates those steps and how the algorithm incorporates them.

The overall structure of the stereo algorithm will be discussed in the following section. In the subsequent sections additional detail about the Displacement and Disparity computational steps will be provided.

Constraints must be imposed on the stereo matching. For instance, opposite contrast sign edges probably should not be matched. When illusory edges are found in images, these should be filtered out as well. The Displacement/Disparity model is as

easily modified as symbolic designs for such constraints. Given the availability of good feature measures such as focus, orientation and contrast in this model, constraints are often easier to implement than in many symbolic designs. On the other hand, how many constraints are necessary to get a good stereo image? It would seem that the best model is one in which relevant constraints are easily incorporated while requiring as few constraints as possible to operate well. The algorithm discussed here is designed with this trade-off in mind. The constraints used in this and other stereo algorithms will be discussed in some detail in this chapter.

Finally, of course, there will be some examples of the stereo algorithm applied to actual stereo image pairs. Although there exist, sadly, few accepted standards for evaluating algorithm designs or comparing performance between competing methods, the central goal is to show that the Displacement/Disparity model is at least as good as the best algorithms, symbolic or continuous, presently in use. The experiments are designed to provide support to this claim.

6.1 Scale-Space Structure

Scale space schemes are usually predicated on the psychophysical work of Wilson that detected multiple (perhaps six) spatial frequency channels in human subjects [93]. One of the first researchers to propose multiple channels in a stereo algorithm were Marr and Poggio [53] and these and other ideas were implemented by Grimson in what is popularly known as the Marr-Poggio-Grimson (MPG) algorithm [28, 29].

The MPG algorithm searches at a given scale for features — in this case, zero-crossings of the Laplacian of Gaussian convolved image — in each image and uses the measured disparities on matched edges to locally shift the search zone of smaller filter sizes.

The basic concept is that at large σ_b filter sizes, only large features — those with low spatial frequency — will dominate the algorithm. This will allow a coarse

disparity measure to be made and locally allow for a coarse prior — $D^+(x,y)$ in the present model — for the next smaller scale.

In practical implementations of the present model, the Displacement function is valid over a ± 2 to $\pm 3\sigma_b$ range. Grimson used a $2\sigma_b$ search range at each scale step. This is generally consistent with the observed Panum's fusional area in humans [78]. Grimson used four scale steps with each step based on some early results of Wilson's experiments using gratings [91]. Later experiments by the same researchers revised the model to six filter sizes [92]. The peak frequency response is fairly uniformly spaced at octave intervals, starting with a excitatory center diameter consistent with a single cell center model posited by Marr et al. [51].

The algorithm described here is very similar. Figure 6-1 shows the overall structure of the algorithm. Five filters sizes are typically used (σ_b = 32, 16, 8, 4, and 2 pixels). Both left and right images are convolved with the filters producing the $I_l(x,y)$ and $I_r(x,y)$ smoothed representations of the image for all scale sizes.

The epipolar Displacements $d_e(x,y)$ and weights $I'_e(x,y)$ are calculated for both left and right smoothed images at the largest scale and passed to the Disparity calculation step. The Disparity computations include stereo Disparity ($D_S(x,y)$) as well as a number of other Disparity representations discussed earlier. The stereo cyclopean Displacement ($d_C(x,y)$) is calculated, as is focus ($\dot{d}(x,y)$). The inverse variance weight $w_D(x,y)$ is derived from the left and right weights. The details of these computations will be discussed in the following sections.

Some constraints are imposed on the algorithm, such as Displacement gradient, contrast sign mismatch and maximum Disparity, although these are intentionally kept as few as possible. Where the constraints are violated, the weight (or inverse variance) measure of the offending interval is set to zero.

Convolution is used to provide a weighted average of the local Disparity estimates as well as a locally optimal measure of Disparity when features are approximately aligned by the scale space approach. This also allows regions with little or no measure

locally to use neighboring valid measures for Disparity estimation. This is needed especially in those regions that fail the constraint tests.

The resulting Disparity measure is used to locally align the Disparity calculation of smaller scales. Each scale Disparity step thus provides a (usually small) correction to the larger step's estimate. The total Disparity is the sum of all the scale Disparity measures. A running sum of these measures is calculated as the output of each stage and used to shift the sampling of the subsequent stage.

This process repeats itself recursively starting with the large filters and ending with the smallest. At the smallest scale, the zero crossings of the stereo Cyclopean Displacement are sampled and, where the stereo Disparity weight is significant, the Disparity measures are taken. These samples of the Disparity at marked edges comprise the output of the Disparity calculation.

The details of the Displacement and Disparity calculation steps of Figure 6-1 are discussed in the next two sections.

6.2 Epipolar Displacement Calculation

The first step of the Displacement calculation is to take the smoothed image $I(x, y)$ and calculate the Laplacian $I''(x, y)$ and Gradient $\mathbf{I}'(x, y)$ representations from it. Simple approximations to the Laplacian and the Sobel operator with small (3x3) support are used for this purpose in this algorithm [36].

Each "eye" image is assumed to be the product of a projective optical model onto a focal plane. This is a very general model and is certainly adequate for all camera imaging systems. This imaging model reduces the stereo problem to an epipolar matching problem. Chapter 5 discussed how these lines are formed in the image planes as the result of the stereo imaging setup. Since the imaging optics restrict scene points to share epipolar lines in the two images, any horizontal line in the Cyclopean scene corresponds to an epipolar pair of the left/right. The average feature

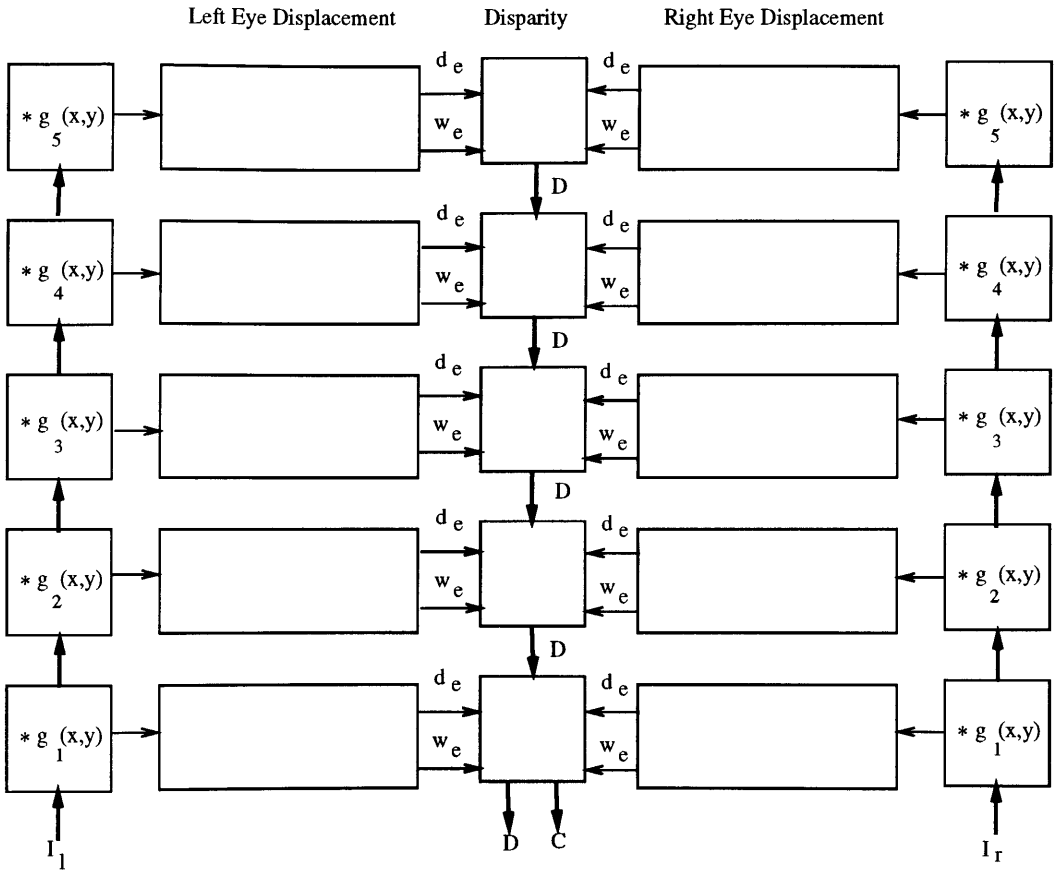

Figure 6-1: Stereo Scale Space Algorithm

position along those lines in the two scenes determines the horizontal position on the Cyclopean line. Depth Disparity is derived from their positional differences.

The Displacement measurement requires knowledge only of the relative orientation of the epipolar lines in the image plane. The Displacement measures feature distance normal to the feature edge. What is needed is the distance to each feature along the local epipolar direction. This was shown in the last chapter (Equation 5.10) to be

$$d_e(x,y) = -\sigma_b^2 \frac{I''(x,y)}{I'_e(x,y)} \tag{6.1}$$

where the inverse variance weight function is (Equation 5.9)

$$I'_e(x,y) \;=\; \mathbf{I}'(x,y) \cdot \hat{\mathbf{e}}(x,y) \tag{6.2}$$

These two functions are calculated for all image points in both left and right scenes.

6.3 Stereo Disparity Calculation

The algorithm must now match features on corresponding 1D epipolars and calculate their relative Disparities. As mentioned in the previous section, the Disparity calculation is based on a Cyclopean domain that itself maps to epipolars in the respective eye images. Such a mapping $\mathbf{M}$ is assumed to exist for each eye based on a calibration step, that provides both the scene sample points as well as the epipolar directions $\hat{\mathbf{e}}(x,y)$. All discussion of samples in the Cyclopean image domain refer indirectly to these mappings, i.e. any point in the Cyclopean domain corresponds to a point (and an epipolar) in each image. At each sample point in the Cyclopean domain, the left and right image sample points are based on both this mapping, as well as any prior on Stereo Disparity at each Cyclopean point.

This previous scale Disparity $D^+(x,y)$ is used to shift the sample in the image domains by moving sample points horizontally along the epipolars. When the Dis-

parity is positive, the left sample is shifted to the right and the sample point in the right image is shifted to the left. This results in an alignment of the image features along the epipolars in proportion to the measured Disparity in each eye.

$$(x_l, y_l) = D^+(x,y)\hat{\mathbf{e}}_l + \mathbf{M}_l(x,y) \quad (6.3)$$

$$(x_r, y_r) = -D^+(x,y)\hat{\mathbf{e}}_r + \mathbf{M}_r(x,y). \quad (6.4)$$

Once the appropriate epipolar Displacements ($d_r(x_r, y_r)$ and $d_l(x_l, y_l)$) and their weights ($I'_{el}(x_l, y_l)$ and $I'_{el}(x_r, y_r)$) are sampled, a number Disparity measures are calculated:

- Focus — An estimate of the slope of the Displacement is needed for both focus and Chevruel illusion filtering. $\dot{d}_l$ and $\dot{d}_r$ are based on simple first difference slope calculation on the epipolar Displacement inputs.

- Focus Match — The difference between the above Displacement slopes is calculated as a measure of focus match: $\dot{D} = \dot{d}_l - \dot{d}_r$

- Stereo Disparity — The difference between the left and right Displacements is scaled to provide a measure

$$D_s = (d_r - d_l)/2 + D^+. \quad (6.5)$$

  The reason for the scaling by 1/2 is that Disparity is defined as the difference between each eye feature position and the cyclopean (average) feature position. Note that the previous scale Disparity D^+ is added, thus providing a total Disparity measure.

- Cyclopean Stereo — As discussed in the last chapter, the Stereo Cyclopean Displacement provides a measure of the location, in the Cyclopean Domain, of matched features. It is needed at the smallest scale to mark valid features in

the image:

$$C_s = (d_r + d_l)/2. \tag{6.6}$$

6.3.1 Maximum Likelihood Disparity Estimation

In chapter 4 a maximum likelihood estimation for the 1D Displacement function was found to be (Equation 4.7)

$$d_o = \frac{\sum_{x=0}^{N} g_w(x) I'(x) x_d(x)}{\sum_{x=0}^{N} g_w(x) I'(x)}.$$

Another approach shown was to use convolution (Equation 4.8)

$$d_{co}(x) = \frac{g_w * [I'(x) x_d(x)]}{g_w * I'(x)}.$$

These are equivalent when $d_{co}(x)$ is sampled at (or near to) x_o. It was also observed, however, that this $d_{co}(x)$ function is pathological since it requires as big a dynamic range as the size of the domain N. It was suggested that this maximum likelihood estimation step could possibly be deferred to the Disparity calculation.

The maximum likelihood estimation of stereo Disparity is:

$$D_o = \frac{\sum_e w_D(x,y) D_s(x,y)}{\sum_e w_D(x,y)}$$

The sum is along the 1D epipolar. The weight function is, again, the inverse variance measure:

$$w_D(x,y) = \frac{\alpha_l g_l(x_l,y_l) I'_l(x_l,y_l) \alpha_r g_r(x_r,y_r) I'_r(x_r,y_r)}{\alpha_l g_l(x_l,y_l) I'_l(x_l,y_l) + \alpha_r g_r(x_r,y_r) I'_r(x_r,y_r)}.$$

It would be nice if convolution could be used as a substitute for g_l and g_r, but unless $\alpha_l g_l(x_l, y_l) = \alpha_r g_r(x_r, y_r)$, this is not possible. Since the contrasts α_l and α_r are not explicitly solved for in this model, the convolution model is not used as a substitute for the Gaussian terms in the variance functions. Instead, $\|\mathbf{I}'(x,y)\|^2$ is used as the variance measure. The maximum likelihood estimation requires the variance weighted estimates to be summed over the region where the operator is valid.

$$W(x,y) = \frac{I_l'^2(x_l, y_l) I_r'^2(x_r, y_r)}{I_l'^2(x_l, y_l) + I_r'^2(x_r, y_r)} \tag{6.7}$$

$$\tag{6.8}$$

The weight function W is calculated from the right and left epipolar gradients $I_e'(x,y)$ discussed earlier. Prior to the convolution, however, some left and right image properties are tested to check for consistency.

6.3.2 Constraints

As mentioned earlier, there is a need to avoid false correspondences in any stereo algorithm. With this representation, however, substantial information is on hand to help avoid such mistakes. The following criteria are be used to detect mismatches and, if so, zero the weight function $W(x,y)$ over the offending intervals:

- Where the contrast signs are mismatched (the signs of the I_e' functions do not match)
- Where a real edge is being matched to an illusory edge. A negative slope $\dot{d}$ indicates an illusory edge.
- Where the blurs are severely mismatched (resulting in a sloped D function near the feature)
- Where the absolute Disparity is too large.

Note that Disparity slope and many other constraints are *not* included here. The intent of the design is to impose as few as possible constraints as needed to the algorithm. Some others *could* have been added as other researchers have in symbolic model stereo methods. Some of these will be discussed in the following section. It is useful to note that, like the symbolic designs, constraints *can* be incorporated into the Disparity model. In many continuous domain models it is less clear how one could do so.

Once this constraint filtering takes place, the functions W and D_S are both convolved with a Gaussian of width σ_w (see Chapter 4). This operation effectively interpolates between features and "fills in" Disparity estimates where mismatches were detected and their erroneous disparities rejected. Where features exist, the convolution serves to provide the best estimate of Disparity by using all the local Displacement information.

$$D(x,y) = \frac{g_w * [D_S(x,y)W(x,y)]}{g_w * W(x,y)} \tag{6.9}$$

The convolved Disparity, then, is fed to the next smaller scale calculation. The convolved weight function — a contrast measure — is also passed down such that at any scale, should there be absolutely no feature information (or mismatched feature information only) the larger scale information can be preserved.

At the smallest scale the weight function, $W(x,y)$, is sampled where the "cyclopean" Displacement function has zero crossings. This function:

$$C(x,y) = (d_r(x_r,y_r) + d_l(x_l,y_l))/2$$

is the stereo Cyclopean Displacement function discussed earlier. The Disparities $D(x,y)$ associated with significant weight values $W(x,y)$ at zero crossings in the cyclopean Displacement $C(x,y)$ are the sparse stereo Disparity representation desired.

Thus, to summarize, the basic stereo Disparity algorithm at scale n is:

1. Given the prior disparity $D^+(x,y)$ from scale $n-1$, and a chosen epipolar pair and sample position, offset the left and right epipolar sample positions by $D^+(x,y)$. This is the scale space alignment step.

2. Get the epipolar Displacements, d_e, and weights, w_e. Calculate the stereo Disparity $D_S(x,y)$, and Disparity weights, $W(x,y)$, using the above relations.

3. Check the above constraints for invalid matches. If found, zero the weight W at those points.

4. Use the convolution estimation model to derive a dense Disparity map D for the next smaller scale.

6.3.3 Constraints in Stereo Algorithms

There have been many stereo algorithms using symbolic feature sets that have imposed many constraints on the solution to insure that false correspondences do not take place. It is often difficult to assign unique correspondences, even for human viewers, and when features are identical, or missing from one view, any algorithm will be error prone. A computational model can be judged by both how many constraints *need* to be added to make it work on as well as how many *can* be added on. Continuous domain models invariably have no added machinery to filter false correspondences. One reason for this is it would be very difficult to figure out how such constraints could be imposed on these very black-box approaches.

With this model, given its genesis as a continuous domain symbolic feature approach, many constraints can be incorporated into algorithms. It is useful to look at how this compares with some previous symbolic approaches.

Uniqueness — One of the first constraints proposed by Marr and Poggio was "Uniqueness"[52]. Uniqueness requires a one-to-one correspondence between left and

right image features. A distinction is sometimes made between allowing multiple matches and unmatched features but this is an almost ubiquitous restriction in symbolic algorithms (see [1, 29, 59, 64]). The present algorithm imposes no such constraint. Multiple features could match a single feature, since this is a legitimate, although rare, solution in the real world. The main reason it is not done in this model is because it would be difficult to implement and is not obviously needed.

Continuity — Along with uniqueness, Marr and Poggio proposed another constraint — "continuity"[52]. The argument here is that neighboring features should exhibit relatively minor variations in disparity. This is similar to assumptions of smoothness in Regularization theory. Many algorithms use this constraint [1, 2, 62]. Again, this algorithm imposes no such constraint. It might appear that the convolution of $D_S(x,y)$ with g_w would cause some interpolation but the scale space structure insures that arbitrarily steep Disparity discontinuities will be allowed without smoothing. Indeed, they occur frequently.

Disparity Gradient — Similar to continuity is the restriction of the disparity gradient, either based on physical principles, where the maximum gradient for non transparent scenes is $|dD(x)/dx| = 1$, or based on psychophysical arguments, where only smaller gradients are found[62, 64]. This algorithm has no such gradient restriction explicitly. On the other hand, the minimum slope for Displacements is 0, and, at small scales where features are isolated, the maximum is one. Therefore disparity gradient limits are intrinsic to the Displacement/Disparity model in this sense.

Orientation Match — Feature orientation match is a very common constraint. Only features that have approximately the same slope can be considered correspondences. Some matching algorithms have successfully used *only* orientation cues for matching [7]. Others prune mis-oriented matches [2, 29, 59, 62]. This model incorporates this constraint in an indirect ways. The weight w_e suppresses all edges misaligned with the epipolars. This tends to favor vertical edges, It would be easy to have the stereo displacement weight scaled by the dot product of the left and right

$\hat{\Theta}$ vectors as well to impose this constraint, but this was not deemed necessary.

Complex features — A stronger restriction to the above constraint is to test endpoints, lengths, or other spatial organization of features [1, 2, 59, 62]. No such machinery is used in this model.

Contrast — Contrast is an example of an optically based constraint [2, 59]. Other than weights based on combined feature contrast, and allowing only like-sign contrast edge matching no other contrast matching is made in the Displacement/Disparity model.

Disparity Limit — An absolute maximum disparity range allowed for correspondence is often imposed at some level in an algorithm [1, 29]. In this model, Disparity estimates are limited to $\pm 3\sigma_b$.

Epipolar Restrictions — One of the most common, if not essential restrictions is limiting the search space to epipolars [1] or, even more restrictive, horizontal epipolars [2, 29, 62, 64]. In this model, any epipolar arrangement is acceptable. There is no performance penalty associated with the projective sensor model or arbitrary sensor configurations.

Vertical Features — It is well known that features aligned nearly parallel to the epipolars are quite sensitive to calibration errors [1, 29]. This is built into the variance model and the epipolar weight reflects this variance. Thus horizontal features do not get matched.

Focus Match — As mentioned earlier, focus is a Disparity measure that is easy to calculate and is useful for matching. It also is a measure of illusion edges ("negative" focus slope) that must be pruned from the match.

Some of the constraints that were used in this algorithm were tested to see how much they affected the matching error rates. These will be discussed at the end of the chapter. The next section discusses the performance of the algorithm on a series of image pairs.

6.4 Algorithm Summary

The stereo algorithm described in the preceding sections of this chapter is extremely simple to implement. The basic steps involved at each σ_b scale size are:

1. Convolve the left and right images with the 2D Gaussian of width σ_b.

2. Calculate the gradient and Laplacian image representations from the Gaussian smoothed images. These can be arrived at using extremely small support operators (see [36, 4]).

3. Calculate the epipolar Displacement and Displacement weight representations for the individual images (Equations 6.1 and 6.2).

4. For each pixel in the cyclopean image plane, calculate the corresponding sample points in the left and right image planes based on the epipolar mappings and (any) previous Disparity estimate available from a large scale (Equations 6.3 and 6.4). Get the Displacement measures at these points.

5. Calculate the Cyclopean Displacement (Equation 6.6), stereo Disparity (Equation 6.5) and the weights (Equation 6.7) using the above Displacement information.

6. Constrain the stereo matching by rejecting any points where the Focus Disparity is not close to 1.0, doesn't match well, or contrasts mismatch. This is accomplished by zeroing the local stereo weight value at bad match points.

7. Convolve the weighted Disparity to arrive at a best estimate of Disparity using local Disparity estimates and their variances (Equation 6.9).

At each scale step, the Disparities from the previous (larger) scale are used for each Cyclopean location in the mapping step (4). At the largest scale, this prior

Disparity is usually assumed to be zero. The algorithm can be iterated at any scale to allow the Disparity estimates to converge, but this is usually not needed.

At the smallest scale, the Disparity is sampled at the zero-crossings of the Cyclopean stereo Displacement image where the stereo weights exceed some preset threshold. This can be based on the signal to noise formulation derived in Equation 4.6.

The calculations involved in all of the above steps are simple and involve local support (at most 3 by 3). The most computationally intensive steps are the convolutions with Gaussians in the first and last steps. These can be decomposed or performed using FFT techniques, as well as by other very efficient methods.

6.5 Experiments

The stereo algorithm described in the preceding section was tested using a number of synthetic and real images. The synthetic imagery as well as some monocular images that were manipulated to create artificial stereo disparity maps were useful to allow testing of many of the issues of a nonlinear model such as this.

For instance, a large variety of random edge profiles shifted relative to each other needed to be run to determine if the scale-space concept would indeed disambiguate the correspondences. It was found that only in the instance of Disparity gradients approaching one (i.e. where two features in one image merge while remaining separate in the other) would the algorithm drop the correspondence — it would simply mark the feature unmatched.

This is a very reasonable result, since the alternative would be to have the algorithm mark any neighboring like-signed feature in one eye to an unmatched feature in the other as the match. Every occluded feature would then be assigned to the occluding edge, thereby creating a false set disparity assignments.

Although this is not a totally unreasonable inference (human subjects appear to do this to a limited degree with occluded views), it is at least as good a solution to

leave such ambiguously matched features left unmatched.

The most interesting tests, however, were of stereo image pairs. Like all stereo algorithms, this algorithm requires some information on the epipolar alignments. When this information is not available, then the usual assumption is that the epipolars are horizontal and that the pixels correspond in the fixation plane, i.e. the epipolar mappings $\mathbf{M}_l$ and $\mathbf{M}_r$ are simple identities. This is only true of carefully aligned camera set-ups. In fact, no image pair was found in the relatively large image store at the A.I. Laboratory where there were not substantial errors in the image registrations and epipolar field assumptions.

This being the case, the algorithms were tested using rough estimation of alignments. Horizontal epipolars were still assumed, which was sometimes a bold assumption, but in the absence of a calibration method, a necessary one. The stereo algorithm was run on the image pairs and depth cues were used to adjust the vertical offset of the images. When a known flat surface, for instance, demonstrated disparity variations based entirely on edge orientation, the problem usually rests with vertical registration errors.

Three major classes of errors are found in stereo algorithms; calibration errors, correspondence errors and Disparity errors. Calibration errors include both false correspondences due to faulty epipolar field assumptions as well as errors in Disparity measurements on correctly matched features due to faulty epipolar mappings. Correspondence errors are features that are mismatched and assigned Disparities instead of correctly matching or leaving a feature unmatched. Disparity errors are simply the errors in measuring Disparities in correctly matched features.

Correspondence errors are, in general, easy to detect. A false correspondence usually results in a large measurement error. The generally low overall error rates of the algorithm make it possible to count the total false correspondences even in very dense Disparity estimations. Since the calibration used in the experiments was very coarse, there were often quite a few mismatches due to calibration errors. The

balance of the mismatch count is considered to be due to the algorithm.

Disparity errors of correctly matched features are somewhat difficult to measure without some estimate of "ground truth" and a model of imager noise. This section focuses mostly on the issue of non-calibration matching error rates in the stereo algorithm.

Three image pairs are presented. The first, the Jet and Decoy image is a difficult test because it tests the overall sensitivity of the algorithm to small Disparity measures. The second, the Hallway, has the opposite character — very large Disparity ranges and large Disparity gradients. The last, the UBC campus, has a wide variety of image feature detail and character and is used to demonstrate a relatively dense scene interpretation.

In all three examples, only the weight threshold for sampling Disparities and the vertical image registration is altered between tests. No fine tuning of the algorithm was attempted.

Images are run through the stereo algorithm, and samples taken at the zero crossings of the stereo Cyclopean image where the W weights exceed some threshold. This is proportional to the inverse variance of the feature Disparity measure. Based on the noise model of Chapter 4 Equation 4.5, the scaled Disparity variance from uncorrelated Displacement measures (where the image noise levels are comparable) is

$$\sigma_D(x) = \frac{\sigma_n \sigma_b^2}{4\sqrt{W(x)}}.$$

To know the actual pixel scaled variance from this figure, some estimate of image noise σ_n is required, which is not readily available. In the experiments, the threshold weight value for W is chosen based on observed error rates.

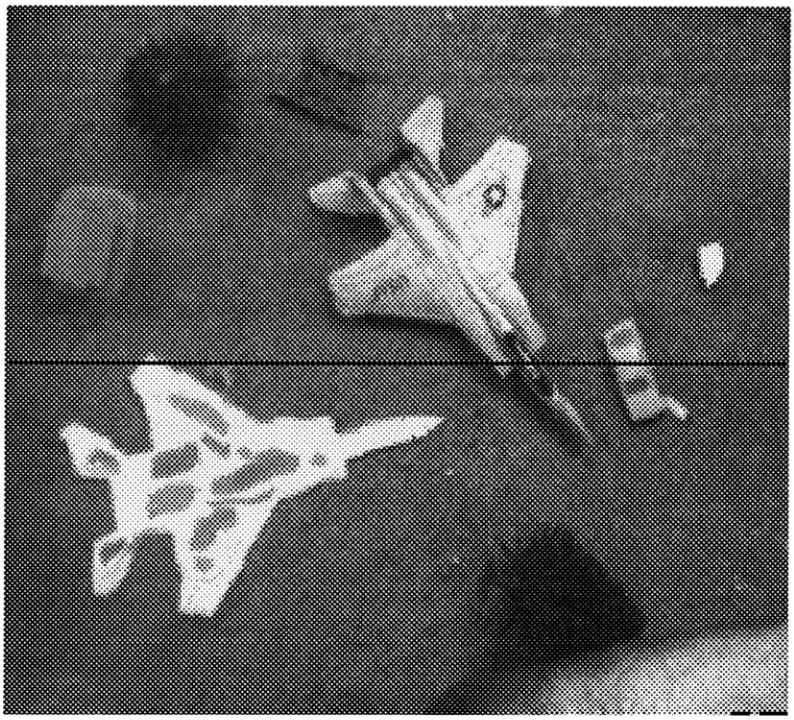

Figure 6-2: jet grey

6.5.1 Jet and Decoy Images

The jet and Decoy image pair scene consists of a "real" jet (actually a toy model) and a "decoy jet" cut out of paper. The concept behind such an image problem is to see whether 3D stereo cues can be used to discriminate between such seemingly similar 2D images.

The images were taken with a "head-eye" system that provides stereo images with adjustable vergence, pan and tilt. This also results in images that have very non-parallel epipolars. Calibration software was used to "warp" the scenes such that the epipolars are approximately parallel. To do this, a large amount of interpolation was used to generate the pixel intensity values. There is very little likelihood that such a distortion will result in good spatial accuracy for the Displacement edges at the sub-pixel level, since such interpolation methods will not preserve edges well.

On the other hand, the differences in disparity between the jet and the decoy is, on average, only about a pixel. This makes this image a very difficult test for any

algorithm. Figure 6-2 shows one of the two stereo images. Note that besides the jet and the decoy, there are some other features in the scene. The car to the right is about the same height as the jet. There is a large rock in the lower right corner, a house in the upper left and two tree-like objects at the top and bottom of the scene. All of these objects cast shadows.

The results are plotted in Figure 6-3. The grey scale image data is greatly compressed while the Disparity data is superimposed on the image in grey-scale coding. White indicates high Disparity and the darker pixels indicate lower values.

Note that all the Disparity measures occur at intensity edges, as expected. There are 1060 depth estimates in this image. There were no correspondence errors found (including any caused by calibration errors). The estimations are dense around the edges of the jet and decoy outlines. It is easy to distinguish the jet and the decoy based on these Disparity measures despite the calibration errors. Note also that the jet disparity estimates are notably higher (lighter) than the decoy. This is especially noteworthy given that the mean disparity difference between the two objects is on the order of 1-2 pixels and that the interpolation calibration used would be expected to introduce noise into the measures of the order of 1 pixel. Note the e jet shading and its shadow. There are clearly some small fluctuations in depth on the jet wings (especially on the right side of the figure). These are due to the calibration interpolation method used to generate the images.

The dark (low) estimations found on the tail of the real jet are specular reflections of the vertical stabilizer — a correct Disparity interpretation of the images. Although it is difficult to see in the Figure, numerous estimates were also found on the vertical stabilizer (on the right side) in spite of the fact that it is almost impossible to detect the stabilizer by simple inspection of the 2D images.

It is also informative to examine some of the signal representations of the continuous domain model. The black line on Figure 6-2 is a sample epipolar on which all of the important intermediate representations are shown in Figure 6-4.

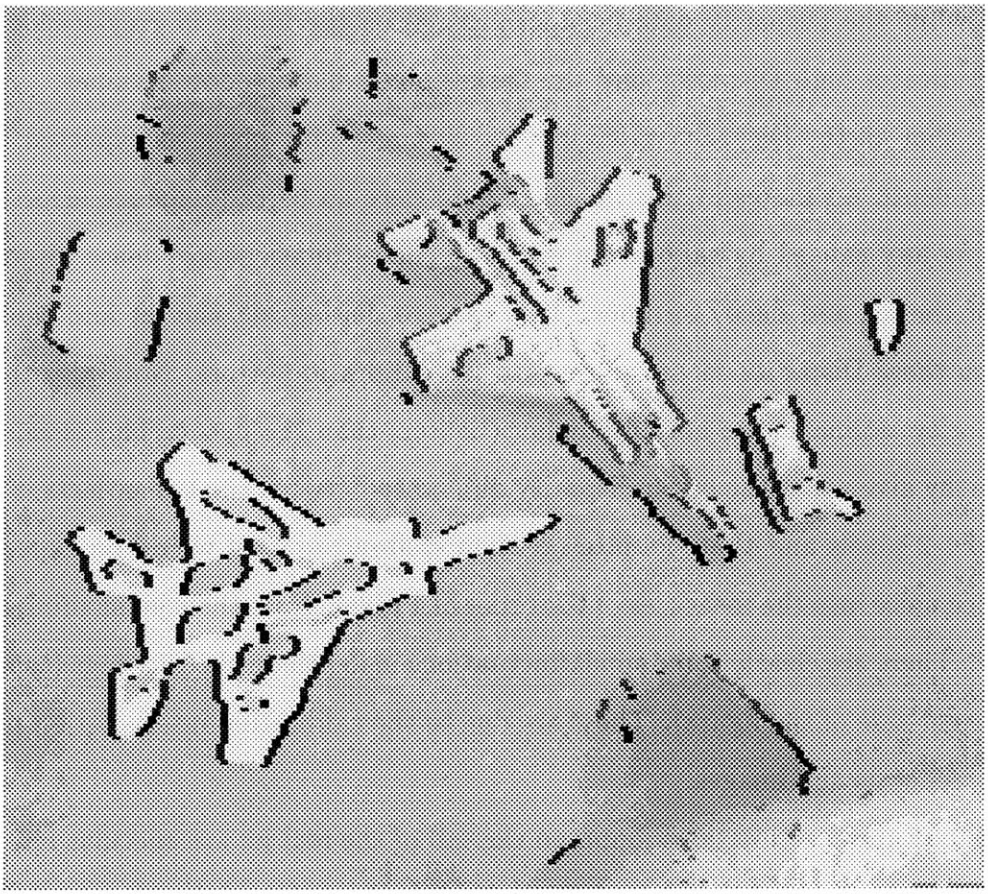

Figure 6-3: Depth Map for Jet and Decoy

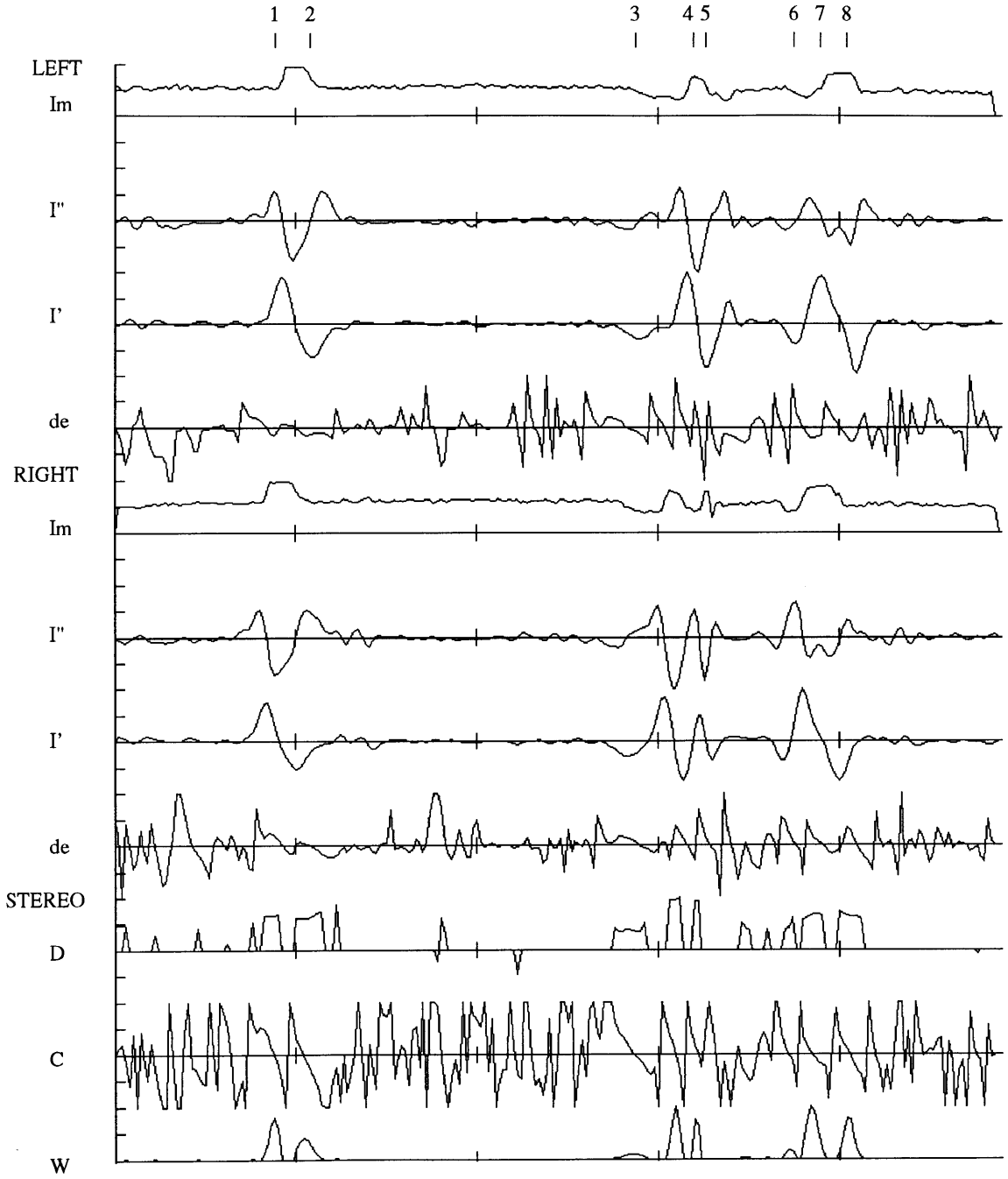

Figure 6-4: Jet and Decoy Signal Formats

The top four signal traces show the left eye Intensity image $I_m(x, y_e)$, the Laplacian of Gaussian convolved image $I''(x, y_e)$, the Gradient of Gaussian convolved image epipolar weight $I'_e(x, y_e)$ and the epipolar displacement $d_e(x, y_e)$ for the fixed sample line $y_e = 110$. Note the first signal steps correspond to the decoy on the left side of the image (indicated by markers 1 and 2 at the top of the figure) while on the right side the jet (4-5) and a car (7-8) are found. Shadows are also cast on the ground by the jet (3) and car (6).

The next four traces are identical to the first four, except that they are for the right image line. The bottom three traces are, respectively, the Disparity $D(x, y_e)$, the stereo Cyclopean image $C(x, y_e)$ and the Disparity weight function $W(x, y_e)$. Where the zero crossings of $C(x, y_e)$ correspond to significant weights in $W(x, y_e)$ the depth map is labeled with the estimate from $D(x, y_e)$.

Note that the representations are dense up to the point of the $D(x, y)$ and $W(x, y)$ steps. For all scales except for the very smallest, the Displacement is also valid at every point in the scene. It is only when the width of g_w vanishes that the representation becomes sparse.

6.6 The Hallway

The next image pair are of a long corridor. One of these images is shown in Figure 6-5. This image pair is challenging due to the very large disparity range. The closest points near the sides of the images have 20 pixels of crossed Disparity while the points in the middle of the picture further down the corridor exhibit more than 20 pixels of un-crossed Disparity. At many points in the scene, such as at the boundaries of the door windows and at various other door edges, the Disparity gradient is at the limit (1.0). Numerous features are occluded and there are plenty of opportunities for false matches.

The superposed depth map of Figure 6-6 shows the nearby points as light and the

Figure 6-5: Hall Grey

distant ones as dark. The images in the windows of the overhead lamps are actually behind the door — not reflections — as the deeper estimates correctly indicate. This, like the jet reflection, is a correct Disparity interpretation. Note the reflection on the floor in the lower right. It preserves the depth of the shelf which cast it, as it should.

At the many occluding boundaries in the image, the Disparity correctly matches despite the presence of the maximum possible Disparity gradient. At almost all points in the scene, the data is correctly interpreted with the exception of only a few (3) points on the left side of the image which assign a wall point greater depth than would be expected. This too *could* be a reflection, but is assumed to be a matching error.

There are 1710 features plotted in this image. There are no correspondence errors due to calibration in the image, presumably due to the camera setup. Algorithm matching errors therefore account for less than 0.2% of the total estimations. This is a *very* low contrast and low acuity image as well that results in edges appearing much more ragged than is usually found in stereo pairs.

6.7 The Campus

The final example is an aerial shot of the UBC campus. This is an example where the epipolar field is decidedly non-horizontal. Two shots of the campus are taken from a plane. In one the heading is slightly rotated from the other. The small detail in the image, such as the cars in the parking lots, guarantee that even the smallest mis-registration in the image will result in false correspondences.

A notable feature of this image pair is the plaza inside the central building, which produces a Disparity gradient of almost 1.0. This presents a number of non-corresponded features in the images to the algorithm. There are extremely subtle variations in height, such as the sloped shadow next to the central building, and the slightly depressed roadway in the upper right corner which are readily detectable, at least in the data and with false color plots. It is a bit more difficult to detect these

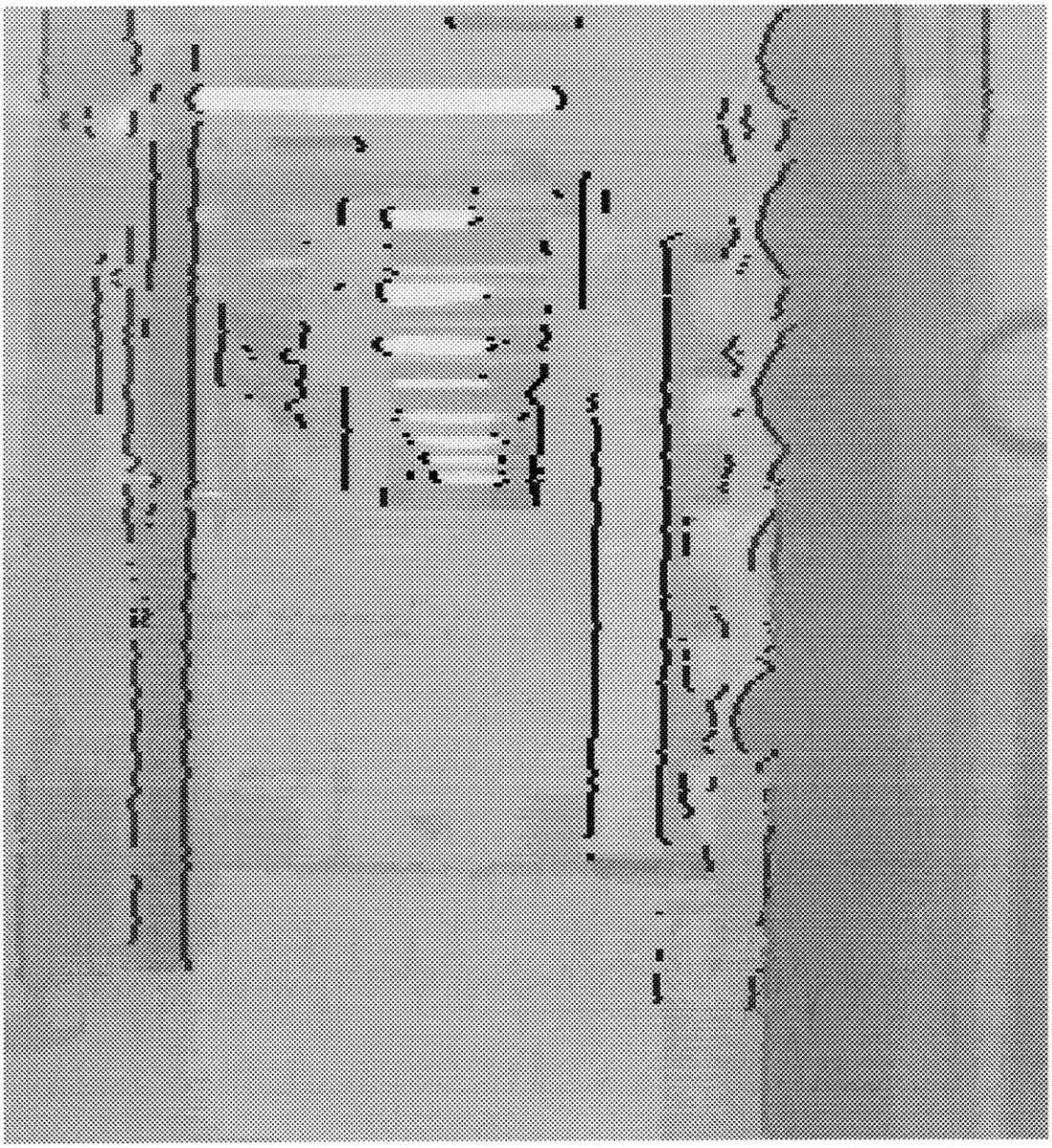

Figure 6-6: hall depth

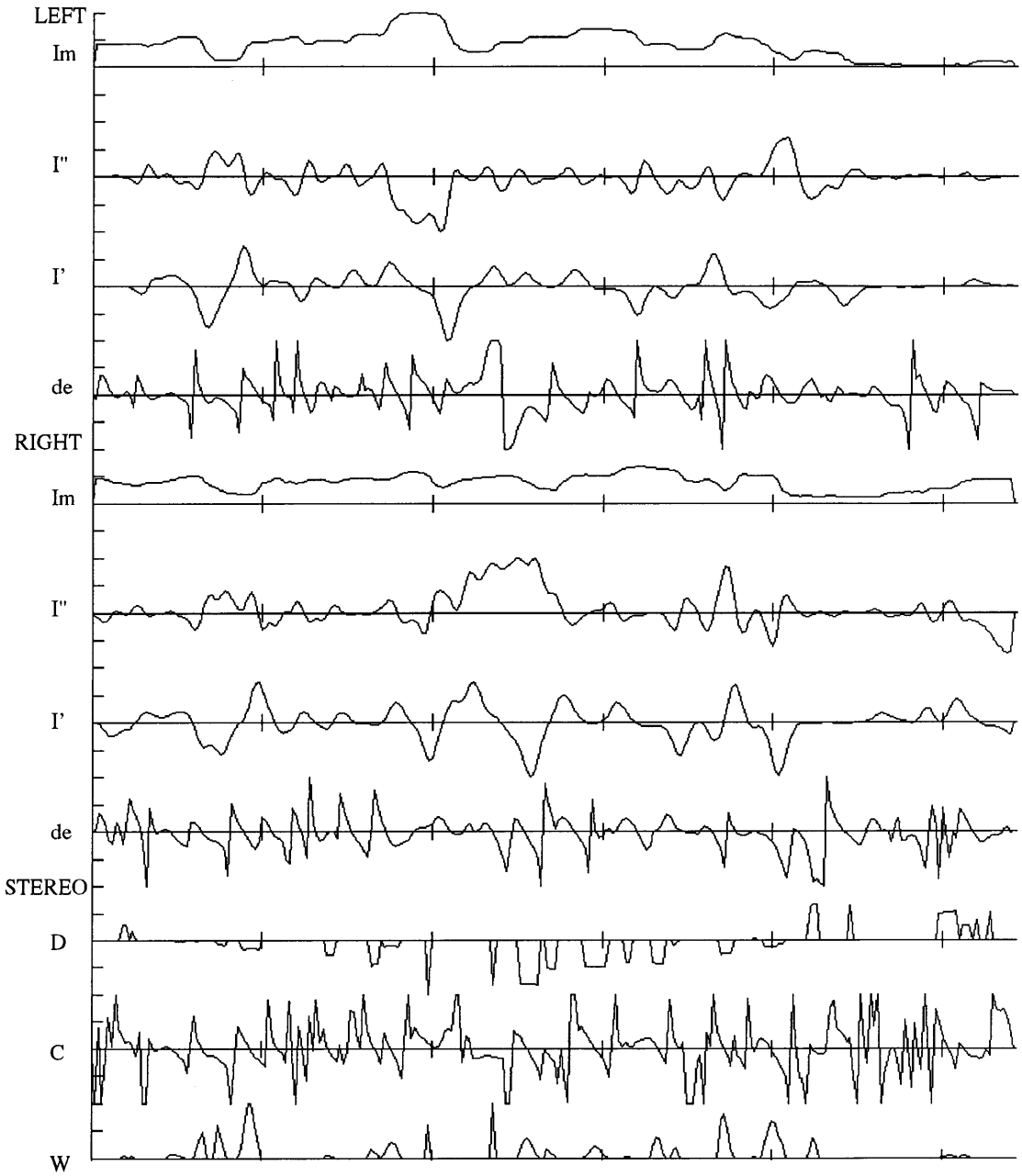

Figure 6-7: Hall Signal Formats

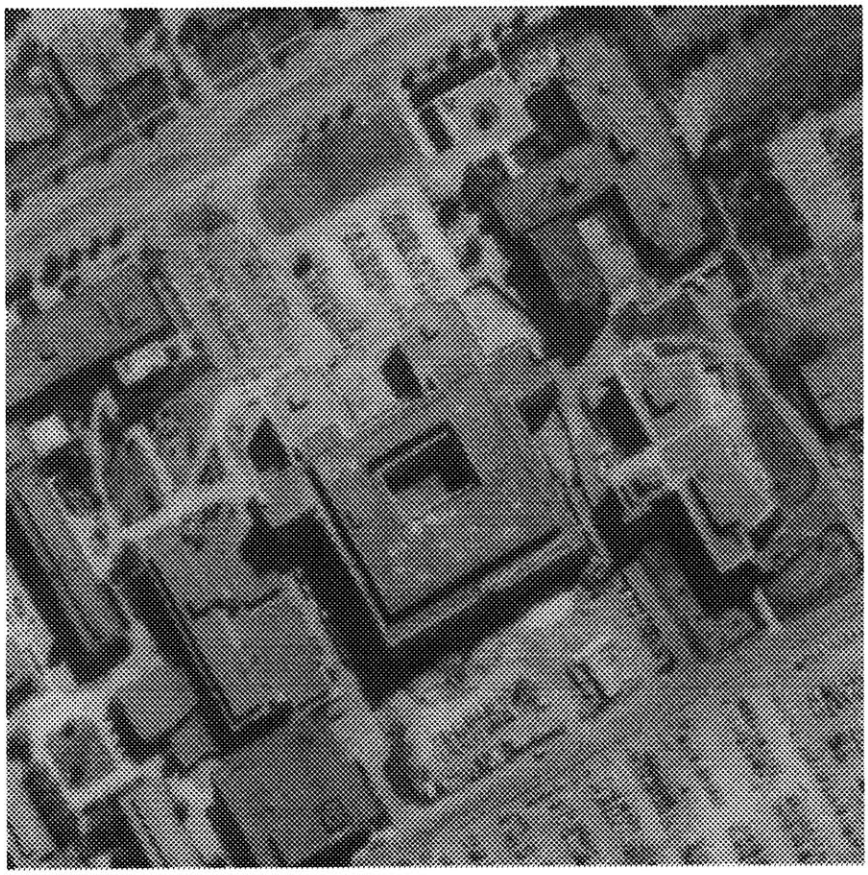

Figure 6-8: ubc grey

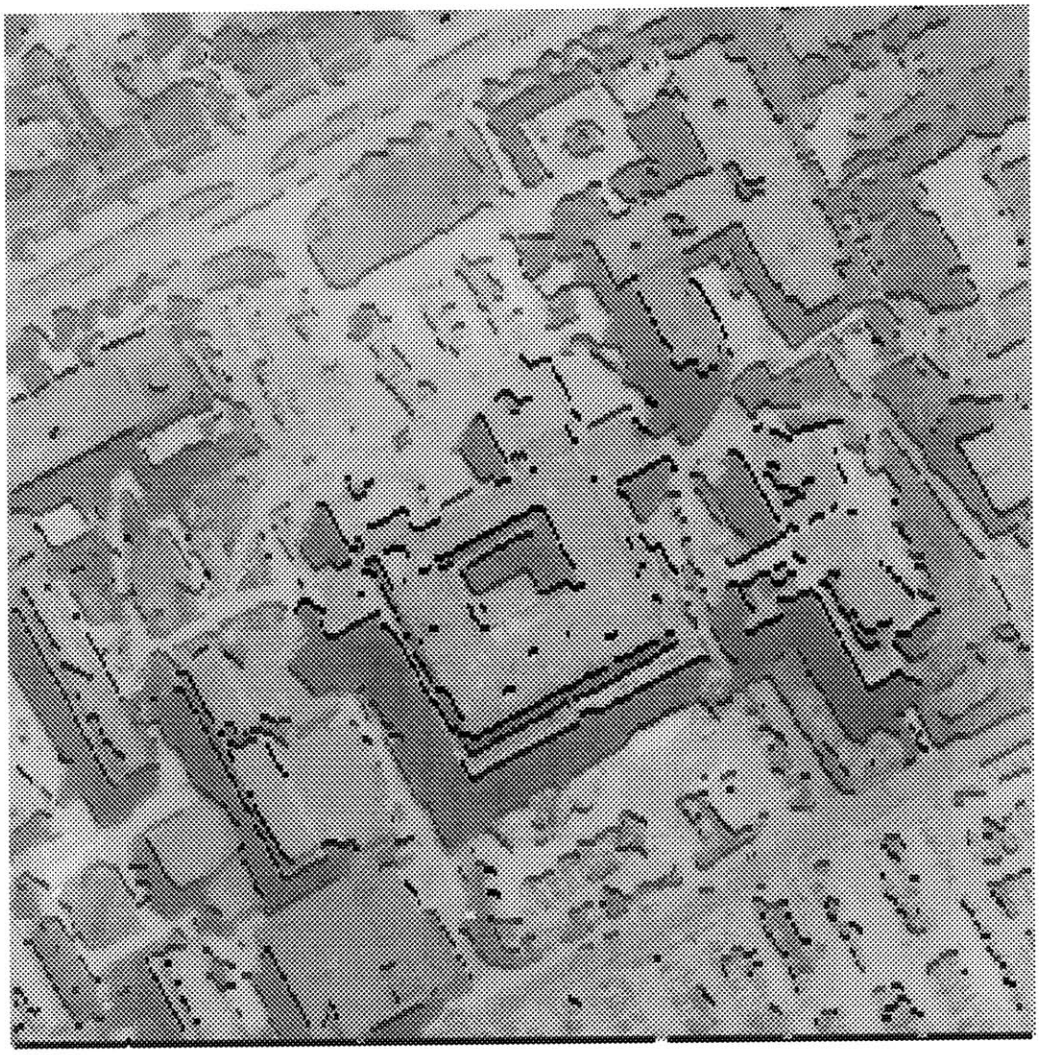

Figure 6-9: ubc depth

in the false grey plots presented here.

The net result, however, is that when up to 5,400 features are extracted from the image pair, many of which are on small randomly structured objects like trees or cars, there are extremely few (0.2-0.4%) false correspondences that are not clearly due to the incorrect epipolar assumptions of the model (which account for 40 errors in the image — 0.7%). It would be interesting to devise a correction to the calibration based on *vertical* disparities in the image, but this has not been tried yet.

6.7.1 Other Constraints and Parameters

The stereo algorithm has both the capability to have constraints imposed on it, much like most symbolic approaches, and can do very well with a minimum number of such constraints. It is not possible to test all combinations of constraints and parameters on the above algorithm, but some tests were run to examine the influence of some of them.

The Convolving Width — g_w

The Gaussian g_w used to bound the maximum-likelihood variance weighting of Disparities does not serve precisely the same function as the 1D noise model convolver, since in that model the Gaussian was *part* of the variance model and in the 2D case it merely serves to render the estimation local. Some tests were run to determine whether a 1D or 2D Gaussian convolution was best and whether the width of the convolver should indeed reflect the widths g_w determined in Chapter 4.

It was found that the width of the convolver should be approximately σ_b and that the widths σ_w appear to give the best results. A 2D convolver was also found to produce the best results.

Combining Scale Disparities

As will be mentioned in the next section, a provocative finding by Mayhew and Frisby is that the loss of spatial frequency information in any one channel results in loss of stereo fusion in random dot stereograms [56]. This would seem to indicate that a structure where each scale directs the next smaller scale directly might be at work.

Another approach considered was to have each scale take the weighted average of Disparity estimates based on the variance measure $W(x, y)$. Thus, if any scale found little or no Disparity information (based on its own $W(x, y)$ measure), then the larger scale estimates would be allowed to dominate. It was found that such an approach did indeed seem to help in some images, although the improvement was marginal.

Iterating the Estimation

When a Disparity is estimated at any point and at any scale, when the images are shifted locally to align them and the same computation of Disparities calculated, the resulting Disparity field will not be zero. Given this, it might be useful to repeat the calculation for some number of trails at any given scale before moving on to the next smaller scale.

This indeed did seem to improve the performance, but no noticeable improvement was seen for more than two passes at any scale.

Orientation and Focus Mismatch

These constraints are easy to implement, so some testing was performed on how much they improved the results. Focus mismatch ($\dot{D}$) restriction was found to reduce the number of features found significantly (maybe 50%) while it improved the error rate marginally. In other words, mis-corresponded features were more likely to be pruned by the focus filter than correctly matched features. The same was true of using an edge orientation filter.

The utility in using these kinds of filters with the tradeoff between error rate (which never exceeded 1% anyway) and sample density. If extremely low error rates are necessary, matching constraint filters indeed seem to help. On the other hand, the sacrifice in sample density to achieve such low error rates may not make such filters desirable in some applications.

The scene character may also be an important consideration. Where the image is poorly focused or where the depth of field is very large relative to the depth disparity range, focus will not be a very useful cue. In scenes where edges are locally aligned — i.e. just about any indoor scene — orientation cues are of little help. In natural scenes, especially with dense random detail, it could be a critical matching criteria, especially with poor calibrations.

Chapter 7

Natural Early Vision Models

The computational model for Displacement and Disparity representations is designed primarily to provide a good early vision model to be used in computational vision research. It has also been partly motivated by a desire to have the computations performed be consistent with what is known of the biological processing of images, particularly in primates.

Of course, any attempt to propose a computational model of cortical function is fraught with risk. Despite the huge amount of anatomical, psychophysical and neurophysiological research on primate vision in recent years, it is impossible to claim with any confidence that the evidence supports any one particular model of computation.

Despite this, there is still every reason to examine any computational model in the context of the biological system, even if they are clearly different. Whenever a model differs drastically from what is known of biological systems, it is useful to understand the consequences of that departure. For instance, symbolic models usually abandon precise positional measures in order to reduce the representation size. If this were consistent with natural vision models, then human acuity would suffer from such effects. The fact that no such degradation is found has led to many proposals for special mechanisms to "correct" the symbolic models.

It is therefore useful to check back with the model that motivates most compu-

tational vision in the first place — the biological vision model — and see what the biological research may say about the Displacement/Disparity model, as well as what the Displacement and Disparity concepts might predict in the natural vision research areas.

This chapter selects a variety of topics typically found in the biological research literature. It is not possible to cover all the possible relevant research areas or even all the research in the few areas selected. There has been an effort to select topics that are central to the model design.

Wherever the model supports the research, this will be pointed out. Perhaps, though, the most interesting issues are the areas where the model predicts behavior that either is not clearly documented or has simply never been tested for. These are the areas that deserve the most attention.

7.1 Hyperacuity

Acuity is the ability to detect small changes. Spatial acuity is the ability to detect small changes in the spatial position of features. Hyperacuity is the term applied to the phenomenon where humans are able to detect image positional differences that are much smaller than the diameter of the smallest foveal cell diameter [87, 89, 90]. Vernier acuity, the ability to detect a small lateral displacement between two aligned vertical bars, is of the order of 5// while the smallest cone diameter is of the order of 30//. The smallest receptive field is, at best, consistent with a single cell diameter corresponding to the central region of the Laplacian of Gaussian [51] that would correspond to a diameter of $2\sqrt{2}\sigma_b$. Thus human acuity is of the order of $0.5\sigma_b$ of the smallest spatial channel.

It is not entirely clear what the mechanism of acuity threshold detection is. In other words, it is not obvious what measure to use on any given model to compare with human threshold experiments. One simple measure could be the standard de-

viation of a positional measure. Early experiments by Watt and Morgan suggested a compressive power law of -0.5 for the contrast/acuity threshold relationship [88] but Wilson and others found a power law relationship much closer to -1.0 over a broad range of contrasts [90]. In other words, the acuity threshold is inversely proportional to contrast.

Recall the findings of chapter 4 Equation 4.5. This shows the relationship between contrast ($\frac{\alpha}{\sigma_n}$) and the standard deviation of the Displacement function:

$$\begin{aligned} \sigma_d(x) &= \frac{\sigma_n \sigma_b^2}{\alpha g(do(x))} \\ &= \frac{\sigma_n \sigma_b^2}{I'(x)} \end{aligned}$$

This model suggests that the -1.0 power law would be expected for contrast variations as Wilson's experiments suggest.

Similar results can be found in experiments for detecting acuity thresholds using blurry edges, where Watt and Morgan report an average power law between the blurring width and the threshold acuity of 1.5. When the blurring is Gaussian, their experimental data supports a power law of 2.0, meaning that the ability to resolve edge alignments rapidly degenerates in such tasks [88].

The effects of blur on the threshold model will be complex since blurring affects both Displacement gain ($\nabla \cdot \mathbf{d}$) as well as σ_d. Presumably the detection threshold should be proportional to the sensitivity of the Displacement signal to the displaced edge and inversely proportional to the standard deviation of the Displacement errors due to noise. Experiments were run on this threshold model and the data is plotted in figure 7-1. Although the power law is not precisely quadratic, it is reasonably close, especially given the limited data in the psychophysical experiments.

Perhaps the most important observation that can be made about the research is the large amount of evidence of hyperacuity in a variety of vision domains. Edge location [87], motion [20] and stereo [43, 68] are a few good examples of spatial tasks

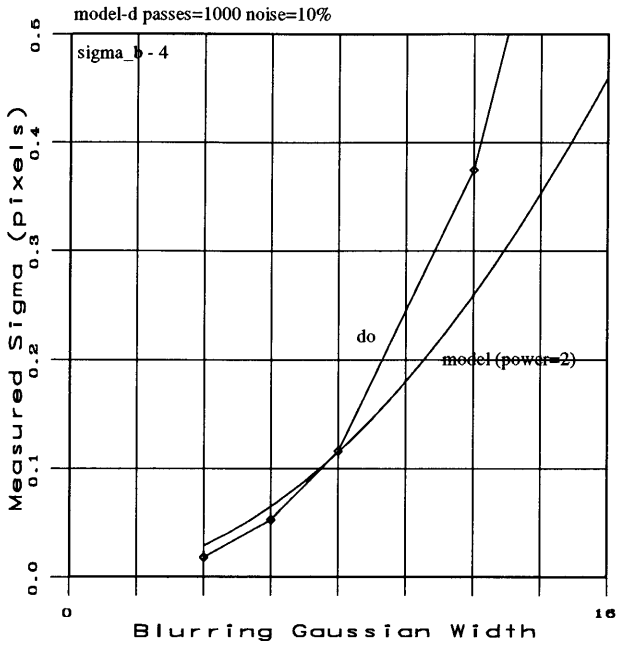

Figure 7-1: Effects of Blur on Displacement Thresholds

in that humans demonstrate superb sensitivity to very small stimuli. The consistent preservation of acuity argues against special mechanisms such as "granular cells" alongside pixel acuity mechanisms for stereo or motion. No evidence exists that the cortex uses any special mechanism to restore acuity to otherwise pixel-acuity vision representations.

Some other continuous domain computational models have been proposed to provide acuity [72, 93] but these are still aimed specifically at performing spatial acuity tasks, not general early vision.

7.1.1 Sampling at Zero-Crossings

One interesting side note of the acuity research is that Watt and Morgan [88] argue that features can only be resolved when there exist distinct zero crossings in the Laplacian of Gaussian retinal model, thus supporting that model of retinal computation, and possibly the idea of sampling at zero-crossings.

The zero-crossings of the Laplacian representation correspond to the zero-crossings of the monocular Displacement and the Cyclopean fused vision Displacement representations. The present model suggests that sampling take place *after* low level continuous domain calculations are performed, however. Thus the evidence for zero-crossings supports the present model as well.

7.2 Retinal Sensor Models and Gaussian Receptive Fields

The retina transforms the intensity image into a representation that can be modeled accurately as either a convolution with a Laplacian of Gaussian or a difference of Gaussians. Note that both approaches use Gaussian Receptive Fields (RFs), which are generally accepted models of the early representations both in the retina and the visual cortex [7, 94]. The Difference of Gaussian (DOG) model of retinal and cortical RFs is most noted for the ability to precisely match the responses of cells to the model [18, 91]. There are not many computational models that make use of the DOG model, mainly because it does not appear to lend itself readily to any obvious analytical technique. On the other hand the Laplacian model of Marr and Hildreth has received a great deal of attention as a computational model [50, 53, 54, 77]. It is not as good as DOG models in precisely matching RF properties — for instance, the center-surround cells have non-zero response to uniform illumination fields, Laplacian operators do not — but the closeness of the approximation and the utility of the functional form in analysis help explain its popularity.

Two RF representations are found at the retinal Ganglion cells. One responds strongly to a bright stimulus in the center of the RF and is inhibited by the response just to the side of the central region[1]. These are called ON-center cells. Many cells

[1]RFs can be determined by 1) measuring the excitatory and inhibitory cell responses to some specific stimulus or 2) by changing the nature of the stimuli — such as by noting cell responses to

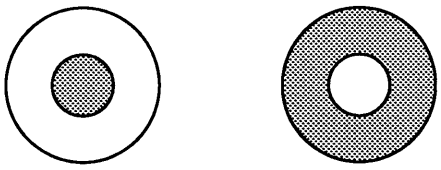

Figure 7-2: Retinal Ganglion Receptive Fields

also have the opposite response — the cell response is diminished by a stimulus in the center of the RF and excited by the surround. These are called OFF-center cells. These RFs are shown graphically in Figure 7-2 where the light region indicates excitatory response to light stimulus and the shaded region inhibited response.

This kind of paired, or complementary, RF representation in the visual system is not uncommon. It is also not uncommon in analog systems. Complementary signals, often called differential pairs, are useful in low noise designs.

The Displacement/Disparity model makes use of the Laplacian of Gaussian model of retinal processing. In Chapter 4 additional evidence was provided to support the Laplacian retinal model on the basis of signal to noise analysis. The basic conclusion was that when only one signal format could be sent over a noisy channel — in this case the optic nerve, Lateral Geniculate Nucleus and optic radiation — between the sensor (eye) and the processor (visual cortex), the highest order spatial derivative needed in the subsequent processing should be sent. Lower order forms, as needed, should be reconstructed at the processor (cortical) stage. If the gradient was the highest order representation needed in vision, then that would presumably be the representation sent between the eye and brain.

When motion is involved, the 1D model suggests the highest order is $\dot{I}''$, or the temporal derivative of the Laplacian of Gaussian. This may indeed be the function of transient retinal (Y) cells [75, 76, 54]. The motion model is not discussed in adequate detail, however, to attempt to predict any cortical instantiation.

The Displacement/Disparity model is based very much on the Marr-Hildreth reti-

---

light or dark bars or spots.

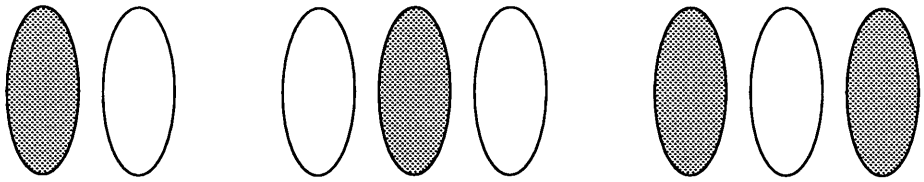

Figure 7-3: Hubel's Simple Cell Receptive Fields

nal model. While many symbolic models are as well, most approaches almost immediately convert to a symbolic scheme by marking the edges at the zero-crossings in the retinal image. In this model, the continuous domain is preserved in the subsequent processing steps.

7.3 Linear Cortical Models — simple cells

The very first stages of processing the retinal signal are probably in cortex, although undoubtedly some form of signal conditioning takes place at the Lateral Geniculate Nucleus (LGN). The LGN is a piece of the Thalamus that passes the signal between the optic nerve and the nerves that carry the center-surround RF representation to the rear of the brain — the visual cortex, or V1.

Hubel and Wiesel discovered in 1962 that there are cells in V1 that, like the retinal RF cells, combine retinal inputs in ways that are linear — you can map the cells by using simple stimuli and the response to complex stimuli can be predicted easily through the superposition property of linear representations [39]. Moreover, of the cells studied, the vast majority fell into three basic RF categories. These cells were orientation selective to bar stimuli and had two or three regions of response to light and dark bars. A number of studies on cats and primates have confirmed these findings [38, 39, 55]. Figure 7-3 shows a simplified plot of the three RFs Hubel and Wiesel discussed. A response to a light bar is indicated by the light areas, response to a dark bar by the shaded areas.

Recent research indicates that these 2 and 3 lobed cells comprise 90% of the simple

cells. Most of the rest are motion selective[55]. Note that the two three lobed RFs are simply, once again, complementary representations. Complementary RFs do not constitute unique representations, just as the two retinal cell RFs are both modeled as Laplacian of Gaussian. Thus there are really only three simple representations; the center-surround, the two lobed RF and the three lobed RF. The first one has a model; the Laplacian of Gaussian.

So a major question is; what are the two and three lobed simple RF cells doing? What might they have to do with the Displacement model? The proposal of the following section is that they represent the steps needed to construct the gradient representation.

7.3.1 Gradient Representations in Cortex

In the development of the 2D Displacement model (Chapter 5) it was necessary to use the gradient of Gaussian convolved image. Since the Laplacian of Gaussian RF exists, a method was needed to reconstruct the gradient representation $\mathbf{I}'(x,y)$ from the Laplacian $I''(x,y)$. In that chapter (Section 5.2.4) a method was developed that uses a local recursive filter.

That filter reconstructs the gradient by generating local estimates of the curl and divergence of the gradient field and testing those against the Laplacian input. The local curl estimate should be zero, the local divergence should be equal to the Laplacian. These calculations involve taking partial derivatives of the gradient components I'_x and I'_y.

Figure 7-4 shows schematically this recursive filter calculation. The RF responses of the various calculation representations are also shown. The boxes indicate signal representations and the circles represent simple linear calculations; summation and spatial derivatives. The gradient representation is in the middle of the figure. The gradient of Gaussian RF will have two lobes. The RF of the representation as revealed by a spot stimulus is shown in Figure 7-5.

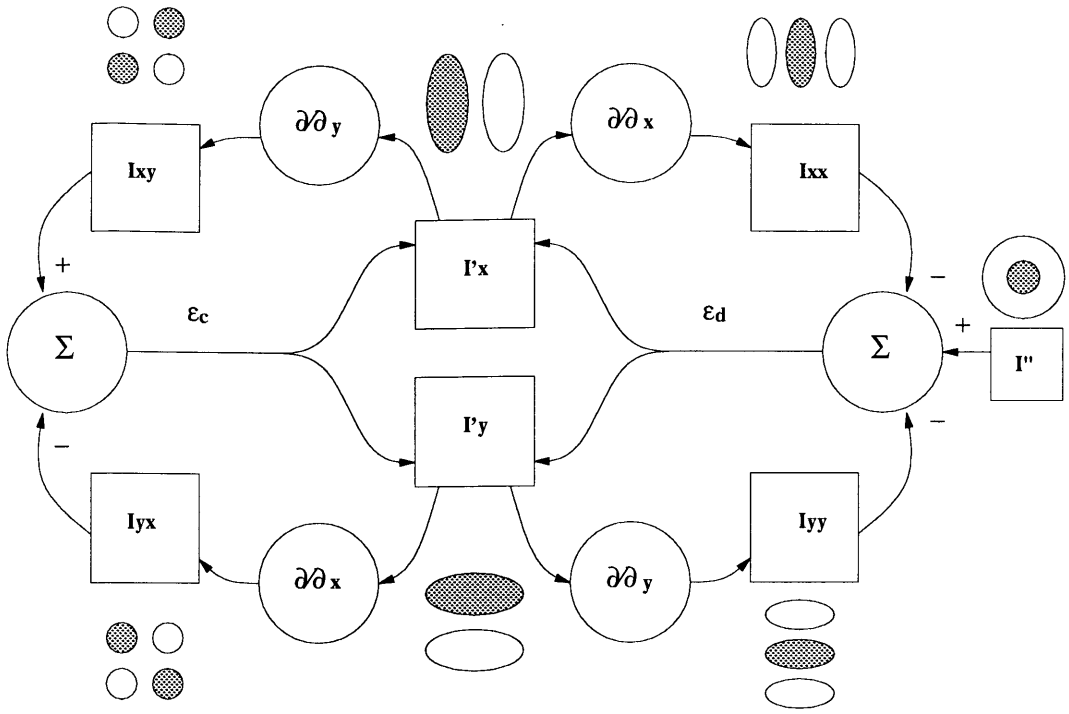

Figure 7-4: Gradient Inverse Calculation

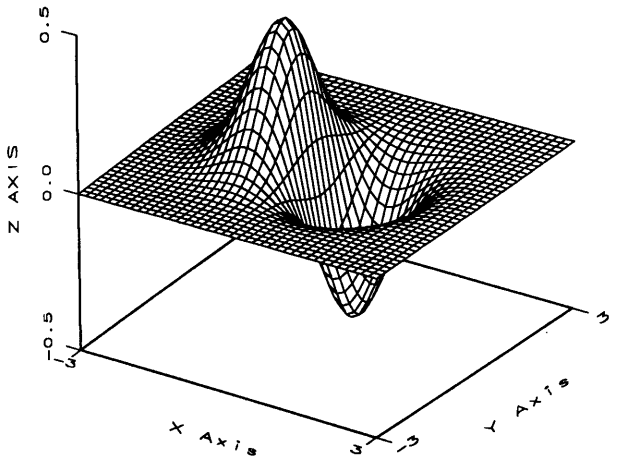

Figure 7-5: Gradient Cell RF

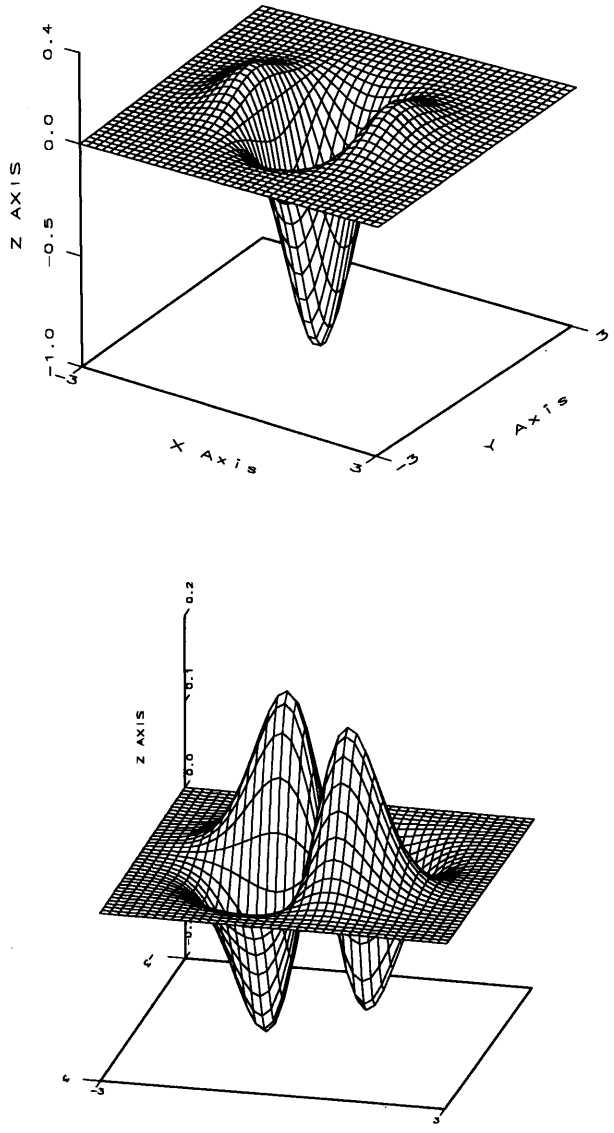

Figure 7-6: I_{xx} and I_{xy} Cell RFs

The Laplacian input is shown on the right ($I''(x,y)$) of the diagram with its center-surround RF. The filter operates by calculating two corrections — the curl error ϵ_c and divergence error ϵ_d. These, being error signals, have no RF since they should be zero for any stable solution. The error signals are used to correct the gradient estimation (shown only as a feedback path here — for details, see the discussion in the 2D chapter).

The calculation of ϵ_c and ϵ_d require four new representations. The divergence calculation — shown on the right — takes the partial $\frac{\partial}{\partial x}$ of the gradient I'_x component. This is shown in the upper right (I_{xx}). Similarly the y partial of $\mathbf{I}'_y$ renders I_{yy} (lower right). Note the RFs of these representations. These are three lobed cells. The sum of I_{xx} and I_{yy} should be equal to the Laplacian, so these three signals generate ϵ_d. The RFs of these cells are shown in Figure 7-6.

On the left side, the curl calculation is needed to complete the solution for the gradient. The partial with respect to y of $\mathbf{I}''_x$ is shown in the upper left as I_{xy}. The partial with respect to x of $\mathbf{I}'_y$ is in the lower right (I_{yx}). These should be equal, so their difference is the curl error ϵ_c. Note, however, the receptive fields of these two representations. On the face of it, there would appear to be no such RF recorded in the literature.

Most simple cells are mapped using bars of light. The mapping process involves finding the "preferred orientation" of the cell by rotating the bar orientation until the cell has a maximum response, then the cell is mapped by noting the response of the cell to moving the oriented bar around the RF. The cells of the curl calculation would, under such a procedure, strongly "prefer" a diagonal bar orientation. They would also appear "sharply tuned", i.e. very sensitive to bar orientation, as compared to the other oriented cells. This response may very well have been classified as an elongated sharply tuned three lobed RF with the methods used [82].

Thus it is possible that all of the non motion selective simple cell representations found in the cortex can be explained by the inverse gradient calculation model and

it would appear that no necessary computational representation is inconsistent with the cortical RFs. Thus the gradient of Gaussian is consistent with the representations found. It is a well accepted representation in computational vision and some researchers argue that it is also an essential component in natural visual processing [8].

7.3.2 Phase Models — Gabors?

There is evidence that there are "paired"(adjacent) phase representations in V1 that exhibit phase shifts of precisely 90^o [65]. This is sometimes used to argue for some sine/cosine Fourier model implementation. Gabor models do indeed use quadrature (90^o phase shifted) basis functions.

It should be noted, however, that *any* computational model that incorporates orthogonal basis RF operators will be supported by this evidence. The Laplacian of Gaussian retinal model is orthogonal to both of the gradient of Gaussian signals — they will exhibit this 90^o phase shift for all spatial frequencies.

There are any number of other functions that would produce such a result, but since all the above proposed representations would predict this quadrature phase relationship relative to the Marr-Hildreth retinal model, this is an important observation.

7.3.3 Orientation Preference

When the orientation preference of orientationally selective simple cells is examined in the visual cortex, there is a surprising amount of disagreement about whether or not there is some manner of orientation preference.

Hubel has produced substantial evidence that when neighboring cell orientation preferences are examined in V1, the distribution of orientations preferences is continuous. No orientation is preferred [38]. Some have modeled this orientation preference arrangement as something of a radial array of preferences about the cortical columns. There is some question as to how precise the orientation mapping can be using Hubel's

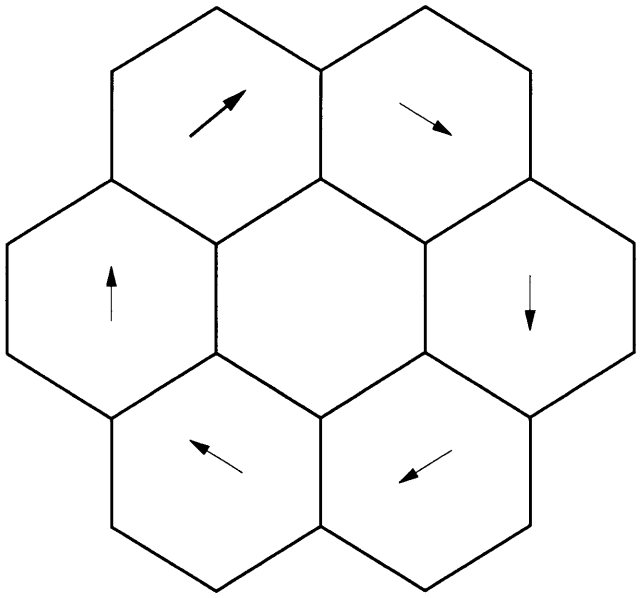

Figure 7-7: Hexagonal Tessellation Curl Gradient Directions.

methods, but there does appear to be a wide array of orientations used in the simple cell computations — certainly more than two.

Other researchers, most notably Mansfield, argue that there is strong evidence for orientation selectivity and, at least in the foveal region, locally paired horizontal and vertical orientation preference [86, 48, 46, 47]. Some studies indicate almost *all* cells in some regions of V1 are not only selective, they are selective exclusively to vertical and horizontal orientations. There is no small amount of acrimony over these conflicting findings. The Displacement model may serve to reconcile some of the seemingly contradictory evidence, especially when coupled with a biologically sensible sensor model.

The retina is best modeled as a hexagonal tessellation of cells. This is how cells are arranged when densely packed. The previous discussions of the gradient model used spatial derivatives of the gradient to calculate the curl and divergence of the gradient image estimation. When hexagonal tessellation exists, however, the model is a bit more involved. The curl calculation, for example, will sum the directional derivatives of the six neighboring cells as shown in Figure 7-7. The individual derivatives are

simply scaled sums of the gradient x and y components of the neighboring cells. They will still have the same RF form, just rotated. The divergence derivatives will also be radially oriented, except that the arrangement will be like radial spokes.

The gradient is, however, still a paired orthogonal representation and there is every reason to presume that all cells will (at least locally) use the same orientation. The reason for horizontal and vertical preferences is that when looking at a distant object, the epipolars are horizontal. So too, presumably, should be the gradient representation alignment. Unlike some past theories, this logic has nothing whatever to do with anthropological or cultural arguments of orientation preference.

Hopefully, since the gradient representation is logically and physically distinct from the curl and divergence representations that solve for it, there is some hope that careful review of these two findings will prove that they are indeed compatible. Hubel may have been examining the inverse gradient solving mechanisms and Mansfield the Gradient itself. Perhaps the answer is as simple as that these could be found on different layers of the cortex.

7.3.4 Scale Space Models

The motivation for scale space computer algorithms is usually based on the psychophysical research of Wilson who suggested that there are between four and six frequency selective channels in the human visual system [92, 93]. There is a lot of other evidence from both psychophysical and neurophysiological research for a multiple channel system involving various RF sizes and frequency tunings [53, 38].

One of the more interesting findings that not only lends support to a multiresolution scheme, but a hierarchical scale-space mechanism as suggested here is the finding of Mayhew and Frisby that when all the frequency content of a single channel is filtered out of the image, stereo fusion is lost [56] as well as the finding by Julesz and Miller that noise corruption of a channel has the same effect [44].

Although this does not exclude other scale space schemes, the simplest "big leads

small" design proposed in Chapter 5 is probably best supported by this finding. One modification tested, for instance, was a weighted sum of scales based of each scale variance measure. This design would not appear to be compatible with the above findings, in spite of the fact that it seemed to work at least as well as the naive design on real images.

7.4 Nonlinear Models

Complex cells are characterized by having nonlinear response characteristics such that direct mapping of excitatory and inhibitory regions using simple stimuli is no longer possible. One important aspect of complex cells is that they respond the same to stimuli regardless of contrast reversals [38, 60]. This is what we would expect from the contrast invariant Displacement distance representation.

Since complex cells are nonlinear and, indeed, complex, linear analysis of their response to stimuli is not very enlightening. Unfortunately, much published work attempts to use Fourier methods to map these cells' RFs[24]. This results in a rather impoverished understanding of these cells' behavior.

This section touches on some of the characteristics that might be expected of an instantiation of the nonlinear Displacement and Disparity computations in cortex.

It has been noted how sometimes simple cells have complementary representations. Another cellular representation issue is how "positive" and "negative" signal content is represented. Retinal ganglion cells have a background firing rate such that negative signals are represented by firing rates less than background and positive signals have rates above background. There is apparently also another form of cellular signal representation where the positive and negative portions of the signal are separated into two signals (see, for example[75, 76]). This can be called a *rectified* representation scheme. In analog signal processing, this manner of splitting signal phases is a common phenomena, especially with nonlinear devices. It seems likely

that the nonlinear Displacement/Disparity representations would be represented in such a rectified form.

7.4.1 Displacement Representations in Cortex

Just as was done with the simple cell population, it would be useful to predict what the RF of a Displacement "cell" might look like. Simply looking at the functional forms is not much help because $d_o(x,y)$ is unbounded. Thus the Displacement RF, even when rectified, is not at all local.

$$\begin{aligned}\mathbf{d}(x,y) &= -\sigma_b^2 \frac{I''(x,y)}{\|\mathbf{I}'(x,y)\|^2}\mathbf{I}'(x,y) \\ &= -\sigma_b^2 \frac{I''(x,y)}{\|\mathbf{I}'(x,y)\|}\hat{\mathbf{\Theta}} \\ &= d_o(x,y)\hat{\mathbf{\Theta}}\end{aligned}$$

The Displacement function, however, is only valid over a region with a radius of some small multiple of the filter σ_b size. There is no need for the function to continue to grow past a certain distance from the edge $\|\mathbf{d}(x,y)\|$. What should the RF look like past this point? It does not really matter. What matters is how the Displacement function behaves near the *boundary* of its RF where $\|\mathbf{d}(x,y)\|$ is near zero — i.e. near the edge feature.

Ironically, this turns the normal notion of RF mapping on its head. The RF of a complex Displacement representation would be maximum at the point where the cell would cease to measure distance correctly. At this point it no longer matters what the RF response is. At some boundary points of the RF the response will correspond to the critical edge $d_o(x,y)$ response shown in the above definition. At the rest of the RF boundary response points, the cell response and RF form would be utterly irrelevant.

One model for bounding the RF of the displacement function is to bound the

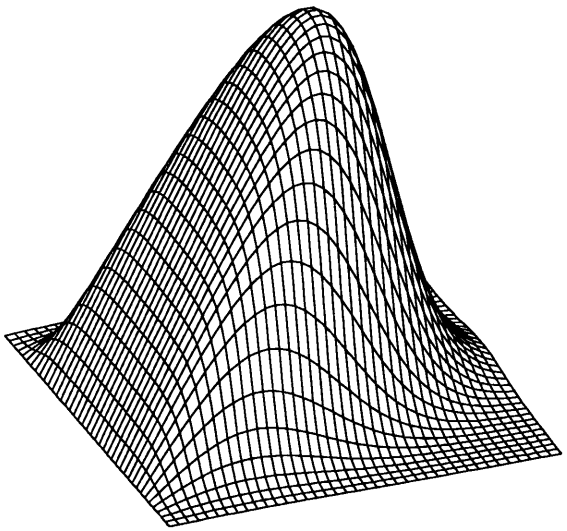

Figure 7-8: Possible Displacement Rectified RF — Bar Stimulus

minimum gradient magnitude of the denominator, by adding a small δ term to the rectified RF calculation (disregarding its orientation):

$$d(x,y) = -\sigma_b^2 \frac{I''(x,y)}{\|\mathbf{I}'(x,y)\| + \delta}$$

A sample plot of such a RF is shown in Figure 7-8 to a hypothetical narrow horizontal bar stimulus. The precise stimulus or RF forms are less important than the general form of the RF structure shown here. It is clear that there is a very planar response profile near the edge response region (the left face of the plot). The rest of the RF is where the δ term takes over and the response profile is largely irrelevant. The RF response region away from the edge zone is also irrelevant computationally,

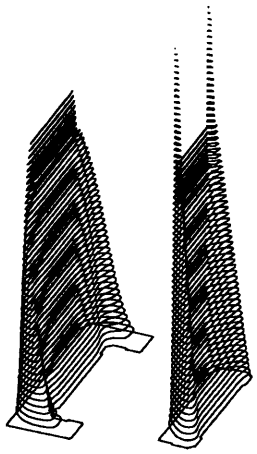

Figure 7-9: Possible Displacement Rectified RF — Box Stimulus

since the weight function is also nearly zero. Near the edge, the effect of the δ term is vanishingly small, making this a reasonable modification to the computational model.

A large stimulus, like a box, with long edges would take on the response profile shown in figure 7-9. The gain of the plot, being arbitrary, makes the Displacement RF signature look remarkably unlike its idealized model. In fact, this plot is taken from the stereo algorithm response to a box input that was correctly matched by the stereo algorithm.

One important observation at this point is that the stimulus used will have great impact on the expected RF response. If simple Displacement responses *do* exist, it has already been demonstrated that the Displacement response to a narrow bar is decidedly pathological (see Section 4.6.2). Unfortunately, this would probably lead the experimenter to assume the singular Delta response demonstrated a "preferred" stimulus form. Perhaps this explains the preference for "bar" stimuli in research. If, indeed, Displacement RFs exist in cortex, it would probably be wise to seek them with simple contrast edge stimuli (such as boxes with large edge lengths) in spite of the probability that the cell response would be somewhat diminished.

As mentioned, the expected Displacement response would look decidedly sloped (at least on one side). There is no unambiguous evidence yet that cells exist in the cortex that produce responses consistent with what could be called a Displacement representation. On the other hand, many complex cells *do* seem to exhibit the very non-uniform responses that are consistent with the expected Displacement response [26, 83, 81]. The researchers in these instances did happen to use the preferred large edge stimuli. On the other hand, Displacement could be a process carried out at the *synaptic* level.

7.4.2 Nonlinear Computation — Division at the Neural Level

The Displacement representation is nonlinear because it involves division. For many years, researchers have argued that some neurons seem to carry out nonlinear steps such as signal multiplication and quasi-logical AND-NOT logical operations through a synaptic process called "shunting inhibition" [45, 71].

The AND-NOT model suggests that the cell will fire only when the excitatory input is active and the inhibitory input is not. Recently, the process has been modeled as division with the numerator N as the excitatory input and the denominator D as the inhibitory [32] and is found in "disparity selective cortical cells". The model has a small offset in the denominator to bound the response — just as was used in the preceding discussion on Displacement RFs:

$$Q = \frac{N}{D + \delta}$$

Therefore, there may be no need to search for Displacement cellular representations in cortex. It may be that the central element in the early vision model — the Displacement representation — may not even have a cell-level representation. It may exist only at the synapses of Disparity RF cells.

7.5 Binocular Fused Vision and Stereo

Early models of stereo vision proposed multiple pools of disparity cells to resolve stereo depth [52]. Gian Poggio provided neurophysiological evidence that instead of multiple disparity "channels" there were really only two classes of Binocular cells; near/far, and tuned excitatory/inhibitory. Much research exists to support this model of stereo and binocular cell populations [21, 67, 69, 66, 70].

Disparity models proposed here transform the left/right Displacement representa-

tions into the stereo Disparity (D) and Cyclopean Displacement (d_C) representations. One is used for depth discrimination, the other for all 2D scene analysis. Both are binocular. The weight functions are also binocular. The fused Cyclopean weight (w_C) responds to either eye input, the stereo Disparity and stereo Cyclopean weight (w_D) respond only to simultaneous input in both eyes. Of these four different binocular representations, only one provides information about depth disparity — $D(x, y)$.

7.5.1 Stereo Disparity — D

Stereo Disparity, using edge, not bar, stimuli, should take on the appearance of the monocular Displacement inputs since, after all, it is simply the difference between two of them. The difference is that when the RF is plotted as a function of lateral position, it should be aligned to the line where the Disparity is constant (see Figure 7-10). The form of the RF may be identical to the Displacement RFs discussed earlier, or it may be weighted by W_D as is suggested in Chapter 5.

The latter is shown conceptually in Figure 7-10. The response is zero along the horopter (shown as $D = 0$). It would increase linearly for increasing disparity up until the cell limits, perhaps at the Panum limit. Panum's fusion region of roughly $\pm 2\sigma_b$ of the largest foveal channel is consistent with the stereo Disparity model performance with real noisy images [78]. The response beyond this disparity is not predicted by the model. Perhaps there is enough certainty in the noisy measures to resolve disparity to the precision of "in front of" and "behind" the horopter, but there is no evidence of that so far.

The RF of Poggio that best matches this predicted response is the near/far representation. It is encouraging to note that he and others report that these cells exhibit "ocular imbalance", or the need for input from both eyes to be active. The stereo model predicts that as well.

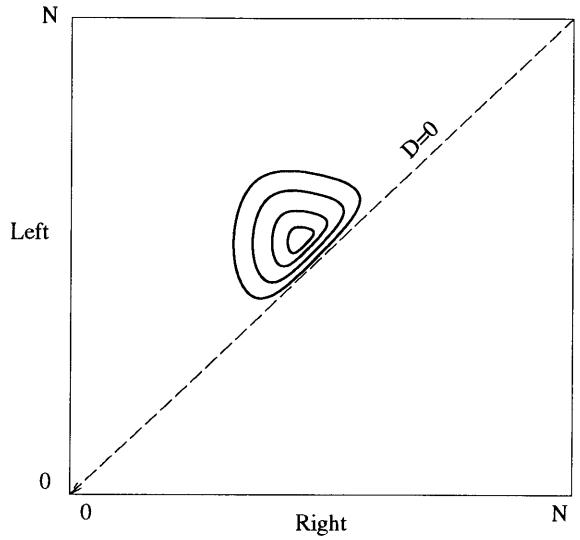

Figure 7-10: Possible Stereo Disparity RF map

7.5.2 Fused Cyclopean Displacement — d_C

Binocular or Cyclopean fused vision is the process whereby two 2D images are combined into one 2D image[42, 43]. Depth is not involved. Indeed the Cyclopean representation would be largely invariant with depth. When one eye is closed, the representation is virtually unchanged. There is certainly little question that humans produce a Cyclopean fused vision representation — this is the "image" we see regardless of which eye is open.

The present model proposes that the fused Cyclopean image is really a fused Cyclopean Displacement representation. The basis for this argument is that, other than stereo Disparity, all other vision tasks should logically rely on a single representation, rather than the two independent representations of the individual views.

The fused Cyclopean Displacement representation RF would be no different than that discussed earlier for the monocular Displacement representation for a single image — with one important exception — it would be binocular. It would accept stimuli from either or both eyes. In fact, it should look just like the stereo Displacement representation D just discussed with one important difference — the RF will be rotated 90^o in the RF plot (see Figure 7-11).

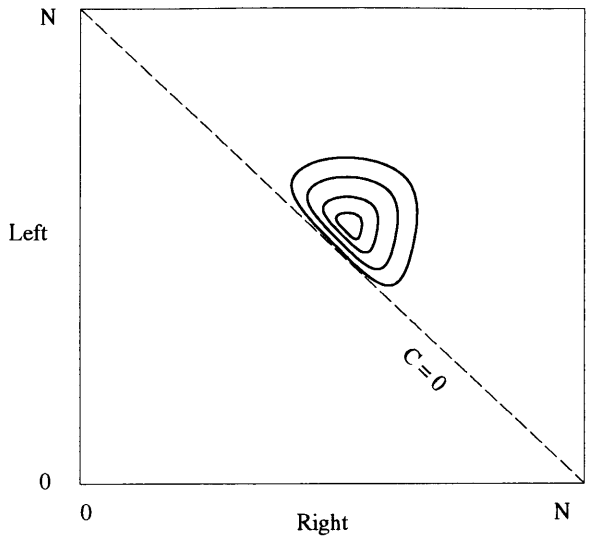

Figure 7-11: Possible Cyclopean RF map

The research indicates that the majority of cells in the striate cortex are binocularly stimulated[67]. Many, if not most, prefer one eye over the other, but this can be understood in the context of the fused Cyclopean weight w_C:

$$w_C = \frac{1}{\sigma_C^2(x,y)} = \frac{1}{\sigma_l^2(x,y)} + \frac{1}{\sigma_r^2(x,y)}.$$

The individual eye variance measures not only measure image properties, such as contrast and focus, but the properties of the individual eye as well. If the eye is badly focused, the other eye will dominate. If the eye has low or no image contrast content, the other eye will again dominate. Ocular dominance is a well known striate cortex property — hence the name [38, 67]. Some effects are long term (such as dominance) and some are short (edge contrast), but the above weight function can easily be cast to incorporate both variance measures.

The tuned excitatory cells reported by Poggio and others may possibly correspond to this RF. They do exhibit the "ocular balance" characteristic where inputs from either eye were capable of activating the cell [67].

7.5.3 Stereo Cyclopean — d_S

The stereo Cyclopean is simply used to sample the stereo Disparity correctly. It has the same weight as the Disparity representation and probably the same representational form (RF) as the fused Cyclopean representation. It has one distinction, however. Like the Displacement representation, it should only be active when both eyes are stimulated

A smaller population of tuned cells, like those postulated for the fused Cyclopean, do display the "ocular unbalance" that would be expected of this representation.

7.5.4 Stereo Disparity and Stereo Cyclopean Weights — W

When two Displacement functions are added or subtracted, their variances add. Thus the inverse of the Stereo weight is the sum of the inverse of the Displacement weights. Since the weights are proportional to the magnitude of the gradient $\mathbf{I}'(x,y)$, the weight takes the form:

$$w_D(x,y) = \frac{\|\mathbf{I}'_l(x,y)\|\|\mathbf{I}'_r(x,y)\|}{\|\mathbf{I}'_l(x,y)\| + \|\mathbf{I}'_r(x,y)\|}$$

There would be no monocular response. The RF would be tuned and centered on the horopter. This representation is a good candidate for the tuned (inhibitory) cell population[67].

Freeman and Ohzawa recently reported finding a cell population that fits this w_D representation precisely. The RF of some of the cells even display a pathological matching of mismatched bars that would be in line with the stereo weight function w_D before the imposition of contrast constraints. This will be discussed more below [24].

7.5.5 Fused Cyclopean Weights — w_C

The inverse variance measure for the Fused Cyclopean representation, which is a weighted average of left and right Displacement representations, is the sum of the component inverse variance measures:

$$w_C(x,y) = w_l(x,y) + w_r(x,y).$$

This quite possibly is a simple cell, since the weights are proportional to the left and right gradient functions. The RF would almost certainly be centered on the horopter line, so it would be a "tuned" response, and since it would accept input from either eye, it would take on the characteristics of the tuned excitatory population[67].

Ohzawa and Freeman recently disclosed such a "simple" cell response with a RF predicted by this model[24].

7.5.6 Constraint Filters

The Displacement/Disparity stereo model begins to impose matching constraints in the form of binary filters. These filters test some signal at every point in the domain and selectively zero the Disparity weight W wherever some match constraint is violated.

Illusion Edges

One such constraint was that false edges should not be matched to real edges. This was accomplished by testing the slope of the Displacement function. False, or "Chevruel" edges are distinguished by negative sloped Disparities ($\nabla \cdot \mathbf{d}(x,y) < 0$). This can be detected also by both the slope of the Laplacian I'' representation and the direction of the Gradient, but this is a nontrivial task and not easily done on the retinal input. Richter and Ullman found that these illusory zero crossings are indeed filtered out of some direction selective cells in the cortex of cats [75, 76, 77].

It was suggested that these cells may be detecting zero-crossings in the Laplacian of Gaussian image $I''(x,y)$. The peak responses do appear to correspond to the zero crossings. On the other hand these cells also appear consistent with the Gradient based Displacement weight function $w_d = \|\mathbf{I}'(x,y)\|$ of this theory, since the gradient is also maximum at the zero crossings of the Laplacian image. Thus the results are consistent with the model of imposing a false edge matching constraint on the gradient based inverse variance weight function as was proposed in the stereo algorithm.

Contrast Matching

Another constraint used in the stereo algorithm is the matching of like contrast signs. This is easily done by insuring that the epipolar weights I'_e are of like sign. Freeman and Ohzawa document the response of two tuned disparity selective cells in the cat visual cortex. In both, a simple tuned response was found when matched contrast bars were presented along the horopter. This is consistent with the "tuned excitatory" cells of Poggio and Fischer [67].

One cell group, however, displayed the decided pathology of producing *two* peak responses when mismatched (light bar / dark bar) stimuli were used. One peak was slightly in front of the horopter position and the other slightly behind. This is what could be expected from the Disparity weight function w_D to such stimuli. In effect, the leading edge of the dark bar matches the trailing edge of the light bar when they are slightly displaced in depth. Neighboring cells however, were found that provided nearly identical responses to the matched inputs but had *no* response to the mismatched bars. This is presumably due to a mechanism that detects mismatched Delta weights, just as the present model corrected for Delta Displacement measures. It would be useful to see if it is possible to explain how such a filter might be realized in the context of the Displacement/Disparity model.

7.6 Summary

The Disparity and Displacement models of early vision computation provide some predictions about what processing steps might be occuring in the first stages in cortical processing. The psychophysical evidence, particularly regarding hyperacuity tasks, as well as the neurophysiological studies of the striate cortex in cats and primates, provide a wealth of evidence which can be tested against the theory proposed in this thesis.

A distinction should be made between evidence which supports the Laplacian and gradient of Gaussian RF representations in the simple cell population. The Displacement model relies on these computations, but any number of other models might also.

On the other hand, the literature appears consistent with both the simple cell predictions of the Laplacian/gradient model as well as the complex cell predictions of the Displacement and the Stereo Disparity models. Given the large body of research devoted to this small area of the cortex, this is a very encourging outcome.

Chapter 8

Conclusion

This thesis has examined two typical computational approaches used for the processing of image data. The continuous domain approaches were found to be rich in scene related information, but they lacked any means to incorporate knowledge of the physics and optics of imaging to constrain the interpretation of image data into 3D scene properties. The symbolic approaches typified by edge-finding algorithms resulted in representations which had the opposite effect. The data is readily incorporated into "top-down" high-level vision algorithms. Yet much of the useful information content relating to the scene is sacrificed in the continuous to symbolic transformation of marking edges. Both of these approaches have strengths which would be useful if they could be incorporated into a single model.

One such model has been developed in this thesis — the Displacement/Disparity model of early vision. It uses two linear image representations, the Laplacian and Gradient of Gaussian convolved images, to generate a Displacement representation which measures local edge feature properties. These three properties are edge distance, orientation, and contrast. Like some continuous domain approaches, such as with Gabor models, the Displacement representation encodes spatial image information. Unlike these models, however, the model is specifically designed to extract individual feature measures. While this is like the symbolic approaches, the Displacement computation

preserves all of the spatial image information related to feature position, orientation, and contrast. Most importantly, useful and powerful low-level representations are computable using simple linear transforms on the Displacement representation.

These transforms produce Disparity representations. Many were discussed in the thesis, such as edge focus, ego motion, Cyclopean fused imagery, and matching. One was developed and tested in detail — the stereo Disparity model. Simple subtraction of Displacement distance measures renders a dense map of stereo estimates. This preserves the "bottom-up" simplicity of the continuous models. Simple criteria based on image information, including other Disparity measures such as focus, can constrain the matching process. This permits incorporation of common-sense and physical knowledge in the continuous domain model. Ultimately, the stereo model renders a sparse yet precise map of the stereo disparity.

This thesis demonstrates that the extremely simple computations involved in the Displacement and Disparity models are capable of preserving the integrity of available spatial and optical properties of the scene features without sacrificing the computational power of the symbolic approaches. Indeed, since so many feature properties are easily measured in this model, there is every reason to prefer the Displacement/Disparity front-end to edge-finders in even the most heavily symbolic vision algorithms.

This model was motivated primarily by the desire to bridge this gap between representational richness and computational utility. It was also motivated by a desire to devise a model consistent with the early visual cortical processes. No claim can be made that the Displacement or Disparity models are precise, or even general, models for cortical function. It does, however, provide some predictions and suggestions about what processes may be involved in the early processing of visual information.

Bibliography

[1] N. Ayache and B. Faverjon, "Efficient registration of stereo images by matching graph descriptions of edge segments," *Int. Journal Computer Vis.*: 107-131 (1987)

[2] H.H. Baker and T.O. Binford "Depth from edge and intensity based stereo" *Int. Joint Conf. A.I.*, 631-636 (1981)

[3] H. Barlow, C. Blakemore, and J. Pettigrew, "The neural mechanism of binocular depth discrimination," *J. Physiol,* **193**: 327-342 (1967)

[4] A. Barr and E. Feigenbaum, *The Handbook of Artificial Intelligence,"* Houristech Press, Stanford, Ca. Vol. 3: 216-223 (1981)

[5] M. Bertero, T. Poggio, and V. Torre, " Ill posed problems in early vision," MIT A.I. Memo 924 (1987)

[6] M. Bertero, T. Poggio, and V. Torre, " Ill posed problems in early vision," Proceedings of the IEEE, **96**, No. 8, 869-889 (1988)

[7] M. Bichsel, "Strategies of robust object recognition of the automatic identification of human faces," Phd Thesis, Eidgenossischen Technischen Hochschule Zurich, Zurich, Germany (1991)

[8] R. Brunelli and T. Poggio, "HyperBF networks for gender classification," *DARPA Image Understanding Workshop Proceedings,* (1992)

[9] R. Brunelli and T. Poggio, "Face recognition: features versus templates," submitted to *IEEE Pattern Analysis and Machine Intelligence,* (Submitted 11/91)

[10] J.F. Canny, "Finding edges and lines in images," Mass. Inst. of Technol., Cambridge, MA, Tech. Rep. TR-720, (1983)

[11] S. Carey and R. Diamond, "From piecemeal to configurational representation of faces," *Science,* **195**, 312-313 (1977)

[12] A. Carlson, *"Communication Systems,"* McGraw Hill, New York, NY, (1975)

[13] T.S. Collett, "Extrapolating and interpolating surfaces in depth," Proc. Royal Soc. Lond. **B** 224, 43-56 (1985)

[14] J. Daugman, "Pattern and motion vision without Laplacian zero crossings," *J. Opt. Soc. Am.* **A**, 5(7):1142-1148 (1988)

[15] R. Deriche, "Using Canny's criteria to derive a recursively implemented optimal edge detector," *IJCV* 167-187 (1987)

[16] R. Duda and D. Hart, *Pattern Classification and Scene Analysis,* John Wiley and Sons Inc. New York, NY (1973)

[17] H. Ellis and A. Young, "Are faces special," *Handbook of research on face processing,* A. Young and H. Ellis Eds. Elsevier Science Publishers, N.Y, N.Y. (1989)

[18] C. Enroth-Cugell and J. Robson, " The contrast sensitivity of retinal ganglion cells of the cat," *J. Physiol. Lond.* 187: 517-522 (1966)

[19] M. Farah, *Visual Agnosia,* MIT Press, Cambridge, Ma. (1990)

[20] M. Fahle and T. Poggio, "Visual hyperacuity: spatiotemporal interpolation in human vision," Proc. R. Soc. Lond., B **213**: 451-477 (1981)

[21] D. Ferster, "A comparison of the binocular depth mechanisms in areas 17 and 18 of the cat visual cortex," *J. Physiol.* **311**: 623-655 (1981)

[22] D. Fleet and A. Jepson, "Computation of component image velocity from local phase information," Int. Jorunal of Comp. Vision, 5:1, 77-104 (1990)

[23] A. Fiorentini, G. Baumgartner, S. Magnussen, P. Shiller, and J. Thomas, "The perception of brightness and darkness," in *Visual Perception,* L. Spillman and J.S. Werner Eds. Academic Press, Chap. 7: 135-137 (1990)

[24] R. Freeman and I. Ohzawa, "On the neurophysiological organization of binocular vision," *Vision Res.,* **30**, No. 11: 1661-1676 (1990)

[25] R. Gonzalez and P. Wintz, *Digital Image Processing,* Addison Wesley Publishing Co. Inc., Reading, Mass. (1977)

[26] P. Gouras and J. Kruger, "Responses of cells in foveal visual cortex of the monkey to pure color contrast," *J. Neurophysiol.*, **42**, No 3.: 850-860 (1979)

[27] W.E.L. Grimson, *From Images to Surfaces: A Computational Study of the Human Early Visual System,* The MIT Press, Cambridge, Ma. (1981)

[28] W.E.L. Grimson, "A computer Implementation of a theory of Human Stereo Vision," Phil. Trans. Royal Society London, Vo. B 292, 217-253 (1981)

[29] W.E.L. Grimson, "Computational experiments with a feature based stereo algorithm," IEEE Trans. Pattern Anal. Mach. Intell. PAMI-7, No 1: 17-34 (1985)

[30] W.E.L. Grimson and E. Hildreth, " Comments on digital step edges from zero crossings of second directional derivatives," *IEEE Trans. PAMI,* **7** (1): 121-129 (1985)

[31] C. Gross, R. Desimone, T. Allbright, and E. Schwartz, "Inferior temporal cortex and pattern recognition," *Pattern Recognition Mechanisms,* 179-201 (1986)

[32] N. Grzywacz and T. Poggio, "Computation of Motion by Real Neurons," In: *An introduction to neural and electronic networks*, Eds. S. Zornetzer, J. Davis, and C. Lau, Academic Press, Orlando, Fl., (in press)

[33] R. Hamming, "Integrals and differential equations," In: *Numerical Methods for Scientists and Engineers*, McGraw Hill, New York, NY, Chap 36: 574-591 (1973)

[34] E. Hildreth, "The detection of intensity changes by computer and biological vision systems," *Computer Vision and Graphics Image Processing,* **22**: 1-27 (1983)

[35] E. Hildreth, "Edge detection," MIT A.I. Memo No. 858 (1985)

[36] B.K.P. Horn, *Robot Vision*, The MIT Press, Cambridge, Ma. (1986)

[37] B.K.P. Horn, Personal communication, (1992)

[38] D.H. Hubel *Eye, brain, and vision,* Scientific American Library Series No. 22, New York, N.Y. (1988)

[39] D. Hubel and T. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *J. Physiol.,* **160**: 106-154 (1962)

[40] D. Hubel and T. Wiesel, "Stereoscopic vision in macaque monkey," *Nature*, **225**: 41-42 (1970)

[41] A. Jepson and M. Jenkin, "The fast computation of disparity differences," *Proc. Conf. of Computer Vision and Pattern Recognition 1989,* San Diego, Ca. 398-403 (1989)

[42] B. Julesz, *Foundations of cyclopean perception*, Chicago University Press, Chicago, Ill. (1971)

[43] B. Julesz, "Stereoscopic vision," *Vision Research*,Vol. 26, No. 9, 1601-1612 (1986)

[44] B. Julesz and J. Miller, "Independent spatial-frequency-tuned channels in binocular fusion and rivalry," *Perception,* 4:125-143 (1975)

[45] C. Koch and T. Poggio, "Biophysics of computation: neurons, synapses and membranes," MIT A.I. Memo 795, (1984)

[46] R. Mansfield, "Neural basis of orientation perception in primate vision," *Science,* **186**: 1133-1135 (1974)

[47] R. Mansfield, "Role of the striate cortex in pattern perception in primates," in *Analysis of visual behavior*, D. Ingle, M. Goode, R. Mansfield eds., MIT Press, Cambridge, Ma. 443-482 (1982)

[48] R. Mansfield and S. Ronner, "Orientation anisotropy in monkey visual cortex," *Brain Res.*, **149**: 229-234 (1978)

[49] D. Marr, *Vision,* San Francisco, CA: Freeman (1982)

[50] D. Marr and E. Hildreth,"Theory of edge detection," *Proc. Royal Soc. London B 207*: 187-217 (1980)

[51] D. Marr, T. Poggio, and E. Hildreth, "Smallest channel in early human vision," *J. Opt. Soc. Am.*, **70**, No. 7: 868-870 (1980)

[52] D. Marr and T. Poggio, "Cooperative computation of stereo disparity," *Science,* **194**: 283-287 (1976)

[53] D. Marr and T. Poggio, "A computational theory of human stereo vision," *Proc. R. Soc. Lond.* B. **204**: 301-328 (1979)

[54] D. Marr and S. Ullman, "Directional selectivity and its use in early vision processing," *Proc. Soc. Lond.* **B** 204: 301-328 (1981)

[55] R. Maske, S. Yamane, and P. Bishop, "Simple and B-cells in cat striate cortex. Complementarity of responses to moving light and dark bars," *J. Neurophysiology,* **53**, No. 3: 670-685 (1985)

[56] J. Mayhew and J. Frisby, "Rivalrous texture stereograms," *Nature,* **264**: 53-56 (1976)

[57] J. Mayhew and J. Frisby, "Psychophysical and computational studies toward a theory of human stereopsis," *Artificial Intell.,* **17**, 349-385 (1981)

[58] J.L. McClelland, D.E. Rummelhart, and the PDP Research Group Eds., *Parallel distributed processing,* MIT press, Cambridge, Ma. (1986)

[59] G. Medoni and R. Neviata, "Segment based stereo matching," *Computer Vision, Graphics, and Image Processing*, **31**: 2-18 (1985)

[60] J. Movshon, I. Thompson, and D. Tolhurst, "Receptive field organization of complex cells in the cat's striate cortex," *J. Physiol.*, **283**: 79-99 (1978)

[61] J. Movshon, E. Adelson, M. Gizzi, and W. ¯Newsome, "The analysis of Visual Motion" in *Pattern Recognition Mechanisms* C. Chagas, R Gattass amd C. Grfoss Eds., Pontifica Academia Scientarvm, Rome. (1983)

[62] Y. Ohta and T. Kanade "Stereo by intra- and inter-scanline search using dynamic programming," *IEEE Trans. Pattern Anal. Mach. Intelligence,* PAMI-7, No. 2: 139-154 (1985)

[63] I. Ohzawa, G.C. DeAngelis and R.D. Freeman, "Stereoscopic depth discrimination in the visual cortex: neurons ideally suited as disparity detectors," *Science,* **249**: 1037-1041 (1990)

[64] S.B. Pollard, J.E.Mayhew and J.P. Frisby, "PMF: A stereo correspondence algorithm using a disparity gradient limit" *Perception*, **14**: 449-470 (1985)

[65] D. Pollen and S. Ronner, "Phase relationships between adjacent simple cells in the visual cortex," *Science,* **212**: 1409-1411 (1981)

[66] G. Poggio, "Processing of stereoscopic information in primate visual cortex," in *Dynamic Aspects of Neocortical Function,* G. Edelman, W. Gall and W. Cowan Eds. John Wiley and Sons, N.Y. NY. 613-635 (1984)

[67] G. Poggio and B. Fischer, "Binocular interaction and depth sensitivity in striate and prestriate cortex of behaving rhesus monkey," *J. Neurophysiology,* **40**, No. 6: 1392-1405 (1977)

[68] G. Poggio and T. Poggio, "The analysis of stereopsis," *Ann. Rev. Neurosci.*, **7**: 379-412 (1984)

[69] G. Poggio, B. Motter, S. Squatrito, and Y. Trotter, "Responses of neurons in visual cortex (V1 and V2) of the alert macaque to dynamic random-dot stereograms," *Vision Res.,* **25**, No. 3: 397-406 (1985)

[70] G. Poggio, F. Gonzalez, and F. Krause, " Stereoscopic mechanisms in monkey visual cortex, correlation and stereo selectivity", *J. Neurosci.*, **8(12)**: 4531-4550 (1988)

[71] T. Poggio and V. Torre (Eds.), "A theory of synaptic interactions," in *Theoretical Approaches In Neurobiology*, MIT Press, Cambridge, Ma.: 28-38 (1980)

[72] T. Poggio, M. Fahle and S. Edelman, "Synthesis of visual modules from examples learning hyperacuity," MIT Artificial Intelligence Laboratory Memo No. 1271 (1991)

[73] W. Pratt, *Digital Image Processing,* John Wiley and Sons Inc., New Yror, NY (1978)

[74] D. Regan, J.P. Frisby, G.F. Poggio, C.M. Schor and C.W.Tyler, "The perception of stereodepth and stereomotion," in *Visual perception,* L. Spillman and J.S. Werner Eds. Academic Press, Chap. 12 317-347 (1990)

[75] J. Richter, S. Ullman, "A model for the spatio-temporal organization of X and Y - type ganglion cells in the primate retina," MIT A.I. Memo No. 573 (1980)

[76] J. Richter, S. Ullman, "A model for the temporal organization of X- and Y-type receptive fields in the primate retina," *Biol. Cybern.*, **43**: 127-145 (1982)

[77] J. Richter and S. Ullman, "Non-linearities in cortical simple cells and possible detection of zero crossings," *Biol. Cybern.*, **53**: 195-202 (1986)

[78] D. Regan, J.P. Frisby, G.F. Poggio, C.M. Schor and C.W.Tyler, "The perception of stereodepth and stereomotion," in *Visual perception,* L. Spillman and J.S. Werner Eds. Academic Press, Chap. 12 317-347 (1990)

[79] I. Rock, "The perception of disoriented figures," in *Image, Object, and Illusion,* Readings from *Scientific American,* San Francisco, CA. (1974)

[80] T. Sanger, "Stereo disparity computation using Gabor filters," *Biological Cybernetics, 59*: 405-418 (1988)

[81] P. Schiller, B. Finlay, and S. Volman, "Quantitative properties of single cell properties in monkey striate cortex I. Spatio-temporal organization of receptive fields," *J. Neurophysiology,* **39**: 1288-1319 (1976)

[82] P. Schiller — Personal communication (1992)

[83] R. Szulborski and L. Palmer, "The two-dimensional spatial structure of nonlinear subunits in the receptive fields of complex cells," *Vision Res.*, **30**, No. 2: 249-254 (1990)

[84] V. Torre and T. Poggio, "On edge detection," IEEE Trans. PAMI-8, No. 2: 147-163 (1986)

[85] S. Ullman, "An approach to object recognition: aligning pictorial description," *MIT Art. Intell. Memo* No. 931 (1986)

[86] T. Vidyasagar and G. Henry, " Relationship between preferred orientation and ordinal position in neurones of cat striate cortex," *Visual Neuroscience*, **5**: 565-569 (1990)

[87] R. Watt and M. Morgan, "Mechanisms responsible for the assessment of visual location: theory and evidence," *Vision Res.*, **23**: 97-109 (1983)

[88] R. Watt and M. Morgan, "The recognition and representation of edge blur: evidence for spatial primitives in human vision," *Vision Res.*, **23**, No 12: 1465-1477 (1983)

[89] Westheimer G. "Spatial sense of the eye," Invest. Optham. Vis. Sci. Vol 18, 893-912 (1979)

[90] H.R. Wilson, "Responses of spatial mechanisms can explain hyperacuity," *Vision Res.,* **26**, No. 3: 453-469 (1986)

[91] H.R. Wilson and J.R. Bergen, "A four mechanism model for threshold spatial vision," *Vision Res.,* **19**: 19-32 (1979)

[92] H.R. Wilson, D.K. McFarlane and G.C. Phillips, "Spatial frequency tuning of orientation selective units estimated by oblique masking," *Vision Res.,* **23**, No. 9: 873-883 (1983)

[93] H.R. Wilson and D.J. Gelb "Modified line element theory for spatial frequency and width discrimination," *J. Opt. Soc. Am.* **A**, Vol. 1, No. 1: 124-131 (1984)

[94] H. Wilson, D. Levi, L. Maffet, J. Rovamo and R. DeValois, "The perception of form," in *Visual Perception,* L. Spillman and J.S. Werner Eds. Academic Press, Chap. 10: 231-272 (1990)

Appendix A

Optimal Filter Design for Gaussian Signals

This appendix covers the detailed derivation for the optimal filter design of Chapter 4. In that chapter, the objective was to design an optimal filter for restoring a Gaussian form 1D image signal which is corrupted by additive uncorrelated noise after being processed by a derivative operator.

Figure A-1 shows, in schematic form, the signal processing steps involved. The Gaussian signal $b(x)$ is processed by a differentiation operator $H = j\omega$. This signal is corrupted by additive noise $n_i(x)$ producing the input signal $i(x)$ to the filter to be designed in this section — H'. The goal is to produce an output signal $o(x)$ which corresponds in a best least-squares sense to the input signal $d(x) = b(x)$, or minimizes the RMS energy of the signal $\epsilon(x)$.

The approach used in this analysis is identical to that described by Berthold Horn in chapter six of his book *Robot Vision*, so the symbology and form of the analysis are chosen to match that used by him [36].

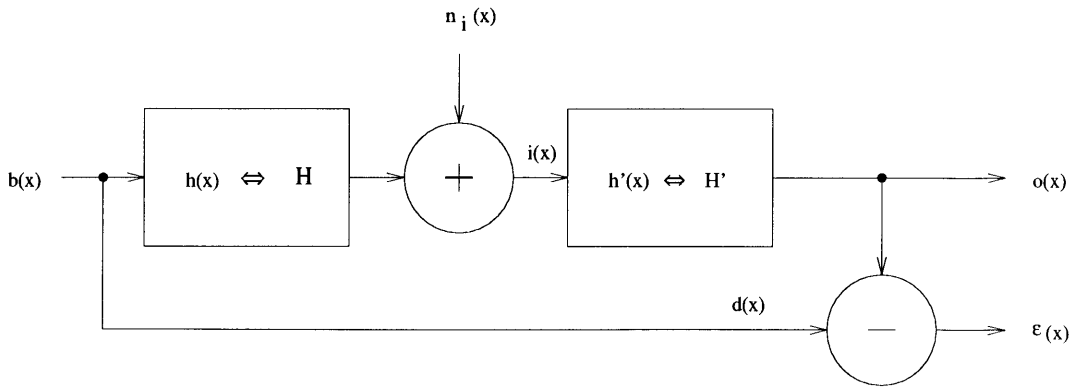

Figure A-1: 1D Optimal Filter Model

A.0.1 The Optimal Filter

In the Displacement model, the input signal $b(x)$ is of Gaussian form:

$$b(x) = \frac{1}{\sqrt{2\pi}\sigma_b} e^{\frac{-(x-x_o)^2}{2\sigma_b^2}}$$

To design the optimal filter, it is important to determine the power spectra of the signal $b(x)$. The power spectra is a function which represents the energy density of the signal as a function of spatial frequency. To simplify the analysis, $x_o = 0$ is assumed throughout. Since the power spectra is insensitive to phase terms, this is a reasonable simplification. Since the Fourier transform of $b(x)$ is

$$B(\omega) = \mathcal{F}[b(x)] = e^{-\frac{\omega^2\sigma_b^2}{2}},$$

the signal has the power spectra $\Phi_{bb} = B^*B(\omega) = e^{-\omega^2\sigma_b^2}$.

The noise is treated as uncorrelated and zero mean with a normal probability distribution:

$$\forall x \quad p(n_i) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{\frac{-(n_i)^2}{2\sigma_n^2}}$$

The noise signal has the power spectra $\Phi_{nn} = \sigma_n^2$.

A measure of the signal-to-noise (SNR) ratio of the signal $i(x)$ can be defined as ratio of the peak amplitude of Gaussian signal to the RMS noise:

$$\rho = \frac{\max\{b(x)\}}{\sigma_n} = (\sqrt{2\pi}\sigma_b\sigma_n)^{-1}$$

so another way of expressing the noise power spectra is in terms of the Signal to Noise Ratio:

$$\Phi_{nn} = (2\pi\sigma_b^2\rho^2)^{-1}$$

When the noise is uncorrelated to the signal, it has been shown that an optimal filter for restoration of the signal $b(x)$ will have the following transfer function [1]:

$$H' = \frac{\Phi_{id}}{\Phi_{ii}} = \frac{H\Phi_{bb}}{H^*H\Phi_{bb} + \Phi_{nn}}$$

so the following transfer function results for the case of Gaussian signals and $H = j\omega$:

$$H'(\omega) = \frac{-j\omega}{\omega^2 - (2\pi\sigma_b^2\rho^2)^{-1}e^{\omega^2\sigma_b^2}}$$

This rather complex looking transfer function is shown in a log-log plot of Figure A-2. A 400 pixel image is assumed so the filter sizes — σ_b — can be estimated using a reasonable nominal image width. The abscissa is thus in log cycles per image (0 means 1 cycle per image, 1 means 10 cycles, and so on). The ordinate is in decibels (db). The 0 db point indicates unity gain.

Four filter sizes are plotted (σ_b = 16, 8, 4, and 2). Note that the σ_b plots simply shift in the frequency domain while the relative shape of the response is fixed with varying smoothing widths. This response "shape" of the log-log plots takes on a

[1] see page 136 of [36]

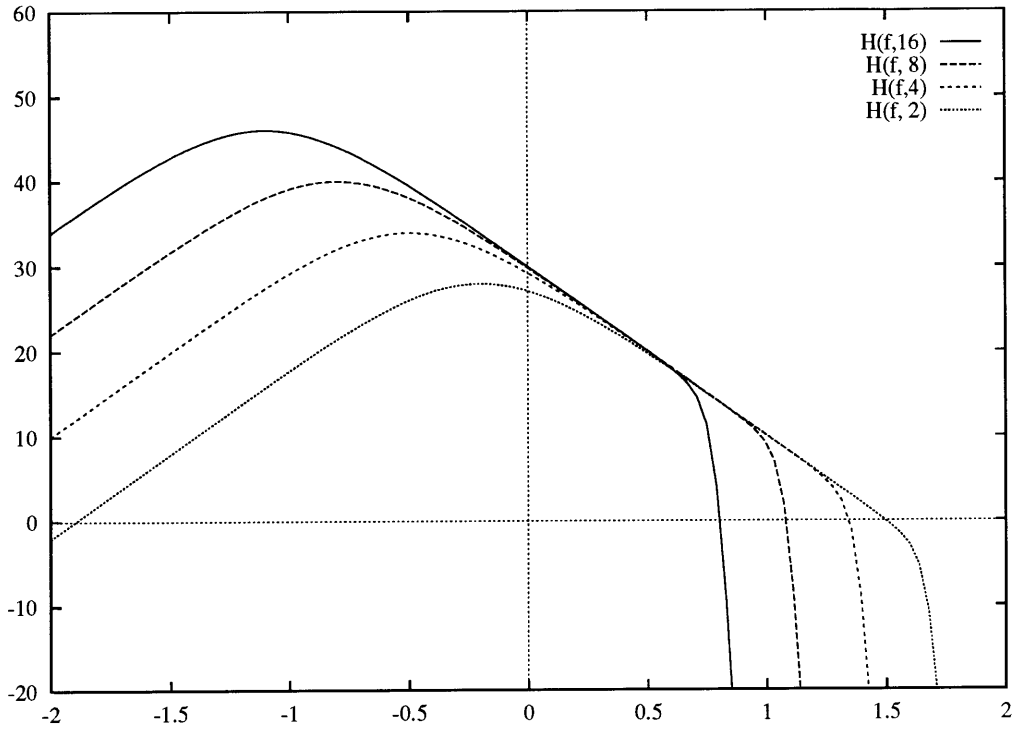

Figure A-2: 1D Optimal Filter Response — Four σ_b sizes are shown (16, 8, 4 and 2) The vertical axis is db gain. The horizontal axis is in log cycles per image (0 = 1 cycle per image)

decidedly piecewise linear appearance typical of filters which can be modeled by lumped first order filters combined in simple networks. There are three such linear gain regions in these plots.

In the central region — for the filter size σ_b of 16, this is the region between 0.1 cycle per image width and approximately 5.0 cycles per image (abscissa reading between 0.5 and 1.0) — the response drops off by a factor of two for every doubling (octave) of spatial frequency. This is characteristic of an integrator, so this region of the response corresponds to the $1/j\omega$ function of the naive design.

The low frequency region — for the same filter size σ_b of 16, this is the region below 0.1 cycle per image width — there is just the opposite response. The gain *increases* at 6db (double amplitude) per octave. This is characteristic of a differentiation function

such as the $j\omega$ operators in the retinal model.

The SNR figure ρ used was 10, a rather large noise content typical of tests run in other parts of this research. When more noise is added, the peak on the curves shifts. Thus the region of integration shrinks with increasing noise, while the region with the opposite slope grows. A reasonably good estimate of the injected noise amplitude is very important in choosing this cutoff frequency. By inspection of the plots for various filter σ_b sizes and SNR ρ values, the peak filter response ω_c is inversely proportional to both of these variables:

$$\omega_c = \frac{c}{\rho\sigma_b}. \tag{A.1}$$

The constant c can also be roughly estimated from the response curves to be approximately $1/2\pi$. This value was used in the implementations of the filter designs.

The third region is characterized by the sharp, indeed almost vertical, drops in gain at the high-frequency end of the response curves. This precipitous drop is a consequence of the Gaussian signal form and has nothing whatever to do with the added noise amplitude. This suggests that above a specific spatial frequency, there is virtually no need for the filter to pass any signal (or noise) component.

Therefore, the ideal filter would have an integration region (for signal reconstruction), a low-frequency roll-off region (for noise suppression), and possibly a high-frequency cutoff region determined by the Gaussian signal width.

A.0.2 Practical Filter Designs

In practice, the above filter is a bit difficult to design. The very sharp cutoff region insures that the phenomenon known as reciprocal spreading will be an important issue. What this means is that by bounding the filter response in the frequency domain, there will be a corresponding spreading, or unbounded response, in the spatial domain of the corresponding impulse response $h'(x)$.

The purpose of this analysis, however, was less to devise a matched filter than to produce a model on which a well posed — stable — approximation to the integrator could be designed. For this purpose, a practical approximation would be to concentrate on the first two regions, namely the regions below where the high frequency cutoff occurs.

The following filter provides an excellent approximation to the above:

$$H'(\omega) = \frac{\omega}{(\omega + \omega_c)^2}$$

since it will have the same peak response value ω_c, and the response slopes on either side are ± 6 db per octave, just as in the optimal design. This filter transfer function is shown in Figure A-3. The only significant difference between this design and the optimal model is in the absence of the high-frequency cutoff region in the modified design.

The impulse responses of this filter $h'(t)$ are easily calculated by taking the inverse Fourier transform of the $H'(\omega)$ transfer function:

$$h'(x) = (1 - \omega_c x)e^{-\omega_c x}$$

Since the simplest way of implementing such a transform in a practical vision application (or, perhaps, in cortex) would be to convolve the incoming signal $i(x) = I''(x)$ with the filter impulse response $h'(x)$, this response is of interest. It is plotted for five filter sizes in Figure A-4.

One of the observations which can be made from this data is that as the size of the filter σ_b increases, the filter behaves more and more like an ideal integrator. This means that the impulse response is more like a unit step function. In practice, this would require a convolution with a spatially unbounded mask. This is impractical. It is also unnecessary. A local filter which does not use convolution produces good results using only immediate neighbors for support. It preserves much of the noise

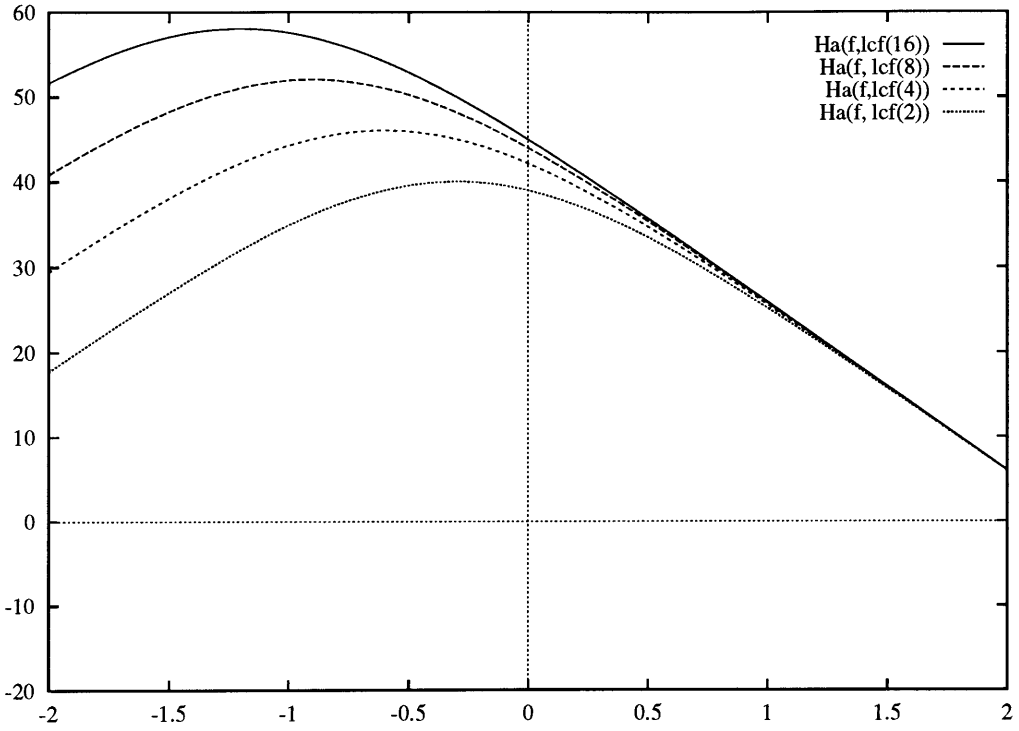

Figure A-3: Response of Filter Approximation — Again, four σ_b sizes are shown (16, 8, 4 and 2) The vertical axis is db gain. The horizontal axis is in log cycles per image (0 = 1 cycle per image)

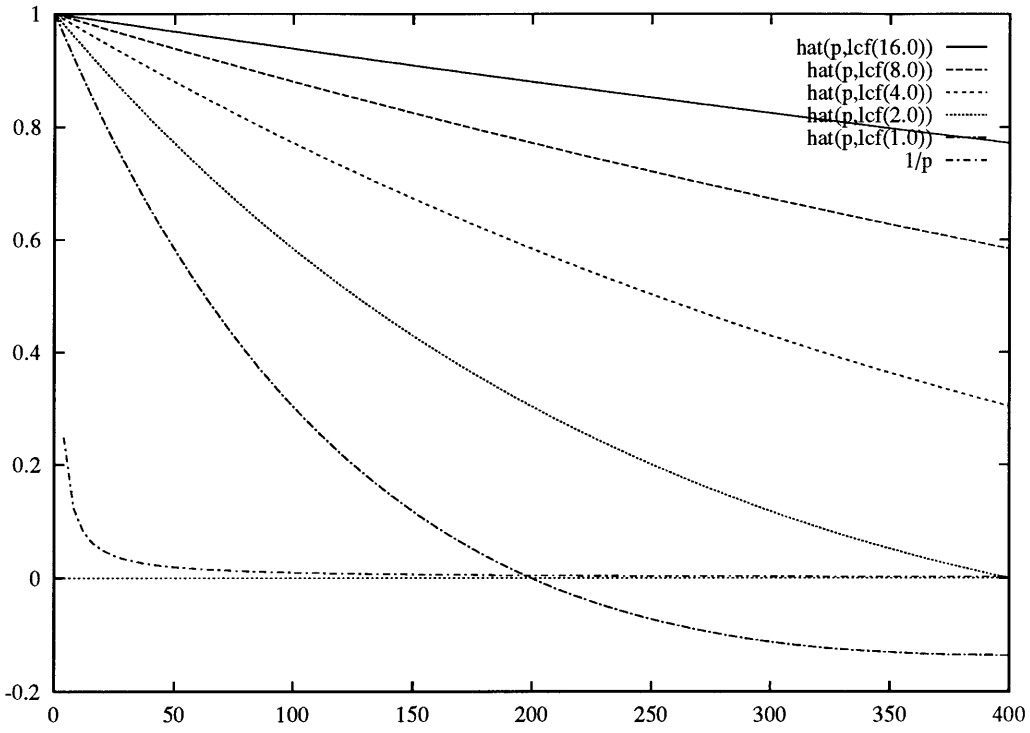

Figure A-4: Impulse Response of Filter Approximation — The four σ_b sizes are shown (16, 8, 4 and 2) The vertical axis is db gain. The horizontal axis is in log cycles per image (0 = 1 cycle per image)

suppressing properties of the above filters without being unstable.

A.0.3 Band Limited Integration

Instead of using convolution to produce the desired H' transform, the above results can be used to design an even simpler, although less optimal, filter — the low-pass integrator.

By selecting the same cutoff frequencies ω_c as the above analysis indicates, but designing a simpler filter, the band-limited integrator or simple low-pass filter, the integration can be rendered stable, noise suppressive, yet preserve the signal integrity.

The transfer function is extremely simple:

$$H'(\omega) = \frac{1}{\omega + \omega_c}$$

and while the impulse response is exponential (and thereby spatially unbounded), it is easily implemented algorithmically using only neighboring samples:

$$o_i \;=\; \gamma o_{i-1} + \frac{i_i + i_{i-1}}{2}$$

The averaging of neighboring inputs insures proper input/output registration by using the "trapezoid rule" of integration [33].

The impulse response for this filter is easily seen to be exponential, i.e. if the input is the Dirac delta function:

$$i_i = \begin{cases} 1 & \text{if } i = 0 \\ 0 & \text{otherwise} \end{cases}$$

then (ignoring the effects of the trapezoidal rule implementation) the impulse response to this will be

$$o_i = \gamma^i.$$

This would correspond, in the case of a continuous first-order lag (low pass) filter, to an impulse response of $o(i) = e^{\frac{-i\omega_c}{2\pi}}$. The appropriate selection for the discrete lag γ given the previous ω_c selections based on the optimal filter response (Equation A.1) is

$$\gamma = e^{-\frac{\omega_c}{2\pi}}.$$

Once again, the results for the optimal design can be used to determine the ω_c values from the noise figure ρ and smoothing filter size σ_b:

$$\gamma \approx e^{-\frac{1}{(2\pi)^2\rho\sigma_b}}.$$

When $\gamma = 1.0$, this is a pure integration ($\omega_c = 0$). When $\gamma < 1$, however, frequency components below ω_c are not integrated. This renders the filter stable. If the same choices are made for ω_c as in the optimal filter, this will insure minimal signal degradation and a stable integration step.

Therefore, the optimal filter can be characterized as an integrator with a low-frequency roll-off region and a high-frequency cutoff point. Practical implementations can be realized by relaxing or eliminating the cut-off region. The above first order approximation replaces the 6 db *gain* per octave region below the integration region of the optimal filter with a flat response. This will permit more low frequency noise contamination than the optimal model, but has the clear advantage of ease of implementation, locality, and speed.

The optimal model is invaluable in determining the other designs' cutoff frequency ω_c in order to minimize noise content while preserving stability without degrading the signal form.

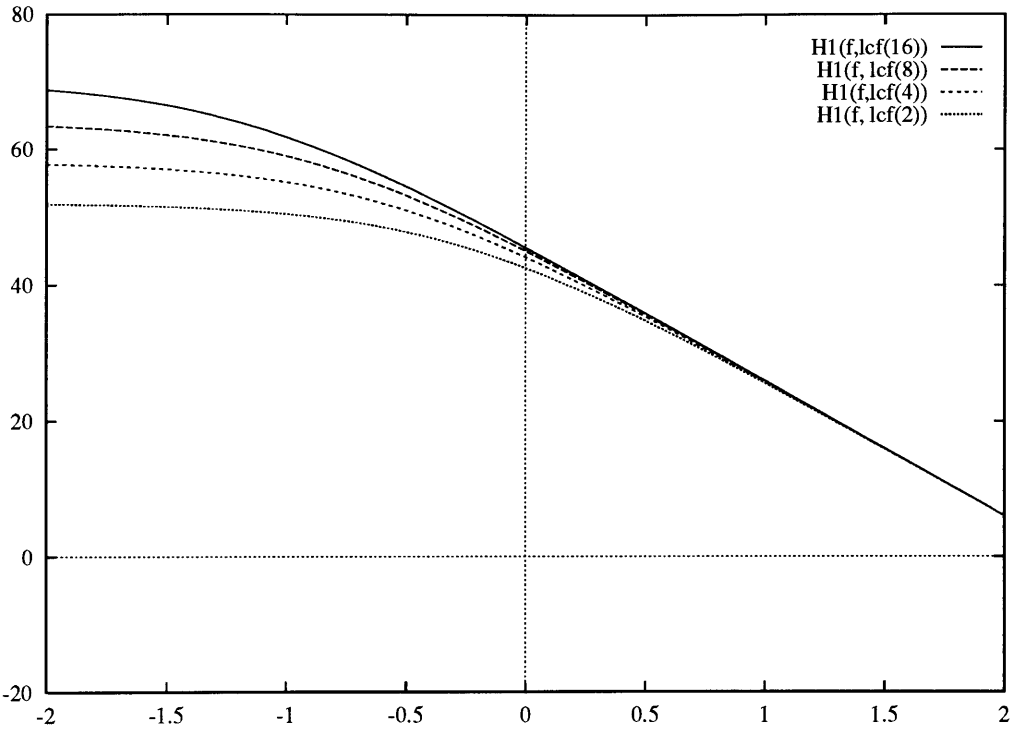

Figure A-5: Response of the lag filter — Again, four σ_b sizes are shown (16, 8, 4 and 2) The vertical axis is db gain. The horizontal axis is in log cycles per image (0 = 1 cycle per image)

Appendix B

Least Squares Analysis on Gaussian Signals

This Appendix deals with arriving at best least squares estimates of signal parameters given noisy 1D and 2D signals with Gaussian forms. The results of this Appendix are used in a variety of points in the thesis ranging from model and filter design approaches and maximum likelihood estimation of Displacement functions to 2D noise model analysis.

The 1D Gaussian signal $I'(x)$, which is one of the two representations used in the 1D Displacement model is analyzed first. It is useful to not only determine the signal location x_o but also the width σ_b, and the contrast α as well. The σ_b measure is helpful in estimating discrete models of variance while α provides a feature contrast measure. The first section deals with these derivations.

The second section derives the more general 2D edge model $\mathbf{I}'(x, y)$ position measure r_o.

B.1 1D Model

B.1.1 Estimation of x_o

The Gaussian signal underlying $I'(x)$is $g(x)$. $I'(x)$ is based on the edge position function $d_o(x) = x - x_o$, the signed edge contrast α and added uncorrelated noise $n_i(x)$:

$$\begin{aligned} g(x) &= \frac{1}{\sqrt{2\pi}\sigma_b} e^{-\frac{x^2}{2\sigma_b^2}} \\ I'(x) &= \alpha g(d_o(x)) + n_i(x) \end{aligned}$$

The indefinite integral notation used throughout indicates integration over infinite bounds.

The least squares solution for x_o given $I'(x)$ is based on minimizing the integral of the squared error term $\epsilon(x)$ with respect to x_o :

$$\begin{aligned} \epsilon(x) &\equiv I'(x) - \alpha g(d_o(x)) \\ \frac{\partial}{\partial x_o} \int \epsilon^2(x) dx &= 0 \\ -2\alpha \int \epsilon(x) \frac{\partial}{\partial x_o} g(d_o(x)) dx &= 0. \end{aligned}$$

Solving for $\frac{\partial}{\partial x_o} g(d_o(x))$ is direct:

$$\frac{\partial}{\partial x_o} g(d_o(x)) = \frac{d_o(x)}{\sigma_b^2} g(d_o(x)) = -g'(d_o(x)).$$

Thus the error integral becomes:

$$\begin{aligned} \int (I'(x) - \alpha g(d_o(x))) g'(d_o(x)) dx &= 0 \\ \int I'(x) g'(d_o(x)) dx &= \alpha \int g'(d_o(x)) g(d_o(x)) dx. \end{aligned}$$

The right hand side involves the integral of an even function times an odd function when a change of variable ($\eta = x - x_o$) is used instead of the dummy variable x. Such indefinite integrals evaluate to zero. Using this simplification, a useful relation is produced:

$$\int I'(x)g'(d_o(x))dx = 0. \tag{B.1}$$

When $g'(d_o(x))$ is expanded, the solution for x_o can be arrived at:

$$\begin{aligned} \int \left(-\frac{x - x_o}{\sigma_b^2}\right) I'(x)g(d_o(x))dx &= 0 \\ \int xI'(x)g(d_o(x))dx &= x_o \int I'(x)g(d_o(x))dx. \end{aligned}$$

Assuming $\int I'(x)g(d_o(x))dx \neq 0$ — i.e. the contrast is non-zero — the solution is

$$x_o = \frac{\int xI'(x)g(d_o(x))dx}{\int I'(x)g(d_o(x))dx}. \tag{B.2}$$

Thus the best estimate of x_o, is the first moment of $I'(x)g(d_o(x))$. Similar analysis shows that the best least squares estimate of contrast α is simply the zero moment (average) of the same function.

B.1.2 Estimation of α

Following the same approach as was used on x_o we minimize the square error for α.

$$\begin{aligned} \frac{d}{d\alpha}\int \epsilon^2(x)dx = 0 &= -2\int \epsilon(x)g(d_o(x))dx \\ 0 &= \int (I'(x) - \alpha g(d_o(x)))g(d_o(x))dx \\ \int I'(x)g(d_o(x))dx &= \alpha \int g^2(d_o(x))dx \\ &= \alpha \int g^2(x)dx \end{aligned}$$

The integral on the right hand side can be solved analytically (or from tables) to be a constant:

$$\begin{aligned} \int I'(x)g(d_o(x))dx &= \alpha\sqrt{\pi}\sigma_b \\ \alpha &= \frac{1}{\sqrt{\pi}\sigma_b}\int I'(x)g(d_o(x))dx \end{aligned} \tag{B.3}$$

B.1.3 Estimation of σ_b

We also minimize $\int \epsilon^2(x)dx$ over σ_b:

$$\begin{aligned} \frac{d}{d\sigma_b}\int \epsilon^2(x)dx &= 0 \\ 0 &= -2\alpha\int \epsilon(x)\frac{d}{d\sigma_b}g(d_o(x))dx \\ \frac{dg(d_o(x))}{d\sigma_b} &= g(d_o(x))\frac{d}{d\sigma_b}\left[\frac{-d_o(x)^2}{2\sigma_b^2}\right] = \frac{d_o(x)^2}{\sigma_b^3}g(d_o(x)) \\ 0 &= \int (I'(x) - \alpha g(d_o(x)))\frac{d_o^2(x)}{\sigma_b^3}g(d_o(x))dx \end{aligned}$$

$$\begin{aligned} \frac{1}{\sigma_b^3}\int I'(x)d_o^2(x)g(d_o(x))dx &= \alpha\sigma_b\int\left[-\frac{d_o(x)}{\sigma_b^2}g(d_o(x))\right]^2 dx \\ &= \alpha\sigma_b\int g'^2(d_o(x))dx \end{aligned}$$

As before, the right side integral can be solved analytically or from tables:

$$\begin{aligned} \frac{1}{\sigma_b^3}\int I'(x)d_o^2(x)g(d_o(x))dx &= \alpha\sigma_b\left(\frac{\sqrt{\pi}}{2\sigma_b}\right) \\ \frac{1}{\sigma_b^3}\int I'(x)(x^2 - 2xx_o + x_o^2)g(d_o(x))dx &= \frac{\alpha\sqrt{\pi}}{2} \\ \int x^2 I'(x)g(d_o(x))dx - 2x_o\int xI'(x)g(d_o(x))dx + x_o^2\int I'(x)g(d_o(x))dx &= \frac{\alpha\sqrt{\pi}\sigma_b^3}{2}. \end{aligned} \tag{B.4}$$

At this point it is useful to introduce three integrals:

$$\begin{aligned} M_0 &= \int I'(x)g(d_o(x))dx \\ M_1 &= \int xI'(x)g(d_o(x))dx \\ M_2 &= \int x^2 I'(x)g(d_o(x))dx \end{aligned}$$

Using these, the expressions can be stated more compactly. Equation B.4 becomes

$$\frac{\alpha\sigma_b^3\sqrt{\pi}}{2} = M_2 - 2x_oM_1 + x_o^2M_0.$$

Since $M_1 = x_oM_o$ (from Equation B.2)

$$\begin{aligned} \frac{\alpha\sigma_b^3\sqrt{\pi}}{2} &= M_2 - 2x_o^2M_0 + x_o^2M_0. \\ &= M_2 - x_0^2M_0 \\ &= M_2 - \frac{M_1^2}{M_0} \end{aligned}$$

Since $\alpha\sqrt{\pi}\sigma_b = M_0$ (Equation B.3)

$$\frac{\sigma_b^2}{2} = \frac{M_2}{M_0} - \left(\frac{M_1}{M_0}\right)^2$$

and finally our result...

$$\begin{aligned} \sigma_b^2 &= 2\left(\frac{M_2}{M_0} - x_o^2\right) \\ &= 2\left(\frac{\int x^2 I'(x)g(d_o(x))dx}{\int I'(x)g(d_o(x))dx} - x_o\right). \end{aligned} \tag{B.5}$$

B.2 2D Gaussian Case — Estimation of r_o

When 2D models are considered, the gradient of Gaussian convolved edge image has a 1D Gaussian cross-section as derived in the 2D Model Chapter (5). The 1D Gaussian function $g(x)$ composed with the edge function $d_o(x,y) = x\sin\theta + y\cos\theta - r_o$, the

edge contrast constant α, and orientation normal unit vector $\hat{\mathbf{\Theta}} = \sin\theta\hat{\mathbf{i}} + \cos\theta\hat{\mathbf{j}}$ are all that is needed to represent the noise-free gradient image (see Equation 5.4):

$$\mathbf{G}(x,y) = g(d_o(x,y))\hat{\mathbf{\Theta}}.$$

The actual image is corrupted by additive uncorrelated noise

$$\mathbf{I}'(x,y) = \alpha\mathbf{G}(x,y) + \mathbf{n}_i(x,y).$$

So the least squares problem is to find the best estimate for the edge distance r_o given this noisy signal and it's functional form

$$\begin{aligned} \epsilon^2(x,y) &= \|\mathbf{I}'(x,y) - \alpha\mathbf{G}(x,y)\|^2 \\ \frac{\partial}{\partial r_o}\int\int_R \epsilon^2(x,y)dR &= 0 \\ -2\int\int_R \frac{\partial\epsilon(x,y)}{\partial r_o}\epsilon(x,y)dR &= 0. \end{aligned}$$

In least-squares analysis, the integrals are unbounded, so R is the full 2D domain. The partial derivative $\frac{\partial\epsilon(x,y)}{\partial r_o}$ is rather unwieldy

$$\frac{\partial\epsilon(x,y)}{\partial r_o} = \frac{-\alpha g'(d_o(x,y))\left[(I'_x(x,y) - \alpha G_x(x,y))\sin\theta + (I'_y(x,y) - \alpha G_y(x,y))\cos\theta\right]}{\|\mathbf{I}'(x,y) - \alpha\mathbf{G}(x,y)\|}$$

but can be expressed in a more compact form and substituted back into the error integral

$$\begin{aligned} \frac{\partial\epsilon(x,y)}{\partial r_o} &= \frac{-\alpha g'(d_o(x,y))(\mathbf{I}'(x,y) - \alpha\mathbf{G}(x,y))\cdot\hat{\mathbf{\Theta}}}{\|\mathbf{I}'(x,y) - \alpha\mathbf{G}(x,y)\|} \\ \int\int_R \frac{\partial\epsilon(x,y)}{\partial r_o}\epsilon(x,y)dR &= -\alpha\int\int_R g'(d_o(x,y))(\mathbf{I}'(x,y) - \alpha\mathbf{G}(x,y))\cdot\hat{\mathbf{\Theta}}dR = 0 \\ \int\int_R g'(d_o(x,y))\mathbf{I}'(x,y)\cdot\hat{\mathbf{\Theta}}dR &= \alpha\int\int_R g'(d_o(x,y))\mathbf{G}(x,y)\cdot\hat{\mathbf{\Theta}}dR. \end{aligned}$$

The right hand integral reduces to a scalar integral and the same arguments about even and odd function products apply here as were used in the 1D analysis. When a change of variables is introduced to align the edge with the ordinate by translating and rotating the domain (x,y) into (η,ν) where $\eta = \sin\theta x + \cos\theta y - r_o$ and $\nu = \cos\theta x - \sin\theta y$, the integral reduces to

$$\begin{aligned}\int\int_R g'(d_o(x,y))\mathbf{G}(x,y)\cdot\hat{\mathbf{\Theta}}dR &= int\int_R g'(d_o(x,y))g(d_o(x,y))dR \\ &= \int\int_{R'} g'(\eta)g(\eta)dR' \\ &= 0.\end{aligned}$$

The balance of the calculation follows the same tack as the 1D analysis:

$$\begin{aligned}\int\int_R g'(d_o(x,y))\mathbf{I}'(x,y)\cdot\hat{\mathbf{\Theta}}dR &= 0 \\ -\int\int_R \frac{d_o(x,y)}{\sigma_b^2} g(d_o(x,y))\mathbf{I}'(x,y)\cdot\hat{\mathbf{\Theta}}dR &= 0 \\ -\int\int_R \frac{d_o(x,y)}{\sigma_b^2}\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR &= 0\end{aligned}$$

$$\int\int_R (x\sin\theta + y\cos\theta)\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR = r_o\int\int_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR.$$

The solution is once again a first moment calculation. The difference is that the moment is measured along the direction normal to the edge, and the moment is of the scalar product $\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)$:

$$r_o = \frac{\int\int_R (x\sin\theta + y\cos\theta)\mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR}{\int\int_R \mathbf{I}'(x,y)\cdot\mathbf{G}(x,y)dR}. \tag{B.6}$$