# MEASURING VISUAL MOTION FROM IMAGE SEQUENCES

Padmanabhan Anandan

COINS Technical Report 87-21

March 1987

This research was supported by the Defense Advanced Research Projects Agency under contract number N00014-82-K-0464.

# Measuring Visual Motion From Image Sequences

A Dissertation Presented

By

Padmanabhan Anandan

Submitted to the Graduate School of the
University of Massachusetts in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

May 1987

Department of Computer and Information Science

This research was supported by the Defense Advanced Research Projects Agency under contract number N00014-82-K-0464.

MEASURING VISUAL MOTION FROM IMAGE SEQUENCES

A Dissertation Presented

By

PADMANABHAN ANANDAN

Approved as to style and content by:

Edward M. Riseman, Chairperson of Committee

Allen R. Hanson, Member

Peter J. Burt, Outside Member

Haluk Derin, Outside Member

W. Richards Adrion, Department Chair
Computer and Information Science

*Dedicated to the memory of my grandfather,*

*PROFESSOR D. GOPALAN*

எப்பொருள் எத்தன்மைத் தாயினும் அப்பொருள்
மெய்ப்பொருள் காண்ப தறிவு.

திருக்குறள்

## ACKNOWLEDGEMENTS

I wish to thank Professors Ed Riseman and Al Hanson for their continued advice and support throughout the course of this research, and for their comments on the earlier drafts of this thesis. Their continued commitment towards the motion project and their non-dogmatic attitude to research made it possible for me and others to explore novel and interesting approaches for the analysis of visual motion. Ed provoked me often to state my case clearly and concisely, and is largely responsible for the comprehensibility of this thesis. I wish to thank Dr. Peter Burt for valuable technical advice on numerous occasions, and Prof. Haluk Derin for his interest in this thesis and for being on my committee. My special thanks also go to Prof. Andy Barto for having me as a guest member of his research group, and to Prof. Michael Arbib for his professional advice and support during my stay at UMass.

Many of my colleagues have helped me in countless ways. Daryl Lawton pioneered the motion research at UMass, and paved the way for the rest of us. Frank Glazer started me in this direction of research, and I benefitted from his analytic skills and his expertise in system building. More recently Gilad Adiv (while he was still here), Mark Snyder, Poornima Balasubramanyam, Phil Kahn, Raj Bharwani, and Igor Pavlin provided an interesting forum to discuss ideas specific to motion research.

As good friends and gentle critics, Debbi Strahman and Michael Boldt provided me the necessary dose of skepticism. George Reynolds was willing to explain mathematical concepts, and also previewed most of my papers during my stay at UMass. Rich Weiss helped me analyze the functional minimization problem, and

verified the accuracy of the mathematical portions of this thesis. Les Kitchen, Brian Burns, John Dolan, Lance Williams, and Harpreet Sawhney were always available for provocative discussions on vision, and provided a lively intellectual environment.

John Dolan helped me with a number of drawings in this thesis, while Michael Boldt helped me with the layout. Of course, none of my experiments would have been possible without the skills of the people who built and maintained the VISIONS system, especially Robert Heller. My thanks also go to Janet Turnbull and Laurie Waskiewicz for various types of administrative support.

I wish to thank my mother and my uncle for their love and encouragement, and my brother Sundararaman, whose dreams for his siblings have been more than a source of inspiration for me. Thanks are also due to Ruki and TV for their continued friendship, and to Debbi, Les, and Jagu for their hospitality during this last and crucial period of my thesis work. Finally, I wish to thank the "amazing woman" for her enduring patience, countless sacrifices, for her moral and material support, and for giving me the benefit of her clarity of thought and her pragmatic attitude towards life.

ABSTRACT

MEASURING VISUAL MOTION FROM IMAGE SEQUENCES
May 1987

Padmanabhan Anandan

B.Tech., Indian Institute of Technology, Madras

M.S., University of Nebraska

Ph.D., University of Massachusetts

Directed by: Professor Edward M. Riseman

Motion is an important and fundamental source of visual information. It is well known that the pattern of image motion contains information useful for the determination of the 3-dimensional structure of the environment and the relative motion between the camera and the objects in the scene. However, the accurate measurement of image motion from a sequence of real images has proven to be difficult.

In this thesis, a hierarchical framework for the computation of dense displacement fields from pairs of images, and an integrated system consistent with that framework are described. Each input intensity image is first decomposed using a set of spatial-frequency tuned channels. The information in the low-frequency channels is used to provide rough displacements over a large range, which are then successively refined by using the information in the higher-frequency channels. Within each channel, a direction-dependent confidence measure is computed

for each displacement vector, and a smoothness constraint is used to propagate reliable displacement vectors to their neighboring areas with less reliable vectors.

For our integrated system, Burt's Laplacian pyramid transform is used for the spatial-frequency decomposition, and the minimization of the sum of squared differences measure (SSD) is used as the match criterion. The confidence measure is derived from the shape of the SSD surface, and the smoothness constraint is formulated as a functional minimization problem. Results of applying our system to several image-pairs containing complex camera motion as well as independently moving objects are included.

A number of well-known gradient-based and matching techniques are also shown to be consistent with our framework. The mathematical relationship between the gradient-based techniques and a class of correlation techniques is established.

This thesis also includes several proposals for extending our approach for multiple-frame analysis. Of particular interest is an approach which involves the decomposition of the input images according to orientation as well as scale. This new approach unifies the spatio-temporal energy models, which are currently popular in psychophysics, with the gradient-based and the matching techniques, and appears biologically feasible, and ideally suited for connectionist models of computation.

TABLE OF CONTENTS

## LIST OF FIGURES

## List of Tables

# CHAPTER I

# INTRODUCTION

Dynamic behavior is inherent to the nature of our physical environment. The ability to perceive the dynamic aspects of an environment is important for any biological or artificial system attempting to function in that environment. If vision is used as a means of perceiving the nature of the environment and constructing an internal model of the external reality of the system, the perception of dynamic visual events is important.

Dynamic visual events arise when there is any relative movement between the visual sensor and any part of the environment. Since visual perception is based on the projection of the environment on the image plane, the perception of dynamic visual events will be founded upon the measurement of the movement of image events on the plane of the image. Hence, the measurement of visual motion is a necessary part of any perceptual system.

This thesis is concerned with the problem of measuring motion from image sequences. In our view, the measurement of motion is an early visual process, and should require minimal preprocessing of the input images. Hence, these measurements should be based on image intensity structures or primitive tokens extracted from the image. For the video-mode of input, a natural choice for the definition of the measurement of motion is the set of inter-frame displacements of the image points.

In this thesis, a hierarchical framework for the computation of dense displacement fields from a pair of real images and an integrated system consistent with

that framework are described. The key idea underlying our framework is the separation of computations according to scale. This idea is based on the following observation: usually, the large scale (or low spatial frequency) intensity variations can provide imprecise measurements over a large range of magnitudes of motion, while the small-scale (or high spatial-frequency) variations can provide more accurate measurements over a smaller range. This leads to three components of our framework: **spatial frequency decomposition**, which is the method of separating the intensity variations according to scale, a local, parallel **match criterion** within each scale, and a **control strategy**, which is a method for controlling the measurement processes at the different scales and combining their results.

Although the scale based separation of computation provides a useful principle for processing scenes containing large displacements, there will always be situations when an image area lacks sufficient local information for displacement computation at a particular scale. Also, since the image displacement is a vector quantity, its reliability can vary according to direction. Therefore another essential component of our framework is a direction-dependent **confidence measure**. The presence of unreliable displacements also means that in order to obtain a dense displacement field, it may be necessary to propagate the reliable displacements to their less reliable neighbors. This leads to the last essential component of our framework: a **smoothness constraint**, which specifies the criterion for the propagation of reliable displacements.

Although the various components of our framework can be found in different well known techniques for the measurement of motion, until now, these components have not been unified into a coherent computational framework. The explicit specification of a framework allows us to unify and compare a wide range of current techniques for the measurement of motion. In addition, the task of designing a working system is reduced to that of making proper implementation choices for the various components of the framework.

During the design of our system, the implementation choices for the spatial fre-

quency decomposition, match criterion, and the control strategy were made from techniques that have been developed by other researchers in computer vision. The primary criteria for our choices were ease of implementation and efficiency of processing. On the other hand, the algorithm for computing a direction-dependent confidence measure was newly developed, since there has been very little effort in computer vision research to develop such a measure. In addition, we have also developed what appears to be the first rigorous mathematical formulation of a smoothness constraint for a correlation-based matching approach. A key contribution of our research is the integration of these components into a robust system, which has been tested successfully over a number of real image pairs. This thesis includes the results from a few of these tests.

Following the development of our system for processing a pair of frames, we have also considered the methods of extending our approach for processing multiple-frames. These considerations lead to a novel, connectionist scheme for the measurement of motion from a sequence of images, which is also described in this thesis.

## I.1 Defining the measurement of motion

In its most general sense, the measurement of visual motion is a qualitative or a quantitative statement regarding the *motion* of the events on the image plane. For example, the following are all measurements of visual motion: the direction of movement of an image feature at a particular point, the instantaneous velocity of a point on the image plane, the displacement of an image event such as a point, line or an area during a certain time interval, and the change in location and/or the shape of a geometric structure extracted from the intensity image. However, not all statements regarding the dynamic behavior of the image intensity are measurements of motion. For example, the rate of change of intensity at a point is clearly a statement regarding the dynamic intensity image, but is not a

measurement of motion.

In general, the choice of a particular type of measurement should be based on the manner in which it is expected to be used. For the measurement of visual motion, this choice depends on the overall view of the process of analyzing visual motion. In computer vision, the most popular approach for motion analysis has been to measure "image-flow" prior to any determination of geometric structures, and then use these measurements for recovering the 3-dimensional structure of the environment and for determining the relative 3-D motion between the camera and the objects in the scene. This approach is based on psychophysical studies which indicate that in the human visual system, the measurement and the interpretation of motion are early processes. For example, Gibson [38] suggested that the flow of the intensity patterns on the image plane can be a useful source of information for navigation and the determination of environmental structure. Similarly, a number of recent studies (for example, see [111,115]) also suggest that the measurement of motion may precede the process of segmenting the image into coherent regions and can actually aid the image segmentation process and the determination of the 3-D structure of the environment.

More specifically, in computer vision, the primary emphasis has been on the determination of instantaneous image-velocities and the displacements of points between successive frames, although a few techniques use symbolic representations of lines and regions in the image as the basis for the measurement of motion. Recent developments in psychophysics, however, have focused on the determination of the speed and direction of motion of a one-dimensional visual signal.

Most of the computer vision techniques for computing the image-velocity utilize the following mathematical relationship:

$$I_x u + I_y v + I_t = 0$$

where $I_x, I_y$ and $I_t$ are the spatial and temporal derivatives of the image intensity function at a point on the image plane, and $u$ and $v$ are the $x$ and $y$ components

of the image velocity of that point. These techniques are known as *gradient-based techniques*, since they use the above-mentioned relationship between spatial and temporal image gradients. When applied to real image sequences, these techniques work most effectively when the displacements are small compared to the distances over which significant intensity changes occur.

The techniques for computing image displacements use a matching approach for determining the correspondence of points from a pair of frames. These consist of *token matching approaches* which attempt to determine the correspondences between primitive tokens extracted from the intensity images and *correlation-based approaches* which determine the match for a point in the first image by optimizing a match measure over a set of candidates from the second image. The match measure between two points is usually based on the cross-correlation of the image intensities around the points.

The techniques described in the psychophysics literature for processing one-dimensional signals usually consist of two computational units which are tuned to a single spatial frequency but to different directions of motion. One unit responds maximally to leftward movement of the signal at a particular speed[1], whereas the other unit responds maximally to rightward movement at the same speed. Sometimes, a third unit which is tuned to stationary signals is also included. The direction of motion (left, right, or stationary) can be determined by noting which of the three units has the maximum response, while the speed can be determined by combining their response strengths. Based on the fact that these units can also be described as localized energy detectors in the spatio-temporal frequency domain, these techniques are called *spatio-temporal energy models* [2]. The application of these techniques to a two-dimensional signal has not been considered in detail, but a preliminary effort can be found in the work of Heeger [47].

[1] It appears that the spatial frequency and the speed to which a unit is tuned are inversely related to each other [114].

The above categorization of the techniques for measuring visual motion is based on the type of measurement they compute. It should be noted that are a number of techniques within each of these categories. It is also common to use a discrete version of the gradient-based techniques to measure displacements. Chapter II contains a detailed review of several techniques belonging to each category and their relationship to one another.

In this thesis, the inter-frame displacement of points on the image plane has been chosen as the measurement of motion. As noted before, this choice fits naturally into the video-mode of input, which consists of a sequence of temporally separated images. Given this definition of the measurement of motion, our primary goal has been the development of a robust system for the accurate determination of image displacements between a pair of frames.

The remainder of this chapter contains a definition of the the computational goals of the matching process, an overview of the framework, a description of its components, an overview of the system developed by us which is consistent with this framework, and an outline of the methods of extending our approach for processing multiple-frame sequences.

## I.2 The computational goals

The goals of the process for computing image displacements are determined by three major factors: the nature of its input, the requirements on the output, and computational efficiency constraints. The input is a pair of digitized frames belonging to a discrete image sequence. The image displacements arise due to a general 3-dimensional motion of the camera as well as due to independently moving objects in the scene. The output required is a dense field of displacement vectors with associated confidence measures. All the computations must be pixel-parallel and use local image information.

## The input

In typical video sequences, the inter-frame displacements are usually considerably larger than a pixel. Due to the presence of high-frequency (i.e., small scale) intensity information at edges and corners, the amount of displacement can be large compared to the scale of intensity changes. This implies that the gradient-based techniques for velocity computations cannot be directly applied. In addition, due to the presence of independently moving objects, often no single set of image transformation parameters is valid for the entire image. This means that techniques which may be highly successful for a restricted class of camera motions, such as Lawton's translational algorithm [61] or image-registration algorithms, cannot be directly applied. Instead, a technique that individually determines the displacements of local image neighborhoods must be used.

Although our approach does not contain some of the usual restrictions on the type of motion and the magnitude of the displacements, a few assumptions concerning the environment are necessary. First, the objects in the environment are assumed to be composed of continuous and opaque surfaces. Second, it is assumed that within the image area covered by a single surface, the displacement field varies smoothly, and that the magnitudes of the displacements along any direction are small compared to the width of the surface along the same direction. Finally, it is also necessary to assume that the image motion can be described as "locally translational", i.e., within a small area of the image, the displacement field can be approximated by a translational flow field. These assumptions are satisfied in a large class of real images. The major exceptions arise when the images contain transparent and/or "fence-like" surfaces, when the points on an object undergo chaotic (or turbulent) motion, and when the amount of rotation is significantly large.

## The output

The requirement that the output should be a dense displacement field with confidence measures is derived from the conclusions of the various studies concerning the problem of extracting structure from motion [3,5,32,63,90,91,106,116]. These studies can be divided into three classes: those that address the problem of determining the minimum configuration of points needed in order to determine the 3-D structure of rigid objects [91,106], those that use local differential properties of the image flow field [63,90,116], and those that use non-local grouping methods [3]. A common characteristic of all the studies is that they assume that the environment is composed of rigid moving objects with opaque surfaces and that the amount of rotation is small.

The studies concerning the minimum configuration of points needed for the determination of structure [91,106] suggest that such processes are sensitive to small errors in the displacements, unless a large number of displacements are used. In addition, there should be an indication of the reliability of the displacements, so that the inaccurate ones can be ignored. Similarly, the techniques that use the differential properties of image flow also require a dense displacement field in order to compute the derivatives of the image flow field.

Perhaps the only technique which has been successfully demonstrated to interpret the image flow extracted from real images containing independently moving objects is that of Adiv [3]. Adiv's algorithm, which employs a non-local grouping method, also requires a large number of reliable displacements. Since there is usually no *a priori* indication of object locations, the density of the displacement vector field should be large uniformly across the image. These conclusions are further supported by his analysis of the inherent ambiguities in the interpretation of motion [5].

The assumption regarding the rigidity of the environmental objects is implied in the types of geometric analyses performed by all three classes of techniques.

These analyses also show that the rotational components of the relative motion between the camera and an object contains no information regarding the 3-D structure of the object. Hence, large rotations often confound the problem of determining the 3-D structure and cause ambiguities in the interpretation of image motion [5]. This leads to the requirement that the rotations be small.

In summary, all of the current approaches undertaken to study the "structure from motion" problem require a large number of reliable image displacements as input. Based on these studies, we require that our output should be a dense displacement field with confidence measures.

### Computational considerations

The considerations of computational efficiency, ease of implementation, and the fact that the image motion lacks global coherence suggest that the following three properties are desirable for all of our computations: *parallelism*, *uniformity*, and *locality*.

Parallelism simply means that it should be possible to perform all computations simultaneously at all locations on the image plane. Uniformity implies that the process should be similar at all locations. It should be possible to describe any differences between the computations at different locations in terms of a few simple parameters. Locality means that the computations at any point on the image should be based on information local to that point.

## I.3 The computational framework

### An overview of the framework

Each input image is decomposed into its spatial frequency components by using a set of spatial frequency channels. The information in the low-frequency channels is used first to provide rough displacements over a large range. For each

displacement vector within each channel, a confidence measure is computed which indicates the reliability of the displacement vector. A smoothness constraint is then used to propagate reliable displacement vectors to their neighboring areas with less reliable vectors. The smoothed displacement vector field is projected to the next higher spatial frequency channel where the displacements are refined by using the information at that higher-frequency. A visual illustration of this framework is provided in Figure 1.

For the purpose of describing our framework, it is convenient to identify its five major components: (i) spatial frequency decomposition, (ii) the match criterion, (iii) the control strategy, (iv) the confidence measure, and (v) the smoothness constraint. The following sections describe the motivation for choosing each of these components and the properties that are required of them.

### Spatial frequency decomposition

As noted briefly at the beginning of this chapter, The key idea underlying our computational strategy is the separation of computation on the basis of scale. This idea is in the form of a constraint involving the scale of the image intensity variations, and the range, the accuracy, and the resolution (i.e., sampling density) of the estimated displacements.

Intuitively, it is clear that while small scale intensity structures can be used to measure displacements over a short range, they may have many duplicate matches over a large range. This leads to ambiguities in matching. Therefore, in order to process large displacements, large scale intensity information must be used. However, a single displacement computed on the basis of a large scale intensity structure will be an average of the displacements over the area covered by that structure. This is roughly equivalent to spatially "smoothing" the true displacement field. Such a smoothed displacement field will vary slowly over the image plane. This means two things. First, due to the averaging effect, the accuracy of the displacements will be low, and second, such a field can be sampled

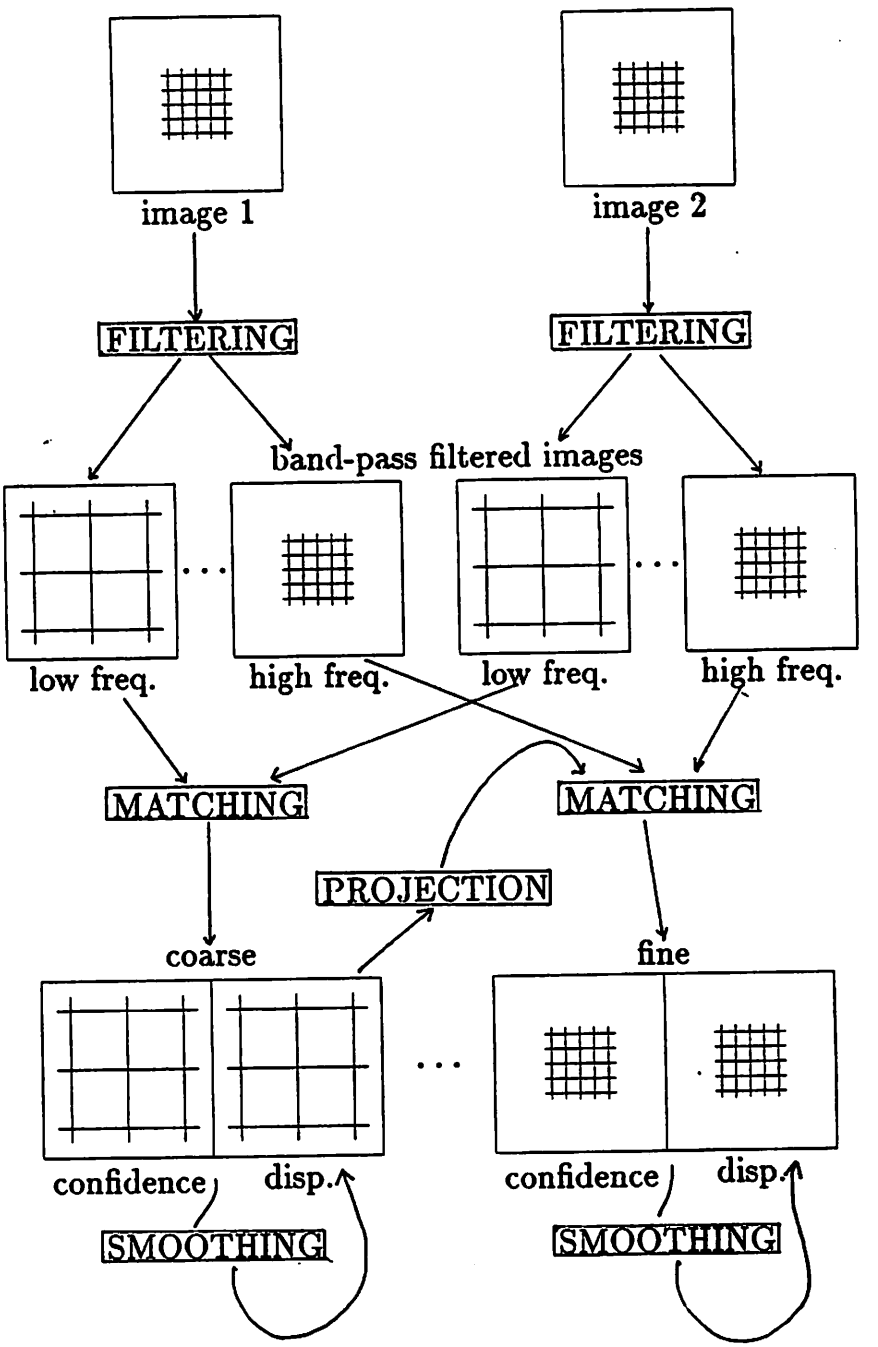



**Figure 1:** The hierarchical computational framework

at a lower rate without loss of information about the displacements.

These observations suggest the following principle: large scale image structures can be used to measure displacements over a large range with low accuracy and at a low sampling density, while small scale image structures can be used to measure displacements over a short range with higher accuracy and at a higher sampling density. An obvious way to enforce this principle is to decompose the image into its spatial frequency components. Such a decomposition and the subsequent processing can be achieved by using a set of *spatial frequency channels*[2].

Since the lower-frequency information can be sampled at a lower rate without any significant loss of information, the spatial frequency decomposition process is usually accompanied by a corresponding reduction of resolution [22,125]. Such an approach leads to a hierarchical representation of the spatial frequency channels and fits naturally into a pyramid [57,103] or a processing-cone [46] architecture.

Although a pyramid representation scheme is often useful for reducing the computational cost of the matching process, there are situations when it is more efficient to maintain the images at full resolution. For example, if a mesh connected computer (MCC) [71] is used, the pyramid representation is inefficient due to the the local connectivity constraints of the MCC architecture. It is important to note that our computational framework is not specific to any architecture. Such issues as image resolution and the method of passing information between pixels are specific to a given architecture. Therefore, a discussion of these issues is postponed until chapter III, where two distinct algorithms suited for two different architectures are considered.

### The match criterion

As noted earlier, the *match-criterion* is a method for determining the displacements within each channel. A number of different approaches are suitable

[2]The confirmation of the existence of spatial frequency channels in the human visual system [122] lends additional motivation for our approach.

for measuring the displacements within a channel. Since, the displacement measured within a channel is small with respect to the scale of the intensity variations, a gradient-based approach can be used (see [31,41]). Alternatively, a correlation-matching approach [20,40] or a symbolic matching approach based on primitive tokens [43,65] can also be used. As it will be explained in chapter II, the choice of a match criterion is often an important distinguishing characteristic for the different techniques which are unified within this framework.

The separation of matching according to scale implies that the match criterion should have a *scaling* property, i.e., the measurement processes within different channels should be scaled versions of each other. For example, if a gradient-based approach is used, the distances over which the image derivatives are computed should increase with scale. Similarly, if a correlation matching approach is used, the image area covered by the template windows should increase with scale. Note that if a pyramid representation is used, these are automatically achieved by simply maintaining constant sizes in terms of the number of pixels at each scale. On the other hand if the image at each scale is maintained at full resolution, then the sizes (in terms of the number of pixels) should increase with scale.

### The control strategy

The control strategy determines how the measurement processes at different scales are controlled and their results are combined. In our framework, the control strategy is based on a *spectral continuity* principle, which can be described as follows: Usually, it can be assumed that the projections of points on different environmental surfaces do not overlap in the image. Hence, the displacement estimates at corresponding image locations in the different channels are due to relative motion between the camera and the same surface; therefore, they must be similar. This means that at any image location, a displacement computed from a high-frequency channel must be consistent with the estimates from the low-frequency channel at the corresponding image location. A similar spectral-

continuity principle has been suggested for stereopsis in [43,69,120].

A simple way of enforcing the spectral continuity principle is by a "coarse-to-fine" control strategy. In this strategy, the processing proceeds from the low- to the high- frequency channels. Since the low frequency channels are expected to provide only displacements to a low accuracy, the displacement range is sampled coarsely, i.e., the match is determined with a large amount of uncertainty. At any pixel, the displacement estimate from one channel determines the center of the search area for the pixels in its vicinity in the adjacent higher frequency channel. The scale-invariance property of the measurement process suggests that the radius of the search areas between two adjacent channels should be proportional to the scale factor in order to ensure scale invariance of the computations. Once again, note that such scaling is automatically achieved in the pyramid representation by maintaining constant sizes in terms of number pixels at a particular scale.

Finally, it should be noted that the non-overlap assumption is invalid if the image contains transparent or fence-like surfaces. This was the reason for the assumption stated earlier that the input images do not contain such surfaces. Also note that the spectral-continuity constraint is invalid at object and surface boundaries in the image, since at such boundaries, the image motion is discontinuous, and the low-frequency information in the two frames will not be consistent. The problems encountered in processing such discontinuities will be discussed in greater detail in chapter III, where it will also be explained how our implementation of the coarse-to-fine control strategy deals with the violations of the spectral continuity constraint at such locations.

### The confidence measure

In general, there will be areas of the image with insufficient information at a particular scale for the local determination of displacements. Therefore a confidence measure should be computed along with each match at each scale to indicate whether or not to accept that match for further processing.

Since the image displacement is a vector quantity, it is possible that different directional components of the displacements may be locally computable with different degrees of reliability. For instance, it is intuitively clear that in a homogeneous area of the image no component of the displacement can be reliably estimated. On the other hand, at a point along a line (or an edge), although the component perpendicular to the edge can be reliably computed, the component parallel to the line (or the edge) may be ambiguous. Finally, at a point of high curvature along an image contour it may be possible to completely and reliably determine the displacement vector based on local information. These observations suggest that the confidence measure should be directionally selective, i.e., that it should associate different confidences with the different directional components of the displacement vector. In addition, while an area may be homogeneous at one scale, it may have information useful for reliable matching at a different scale. Hence, the confidence measures should be separately computed within each spatial frequency channel.

### Smoothness constraint

Since a primary computational goal of this approach is the determination of dense displacement fields, it may be necessary to "fill in" areas with unreliable displacements based on the reliable displacements in their neighborhood. Such a filling in process can be based on the assumption that the displacement field varies smoothly over the image area covered by a single surface.

The smoothness assumption is violated at locations of discontinuities in the image motion. As noted earlier, such discontinuities arise at surface boundaries (due to depth discontinuities) of a single object, or at object boundaries due to independent movement of two different objects. Any scheme that uses the smoothness assumption should also consist of mechanisms of detecting and processing such violations, although this has proven to be difficult in practice.

The most common use of the smoothness assumption can be found in gradient-

based techniques for determining image-velocities, which incorporate a smoothness constraint. This constraint is usually formulated as a variational problem involving the minimization of an error associated with a velocity field. In our framework, we suggest the use of a similar smoothness constraint within each channel. After the displacements and the associated confidences are computed within each channel, the displacement field should be smoothed before it is projected to the next higher-frequency channel. During the smoothness process, the confidence measures should be used to retain the reliable displacements while allowing the less reliable estimates to change. The various current approaches for the formulation of the smoothness constraint and their relative merits are described in detail in chapter II.

## I.4 An overview of our system

By definition, a computational framework is simply a skeleton. Although it captures the essence of the computational process, it needs to be developed into a full algorithm before it can be evaluated. This process of algorithmic development and the subsequent evaluation is also a way of refining and improving the framework.

As mentioned earlier, this thesis also describes an integrated system which is consistent with our computational framework. There are two implementations of this system, one of which is suited for a general purpose pyramid processor [103] while the other is suited for implementation on a mesh connected computer [71]. This section contains a brief overview of our system.

In our system, the spatial frequency decomposition process is achieved by using Burt's Laplacian pyramid transform [22]. This transform involves using a set of difference-of-Gaussian filters to implement a set of spatial frequency channels. Our match criterion is the minimization of the sum of squared difference

(SSD) measure, i.e.,

$$\sum_{i,j=-n}^{n} W(i,j)(I(x_0+i,y_0+j) - J(x_0+\delta x+i,y_0+\delta y+j))^2) ==> min$$

where $I$ and $J$ are the intensity functions describing the the first and second images respectively, $W$ is a weighting function, $n$ is the radius of the window, and $\delta x$ and $\delta y$ are $x$ and $y$ components of the displacements of the pixel located at $(x_0, y_0)$ in the first image along the $x$ and $y$ directions respectively. In our case, $W$ is chosen to be a Gaussian and $n$ is chosen to be 2. The control strategy involves projecting the displacement values obtained for a pixel in low-frequency channel to the corresponding pixels in the next higher frequency channel. These estimates are then refined by searching for the match of each pixel in the first image in an area of the second image centered around the location indicated by the corresponding low-frequency displacement estimate. It is shown that only a $3 \times 3$ set of candidate matches within this area need to be considered.

The confidence measure is based on the shape of the "SSD-surface", which is defined to be a surface whose height at each displacement is the SSD value corresponding to that displacement. The shape of this surface is measured in terms of its directional curvatures around the estimated displacement. In particular, the two principal curvatures (called $C_{max}$, and $C_{min}$) of the SSD surface and the directions $\hat{e}_{max}$ and $\hat{e}_{min}$ of the associated principal-axes completely specify the confidence measure. The smoothness constraint is formulated as a variational problem which involves minimizing the sum of two "errors" $E_{smooth}$ and $E_{approx}$, where $E_{smooth}$ measures the spatial variation of a given displacement field, and $E_{approx}$ measures how well this field approximates the displacement estimates computed according to the match criterion. The confidence measures are used to control the contribution of each pixel to the $E_{approx}$ term. The variational problem is solved by using a finite-element approach, which leads to an iterative relaxation algorithm.

Chapter III and IV together describe our system and explain the differences

between its two versions. Chapter III describes the choices made for the spatial frequency decomposition, the match criterion, and the control strategy. Chapter IV describes the choice of a confidence measure, the exact formulation of the smoothness constraint, and the smoothness algorithm derived from that constraint. Chapter V contains the demonstration of the performance of our system on several pairs of real images.

## I.5 Extension to multiple-frames

This thesis also contains several proposals for extending our framework and the system for processing a multiple-frame image sequence. The first set of extensions involve matching every successive pair of frames and incorporating a temporal continuity assumption on the displacement field for propagating the influence of all computations over time. The temporal continuity assumption can be either in the form of a "prediction-refinement" strategy [16], or in the form of a variational formulation.

Following the direct extensions for multiple-frames, a slight modification to our hierarchical computational framework is considered. This modification involves decomposing the input intensity images not only according to scale, but also according to the orientation of intensity changes. Each channel tuned to a particular orientation and spatial frequency measures image motion along a corresponding direction and over a corresponding range. The control strategy involves combining measurements not only across scales but across the different orientations as well. The multiple-frame extension of this modified framework is achieved at the level of the measurements computed by each unit, i.e., each unit measures the motion at a point over a small time interval. This proposed extension unifies the spatio-temporal energy models with the gradient-based and the matching approaches into a single computational framework.

The extensions of our framework to multiple-frames are discussed in detail in

chapter VI. It must be noted, however, that all the extensions are the in the form of proposals and have not yet been implemented as a working system.

## I.6 Summary

In summary, the primary focus of this thesis is the determination of dense displacement fields from a pair of real images. Thus far, a hierarchical computational framework has been described. Chapter II reviews related techniques described in the literature and explains how they fit into our framework. Chapter III and IV describe our system which is consistent with the framework, while chapter V demonstrates the performance of this system when applied to several real image pairs. Chapter VI describes the proposed set of extensions including the modified framework, which leads to a broader unified perspective on the measurement of visual motion. Finally, chapter VII summarizes the major contributions of our research and indicates the directions for future research.

# CHAPTER II

## MAJOR CURRENT APPROACHES

As noted in chapter I, most of the current approaches for the measurement of image motion can be divided according to their choice of a measurement: (i) those that use the continuous variations of intensity over space and time to measure *instantaneous image-velocities*, e.g., the gradient-based techniques [31,41,48,51,77], (ii) those that measure the *displacement* of points or primitive image tokens between successive frames of a sequence, e.g., the correlation-based matching techniques [20,40,87,125] and the symbolic-token based matching techniques [12,43,65], and (iii) those which measure the *spatio-temporal energy* of the image intensity function in a small area during a small period of time to determine the direction (and possibly the speed) of motion image points, e.g., the techniques described in [2,114].

This chapter contains a review of selected techniques from each of these three categories. An important characteristic common to all the techniques selected for this review is the possibility for parallel computation based on local image information. Techniques which do not possess this property are not suitable for "early" visual processing, and are therefore not considered in this review.

Typically, in any sequence of digitized images, both the spatial and temporal dimensions are discretized. If the spatial and temporal sampling rates of the time varying intensity image are high, the discrete approximations of the continuous formulations of the current techniques will still be useful. The exact condition required is that the sampling rate should be sufficiently high in order to avoid

aliasing effects. While the spatial sampling rate is usually high enough to meet this requirement, as noted in chapter I, the inter-frame image displacements are usually considerably larger than a pixel. This means that the temporal sampling rate is too low and the corresponding approximations are not valid. The difficulties due to temporal aliasing are usually addressed by using the same type of reasoning as in the development of our computational framework in chapter I, which leads to hierarchical, multi-resolution formulations which are consistent with our framework. Such hierarchical formulations have been used only for the first two categories mentioned above. Since the energy models have been developed as computational models for the human visual system, their application to computer vision is not yet fully understood.

This review is divided into three sections corresponding to the three categories mentioned above. All the velocity measurement techniques considered here are based on the gradient-based approach. Therefore, the principles underlying this approach are first described. Following this, examples of single- and multi-level gradient-based techniques are considered. The displacement computation techniques consist of correlation matching techniques as well as symbolic token matching techniques. First, the general principles underlying each of these two types of matching techniques are discussed. This discussion is followed by a description of several single-level and multi-level matching techniques. Since the energy models are somewhat novel, especially for computer vision, the principles underlying such models are described in greater detail. Two specific energy models are briefly outlined.

An important purpose of this review is to show the relationships of these techniques to our computational framework. Therefore, such relationships are discussed in detail following the description of the techniques in each category.

# II.1 INSTANTANEOUS VELOCITY COMPUTATION

## II.1.1 The principles underlying the gradient-based approach

### The intensity constraint

Almost all the techniques for measuring the instantaneous image-velocities use the gradient-based approach. This approach is based on the assumption that the intensity of light reflected by a point on an environmental surface and recorded in the image remains constant during a short time interval, although the location of the image of that point may change due to motion. This can be mathematically stated as,

$$I(x,y,t) = I(x+u\delta t, y+v\delta t, t+\delta t) \qquad \text{(II.1)}$$

where $\vec{U} = (u,v)$ is image-velocity vector at the point $(x,y)$, assumed to be constant during the interval $(t, t+\delta t)$. In the limit, when the length $\delta t$ of the time interval tends to zero, the intensity-constancy assumption leads to the equation noted in chapter I: $I_x u + I_y v + I_t = 0$, where $I_x$, $I_y$, and $I_t$ are the spatial and temporal derivatives of the intensity function. This equation is called the *intensity-constancy constraint*, or simply the *intensity constraint*. Figure 2 shows the geometric interpretation of this constraint. All points on the line shown in this figure satisfy the intensity constraint, hence, they are all possible candidates for the image-velocity at the point $(x,y)$.

At any point, if we denote the component of the image velocity tangential to the iso-intensity contour passing through that point by $U^T$, and the component perpendicular to the iso-intensity contour by $U^\perp$, we can rewrite the intensity constraint as

$$|\nabla I| U^\perp = -I_t$$

where $|\nabla I|$ is the magnitude of the intensity gradient vector $\vec{\nabla} I = (I_x, I_y)$. Note that this constraint involves only the "normal-flow" component $U^\perp$. The lack

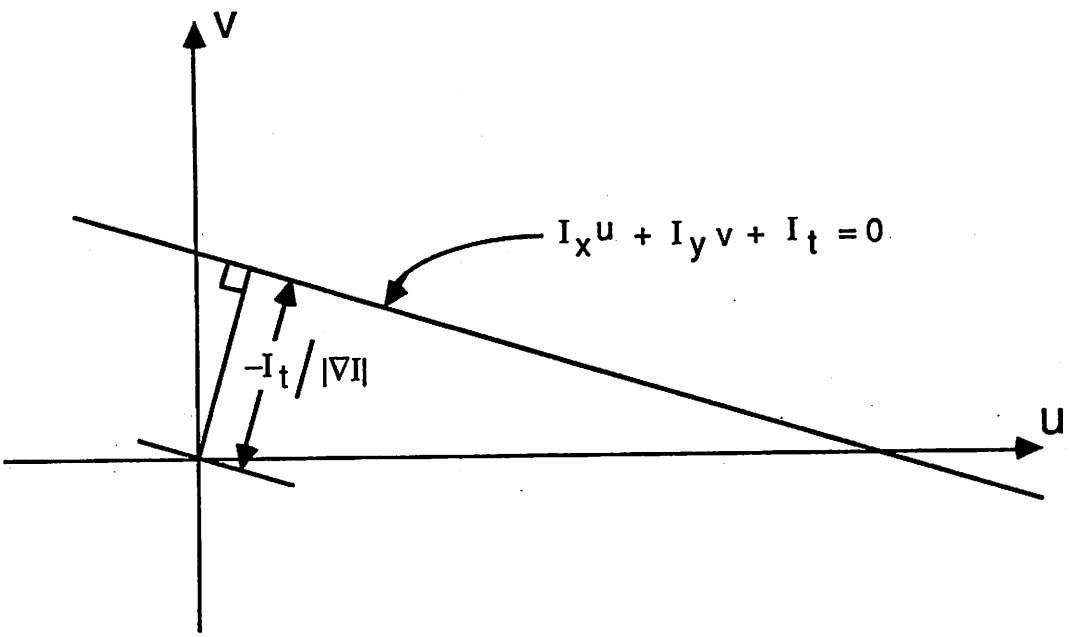



**Figure 2:** The intensity constancy constraint

of information regarding $U^T$, which is called the tangential-flow component, is known as the "aperture problem".

Note that the above formulation of the intensity constraint involves only the first-order derivatives of the image intensity function. Based on the consideration of the second-order intensity variations, Nagel [75] has suggested a slightly different formulation of the intensity constraint, which will be discussed later in this section.

Recently Kahn [54] has suggested an alternate method for measuring the normal-flow. His approach is based on the detection of "temporal-edges", which arise due to the movement of static image edges. The normal-flow is determined by comparing the order and the times of appearance of an edge at three discrete image locations that are equidistant from each other. The detection of a temporal-edge at a location is communicated immediately to its neighbors in order to determine the normal-flow. Such an "event-based" communication technique does not assume a temporal synchronization of the detection processes at neighboring locations on the image.

### Using a global constraint

Since the local intensity information determines only one component of the image-velocity at a point, an additional constraint is necessary to completely determine the velocity vector. The different techniques which are based on the gradient-based approach differ according to the type of additional constraint used by them.

Historically, the first attempts to incorporate a global constraint can be found in the techniques of Fennema and Thompson [33] and Glazer [39]. Fennema and Thompson assumed that the image is composed of a few distinct large regions whose motions are translations on the image plane. A modified Hough-transform clustering algorithm was employed to determine the distinct image velocities of these regions. Glazer assumed that the image-velocities are constant over a small

local neighborhood. Hence, he used a pseudo-intersection method to determine a unique velocity vector from the constraint lines of the points in that neighborhood. Since their underlying assumptions are often violated by the presence of 3-D motion and changes in depth, neither of these techniques seem directly applicable to real image sequences.

The remainder of this section reviews a few single-level and a few multi-level techniques which use slightly more realistic assumptions. Although the single level techniques are not directly applicable to scenes containing large image motions, each multi-level technique is based on a particular single level technique. In order to explain the principles underlying these computational methods, the single-level techniques are first described.

### II.1.2 Single-level gradient-based techniques

#### Horn and Schunck's approach

Horn and Schunck [51] assumed that the image-velocity field varies smoothly across the image plane and formulated a variational problem to determine the image-velocity field. For every field $\{\vec{U}(x,y)\}$, two errors are defined: one is called the smoothness error (or smoothness constraint), and defined as,

$$E_{sm}(\{U\}) = \int\int (|\nabla u|^2 + |\nabla v|^2)\, dx\, dy$$

while the other is called the intensity error and defined as,

$$E_{int}(\{U\}) = \int\int (|\nabla I|U^\perp + I_t)^2\, dx\, dy$$

Note that the integrands in the definitions of $E_{sm}$ and $E_{int}$ are functions of $\vec{U}(x,y)$. The quadratic form of the integrands guarantees that the two errors will be non-negative, and if considered individually, the minimum value of each is zero. Also

note that $E_{sm} = 0$, only for a constant velocity field, and when $E_{int} = 0$, we get back the intensity constraint. For future reference, we will rewrite $E_{int}$ as,

$$E_{int}(\{U\}) = \iint |\nabla I|^2 (U^{\perp} - V^{\perp})^2 \, dx \, dy \qquad \text{(II.2)}$$

where $U^{\perp}$ is the normal component of $\vec{U}$ (i.e., its component along the direction of the gradient vector), and $V^{\perp}$ is the local normal-flow estimate obtained from the intensity constraint. The smoothness error can be rewritten as,

$$E_{sm}(\{U\}) = \iint trace\{(\nabla U^T)^T (\nabla U^T)\} \, dx \, dy$$

where

$$\nabla U^T = \begin{pmatrix} u_x & v_x \\ u_y & v_y \end{pmatrix}$$

The problem of computing a *dense* image-velocity field is defined as that of minimizing a weighted sum of the two errors,

$$E(\{U\}) = \alpha^2 E_{sm}(\{U\}) + E_{int}(\{U\})$$

where $\alpha^2$ determines the relative contributions of the two errors. Horn and Schunck used Euler's equations to transform this minimization problem into the following set of differential equations:

$$(\nabla I)(\nabla I)^T U + I_t(\nabla I) - \alpha^2 \begin{bmatrix} u_{xx} + u_{yy} \\ v_{xx} + v_{yy} \end{bmatrix} = 0 \qquad \text{(II.3)}$$

These equations are then solved using a relaxation algorithm in which the image-velocities at all the points on a square-grid are iteratively updated based on their neighboring values.

### Hildreth's approach

Hildreth [48] also used a variational principle, but assumed that the smoothness constraint should be propagated only along image contours. Based on this

assumption, she formulated the error to be minimized as

$$E(U) = \int \left(\frac{\partial u}{\partial s}\right)^2 + \left(\frac{\partial v}{\partial s}\right)^2 + \beta \left(U^{\perp} - V^{\perp}\right)^2 ds$$

where $s$ is the arc-length along an image contour, and at any point along the contour, $U^{\perp}$ is the component of the unknown image-velocity vector $\vec{U}$ along the direction of the normal to the contour at that point, and $V^{\perp}$ is the value of the perpendicular component suggested by the gradient-based (or a similar) analysis.

If the contours are chosen to be the level-sets, which are the set of connected points over which the intensity value remains constant, then at any point, the perpendicular direction to the contour can be shown to be the same as the direction of the gradient vector. In this case, the intensity-constraint can be used to obtain the normal-velocity, and the third term of the error defined above can be shown to be the same as the intensity error used by Horn and Schunck. However, the general nature of Hildreth's formulation permits the use of a wider range of approaches for the estimation of the normal-flow.

### Nagel's approach

Nagel [75,76,77,78] has also formulated the problem of computing an image-velocity field as that of minimizing the sum of an intensity error $E_{int}$ and a smoothness error $E_{sm}$. However, noting that Horn and Schunck's definition of the intensity error was derived from equation II.1, Nagel used the difference between the left- and the right-hand sides of that equation and defines $E_{int}$ as,

$$E_{int} = \iint (I(x,y,t) - I(x + u\delta t, y + v\delta t, t + \delta t))^2 \, dx \, dy$$

Nagel also modified the smoothness error as,

$$E_{sm} = \iint trace\{(\nabla U^T)^T W (\nabla U^T)\} \, dx \, dy$$

where $W$ is a $2 \times 2$ symmetric, positive-definite weight matrix defined as follows:

$$W = \frac{F}{trace\{F\}}$$

with

$$F = \begin{pmatrix} I_y^2 + \beta^2(I_{xy}^2 + I_{yy}^2 & -I_x I_y - \beta^2 I_{xy}(I_{xx} + I_{yy}) \\ -I_x I_y - \beta^2 I_{xy}(I_{xx} + I_{yy}) & I_x^2 + \beta^2(I_{xx}^2 + I_{xy}^2 \end{pmatrix}$$

In Nagel's approach, the modification of the intensity constraint becomes noticeable only when Euler's equations are applied to the the minimization problem. In particular, the following set of differential equations are obtained:

$$AU + b - \alpha^2 \begin{bmatrix} trace\{W\nabla\nabla u\} \\ trace\{W\nabla\nabla v\} \end{bmatrix} = 0 \tag{II.4}$$

where

$$A = (\nabla I)(\nabla I)^T + \bar{x^2}(\nabla\nabla I)(\nabla\nabla I)^T \tag{II.5}$$

and

$$b = I_t(\nabla I) + \bar{x^2}(\nabla\nabla I)(\nabla I_t) \tag{II.6}$$

The $\nabla\nabla$ operator represents the matrix of second derivatives, e.g.,

$$\nabla\nabla I = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix}$$

and $\bar{x^2}$ denotes the size of a small image-window around a point $(x, y)$ which represents that point. Note that if $\bar{x^2}$ is set to zero, and $W$ replaced by the identity matrix Nagel's differential equations are exactly the same as the differential equation II.3 derived by Horn and Schunck.

The change in the smoothness constraint, i.e., setting $W = F/trace\{F\}$, has the following effect: at points on linear image structures and at locations of high-curvatures of image contours, the smoothness constraint is strongly enforced along the direction of the contour, and weakly enforced in the direction perpendicular to

the contour. The approach is based on the view that the image contours and the lines often correspond to object boundaries. Hence, the smoothness constraint is not propagated across such contours or lines. Due to this property, Nagel refers to his formulation as the "oriented smoothness constraint".

### Discussion

The primary difficulty for the gradient-based approaches arises from the fact that they are only suitable when the displacements are small with respect to the scale of the image intensity variations. As noted above, this requirement is rarely met in real image sequences due to the low temporal sampling rates used. In addition, any changes in illumination or contrast between frames cause the intensity constancy assumption to be violated.

Horn and Schunck's smoothness constraint is not valid at image motion boundaries. Although some efforts have been made towards the identification of such boundaries [94], their technique does not include a way of incorporating information regarding boundary locations. Hildreth's restriction of the smoothness process to be along contours, and Nagel's oriented smoothness constraint are both efforts to address the problem of discontinuities in image-motion. They are based on the view that at the boundaries of objects and surfaces, usually there is a discontinuity in the intensity function, which appear as edges and contours. However, often such contours appear due to rapid albedo changes and texture markings which may not be at the boundaries of the surfaces. In these cases, the velocities of the points on the contours are actually useful to constrain the velocities of the points in the adjacent areas. Therefore, it would be more useful to allow the isotropic propagation of the smoothness constraint, and use methods for stopping the propagation at known boundaries. Such an approach is not currently found in any computer vision technique for the measurement of motion.

In all of these techniques, the relaxation algorithms used for the propagation of the smoothness constraint tend to have slow convergence properties. This causes

additional problems for the practical use of these techniques. In the following section, two techniques are considered, which use multi-level processing methods for addressing this problem as well as for processing scenes containing large inter-frame motions.

### II.1.3 Multi-level gradient-based approaches

Recently, Enkelmann [31] and Glazer [41] have each proposed a multi-level gradient-based approach. Both approaches involve preprocessing the input images with a hierarchical set of Gaussian filters. These filters together form a set of low-pass spatial frequency channels. The spatial width of the Gaussian convolution mask is successively doubled, thereby successively halving the spatial frequency width of the corresponding channels. The input images are represented in a $2^n \times 2^n$ pixel array. The filtered images are successively sampled with a two-to-one sampling ratio, thereby providing a pyramid representation. A pyramid formed by using such a set of Gaussian filters is called a *low-pass pyramid* or a *Gaussian pyramid*. In the following discussion, we use the convention that the size of the image at level $i$ of the pyramid is $2^i \times 2^i$ pixels.

Both techniques apply the gradient-based approach for the determination of image displacements from successive frames. For the description below, the two intensity functions are denoted by $I$ and $J$, and $\vec{U}$ is used to denote a displacement vector. Both techniques also use multi-level processing for the propagation of a smoothness constraint, although they differ in their exact methods. The two techniques are individually described below.

#### Enkelmann's approach

Enkelmann's [31] technique is based on Nagel's approach which was discussed above. After constructing a low-pass pyramid from each image, Enkelmann begins the processing at some coarse-level $c$. In the description of his algorithm, he does not specify how $c$ is chosen. At the coarsest level $c$, the initial displacement

field consists of zero vectors. At all other levels $\{i, i = c+1, \ldots, n\}$ an initial displacement field is determined by projecting the final field from level $i-1$. The projection process involves a bi-linear interpolation of the displacement vectors in a small neighborhood of the field at level $i-1$.

At each level, the modifications of the initial vectors are based on the application of Nagel's approach. At level $i$ the images $I_i$ and $J_i$ are the input images. Nagel's intensity constraint is modified as follows:

$$E_{int} = \iint (I_i(x,y) - J_i(x + u_i^0 + du_i, y + v_i^0 + dv_i))^2 \, dx \, dy$$

As it can be seen from the above equation, the modification involves shifting the pixel $(x,y)$ according to its initial displacement $U_i^0 = (u_i^0, v_i^0)$. The field of update vectors $\{\vec{DU}_i = (du_i, dv_i)\}$ is unknown. The oriented smoothness constraint is formulated on the vector field $\{\vec{U}_i = \vec{U}_i^0 + \vec{DU}_i\}$.

In one version of his algorithm, Enkelmann uses the same single-level relaxation process as in the case of Nagel's approach. However, he notes that the convergence of this algorithm is very slow, especially at the finer-levels. Therefore, he provides a modified version where he embeds a multi-level relaxation process within his hierarchical gradient-based approach. After a fixed number of iterations at level $i$, the algorithm is tested for convergence. This test involves measuring the maximum change during one iteration among all the vectors in the image. If this measure is larger than a pre-defined threshold, the smoothed vector field is projected to the coarser level $i-1$, where it is used as the initial vector field for a correction process. A single iteration is performed at level $i-1$ and the update vector field $\{\vec{DU}_{i-1}\}$ is projected back to level $i$, where it is added to the smoothed vector field. Following this, an additional number of iterations are performed at level $i$ to smooth the effect of the correction at level $i-1$. This entire process is repeated for each level from the level $c+1$ to input level $n$. At the coarsest level $c$, a single-level relaxation process is employed.

### Glazer's approach

Although the primary motivation behind Enkelmann's multi-level approach is the concern that Nagel's single-level approach converges slowly, the multi-level approach is also necessary for measuring large inter-frame displacements. This is recognized by Glazer during his development of a multi-level gradient-based approach.

As noted earlier, Glazer also uses a low-pass pyramid representation of the input images[1] and employs a hierarchical version of the Horn and Schunck approach. Given the initial displacements $\vec{U}_i^0$ for every grid-point at level $i$, the intensity constraint is defined as,

$$E_{int} = \iint (|\nabla F_i| DU_i^{\perp} + \hat{J}_i - I_i)^2 \, dx \, dy$$

where

$$\begin{aligned} \hat{J}_i(x,y) &= J_i(x + u_i^0, y + v_i^0) \text{and} \\ F_i(x,y) &= \frac{1}{2}(I_i(x,y) + \hat{J}_i(x,y)). \end{aligned}$$

and $D\vec{U}_i$ is the update to the displacement at level $i$. Glazer also defines a smoothness error $E_{sm}$ as,

$$E_{sm} = \iint trace \left\{ (\nabla U_i^T)^T (\nabla U_i^T) \right\} \, dx \, dy$$

where $\vec{U}_i = U_i^0 + D\vec{U}_i$. Note that as in Enkelmann's case, the intensity constraint only involves the "update vector" $D\vec{U}_i$, whereas the smoothness constraint is formulated on the vector $\vec{U}_i$.

As in the case of Enkelmann's first algorithm, the process begins at a coarse-level $c$ where the initial displacement vectors are chosen to be zero-vetors. Glazer

[1] In his other work with his colleagues, Glazer [40] has described a hierarchical correlation technique, which uses a band-pass pyramid. This work is reviewed later in this chapter.

suggests using $c = n - \lceil \log_2 \Delta \rceil$, where $\Delta$ is the maximum $x$ or $y$ component of the displacement among all the vectors in the field. However, due to the lack of any *a priori* knowledge of $\Delta$, the choice of $c$ is usually made by the user.

The process proceeds in a coarse-to-fine fashion from level $c$ to level $n$. At all levels $\{i, i = c+1, \ldots, n\}$, the initial vector field is the projection of the final vector field from level $(i-1)$. The projection is based on a "quad-tree" connectivity, wherein each pixel at level $(i-1)$ is defined to be the "parent" of four pixels at level $i$. After the projection, the incremental displacement vectors $\vec{DU}_i$ are computed by minimizing the sum of the intensity and the smoothness errors defined above.

Glazer also describes a multi-level approach for the smoothing process. His approach is more complex than Enkelmann's method and is based on recent theoretical work concerning general multi-level relaxation techniques. It involves dynamic switching between the levels (up and down) according to the current rate of convergence during the course of the process. Although the hierarchical gradient-based approach and multi-level relaxation are both described in detail, it is not clear whether the two processes have been integrated into a single scheme for the measurement of motion.

### II.1.4 Relationship to our computational framework

Although the single-level gradient-based approaches are unsuitable for processing real image sequences, the multi-level schemes appear fully suited for practical use. Both the multi-level techniques described above are also consistent with our framework. In particular, the five components of our framework are present in each of these techniques.

**spatial frequency decomposition** - Both Enkelmann and Glazer use a set of Gaussian low-pass filters and a pyramid representation for implementing the spatial frequency channels.

**match criterion** - In both cases the match criterion is based on the intensity constancy constraint. In Glazer's algorithm, the normal displacements are obtained from the intensity constraint equation. Although it is not explicitly present in Enkelmann's algorithm, the match criterion appears in the differential form derived using the Euler's equations. In particular, following Nagel's approach, Enkelmann can determine locally the normal displacement at edge-like points and uniquely determine both components of the displacement at "corner-like" points. For details, see [75,78].

**control strategy** - Both techniques are founded on the use of the coarse-to-fine control strategy. However, the additional use of multi-level smoothing algorithms makes the control usually more complex and data-dependent.

**Confidence measure** - Although a confidence measure is not explicitly present in either technique, such a measure is implicitly used during the formulation of $E_{int}$. In Glazer's technique, the confidence measure for the normal-flow component is $|\nabla I|$, whereas the confidence for the tangential-flow component is zero. In Enkelmann's case, neither confidence measure is zero for all points. Due to the use of second-order image intensity derivatives, at corner points both will be high, at edge-like points the confidence associated with the tangential-flow component will be low, and at homogeneous areas both confidences will be low. However, the implicit use of a confidence measure has not been recognized by Glazer or Enkelmann in the descriptions of their algorithms. There is also a clear mathematical relationship between the confidence measures which have been implicitly used in these two gradient-based techniques and our matching technique described in this dissertation. Since a thorough discussion of this relationship requires a detailed description of our technique, we postpone such a discussion to chapter IV.

**smoothness constraint** – It should be clear from the descriptions of the two techniques given above that both of them use a smoothness constraint. The difficulties at areas of image motion boundaries are inherited by both techniques from the respective single-level techniques upon which they are based.

It should be noted that both these techniques have been developed very recently [31,41]. In our view, along with our integrated system, which will be described in chapters III and IV of this thesis, these two techniques show some of the best results for the problem of determining dense displacement fields from a pair of real images.

## II.2 DISPLACEMENT FIELD COMPUTATION

The measurement of visual motion from a discrete image sequence is usually based on identifying corresponding image events from successive frames. This is called the *correspondence problem* and leads to a matching approach, i.e., one in which image events from successive frames are matched with each other.

There are two classes of matching techniques that are commonly used: correlation based techniques, and symbolic token matching techniques. In this section, first, the common computational principles and motivations underlying each of these classes are described. This is followed by a discussion of a selected set of single-level and multi-level techniques belonging to either class. Finally, the relationships of these selected techniques to our computational framework are explained.

### II.2.1 The principles underlying correlation matching techniques

If the intensity functions of two images are $f(x,y)$ and $g(x,y)$ then the cross correlation function between the two images is defined as (see [92]),

$$C_{fg}(\delta x, \delta y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x-\delta x, y-\delta y)g(x,y)xdy \qquad \text{(II.7)}$$

where $\vec{\delta} = (\delta x, \delta y)$ is the relative shift between the two images, C is the correlation function, and $x$ and $y$ vary over the two images. The best estimate of $\vec{\delta}$ is determined by maximizing the C over a set of candidate values for $\vec{\delta}$.

When a finite sized window from one image is matched against identical windows from the second image, the definition given above is modified such that the integration is limited to the windows. If the image is represented as a discrete pixel array, the above description will still hold, except that "point" is replaced by a "pixel" and the integral is replaced by discrete sum. Since our concern is primarily with digitized images, the following discussion uses the discrete formulations.

There are also other related measures that can be used to determine the match (see below). In general, these measures can also be used as an estimate of the match strength between two *feature vectors*; hence, they are useful as match measures for many different types of image events.

The correlation matching process consists of the following steps:

- An area around a pixel of interest in the first image is chosen as the template window.
- All the pixels in a target area (called the *search area*) in the second image, which is expected to contain the match of the pixel of interest in the first image, are called *candidate match pixels.* An area congruent to the template window is chosen around each candidate match pixel. These areas are called *candidate match windows.*

- For each candidate match pixel, a *match measure* is determined by comparing the image intensities of the pixels in the template window and the corresponding pixels in the candidate match window. The most common match measures are (i) direct correlation, in which the image intensity values of the corresponding pixels in the two windows are multiplied and summed, (ii) mean normalized correlation, in which the average intensity of each window is subtracted from the intensity values of each pixel in that window before multiplication and summing, (iii) variance normalized correlation, in which the correlation sum is divided by the product of the variances of the intensities in each window, (iv) sum of squared differences, in which the sum of the square of the differences between the intensities at corresponding pixels is used, and (iv) sum of absolute differences, which is similar to sum of squared differences, but the absolute values of the differences are used instead of their squares. The mathematical definitions of these measures are provided in Appendix A.

  In some cases, the match measure may be a *weighted* sum of the individual pixel comparisons. Usually, the weights are chosen to increase the contribution of the pixels near the center of the window relative to those of the outlying pixels.

- If either direct, mean normalized, or variance normalized correlation is used to compute the match measure, then the *best match* window is the candidate window that maximizes the match measure. If one of the difference measures is used, then the best match window is the candidate window that minimizes the match measure. The candidate match pixel which corresponds to the best-match candidate window is regarded as the "match" for the pixel of interest in the first image.

In a strict sense, the use of a correlation measure assumes that the displacements of all the pixels in the template window are identical. However, since the

optimization process determines the "best" match among a set of candidates and not an exact match, this method of matching is applicable to more general cases of image motion. In particular, if the image motion can be assumed to be locally translational, these measures are still useful for matching. Such an assumption will be violated if the image area undergoes significant rotation or expansion or if it contains motion-boundaries.

There is a long history of using correlation techniques for image registration, stereo disparity computation, and for motion analysis. However, the techniques which are tailored for solving registration problems are not always suitable for the measurement of motion. Some techniques developed for stereo-matching are of interest here, because the principles underlying their computations are suitable for the measurement of motion.

### II.2.2 The principal motivations for symbolic matching

Symbolic matching techniques use a symbolic representation of geometric structures in the image as the basis for matching. Such an approach is motivated by the view that geometric structures in the image often correspond to "interesting" physical structures in the 3-D environment. These structures are interesting because they may be easily distinguishable from other such structures and are likely to be stable over several image frames. With suitable representations, it may also be possible to find the match for a geometric structure even when the image contains rotations and expansions. In addition, the set of candidate matches can be restricted to distinct image structures, thereby both increasing the reliability and reducing the computational cost. Finally, spatial image structures suitable for image segmentation can be chosen, thereby allowing a combination of static and dynamic image analysis.

The techniques in this category range from those relying on simple point tokens [12,85] or edge tokens [65,43] to more complex structured edges and regions [53,107]. The determination of the complex symbolic tokens tends to be expensive.

Moreover, since they usually describe events in a large area of the image, mistakes in the determination of the tokens tend to seriously affect the matching process. Since our focus is on early processes for the measurement of image motion, only an outline of the techniques which match complex image structures is included in this review.

### II.2.3 Single level matching techniques

#### Gennery's stereo algorithm

Gennery's stereo algorithm is based on maximizing the variance-normalized correlation measure. Since the algorithm is designed for processing stereo images, one component of the displacement is assumed to be known. In particular, two corresponding points in the two images will lie on corresponding *epipolar-lines*[2]. Therefore, only the displacement along the epipolar-line, which is called *disparity*, needs to be determined.

Gennery's primary contribution is a consideration of the reliability of the estimated disparities. Based on a mathematical analysis of the shape of the correlation function, he determined the confidence of a match. His analysis is founded on statistical theories and involves the assumption that the intensity values at the pixels of the image can be considered to be set of uncorrelated random variables with Gaussian distributions. In addition, Gennery also included a method for camera calibration and the estimation of sensor noise parameters.

#### Barnard and Thompson's approach

Barnard and Thompson [12] describe a technique for computing image displacements of point tokens. The points are selected by an "interest-operator" [72]

[2]The epipolar-line is defined as the intersection of the image plane and an *epipolar-plane*. The epipolar-plane is defined as a plane containing the line joining the focal points of two cameras arranged in a stereo configuration. Two epipolar lines in two images correspond to each other if they arise from the same epipolar plane. For the usual situation of parallel axis stereo with a horizontal shift between the cameras, the epipolar lines are the scanlines of the images.

which identifies distinguishable image pixels. All further processing is restricted to the two sets of interesting points derived from the two images.

For each interest-point in the first image, the set of candidate matches consists of all the interest-points in the second image that are within a given distance from the point in the first image. Each interest point in the first image is called a "node" and each of its candidate matches is called a "label". The sum of squared difference measure over $5 \times 5$ windows is used as the match measure. For each label, an initial probability of match is derived from its match measure.

Barnard and Thompson's algorithm includes a smoothness constraint on the displacement field, although this smoothness constraint is used only to modify the displacements of the interest-points. The displacements of the pixels which are not contained in the set of interest points are not computed. A probabilistic relaxation labeling algorithm is used to implement the smoothness constraint. The probabilities of each label for each node are iteratively updated based on the probabilities of the labels of all the nodes in its local neighborhood. The algorithm is terminated after 10 iterations of the relaxation process.

### Other single-level techniques for matching primitive tokens

Prager and Arbib [85] describe a symbolic matching technique which is similar to Barnard and Thompson's approach. The primary differences are in the formulation of the smoothness constraint, and in the fact that Prager and Arbib include a temporal continuity constraint on the displacement. However, the description of their technique does not clearly specify the method of choosing the interest points and the match measure.

Baker [11] and Ohta and Kanade [82] have separately developed edge-based matching algorithm for stereopsis. Baker's algorithm includes a second stage, where the edge matches are used to constrain the search for matches of points near an edge. The point-matches are determined by a correlation-based algorithm. Both these edge-based matching algorithms assume that the corresponding epi-

polar lines are known, and use an "edge-connectivity constraint" to propagate the influence of the match for an edge to the edges connected to it in the adjacent epipolar-lines. The use of this constraint and the exact formulation of the edge matching algorithms are rather specific for stereo-matching and do not generalize easily for motion analysis.

### II.2.4 Hierarchical matching techniques

Single level matching can be computationally expensive, since the search area can be large. Further, for a given template window size, as the search area size increases, there is also a greater potential for duplicate matches. Hence the template window size must increase, leading to further increase in computational cost.

The hierarchical techniques try to reduce this cost by first matching large windows and performing the search at a coarse resolution, and refining the match at a finer resolution. The techniques from this class that are considered in this review are respectively due to Wong and Hall [125], Lucas and Kanade [64], Moravec [72], Burt, Yen, and Xu [20], Quam [87], Glazer, Reynolds, and Anandan [40], and Marr, Poggio, and Grimson [43,65].

#### Wong and Hall's approach

Wong and Hall construct a low-pass pyramid from each input image, and match the images at the corresponding levels of these pyramids. The match is performed at all the pixels at each level of the first image pyramid, using direct correlation as the match measure. In their coarse-to-fine control strategy, all the candidate-matches for a pixel whose match strengths exceed a given threshold are projected to the next finer level. At the finest level, the best-match is selected as the one that maximizes the correlation measure. The dimensions of the template windows increase by a factor of 2 between a coarse level and the next finer level, thereby covering the same area in the finest resolution image.

### Lucas and Kanade's approach

Lucas and Kanade [64] propose a hierarchical iterative approach for image-registration. However, their formulation is also suitable for determining the displacements of individual points between two image frames.

The overall scheme consists of using a set of multi-resolution band-pass filtered images and a coarse-to-fine matching strategy. Their method of determining the inter-image transformation parameters of the neighborhood of a point is based on the minimization of the sum of squared difference measure. A gradient-descent approach is employed to iteratively estimate the transformation parameters. A confidence measure based on the magnitude of the image intensity gradient is included as a part of the iterative update scheme. While the general formulation of this approach allows arbitrary linear transformation of an area between the two frames, only its application to stereopsis has been demonstrated. The relative locations and orientations of the camera are unrestricted, although no independent object motion is allowed. The registration process is used to determine the camera transformation parameters as well as the depth of image points.

### Moravec's technique

Moravec's two-level hierarchical algorithm [72] does not involve an explicit creation of multi-resolution images. Instead, the fine-resolution image is sampled in order to compute correlations and to select candidate matches at a coarse-resolution. This approach was chosen in order to enable the implementation of his algorithm using non-specialized hardware. However, the two-to-one pixel sampling used in his algorithm makes his algorithm easily suitable for a pyramid representation scheme.

Moravec restricts the matching process to a set of interesting points in one of the images. The match measure used is a type of normalized-correlation measure which is similar to the variance-normalized correlation measure, but simpler to compute. At the coarse level, each point is represented by the $6 \times 6$ set of samples

from a 12 × 12 image area. The coarse-level search is performed over the entire second image. The search area for the corresponding interest point at the finer level is restricted to the 12 × 12 pixel area that corresponds to the best match window at the coarse-level. All the 6 × 6 windows that fully fit within this 12 × 12 window are considered candidate match windows. Thus, the candidate match points at the finer level are contained in a 7 × 7 pixel area.

An important aspect of Moravec's work is his concern with the computational speed necessary for a practical system. His matching algorithm is used for the rough estimation of the 3-D structure of the environment of a moving robot vehicle. His work also includes a demonstration of the performance of the vehicle.

### The flow-through algorithm of Burt, Yen, and Xu

Burt, Yen, and Xu [20] use a band-pass pyramid of the input images [22] in their hierarchical "flow-through" matching algorithm. At each level of the pyramid, the search area used for each pixel in the first image is the 3 × 3 pixel window centered around the corresponding image position in the second image. The match measure is the variance-normalized correlation of the values in 5 × 5 template windows. Their approach is called a "flow-through" algorithm because the match processes at the different levels operate completely independently of each other. This is justified by the view that the measurement of large image displacements need not be as precise as the measurement of small displacements.

The directional second derivatives of the cross-correlation function at zero displacement are used as confidence measures for the corresponding directional components of a displacement. A method for combining these directional components based on the associated confidence measures is included.

### Quam's hierarchical warp stereo algorithm

Quam's approach [87] is specific to stereopsis, hence the matching is constrained along epipolar lines. A low-pass pyramid is constructed from each input image using Burt's Gaussian pyramid transformation algorithm [22]. A coarse-

to-fine hierarchical matching strategy is used. At each level, the search interval is $(-2, 2)$ pixels. The match measure is the Gaussian weighted variance-normalized correlation of the values in $13 \times 13$ template windows. A match is accepted if it is not at either end of the search interval, and if the best and next-best match are adjacent. A fast surface interpolation algorithm [99] is used for computing the disparity for pixels without reliable matches. After the coarse disparity estimates are projected to the next finer level, geometric warping of the second image is performed to improve the matches.

### The hierarchical correlation algorithm of Glazer, et al.

Glazer, Reynolds, and Anandan [40] describe a hierarchical correlation matching algorithm. Their algorithm uses Burt's Laplacian-pyramid transformation [22] to construct a set of multi-resolution band-pass filters which are applied to each input image. The match measure used is the direct-correlation of the values in $8 \times 8$ template windows. A coarse-to-fine control strategy is used, in which the search area for a child is the $3 \times 3$ pixel area centered around the projected match of its parent. No confidence measure or smoothness constraint is included.

The work of Glazer *et al.* is of special interest here, because it was the starting point of our own research described in this thesis. The essential differences between Glazer's algorithm our algorithm are are that we use a Gaussian-weighted SSD measure with $5 \times 5$ template windows, and an "overlapped-pyramid" projection algorithm for implementing the coarse-to-fine control strategy. In addition, we also include a confidence measure and a smoothness constraint. These modifications will be explained in detail in chapters III and IV.

### The stereopsis algorithm of Marr, Poggio, and Grimson

Marr and Poggio proposed a theory of *human stereo vision* [65], which was later implemented by Grimson [43] as a computer algorithm. In this approach, spatial frequency channels are formed using a set of band-pass filters, although a multi-resolution representation is not used. The band-pass filters are implemented

as convolutions with a set of four $\nabla^2 G$ masks whose center-widths are 4, 9, 17, and 35 pixels. The zero-crossings of the filtered images are used as "edge-tokens" for a symbolic matching process. The search for a match is confined to the corresponding epipolar lines.

Although an explicit multi-resolution image representation is not used, the filtering process has the effect that the distribution of the zero-crossing becomes sparser with increase of scale (or the center-width of the $\nabla^2 G$ mask). A hierarchical coarse-to-fine control strategy is used, wherein the propagation of the disparities from a low- to the next higher-frequency channel are based on the links between zero-crossing segments in the two channels. Grimson computed a sparse depth map from the finest resolution disparity values, and used a surface interpolation algorithm to obtain a dense depth map [42].

Marr and Poggio's theory was one of the earliest attempts in computer vision to recognize the "range vs. resolution" tradeoff involved in matching. The key idea underlying our framework, viz., the separation of the computations according to scale is essentially a product of this type of consideration.

### II.2.5 Relationship of the matching techniques to our framework

Most of the techniques considered in the review above are at least partially consistent with our framework. Table 1 indicates which of the five components of our framework are included in each of the hierarchical techniques and the manner of their implementation.

Although each component of our framework can be found in one or the other of the approaches reviewed here, it appears that no single approach contains all of them and combines them in the manner suggested in chapter I. Almost all the techniques explicitly use spatial frequency channels, while most of them also include a multi-resolution data-structure. The match criteria include different types of correlation measures and edge-token correspondences. The coarse-to-

| Author(s) | spatial freq. channels |
|---|---|
| Gennery (G) | none |
| Barnard and Thompson (BT) | none |
| Wong and Hall (WH) | low-pass pyramid |
| Lucas and Kanade (LK) | band-pass pyramid |
| Moravec (M) | no filters, but subsampling |
| Burt, Yen, and Xu (BYX) | band-pass pyramid |
| Quam (Q) | low-pass pyramid |
| Glazer, Reynolds, and Anandan (GRA) | band-pass pyramid |
| Marr, Poggio, and Grimson (MPG) | $\nabla^2 G$ filters, no pyramid |

| A(s) | match criterion | control strategy |
|---|---|---|
| G | var. norm. corr. | – |
| BT | SSD | – |
| WH | direct corr. | coarse-fine |
| LK | SSD, using grad. desc. | coarse-fine |
| M | normalized corr. | coarse-fine |
| BYX | var. norm. corr. | flow-through |
| Q | var. norm. corr | coarse-fine |
| GORE | direct corr. | coarse-fine |
| MPG | zero-crossing matching | coarse-fine |

| A(s) | confidence measure | smoothness constraint |
|---|---|---|
| G | distrib. of SSD vals. | none |
| BT | mag. of min. SSD | a relaxation alg., only for interest points |
| WH | none | none |
| LK | gradient magnitude | none |
| M | none | none |
| BYX | directional derivs. of corr. | none |
| Q | a binary selection | warping alg. at each level |
| GORE | none | none |
| MPG | binary sel. of edges | interpolation at finest res. |

**Table 1:** The relationship of matching techniques to our framework. The implementation choices for the five components of the framework are indicated. Due to space constraints the table is divided into three portions.

fine control strategy is found in all techniques except that of Burt, Yen, and Xu.

Gennery, Barnard and Thompson, Lucas and Kanade, and Burt *et al.* all include some type of confidence information in their techniques. However only Burt *et al.* use a measure which is orientation-selective. The binary selection involved in the techniques of Quam and Marr, *et al.* serves more as an interest operator, and cannot be regarded as a scalar valued confidence measure. While the use of such an interest operator may speed up computation and can reduce the number of false-matches, it also causes the algorithm to be non uniform across the pixels, and does not produce a dense displacement field. Moreover, a binary valued confidence measure is usually hard to compute and often involves ad-hoc choices for the interest point selection algorithm and the various thresholds used in such algorithms [72]. The correct match of an interest point cannot be guaranteed to be among the restricted set of candidate matches. Taken together, these considerations suggest that for uniform pixel-parallel processing, it may be advantageous to allow the matching process to occur at all pixels and then compute a confidence measure for each match.

Most of the techniques do not include a smoothness constraint to obtain a dense displacement field. Although Barnard and Thompson's relaxation algorithm is based on a smoothness assumption, only the displacements of the interest points are modified. On the other hand, Quam, and Grimson both include an interpolation process, wherein the sparse set of reliable matches are retained and used to fill in the disparities for the other pixels.

It appears that no smoothing algorithm that uses a directional confidence measure has been employed for the computation of dense displacement fields using a matching approach. This is in contrast to the gradient-based approaches wherein the smoothness constraint is almost always used. As noted in chapter I, an important aspect of our approach is the inclusion of a confidence measure and a smoothness constraint as essential parts of our computational framework.

### II.2.6 Matching based on complex image structures

Matching based on more complex symbolic templates (e.g., regions, lines, etc.) is less common. Two examples of this type of work are due to Tsuji, Osada, and Yachida [107], and Jacobus, Chien, and Selander [53]. Tsuji's approach is applied to cartoon line drawings. Hence the problem of region and boundary extraction is simplified. Each input image is divided into regions, each region being described in terms of the location of its centroid, its perimeter and a list of arcs (its boundary and internal edges). The matching process consists of two stages. First, for each region in the first image, a region is selected from the second image as its best match. Second, the arcs composing the two regions are matched against one another using what is called a "flexible template matching" method.

Jacobus *et al.* use a more complex graph extraction and matching technique, and apply their algorithms to intensity images of single objects obtained from different viewpoints. These graphs, which are called "half-chuncks", encode regions, their bounding contours and edges, and vertices. Image-region statistics are used to create symbolic "features" associated with the nodes of the graph and are also encoded as a part of that representation. A half-chunk graph representation is created for each image and matching is done on these graphs. The feature information is used determine the a match strength and the relationships between the nodes in the graph is used for checking the consistency of matches.

Although matching based on complex symbolic structures can be robust, in practice the determination of symbolic structures suitable for matching has remained a difficult problem. This is because, the problem of extracting such symbolic tokens is similar to the problem of image segmentation, which has remained as one of the more difficult problems in computer vision. Typically, the symbolic structures tend to be unstable as the imaging conditions vary, thereby causing the matching process to be unreliable. In addition, matching schemes are not

easily suitable for parallel processing. Hence, the symbolic matching processes tend to be computationally expensive and do not appear to be suited for early visual processing.

## II.3 SPATIO-TEMPORAL ENERGY MEASUREMENT

### II.3.1 The principles underlying the energy models

Recently, there have been a number of separate efforts among psychophysicists to design models of the human motion measurement system. The different models and their relationships to each other have been described in detail in [2,114,117]. Of particular interest here is the article by Adelson and Bergen [2] which provides a clear and simple unified view of all these models, and groups them under the title "spatio-temporal energy" models. The brief review given here is largely based on that paper.

The spatio-temporal energy models for motion analysis are founded on the view that the visual signal is a function of $(x, y, t)$. For example, consider the the movement of a black rectangle in a white background as shown in Figure 3. This figure shows three views of the same phenomenon. In Figure 3a, the rectangle in shown on the $(x, y)$ plane and is indicated (by the arrow) as moving to the right. In Figure 3b, a 3-dimensional spatio-temporal image cube is shown and what is seen has the shape of a parallelepiped. Figure 3c shows the image on projected on the $(x, t)$ plane by a particular scan-line (i.e., fixed $y$ position) of the figure on the $(x, y)$ plane. It is obvious that as the speed of the rectangle increases, the "vertical" sides of the parallelogram on the $(x, t)$ plane becomes more slanted. It is also apparent that if the direction of motion is to the left, these sides are slanted leftwards and for a stationary object, the projection on the $(x, t)$ plane is a rectangle.

If the image is considered to be a one-dimensional signal and it moves only

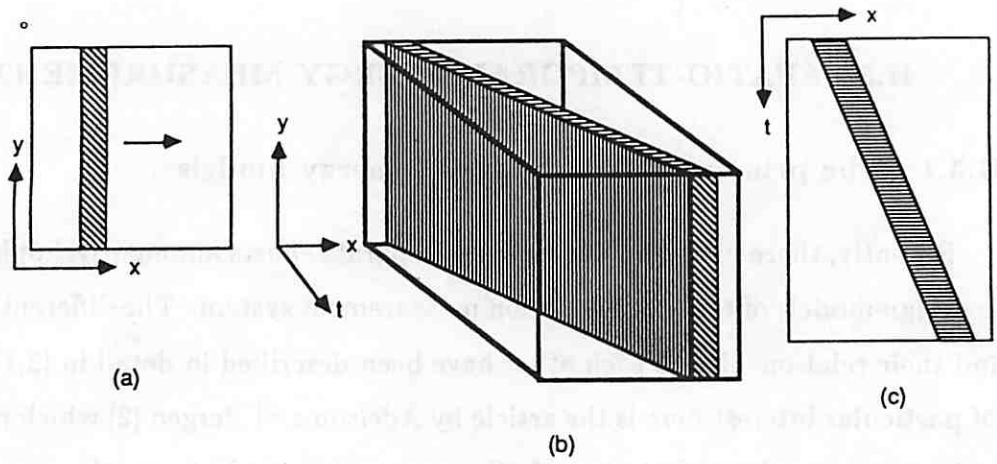



**Figure 3:** Movement of a bar shown in the $(x, y, t)$ domain: (a) the traditional view from the $(x, y)$ plane, (b) the movement in the spatio-temporal image "cube", and (c) the projection on the $(x, t)$ plane.

along that dimension, then it can be completely represented on a single spatio-temporal plane. For instance, consider a restricted case of two-dimensional signals consisting only of grating patterns where the gratings are parallel to the $y$ direction. If this pattern is allowed to move only in the $x$ direction, then the projection of the 3-dimensional space-time cube onto the $(x, t)$ plane is sufficient to describe the signal. In this one-dimensional case, the problem of measuring motion is equivalent to the problem of determining the orientation of the lines in the $(x, t)$ plane. The slope of the line on the $(x, t)$ plane is the velocity of the corresponding image-point along the $x$ direction. Even if the motion is not uniform, i.e., the velocity does not remain the same over time, at any instance of time, the slope of the curves traced in the $(x, t)$ plane indicates the $x$ component of the velocity. The theme common to all spatio-temporal energy models is simply the determination of this slope. Such a slope-detection problem is not unlike standard edge detection problems in static images and can be achieved through the use of simple linear-filters that are sensitive to spatio-temporal "edges" along specific orientations.

Since the maximum rate of change in intensity at a point along a line occurs in the direction perpendicular to the line, the "energy" (which is proportional to the rate of change) will also be maximum along that direction. Hence, it is also possible to regard the slope detection mechanisms as methods of measuring the energy of the signal along different directions. The various energy models that have been described in the psychophysics literature usually consist of two or three detectors: one to detect signals moving to the right, one to detect signals moving to the left, and sometimes one that detects stationary signals.

An alternate way of describing the pattern on the $(x,t)$ plane involves considering the Fourier transform of the signals on that plane. Consider a sinusoid of wavelength $\lambda$ moving at velocity $v$. This can be mathematically described as

$$s(x,t) = C\cos\frac{2\pi(x-vt)}{\lambda}$$

where it is assumed (according to usual conventions) that the initial phase of the sinusoid is zero. From the above definition, it is immediately clear that the temporal frequency of the sinusoid $f_t = \frac{2\pi v}{\lambda} = f_s v$, where $f_s = \frac{1}{\lambda}$ is the spatial frequency of the signal. In general, the spatio-temporal Fourier transform of a moving signal will have the property that the temporal frequency component due to each moving spatial frequency component satisfies the above equation. In short, all the energy in the spatio-temporal frequency domain will be concentrated along the line through the origin shown in Figure 4. This line has a slope that is proportional to $v$.

Although the assumptions necessary to enable this type of frequency domain analysis – i.e., uniform motion, constant velocity across the entire signal, etc. – are usually not valid across the entire image, this approach does provide an alternative way of thinking about the measurement of motion. Moreover, if attention is restricted to a small spatial extent and to a small time interval, the above analysis may serve as a key for designing computational models.

The remainder of this section reviews two specific examples of spatio-temporal

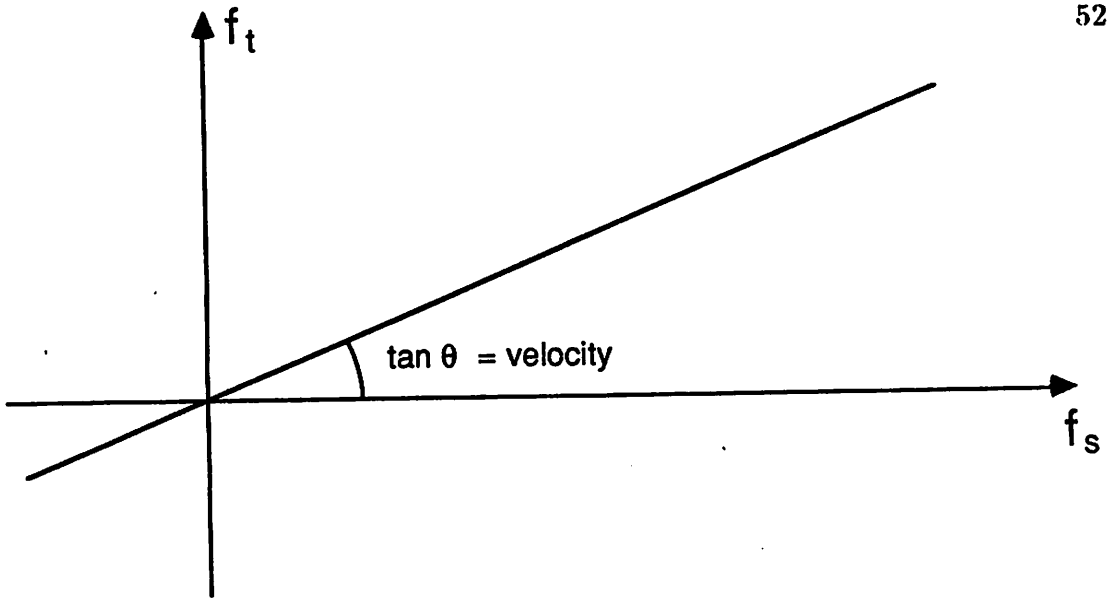



**Figure 4:** The concentration of spatio-temporal energy due to a moving pattern

energy models. Both models assume that the input is preprocessed using a spatial frequency band-pass filter. Following this, the application of the energy models to real image sequences is discussed.

### II.3.2 Examples of energy models

Adelson and Bergen suggest the use of oriented 2-D Gabor filters for the measurement of spatio-temporal energy. Each motion detector actually consists of two filters, whose impulse responses are 90° out of phase, e.g., a sine and a cosine wave under a spatio-temporal Gaussian envelope. The responses of these two are squared and summed in order to obtain the energy associated with a particular direction of movement. Three such detectors are used to measure leftwards motion-energy, rightward motion-energy, and stationary energy.

Van Santen and Sperling [114] suggest a slightly different scheme which they call "elaborated Reichardt detectors". This scheme uses two receptors which are located slightly apart and are tuned to a small range of spatial frequencies. The output from each unit is multiplied with the delayed output from the other unit

and difference between the temporal averages of the two products is computed. Van Santen and Sperling show that with the proper choice of the temporal delay filters and the temporal averaging scheme, the sign of this difference measure will indicate the direction of motion (left or right).

Adelson and Bergen also note that the two models described above can be shown to be mathematically equivalent to each other, although they differ in their exact forms. There is also an energy model due to Watson and Ahmuda [117] which is in many ways similar to the model of Adelson and Bergen. It has also been shown to be equivalent to the other two described above.

### II.3.3 Applying energy models to image-sequences

All of the spatio-temporal energy models that have been considered have been with respect to an one-dimensional signal. As such these models are not applicable to 2-dimensional real images. However, these models may be used by first decomposing the image into a set of one-dimensional signals. One way to achieve such a decomposition is by using a set of orientation-selective filters in the space domain. Each orientation selective channel can measure the one-dimensonal motion component along its orientation; following this the results from the different orientation channels can be combined to determine the true 2-dimensional motion at that image location.

Such an idea can be found in the techniques of Heeger [47], and Watson and Ahmuda [117]. Heeger uses a set of 3-D Gabor filters to measure the energy in a small region of the spatio-temporal frequency space. He also provides a schemes for recombining the motion information obtained from these different energy measurements, and shows preliminary results for synthetic data. Watson and Ahmuda use a set of input linear-filters tuned to specific orientations and combine the results using a constraint that all the channels measure the various directional components of the same image-velocity.

The last issue of concern is temporal aliasing. When the energy models are used on a discrete image sequence, if the sampling rate it not high enough, the temporal aliasing (i.e., inter-frame displacements larger than one pixel) can cause incorrect estimation of the direction of motion. As noted by Adelson and Bergen, it is the presence of the low spatial frequency information that appears to allow for avoiding mistakes due to aliasing. This observation also indicates that the kind of hierarchical approach described in our framework would be useful for the measurement of visual motion. However, as yet there appears to be no effort to use such a hierarchical approach in conjunction with the energy models. In chapter VI, we extend our framework to include the spatio-temporal energy models and propose a unified framework which appears suited for connectionist models of computation.

# CHAPTER III

# TWO BASIC HIERARCHICAL ALGORITHMS

The first two chapters described our computational framework and explained its relationship to some of the current techniques in computer vision. As noted in chapter I, our research has also included the development of an integrated system consistent with our framework. This chapter and chapter IV together contain a complete description of our system. This description consists of detailed explanations of the implementation choices made for the five components of our framework. In this chapter, the implementation choices for the spatial frequency decomposition, the match criterion, and the control strategy are described. The descriptions of the confidence measure and the smoothness constraint are contained in chapter IV.

Any algorithm consistent with a computational framework will also be influenced by the architecture of the machine for which that algorithm is designed. Since as noted in chapter I, pyramid representations and computations are naturally suited for our framework, our first choice is a pyramid architecture. However, no general purpose pyramid machine is currently in existence, although limited efforts can be found in [113,101]. On the other hand, there are a number of current efforts [118] to build a simple mesh connected computer (MCC). Therefore, we have also modified our pyramid-algorithm to be suitable for an MCC. This chapter contains a description of the pyramid algorithm first, and following that a description of the MCC algorithm. Since the two algorithms are largely similar to each other, the description of the MCC algorithm primarily involves noting the modifications made to the pyramid algorithm.

## III.1 An Algorithm for a Pyramid Processor

Although no general purpose "pyramid machine" has yet been built, it has remained a useful conceptual model for computation for a number of years. The earliest references to such a machine can be found in the works of Hanson and Riseman [46], Tanimato and Uhr [103], Tanimato and Pavlidis [102], and Klinger and Dyer [57]. The related notion of an abstract recognition hierarchy has been discussed by Uhr [108,109].

Figure 5 shows the arrangement of processors in a typical pyramid architecture; the dimensions of the processor arrays typically decrease by a factor of 2 between adjacent levels of the pyramid. The levels of the pyramid are numbered ($l = 0, 1, \ldots$), where at any level $l$, the size of the processor array[1] is $2^l \times 2^l$. While processing a digitized image of resolution $N \times N$, where $2^{L-1} < N \leq 2^L$, it is usual to regard level 0 as the "top" of the pyramid and level $L$ as its "bottom".

In a general-purpose pyramid machine, each processor has the capabilities that are typical of any microprocessor; the processor array at each level is a Single-Instruction-Multiple-Data (SIMD) type machine. Apart from processors at its own level, the neighborhood of each processor also contains those at its adjacent levels. Each processor is called a "parent" of a set of processors at the next finer level, which are its "children". In the traditional version of the pyramid, this relationship is strictly one-to-four, and known as the "quad-tree" connectivity scheme. However, there are also *overlapped pyramids* (e.g., see [21]), in which the set of children for two nearby parent pixels may overlap. In the algorithms described here, the connectivity between processors at adjacent levels is explicitly specified wherever necessary.

[1]This numbering scheme, which is used throughout this dissertation follows that of [46]. Although other numbering schemes can also be found in the literature [21,103], the transformations between the different schemes are straightforward.

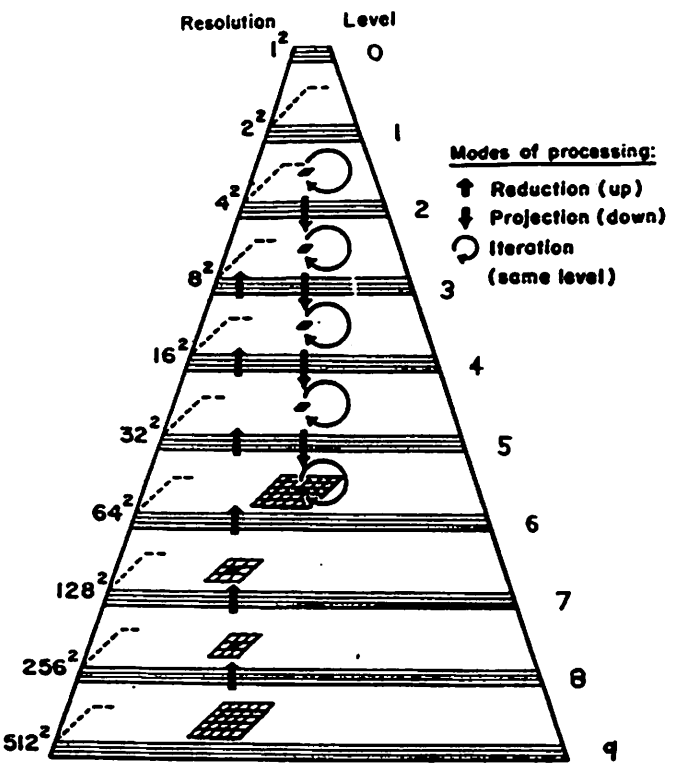



**Figure 5:** The pyramid architecture

### III.1.1 Spatial frequency decomposition and representation

The development of computational algorithms for creating a hierarchical set of band-pass filters has been addressed by a number of researchers [22,28,41,43,66] in computer vision. A detailed discussion comparing different methods of creating band-pass filter pyramids can be found in the recent work of Glazer [41]. Based on a combination of theoretical considerations regarding aliasing effects and practical considerations regarding ease of implementation and efficiency, Glazer chooses a version of Burt's Laplacian-pyramid transform as one of his methods of constructing a band-pass pyramid. Due to the same considerations, we have also chosen Burt's Laplacian-pyramid, although our version of Burt's algorithm differs slightly from that of Glazer. These differences are explained below.

Burt's algorithm for constructing a band-pass pyramid consists of two stages: the first stage involves the construction of a Gaussian low-pass filter pyramid from the input image, while the second stage involves computing the difference between the images at the adjacent levels of the low-pass pyramid to obtain the set of band-pass filtered images.

#### Gaussian pyramid

The finest-level $L$ of the Gaussian pyramid contains the input image. The image at any level $l = L - 1, \ldots, 0$ is constructed by applying a low-pass filter to the image at level $l+1$ and subsampling the filtered image. The low-pass filtering process is achieved through convolution with a localized weighting function, which has non-zero values for only a small number of pixels. The sampling is done simply by selecting every alternate row and column and ignoring the others. Equivalently, the value at each pixel at level $i$ can be regarded as a weighted sum of the values of a small number of pixels below it. Figure 6 illustrates this process for a one dimensional function.

Burt's analysis of different weighting functions suggests that the best approximation to a Gaussian is obtained when windows with "odd" (say $5 \times 5$) dimensions

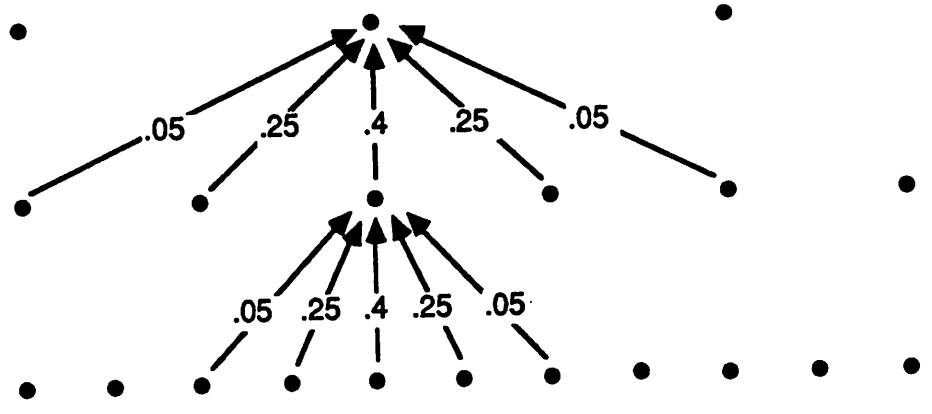



**Figure 6:** The reduce operation in the Gaussian pyramid – the 1-D case.

are used, and for a $5 \times 5$ window, with the following convolution mask.

$$G_5 = \frac{1}{400} \times \begin{bmatrix} 1 & 5 & 8 & 5 & 1 \\ 5 & 25 & 40 & 25 & 5 \\ 8 & 40 & 64 & 40 & 8 \\ 5 & 25 & 40 & 25 & 5 \\ 1 & 5 & 8 & 5 & 1 \end{bmatrix}$$

When this mask is used along with a two to one reduction in resolution, the standard deviation of the Gaussian is approximately 0.56 pixel units at each level.

Note that the weighting function $G_5$ given above is "separable", i.e., it can be expressed as a product of two one dimensional weighting functions,

$$\frac{1}{20}[1\,5\,8\,5\,1] * \frac{1}{20}[1\,5\,8\,5\,1]^T .$$

The convolution with a $w \times w$ separable mask can implemented by an algorithm requiring $O(w)$ steps instead of $O(w^2)$ steps required for a non-separable mask of the same size. This is achieved by first performing a one-dimensional convolution along the $x$ direction (or rows), followed by a one-dimensional convolution along the $y$ direction (or columns).

The effective width of the Gaussian used in Burt's Gaussian pyramid increases as the level-number decreases. In particular, the standard deviation of the Gaussian at any level $l$ is approximately $0.56 \times 2^{N-l}$ pixels, where a "pixel" is measured at the finest-level $L$. It is well known that the frequency-response of a Gaussian weighting function is also Gaussian and that the standard deviations in the space and the spatial-frequency domains are inversely related. This means that in the frequency domain, the standard deviations of the Gaussian-filters decrease by the factor 2 between adjacent levels of the Gaussian-pyramid. Figure 7 shows the power-spectrum of the continuous analogs of the one-dimensional Gaussian filters at the first two levels of the Gaussian pyramid.

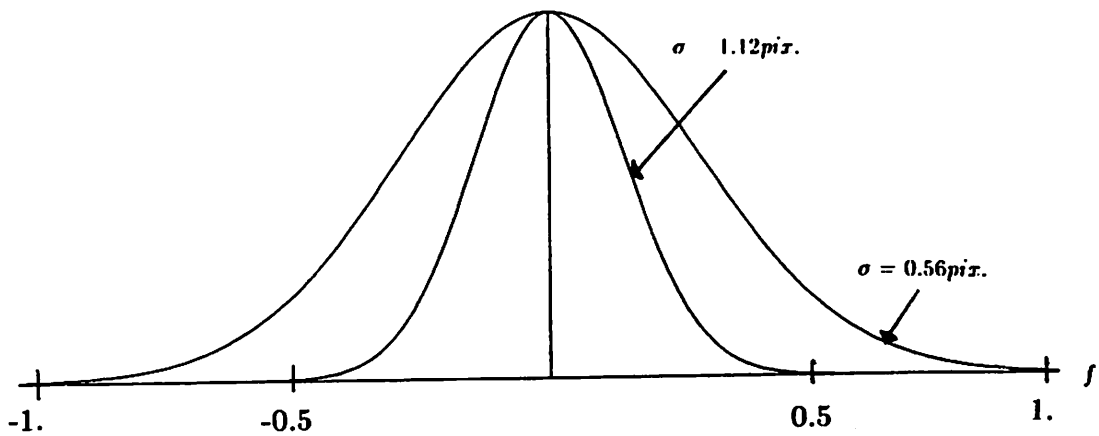



**Figure 7:** The power-spectrum of the effective convolution masks at the two finest levels of the Gaussian pyramid: the one dimensional case.

Finally, it should be noted that Glazer uses the following $4 \times 4$ weighting function in his experiments

$$G_4 = \frac{1}{64}\begin{bmatrix} 1 & 3 & 3 & 1 \\ 3 & 9 & 9 & 3 \\ 3 & 9 & 9 & 3 \\ 1 & 3 & 3 & 1 \end{bmatrix} = g_4 * g_4^T$$

where

$$g_4 = \frac{1}{8}[1\,3\,3\,1]$$

The primary motivation for Glazer's choice appears to be the fact that $g_4$ can be expressed as a recursive convolution $g_1 * g_1 * g_1$, where $g_1 = \frac{1}{2}[1\,1]$. The convolution of the image with $g_1$ can be implemented with no multiplications and division by a power of 2, which implies that the architecture of the underlying machine can be fairly simple. For our algorithm, we have chosen the $5 \times 5$ weighting function because, as noted above, the odd weighting functions appear to be closer approximations to Gaussian functions.

### Band-pass pyramid

Burt's approach to the construction of band-pass pyramids is based on taking the difference between the images at adjacent levels of the Gaussian pyramids. This is equivalent to using a difference-of-Gaussians (DOG) filter, which is a type of band-pass filter. It is also known [66] that the DOG filter is a close approximation to the Laplacian of Gaussian ($\nabla^2 G$) filter. The band-pass pyramid constructed in this manner is also called the *Laplacian-pyramid*.

Since the resolution of the images at the adjacent levels of the Gaussian pyramid are not the same, the difference between the two images cannot be directly computed. Instead, the coarse-level image at level $l-1$ is first projected to level $l$. The band-pass filtered image at level $l$ is obtained by computing the differences between corresponding pixels of the image at level $l$ of the Gaussian pyramid and the image projected from level $l-1$.

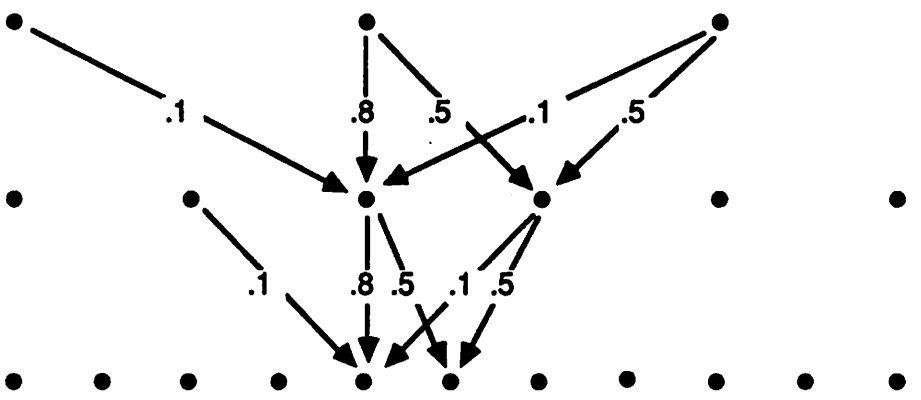



**Figure 8:** The project operation in the Laplacian-pyramid.

The projection of an image from level $l-1$ to level $l$ involves performing an interpolation of the values of the pixels at level $l-1$. In Burt's approach, the weights for the interpolation process can also be expressed as a separable product of two one dimensional weight masks. For any pixel $p$ at level $l$, the contribution from pixel $q$ at level $l-1$ is proportional to the contribution received by $q$ from $p$ during the construction of the Gaussian pyramid. Since the odd-pixels are eliminated during the sampling process, during the projection, the weights for the odd and even pixels appear to be different from each other. Figure 8 illustrates the one-dimensional version of the projection operation.

The interpolation process involved in the projection actually corresponds to performing additional smoothing on the Gaussian-smoothed image at level $l-1$. Therefore, the ratio of the frequency-domain widths of the low-pass filters at adjacent levels of the pyramid is greater than 2. This means that at any level, the width of the band-pass filter is slightly more than an octave. For more details regarding these filters, see [22,23].

For the purposes of illustration, Figure 9 displays the four finest levels of the Gaussian and the Laplacian pyramids computed from a real input image. The input image is the $128 \times 128$ pixel resolution image shown at the finest level of the Gaussian-pyramid.

### III.1.2 Match criterion

As noted in chapters I and II, our framework allows a variety of choices for the match criterion within each channel. For our algorithm, the minimization of Gaussian-weighted sum of squared differences (SSD) of $5 \times 5$ windows was chosen as the match criterion. The motivation for our particular choices for the match-measure and the window sizes are explained below.

#### Choosing the match measure

The choice of the correlation of the intensity values as a match criterion was

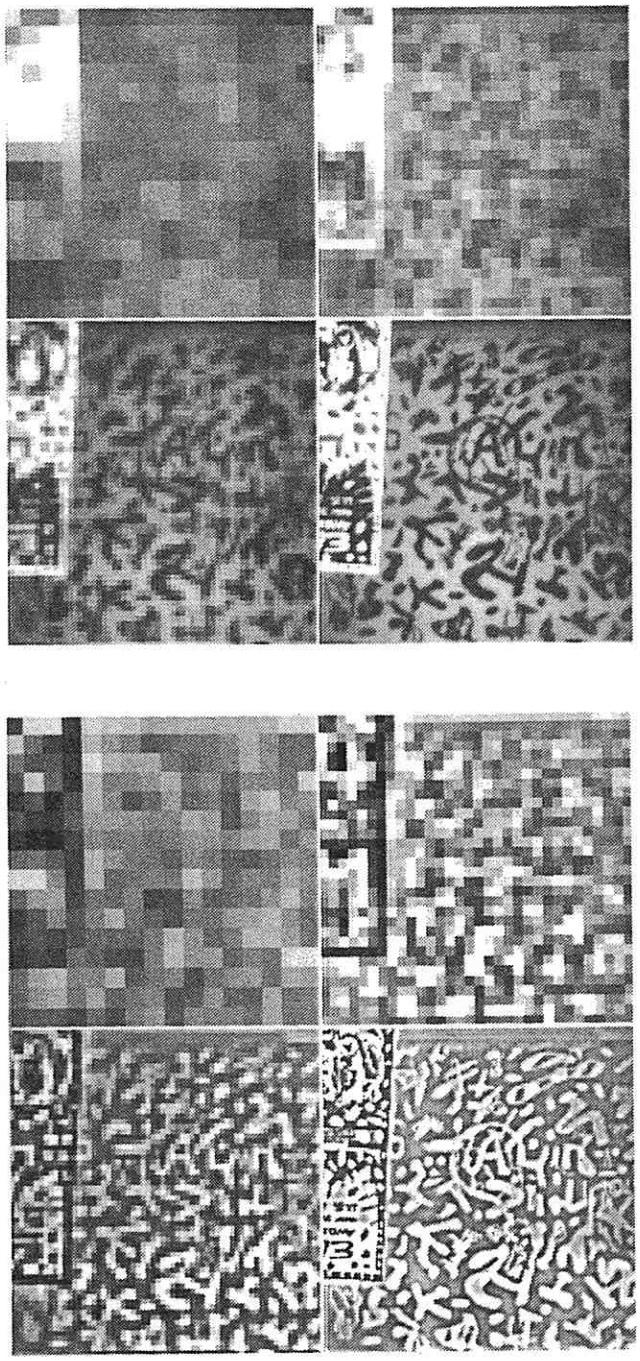


**Figure 9:** The four finest levels of the Gaussian and the Laplacian pyramids computed from a real image. The top figure contains the Gaussian pyramid, while the bottom pyramid displays the Laplacian pyramid. The input image is the finest-level image shown at the bottom-right quadrant of the Gaussian pyramid.

motivated by its simplicity of definition and ease of computation. The choice of the SSD measure over other correlation measures noted in chapter II, viz., direct correlation, mean-normalized correlation, variance-normalized correlation, and the sum of absolute-differences (SAD), was based on a comparison of the cost of computing these different measures, as well as a comparison of their performances. Our evaluation of the performances of the different correlation measures was in turn based on a comparative study of these measures by Burt, Yen, and Xu [19].

Among the different measures mentioned above, the variance-normalized correlation measure is usually the most reliable one, since it is insensitive to changes in the mean-intensity and the contrast between frames [44]. When compared in terms of the complexity of computation, the direct correlation, sum of squared differences (SSD), and the sum of absolute differences (SAD) require fewer operations than variance-normalized correlation. The cost of performing mean normalized correlation lies somewhere in between.

The comparative study of Burt *et al.* indicates that after band-pass filtering, simple direct correlation measure and mean-normalized correlation are nearly as reliable as variance-normalized correlation. As noted earlier, the process of band-pass filtering can also be regarded as subtracting from the value of a pixel the weighted average of its neighbors. Therefore, the mean-value of template windows in the band-pass filtered image tends to be small, provided the windows are larger than the weighting functions used during the filtering process. Hence, performing direct-correlation on the band-pass filtered image is similar to performing mean-normalized correlation, while being less expensive.

As noted in Appendix A, for small search windows, SSD is closely related to direct correlation and hence shows a similar performance. Besides, the SSD measure is always guaranteed to be positive, a fact which will be utilized during the normalization of our confidence measures (to be explained in chapter IV). The SAD measure shows a performance similar to the SSD measure, and is slightly less expensive. However, the absolute-value function has discontinuous derivatives at

zero, thereby making a mathematical analysis of the SAD measure difficult. Since the computation of our confidence measure uses the derivatives of the match-measure, the SSD measure is preferable. Hence, for our algorithm, the SSD measure has been chosen as the match measure.

### Choosing the shape and size of the window function

The computation of the SSD measures involves averaging the squared difference values over the template window, i.e.,

$$S(x,y,\delta x,\delta y) = \sum_{i,j=-n,n} W(i,j)\,[I(x+i,y+j) - J(x+\delta x+i,y+\delta y+j)]^2$$

This averaging process can be regarded as applying a low-pass filter to an image containing the square of the differences between the intensities of the corresponding pixels in the two input images. The weighting function $W(i,j)$ determines the shape of the low-pass filter. The shape of the Fourier spectrum of this filter indicates the influence of the size and shape of the template window on the match process.

The study by Burt *et al.* [19], which was referred to in the previous section, also included an analysis of the shape of the window function and its Fourier transform. The two significant properties of the Fourier transform of a particular correlation window are its *bandlimit* and the reduction of the *side-lobes*. Based on an empirical analysis, Burt *et al.* noted that if most of the significant spectral energy of the matched-images are above the bandlimit of the window function, the error-rate in the matching results are low. This suggests that larger and more broadly shaped template windows are likely to yield superior performance.

Burt *et al.* recommended the repeated application of the hierarchical set of $5 \times 5$ weight functions used in the Gaussian pyramid to create effective correlation windows of varying sizes. For instance, it can be shown that two applications of the $5 \times 5$ window is equivalent to the use of $13 \times 13$ Gaussian window.

For our purposes, the choice of the window function should be based on balancing between two mutually opposing sets of considerations. On the one hand, it is intuitively clear that a larger window is more likely to be representative of a pixel than a smaller window. On the other hand, the window sizes should be small in order for the local-translational approximation to be valid and for reducing the computational cost of the algorithm. For most of our experiments, the compromise chosen was a single application of the $5 \times 5$ Gaussian window.

Appendix A includes a simple ideal-case analysis of the influence of the correlation window size on the range of displacements over which reliable measurements can be obtained. This analysis also suggests that the $5 \times 5$ window may be sufficient for maximizing the displacement-range. The theoretical analysis of a more general case appears to be considerably more complex, and to our knowledge such an analysis has not been described in the image-processing literature.

### III.1.3 Control strategy

The control strategy used in our algorithm is the coarse-to-fine sequential processing of the images in the band-pass pyramid. An important aspect of this strategy is the scheme used for projecting the displacements from a coarse-level to its adjacent finer-level. In our algorithm, an *overlapped pyramid* projection scheme was used. The motivation behind our projection scheme and the details of our control strategy are explained below.

If the maximum image displacement is $\delta$, the processing should begin at level $C = L - \log(\delta)$ to ensure that no pixel moves larger than one pixel distance at that level of the pyramid. Although $\delta$ is usually not exactly known, it is sufficient to obtain an upper-bound for $\delta$. Since there is no simple automatic method of determining this upper-bound, it must be specified by the user.

The processing begins at level $C$ and proceeds via sequential projection to the image level $L$. At the coarsest level, the search area is the $3 \times 3$ pixel area

centered around the corresponding pixel location in the second image. At all other levels, an initial set of displacements for a pixel are obtained by projecting the displacements from the adjacent coarser level. The search area is the $3 \times 3$ area surrounding this projection.

The traditional approach (e.g., see [40]) used for the projection of displacements is based on the quad-tree type of connectivity between adjacent levels of the pyramid. As shown in Figure 10, the displacement at a pixel at level $l - 1$ is projected to the four pixel below at level $l$. However, in this scheme, if the displacement computed for a coarse-level pixel is incorrect, the search areas of its descendants at all subsequent levels will not contain their correct matches. Hence, a single match error made at a coarse level causes a large block of pixels at the image resolution to have incorrect displacements.

The problems due to the quad-tree connectivity scheme are most severe for the pixels which lie along a line (or contour) connecting discontinuities in image motion. The displacement of the coarse-level which straddles such a boundary can be correct for at most one of the two neighboring images areas. If it is incorrect for both, then we have the same problem stated above; i.e., the search areas of the descendants of that coarse-level pixel will not contain their correct matches. Even if it is correct for one of the two areas, the search areas for the descendants contained in the other image area will be incorrect. Thus, the quad-tree connectivity scheme is guaranteed to generate incorrect displacements for at least some of the pixels around boundaries of image motion.

To overcome some of these problems due to the quad-tree connectivity scheme, we have used an an alternate scheme for projection of displacements, called the overlapped pyramid projection scheme. This scheme was suggested by by Burt, Hong, and Rosenfeld [21] in connection with the problem of image-segmentation. The information from a pixel at a coarse-level $l$ is transmitted to all the pixels in a $4 \times 4$ area at the next finer level $l + 1$. Thus, each pixel at level $l + 1$ obtains information from 4 pixels at level $l$, and can be regarded as having four

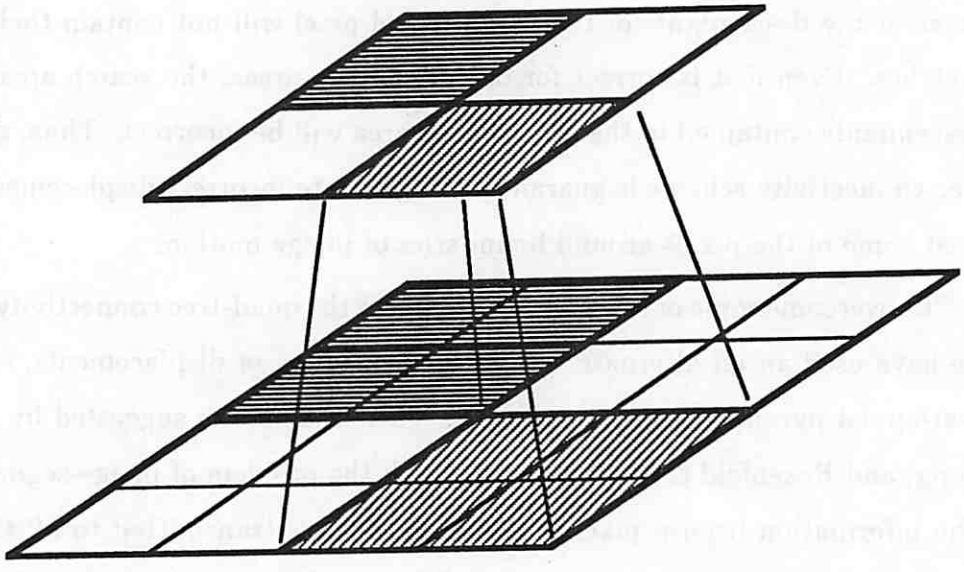


**Figure 10:** The quad-tree connectivity of the usual pyramid projection schemes

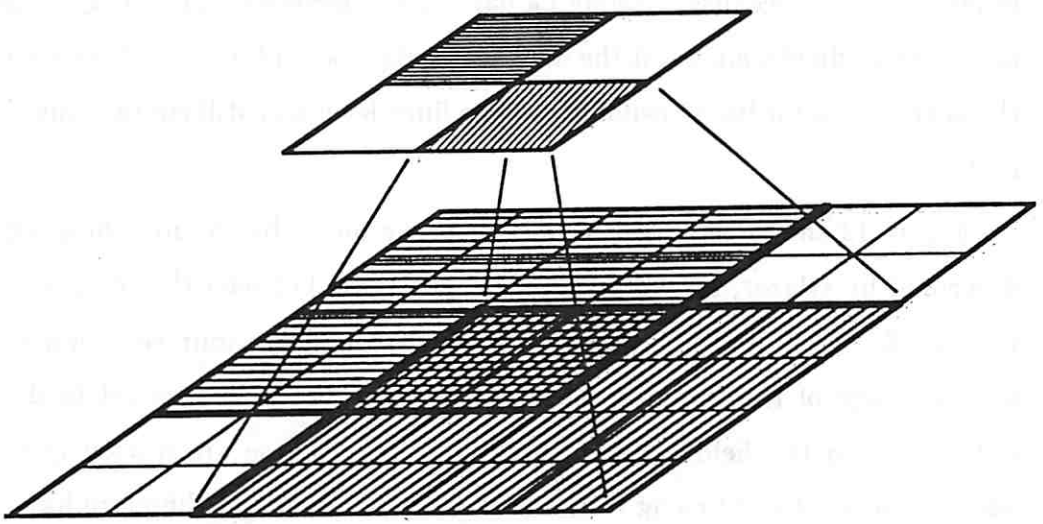


**Figure 11:** The overlapped pyramid projection scheme

potential parents. For our application, the information transmitted is the set of displacement vectors. The displacements of each of the four parents are considered possible initial estimates for the search at level $l$; often, however, two are more of these estimates will be identical. The search area consists of the union of the $3 \times 3$ areas centered around each distinct coarse-level estimate, and the SSD measure is minimized over all the pixels in this expanded search area.

Although the use of the overlapped pyramid implies that the search area size may increase by up to a factor of 4, the resulting scheme is considerably more reliable. This is because, although a particular coarse-level pixel may be assigned an incorrect displacement, if the displacements of any of its neighbors are correct, the search area for its descendants at the finer-level will still contain their correct matches.

Figure 13 shows the displacement field computed by the matching algorithm described by Glazer, Reynolds, and Anandan [40] between the images shown in Figure 12. As it is evident from this figure, there are four rectangular blocks on the image of the textured background where the displacement field appears different from the field in its surrounding area. These are obviously incorrect because these blocks belong to the background and should therefore have similar displacements as their neighbors. Three of these errors are due to the propagation of isolated mistakes made at coarse-levels, while the fourth area is perhaps due to the presence of discontinuities in image motion. In addition, while it is clear from the images that the boundary of the frontal-object is rectangular, the boundary seen in the displacement field is not.

Figure 14 displays the results of using the overlapped pyramid projection scheme. As it is evident from this figure, most of the errors have been removed, and the shape of the boundary is more definitely rectangular.

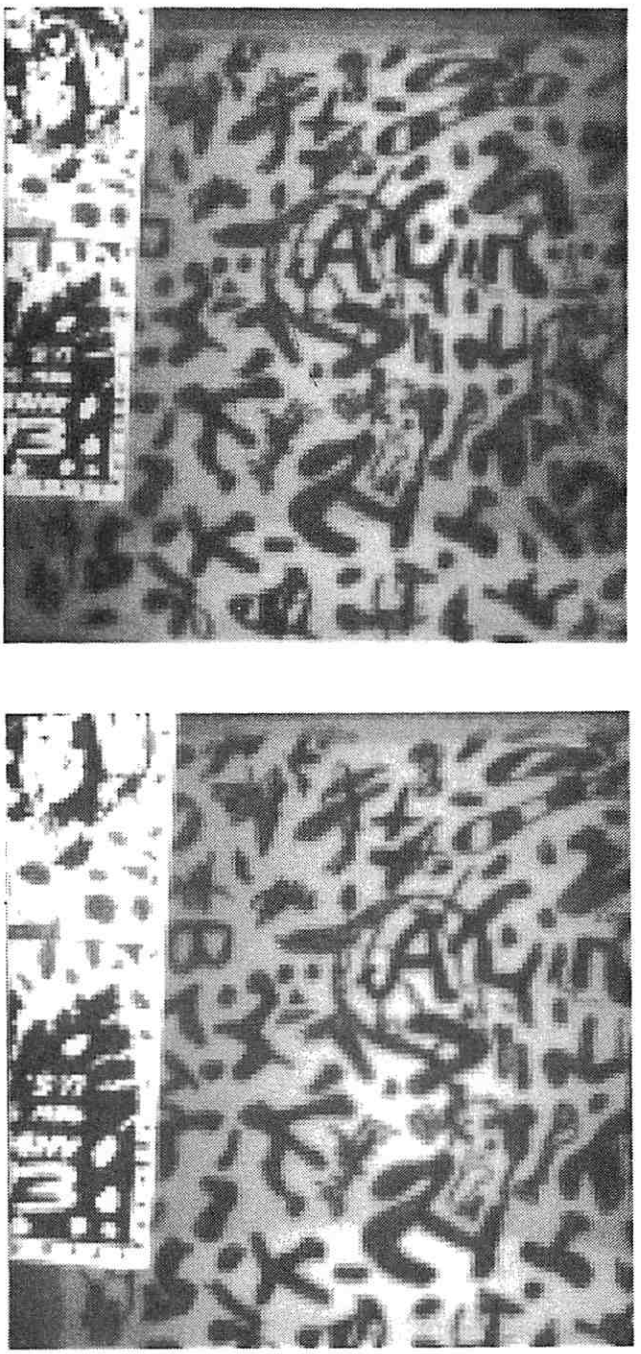


**Figure 12:** The poster input

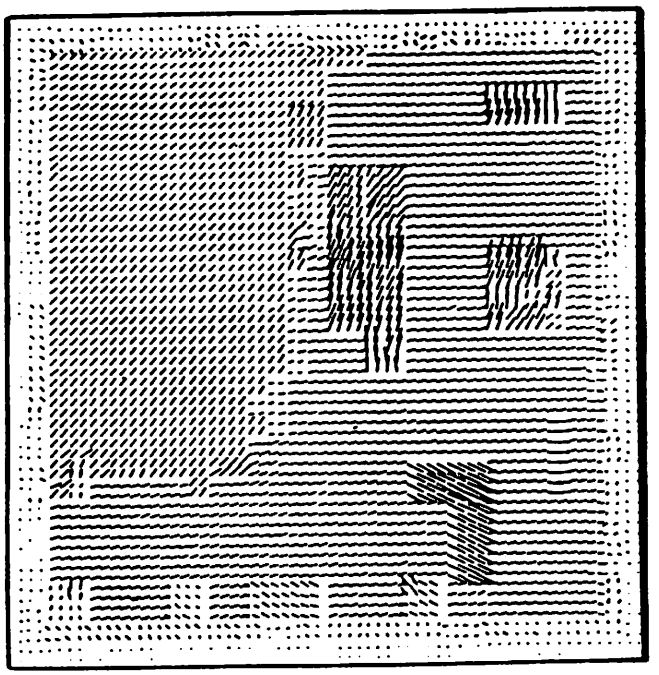


**Figure 13:** The finest level displacement field computed from the poster images, by our algorithm using the non-overlapped projection scheme.

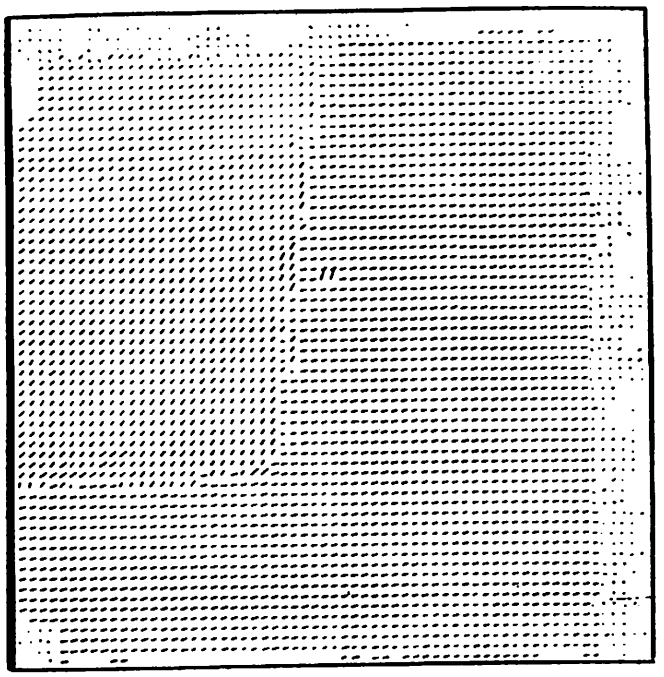


**Figure 14:** The finest level displacement field computed from the poster images, by our algorithm using overlapped projection scheme.

### III.1.4 Complexity analysis

In our pyramid algorithm, the filtering process involves convolutions with $5 \times 5$ masks at all levels. Due to the overlapped projection scheme, the number of candidate matches is between $3 \times 3 = 9$ and $4 \times 3 \times 3 = 36$. The sizes of the template windows are also $5 \times 5$ at all levels. Thus all these parameters are independent of the level number and the maximum displacement. The number of levels is $\log \delta$, where $\delta$ is the maximum image-displacement along either coordinate direction. Hence, the complexity of the pyramid algorithm is simply $O(\log \delta)$.

## III.2 An Algorithm for a Mesh Connected Computer

The pyramid algorithm is interesting because it is highly efficient; however, it relies on rapid communication between distant processors. Such a general purpose pyramid processor is not yet commercially available. On the other hand, a mesh connected computer (MCC) with local connectivity is likely to be available in the near future, since a number of efforts towards this end are currently under way. These include the CLIP-4 and CLIP-7 processors constructed at the University of London [112], the Massively Parallel Processor (MPP) at NASA built by Goodyear Aerospace [14], and the Connection Machine at MIT [50], which are currently in existence, as well as the NON-VON at Columbia University [98], and the CAAPP at the University of Massachusetts, [118], both of which are in advanced stages of development. A detailed review of such machines and parallel processing architectures in general can be found in [118].

The definition of MCC provided by Miller and Stout [71] is used here. The MCC is composed of an array of processors arranged in an $n \times n$ matrix (see figure 15). Each processor has a unique identification number representing the address of that processor in row-major form. From this number, the absolute $x$ and $y$ address of the processor can be computed. Each processor has a constant number

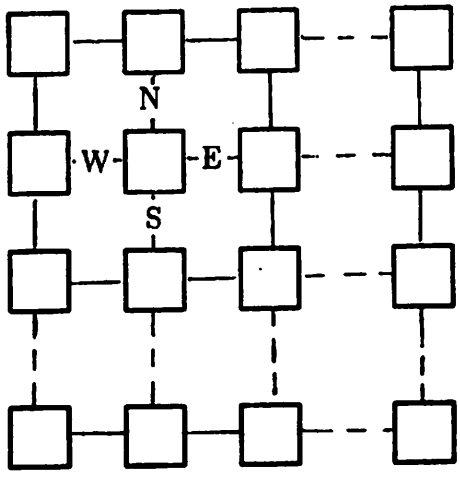



**Figure 15:** The arrangement of processors in the MCC

of registers of word size $\theta(\log n)$ for a total of $O(\log n)$ space. Thus, any register can hold the absolute address of any processor in the matrix. Each processor can ship a single word of data to its east, west, north, or south neighbor in $\theta(1)$ time and can perform standard arithmetic on the contents of its registers in $\theta(1)$ time.

Although a reduced resolution representation of data can be achieved on the MCC by appropriately "shutting off" processors, the transfer of information between adjacent pixels will still have to pass through the intermediate processors. Such an increase in the communication time makes multiple-resolution representation unattractive. Hence, in our pyramid algorithm, the various filtered images are all represented at the same resolution as the input image. For notational convenience, however, the higher spatial-frequency channels will still be called the "finer" levels, and the lower spatial-frequency channels will be called the "coarser" levels. The same level numbering scheme as in the pyramid algorithm will be used, with the understanding that the image-resolution does not change

between the levels.

Since the spatial-frequency decomposition is maintained, there is still a reduction in accuracy and increase in range of the displacements determined from for the lower-frequency channels. Hence, as it will be explained in section IV.2.4, the displacements are sub-sampled according to the spatial-frequencies of the channels.

The remainder of this section contains a description of the conventions used for our MCC algorithm and the specification of the components of that algorithm. A detailed description of the same algorithm was also provided in [121]. Since the primary purpose of this description is to show that our algorithm can be implemented on an MCC, the following description also includes pseudo-code specifications of the various algorithms.

### III.2.1 Notational conventions

The MCC algorithms described in this section are written in a hybrid notation which borrows features from N-PASCAL, which is a language developed for expressing SIMD algorithms for the NON-VON [52]. The principal features of N-PASCAL that are of interest are the *vector* variable type and the parallel conditional form, *where-do-elsewhere*. Statements in N-PASCAL containing references to vector variables operate in parallel at all processors. The *where-do-elsewhere* form allows the execution of a block of code on a processor to be conditional on the value of a vector variable. When a *where* is encountered, execution proceeds at all processors which satisfy the vector variable condition, while it is temporarily suspended at all other processors. An optional *elsewhere* allows a block of code to be executed by those processors which failed the original condition.

The convention used for vector variables, or registers, is a variable name in italic capital letters optionally followed by a compass direction in parenthesis, (e.g., $A$(east)). The legal register transfers permitted within the con-

straints imposed by our definition of MCC are assignments to $A$ from $A$(east), $A$(west),$A$(north) and $A$(south).

Registers $I$ and $J$ contain the two input images, while registers *F1* and *F2* contain the corresponding images in the various spatial frequency channels. The image in register *F1* is always held in registration with the processor array, whereas the other image is moved relative to it. Register *ID* contains the address of each processor, and *ID_X* and *ID_Y* are used to store the $x$ and $y$ coordinates of the address. For each pixel of the image in *F1*, registers *CUR_DX* and *CUR_DY* contain the location of the matching pixel in image *F2*, at the current level of the hierarchy. These are also used during the traversal to maintain the current best match location. At the beginning of the match process within each channel, the initial match estimates from the adjacent coarse level are moved to registers *COARSE_DX* and *COARSE_DY*.

### III.2.2 Spatial frequency decomposition and representation

If a pyramid representation is not used, then there is no particular advantage in using Burt's algorithm for the construction of a hierarchical set of band-pass filters over direct convolutions with appropriate weighting functions. For our MCC algorithm, the well-known family of $\nabla^2 G$ filters have been chosen. The weighting functions are described by

$$\nabla^2 G_{2d}(x,y) = \frac{1}{2\pi\sigma^4}\left[1-(x^2+y^2)\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

where $G_{2d}$ denotes the 2-D Gaussian distribution and the parameter $\sigma$ determines the width of the Gaussian. The center-frequency and the width of the band-pass filter decreases with increasing $\sigma$.

The "size" of the convolution mask is determined by the radius over which the weights are significant. This radius will be proportional to $\sigma$, (and usually chosen to be $4\sigma$, see [66,43]). The number of pixels contained within this radius

is proportional to $\sigma^2$; hence, the time required for combining this information is $O(\sigma^2)$. If, however, the filtering process can be expressed as a combination of two separable one-dimensional convolutions, the time required for combining the information can be reduced to $O(\sigma)$.

Although the 2-D Gaussian function can be separated as a product of two one-dimensional functions,

$$G_{2d}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = G_{1d}(x)G_{1d}(y)$$

where

$$G_{1d}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{\sigma^2}}$$

the $\nabla^2 G$ function cannot be directly expressed as a product of two one-dimensional functions. However, a separable form for $\nabla^2 G$ can be achieved by expressing it as the sum of two separable functions (see [70] for a similar approach),

$$\nabla^2 G_{2d}(\sigma,x,y) = G_{1d}(\sigma,y)\frac{\partial^2 G_{1d}(\sigma,x)}{\partial x^2} + \frac{\partial^2 G_{1d}(\sigma,y)}{\partial y^2} G_{1d}(\sigma,x)$$

where $G_{1d}$ is the one-dimensional Gaussian function. Based on this idea, an MCC algorithm called CONVOLVE, shown in Figure 16 has been developed for the construction of a band-pass filter. CONVOLVE assumes that the image is in register, $I$, and sequentially performs the convolution with each of the two terms in the above expression. The description below explains the convolution with the first term, the convolution with the second term is performed in a similar fashion.

Each processor in the mesh first multiplies the contents of its $I$ register by the value of the $\frac{\partial^2 G_{1d}}{\partial x^2}$ function at the maximum mask radius, $r$. This becomes the initial value of a partial sum that is then passed to its west neighbor. Each processor then multiplies the contents of its $I$ register by the function evaluated

at $r-1$. It is added to the partial sum it *received* from its east neighbor. The new sum is now passed to its west neighbor and the process continues, forming the partial sum as the value is propagated to the west. A mirror image process simultaneously computes a right partial sum, which, when added to the sum from the left, constitutes the one-dimensional convolution for a row. The final two-dimensional convolution is computed by convolving the output of the row convolution in a similar fashion, except along columns, with the Gaussian function. As mentioned above, the second term is computed in a similar fashion and the final output is the sum of the two terms.

In order to obtain the same type of spatial-frequency decomposition as in our pyramid algorithm, the $\sigma$ values should be doubled between adjacent levels. Following the same reasoning as in the pyramid algorithm, the number of spatial-frequency channels required is $\log \delta$, where $\delta$ is the maximum image-displacement.

### III.2.3 Match criterion

The same match criterion as in the case of the pyramid algorithm (i.e., the minimization SSD) is used; however, a key difference arises due to the fact that all the channels are represented at the same resolution as that of the input image. In order to maintain the bandlimit considerations discussed in section III.1.2, it is necessary to increase the window size as the spectral content of the image shifts towards lower frequencies.

Another way of looking at this is as follows: In a pyramid algorithm a $5 \times 5$ window at any level $i$ covers a window whose width is $5 \times 2^{(L-i)}$ pixels at the image level $L$. Since in our representation, the resolution has been maintained constant for all the channels, the widths of the windows should be doubled for each successive lower frequency channel. The standard-deviations of the Gaussian window function should increase with the size of the window in order to meet the bandlimit requirements.

**procedure** CONVOLVE($l$ : integer);

**begin**

$\sigma := 2^l$; $r := \rho * 2^l$;

/* do Laplacian convolution in x-direction */
**for** i := 0 to $(r - 1)$ **do begin**
/* partial sums for east and west halves of Laplacian */
$EPS := EPS(\text{east}) + I * \frac{\partial^2 G_{1d}}{\partial x^2}(\sigma, r - i)$;
$WPS := WPS(\text{west}) + I * \frac{\partial^2 G_{1d}}{\partial x^2}(\sigma, r - i)$;
**end;**

/* add east and west sums */
$LAP := EPS(\text{east}) + WPS(\text{west}) + I * \frac{\partial^2 G_{1d}}{\partial x^2}(\sigma, 0)$;

/* do Gaussian convolution in y-direction */
**for** i := 0 to $(r - 1)$ **do begin**
/* partial sums for north and south halves of Gaussian */
$NPS := NPS(\text{north}) + LAP * G_{1d}(\sigma, r - i)$;
$SPS := SPS(\text{south}) + LAP * G_{1d}(\sigma, r - i)$;
**end;**

/* add north and south sums */
$GAUSS := NPS(\text{north}) + SPS(\text{south}) + LAP * G_{1d}(\sigma, 0)$;

...
/* we omit similar code which computes the second term */
...

**end;**

**Figure 16:** The pseudo-code for the CONVOLVE algorithm

At any pixel, the SSD measure is computed only for a set of candidate displacements that is determined by the coarse-fine control strategy. The match measure between a pixel in the first image and a candidate match in the second image is obtained by first bringing the two pixels into alignment, then computing the squared differences between corresponding pixels, and finally computing the Gaussian weighted sums of the squared difference values within a window. The summing is achieved through a convolution with a Gaussian mask, which as noted in the previous section, is separable as a combination of two one-dimensional convolutions. At all levels, only a $3 \times 3$ set of displacements will be considered for each pixel. The algorithm SSD shown in Figure 17 describes the method of computing the SSD measure.

### III.2.4 Control strategy

Like the pyramid algorithm, the MCC algorithm also uses the coarse-fine control strategy. However, since a pyramid representation is not used, there is no automatic sampling of the displacements. In addition, the local connectivity constraint also implies that some changes must be made to the pyramid algorithm. These modifications are explained below, and a description of the MCC algorithm for the implementation of the coarse-fine control strategy is provided.

If $\delta$ is the maximum displacement in either coordinate direction, then there are $(2\delta + 1)^2$ possible displacements at the image-resolution. At the finer levels, although the search area for any pixel is restricted to a smaller area around the initial estimate obtained from the adjacent coarse-level, the initial estimates for nearby pixels can themselves vary widely over the $(2\delta + 1) \times (2\delta + 1)$ area noted above. In this case, the simplest method of allowing all the processors to communicate with the pixels in their respective search areas is to adopt a generalization of the "uniform-vergence strategy", which was suggested by Williams [120] for stereopsis.

The uniform-vergence strategy involves shifting the second image with respect

```
procedure SSD(level : integer);

begin

/* determine the distance of the current displacement
   from the coarse estimate */
DIST_X := |COARSE_DX - ID_X|;
DIST_Y := |COARSE_DY - ID_Y|;

/* compute the squared difference corresponding values */
C := (F1 - F2) * (F1 - F2) ;

   ...
/* we omit the code for summing the products */
   ...

/* running maximum selection */
where (DIST_X ≤ 2^l) and (DIST_Y ≤ 2^l) do
   where (C < CUR_MAX) do begin
      CUR_MAX := C;
      CUR_DX := ID_X; CUR_DY := ID_Y;
      end;
end;
```

**Figure 17:** The pseudo-code for the SSD algorithm

to the first image through all the relative-positions within the $(2\delta + 1) \times (2\delta + 1)$ area. At level $l$, the radius of the search area around the initial estimate for each pixel is $2^{L-l}$. However, according to the coarse-fine strategy, only a $3 \times 3$ sample of the pixels in this $2^{L-l} \times 2^{L-l}$ need to be considered. Each processor is "on" only when it is aligned with one of its candidate match pixels of the second-image. At this time the SSD measure is computed according to the algorithm described in section IV.2.3, and a running minimum selection is performed.

The uniform shift of the second image relative to the first image can be achieved through a "spiral movement" described in the algorithm SPIRAL shown in Figure 18. Due to the subsampling of the displacements, a set of expanding spirals as shown in Figure 19 can be used. The traversal stops for the computation of the SSD values at the nodes marked in figure. Once again note that a processor will not be "on" during all these stops. As indicated in Figure 19, at all levels, only a $3 \times 3$ set of displacements are considered for each pixel.

Finally, it should be noted that since all the images are at the same resolution as the input image, the concern over the type of connectivity used across adjacent levels is not an issue for our MCC algorithm. At any level $l$, the initial displacement for each pixel is simply the final displacement for the same pixel computed at level $l - 1$, which is also available at the same processor.

### III.2.5 Complexity analysis

The computational complexity of the MCC algorithm is $O(\delta^2 \log \delta)$, where $\delta$ is the maximum image-displacement along either coordinate direction. This increase over the complexity of the pyramid algorithm, which was $O(\log \delta)$, arises due to the local-connectivity constraint. The derivation of the order of complexity of the MCC algorithm is given below.

For the purpose of analysis of the computational complexity, the MCC algorithm can be broadly divided into two sequential stages. The first stage, which

```
procedure SPIRAL(l : integer /* l is the level number */);

begin
num_loops := δ/2^l

/* initialize registers */
COARSE_DX := CUR DX;
COARSE_DY := CUR_DY;
ID_X := ID mod n; ID_Y := ID diff n;

/* each loop cycle below corresponds to one spiral cycle.
   radius contains the radius of the current cycle */
/* at each step we move over 2^l pixels, thus sampling the
   displacements. At the finest level each pixel is traversed,
   whereas at coarser levels they get sparser */

SSD(l);

for radius := 1 to num loops do begin

   /* move left over 2^l pixels once */
   for i := 1 to 2^l do begin
      F2 := F2(east); ID_X := ID_X(east); ID_Y := ID_Y(east);
      end;
   SSD(l);

   /* move up over 2^l pixels 2 * radius - 1 times */
   /* this implements the arm of the spiral moving north */
   for j := 1 to 2 * radius - 1 do begin
      for i := 1 to 2^l do begin
         F2 := F2(south);
         ID_X := ID_X(south); ID_Y := ID_Y(south);
         end;
      SSD(l);
      end;

      ...
   /* similar code for movements east, south, and west over 2^l pixels 2 * radius times */
      ...

   end;
end;
```

Figure 18: The pseudo-code for the SPIRAL algorithm

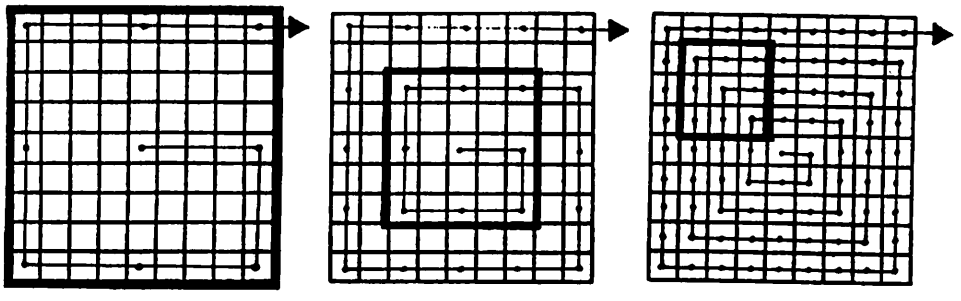


**Figure 19:** The spiral movement in the MCC algorithm. The nodes marked are the ones where the traversal stops for the computation of the SSD. Note that the number of nodes increases by a factor of 4 at each successive level, while the path length increases by a factor of 2. The dark boxes represent the search area for a single processor.

involves the construction of the spatial-frequency channels consists of $\log \delta$ applications of CONVOLVE. At each level, the size of the Gaussian mask is proportional to the corresponding value of $\sigma$. At level $i$, the value of $\sigma$ is proportional to $2^{L-i}$ pixels. The separable convolution technique described in CONVOLVE requires $O(2^{L-i})$ steps. Hence the number of steps needed for CONVOLVE is proportional to

$$\sum_{i=L}^{L-\log \delta} 2^{L-i} = 1 + 2 + 4 + \cdots + \delta$$

which evaluates to $O(\delta)$.

The second stage is the matching stage which involves SPIRAL and SSD. At level $i$, the number of nodes where the spiral traversal stops is $\frac{(2\delta+1)^2}{4^{L-i}}$. However, the expanding spiral traversal requires $\frac{(2\delta+1)^2}{2^{L-i}}$ steps. At each stop, the time for computing the correlation sum is proportional to the width of the template window, since the summing can be expressed as a separable convolution. As noted in section III.2.2, at level $i$, the width of the template window is proportional to $2^{L-i}$. Taken together, the number of steps required at level $i$ for the complete traversal and the computation of the SSD measures is proportional to $2^{L-i} \times \frac{(2\delta+1)^2}{2^{L-i}} = (2\delta+1)^2$. Summing over all the levels $i = L, \ldots, L - \log \delta$, the number of steps required for the matching stage is $O(\delta^2 \log \delta)$.

The above analysis indicates that the matching stage clearly dominates the filtering stage. Hence, the complexity of the complete MCC algorithm is the same as that of the matching stage, i.e., $O(\delta^2 \log \delta)$. Note that a single level matching algorithm on an MCC will require $O(\delta^2)$ steps for the traversal and $O(\delta)$ steps for computing the SSD measure at each stop. Hence, the complexity of the single level algorithm is $O(\delta^3)$. On a sequential machine, the number of steps required will be proportional to the image size, which is usually considerably larger than the maximum displacement. Thus, although the MCC algorithm is less efficient than

our pyramid algorithm, it is still more efficient than a single-level algorithm on an MCC and any implementation on a sequential machine. Recent efforts towards the implementation of our MCC algorithm on a simulator for the CAAPP at UMass [93] suggest that the entire algorithm can be performed at approximately 60 milliseconds, which seems reasonably close to real-time performance.

# CHAPTER IV

# A CONFIDENCE MEASURE AND A SMOOTHNESS CONSTRAINT

Chapter III described the implementation choices made in our system for three of the five components of our framework. This chapter contains a description of the implementation choices for the remaining two components, the confidence measure and the smoothness constraint.

Basically, the confidence measure should indicate whether a pixel in the first image has a match in the second image, and if so, whether the match is unique. Since the SSD measure is an estimate of the similarity of two intensity structures in our band-pass filtered representation, a large SSD value for the best-match pixel indicates that the two structures are not very similar. This would imply that the estimated match may have to be ignored. Since the minimization of the SSD measure is the criterion for determining the matching pixel, the uniqueness of a match estimate depends on the variations of the SSD measure around the match estimate. For instance, if this measure varies significantly in all directions around the best-match pixel, it is likely that the match is unique. If, however, the SSD measure remains nearly constant for some set of pixels including the best match, then all of these pixels are equally likely candidates for the best-match. Based on these intuitions, we have used the directional curvatures of the SSD surface around the best match location to compute the confidence measures for a match.

Our smoothness constraint is similar to those used in the gradient-based approaches, in the sense that it is formulated as the minimization of the sum of two

"error" functionals defined on a displacement field. Our formulation is closely related to the formulations used by Horn and Schunck [51] and Nagel [78], and can be regarded as a vector generalization of the surface reconstruction problem formulated by Grimson [42] and Terzopoulos [104]. Basically, two error functionals are defined: one of these, which is called a "smoothness-error", measures the spatial variation of a given displacement field, while the other functional, which is called an "approximation error", measures the deviation of the displacement field from the set of local estimates determined according to the match criterion. The confidence measures associated with the local estimates are used as weights in the computation of the approximation error. Our solution to the functional minimization problem involves using the finite-element method, which is a well known method [25] for solving such variational problems. In particular, we have adapted the method used by Terzopoulos for solving the surface reconstruction problem.

The gradient-based approaches for the computation of image-velocity fields can be shown to be the mathematical limits of our matching approach for the computation of displacement fields. In particular, it will be shown that the intensity-constraint used in the gradient-based approaches are the limits of our approximation error. As it will be explained in this chapter, this relationship allows us to explicitly identify, for the first time, the match criteria and the confidence measures involved in the gradient-based approaches.

Our algorithms for the computation of the confidence measure and the implementation of the smoothness constraint will be shown to be consistent with the computational considerations of our framework, i.e., parallelism, uniformity, and locality. Once again, the algorithms which we have developed are ideally suited for a pyramid processor, although they can be implemented on an MCC with minor modifications.

The remainder of this chapter is divided into three parts. First, the motivations for our approach for computing the confidence measure, its exact specifi-

cation, and an analysis of its behavior in various situations encountered in real images are described. Second, the mathematical formulation of our smoothness constraint, the relationship of our formulation to those used in the gradient-based approaches, and the algorithm for its implementation are described. Since our primary contribution is the formulation of the minimization problem and not its solution, the mathematical details of the solution method are simply outlined. For a complete and detailed treatment of this method, the reader is referred to [104]. Finally, an outline of the the overall hierarchical algorithm is provided, which includes the explanation of how the confidence measure and the smoothness constraint are integrated with the other three components of our pyramid algorithm, as well as a discussion of the modifications necessary for the implementation on an MCC.

## IV.1 A Confidence Measure

In chapter I, two design requirements were stated for the confidence measure: one is that it should be orientation sensitive and measure the uniqueness of the displacement component along different directions, while the other is that it should be sensitive to occlusion. Thus, two distinct types of information are needed: a measure of the *existence* of a proper match, and a measure of the *uniqueness* of the estimated match. In addition, it is desirable that this process does not require significant computational effort.

In this section, we first illustrate the behavior of the SSD surface[1] by using the results of an empirical study conducted by us [9], and then provide the definition for a confidence measure. The results of our empirical study are included in order to provide a motivation for our definition of the confidence measure. Finally, we discuss the types of behaviors that can be expected of our confidence measure.

[1] Recall that in chapter I, we defined the *SSD surface* as a surface defined over the space of displacements, and whose height is the SSD value corresponding to each displacement.

### IV.1.1 The behavior of the SSD surface

Our empirical study of the SSD surface was performed by creating a pair of synthetic images by digital "cutting and pasting" pieces from two images photographed in the robotics lab at UMass. Gaussian noise of standard deviation of 10, i.e., 20 percent of the standard deviation of the intensity values in the image, was added to the second image. These images are displayed in Figures 20 and 21. Both the input images were first preprocessed using Burt's Laplacian pyramid transform; the finest-level images of Burt's pyramid were then used for the remainder of the experiment. The coffee can in the center part of the image has been displaced by 14 pixels to the right and 4 pixels down. The boundaries of the displaced segment in the two images are shown in Figure 22; the cross-hatched area is visible in the first image but occluded in the second image.

The points chosen to illustrate the behavior of the SSD surface have been highlighted in Figure 20. Figures 23 through 32 show the SSD surface at these points. There are two surface displays for each point: one of these, referred to as the *auto-SSD surface*, is generated when a $5 \times 5$ sample window centered at the point of interest in the first image was matched with similar windows in the same image centered at all points in an $8 \times 8$ area around the point of interest. The other, referred to as the *cross-SSD surface*, is generated when the $5 \times 5$ sample window centered at the point of interest is matched with similar windows centered at all points in an $8 \times 8$ area around the correct match point in the second image.

Note that for occlusion regions, there is no true match point in the second image; however, the true displacement estimate of the background surface (to which they belong) has been used as the center of the cross-SSD surface. This displacement is zero, since the background is stationary.

In order to enhance visibility, the surfaces are shown inverted, i.e., the minimum SSD values are shown as peaks rather than as valleys. In each surface display, the true match point has been marked with an "X" while the point of


$a$ = corner point
$b$ = edge point
$c$ = homogeneous point
$d$ = occluded corner
$e$ = occluded homogeneous area

**Figure 20:** The first frame of the synthetic image pair used to illustrate the behavior of the SSD surfaces.

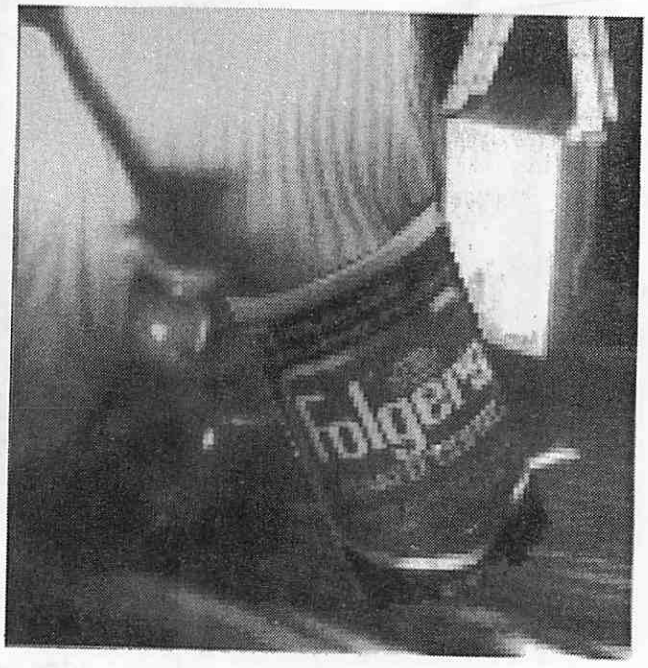



**Figure 21:** The second frame of the synthetic image pair.

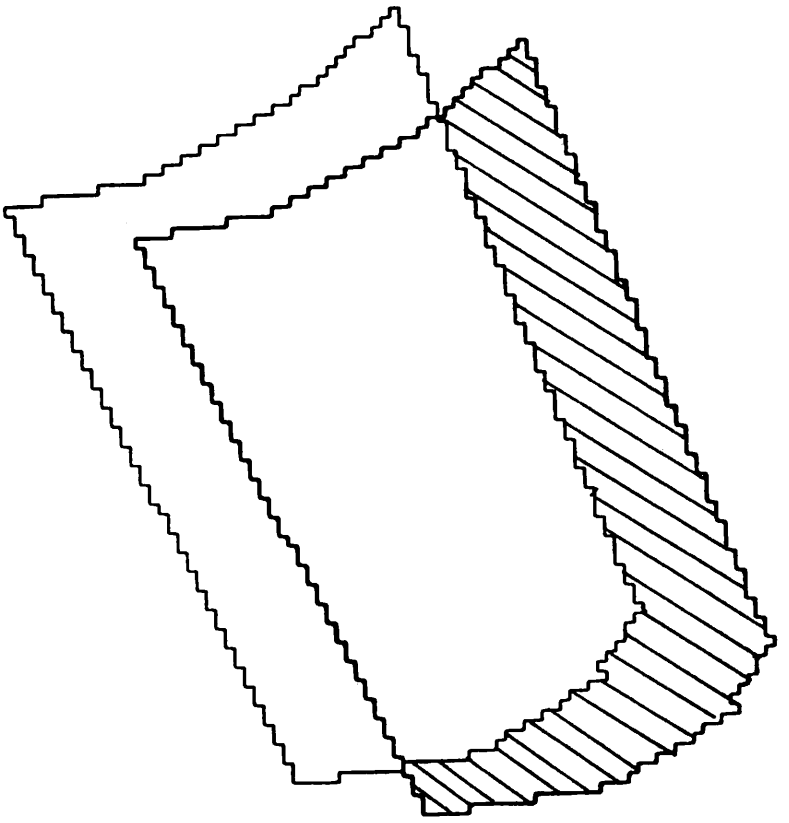


**Figure 22:** The boundaries of the displacement segment in the two Figures 20 and 21.

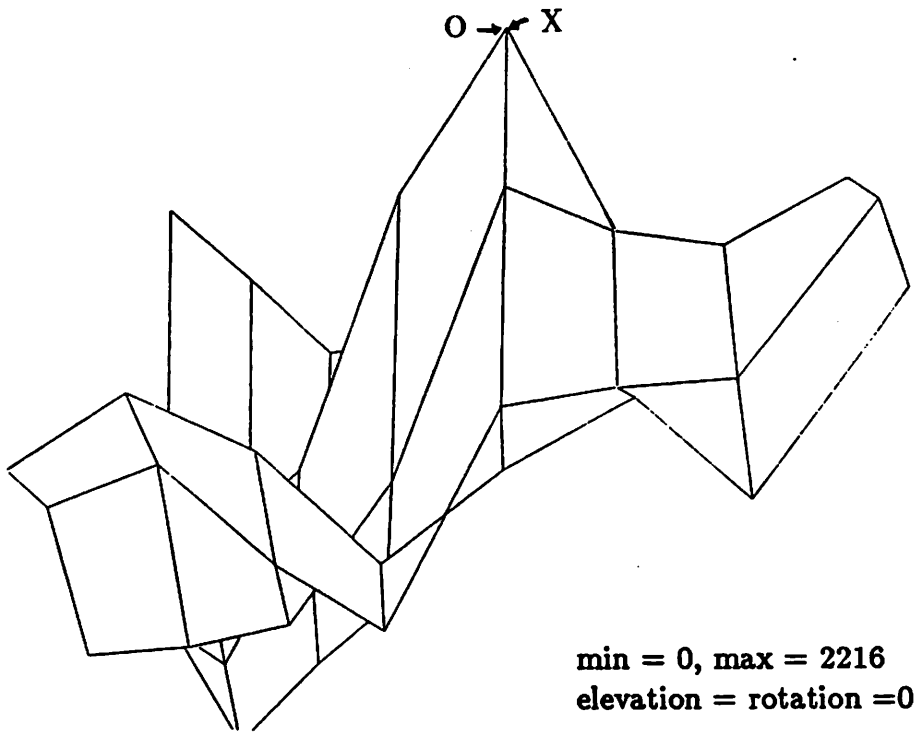



**Figure 23:** The auto-SSD surface at a corner point (point *a* in the first frame).

minimum SSD value has been marked with an "O". Also marked are the minimum and maximum SSD values on each of the surfaces, as well as the view-angle (rotation and elevation). All the surface displays are generated by a perspective projection of the surfaces.

*Corner point* – Figures 23 and 24 display the *auto-* and the *cross-* SSD surfaces at an intensity corner. These correspond to the point *a* in the first image. Note the sharp peak in both the *auto-* and *cross- SSD* surfaces centered at the true displacement value In actuality, this is a sharp valley since these displays have been turned upside-down. This indicates that the match is unique in all directions. Further, note that the shape of the surface is well preserved even though the *cross-SSD* surface was generated using images containing significant amount of noise.

*Along a Straight Edge* – Figures 25 and 26 illustrate the two surfaces at a point

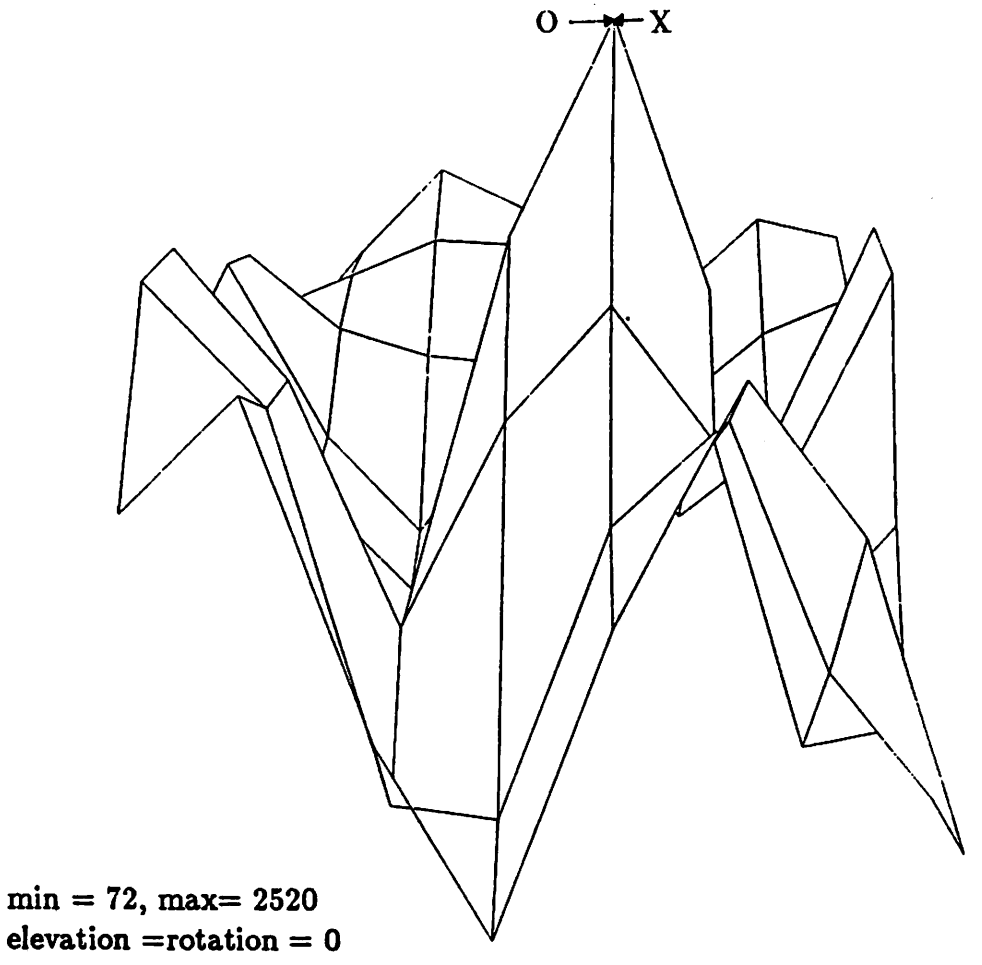



**Figure 24:** The cross-SSD surface at point *a*.

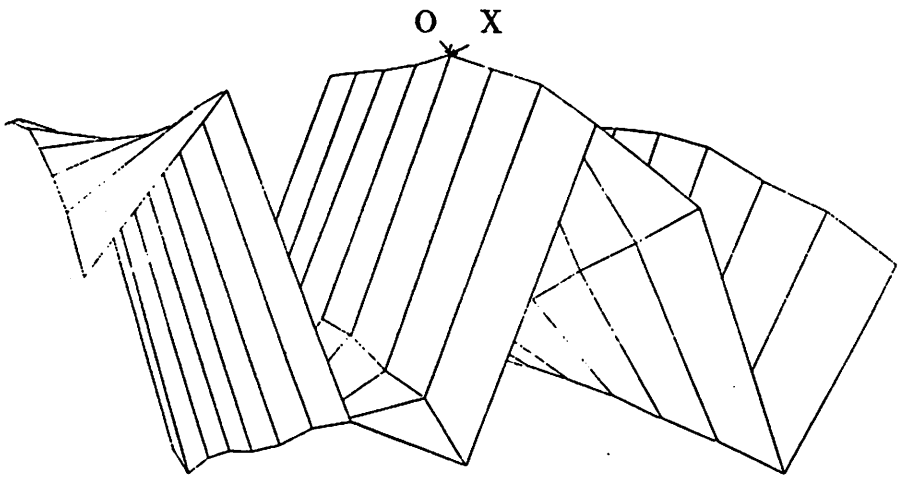



**Figure 25:** The auto-SSD surface at a point along an edge (point *b* in the first frame).

along a straight edge in the image 1 (point *b*). A ridge like structure along the direction of the edge is clearly visible in both the surfaces. This indicates that the match estimate is reliable only in the direction perpendicular to the ridge (or the edge) and that we have no reliable information parallel to the ridge. Note that the peak of the *cross-SSD* surface is shifted away from the correct match-point (i.e., the center of the surface) along the ridge.

*Homogeneous point* Figures 27 and 28 illustrate the SSD surfaces at a homogeneous point (point *c*). In this case the SSD surface is rather flat, especially around the center, i.e., the point of best match. This indicates that the match estimate is unreliable in all directions. Again, the peak of the *cross-SSD* surface does not coincide with the correct match-point. Note that in this case all the values on the *cross-SSD* surface are much smaller than in those for the corner-point described above, implying significantly greater ambiguity in all directions. This behavior is typical at homogeneous areas of the image in the fine-resolution, higher-frequency filtered representations, since at such areas most of the image energy is contained

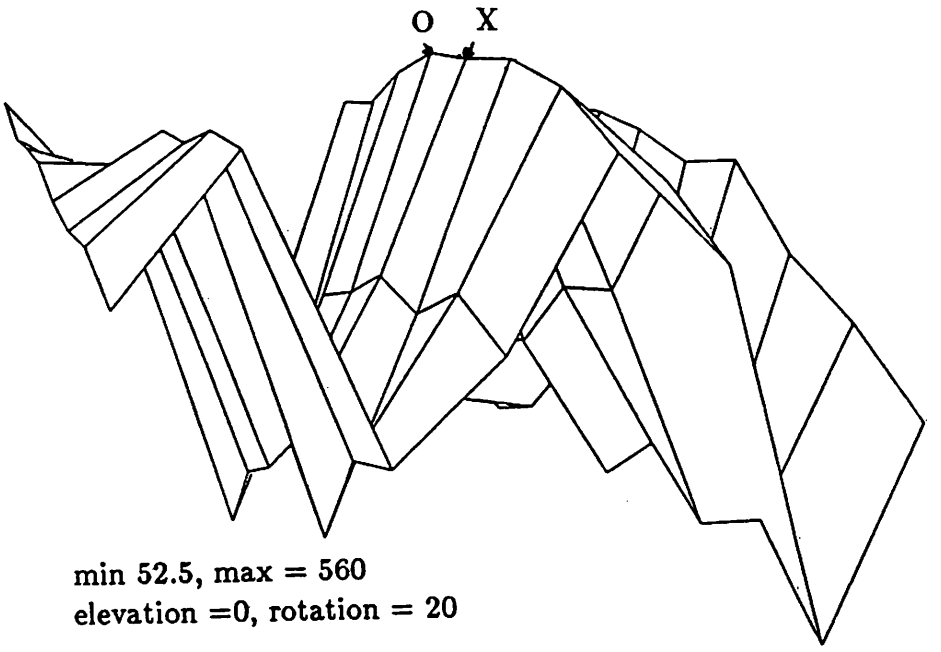



**Figure 26:** The cross-SSD surface at point *b*.

at the lowest spatial frequencies.

*An Occluded Corner Point* - Figures 29 and 30 illustrate the SSD surfaces at an intensity corner (point *d*) which is in the occluded region. In this case, the *auto-SSD* surface display shows a distinct peak, similar to that in Figure 23. Since the area surrounding the corner point is occluded in the second image, there is no window that properly matches the template window surrounding the corner point. The shape of the *cross-SSD* surface does not appear to be similar to that of the *auto-SSD* surface. Also note that the minimum value of the cross-SSD measure for this point is larger than the minimum values for the various non-occluded points discussed above. This is a reflection of the difference in the the intensity structures of the template window which is contained in the occluded area, and the portion of the occluding area which is selected as the best match.

*An Occluded Homogeneous Point* - Figures 31 and 32 illustrate the SSD surfaces at a point (point *e*) in a homogeneous area in the occluded region. Again, the *cross-SSD* surface shows unpredictable and erratic behavior. Once again, note

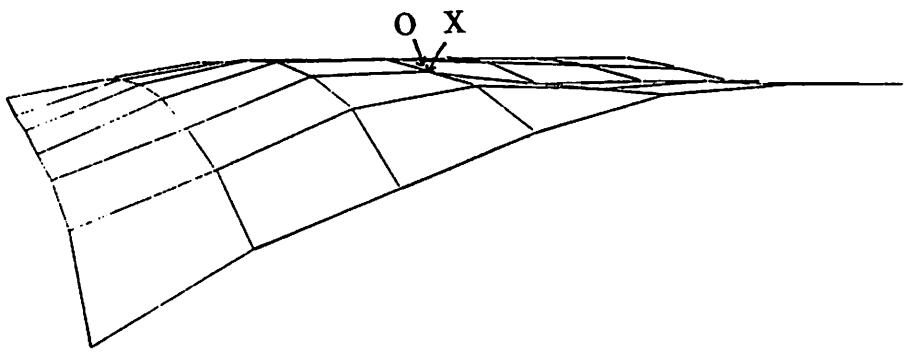



**Figure 27:** The auto-SSD surface at a point in a homogeneous area (point *c* in the first frame).

that the minimum and maximum values of the *cross-SSD* surface are considerably higher than those of the *auto-SSD* surface.

The above demonstrations were intended to show that the SSD surface usually contains much of the information about the image structures as well as occlusion effects. Although only the behavior of the SSD surface at the finest level of resolution was considered, such behaviors are typical at other levels of resolution as well. Where a proper match exists (i.e., the non-occluded regions), the SSD value at the point of best match is low. At occlusion areas this value is usually higher due to the fact that the true match is hidden, and the intensity structure of even the best match area does not resemble the intensity structures in the occluded area. The curvature of the SSD surface along different directions reflects the degree of variation in the image along those directions, and hence the uniqueness of the match estimate along that direction. This suggests that the confidence in the correctness of the displacement component in any direction should vary with the curvature of the SSD surface along that direction, and be inversely proportional

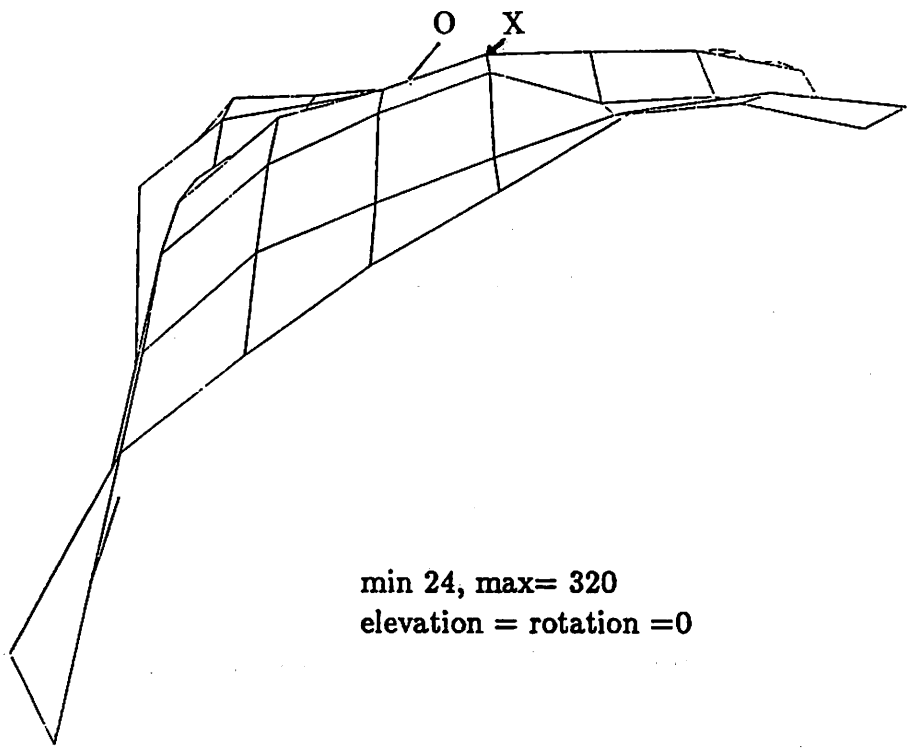



**Figure 28:** The cross-SSD surface at point *c*.

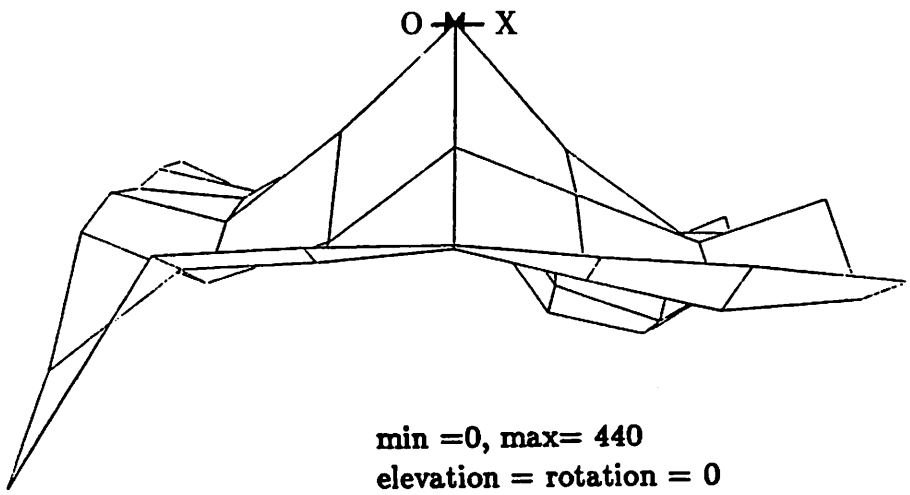



**Figure 29:** The auto-SSD surface at an occluded corner point (point *d* in the first frame).

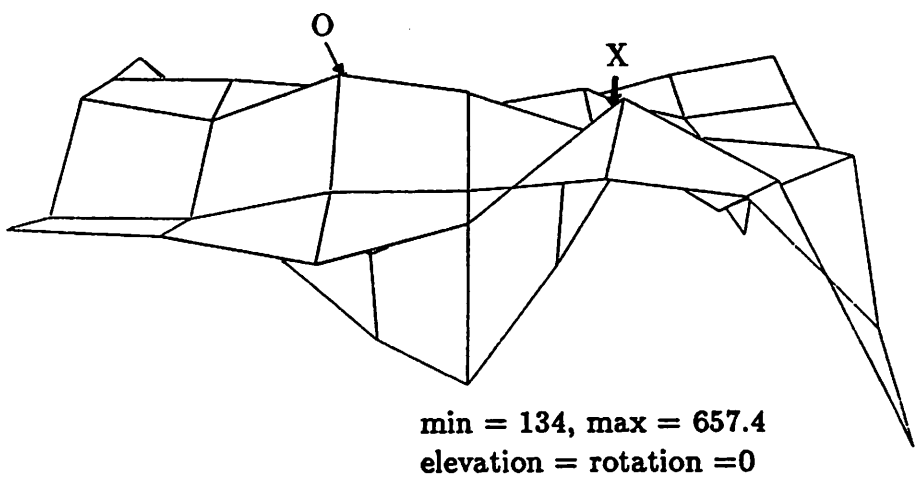



**Figure 30:** The cross-SSD surface at point *d*.

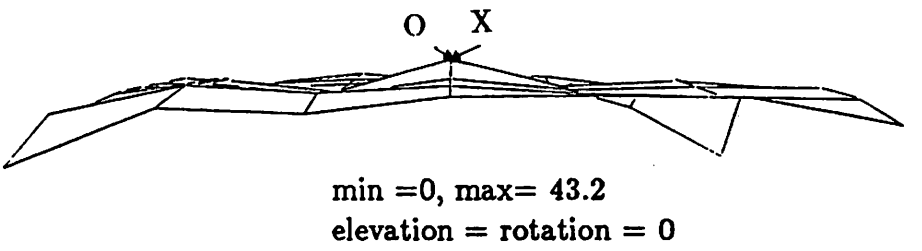



**Figure 31:** The auto-SSD surface at an occluded homogeneous point (point *e* in the first frame).

to the SSD value at the point of best match.

When the image-motion is not locally translational, the presence of a large value for the minimum of the SSD surface can also be due to the deformation of the local image area. A more detailed discussion of the expected behavior of the SSD surface under a wider range of conditions is included in section IV.1.3.

### IV.1.2 Computing confidence measures

The curvature of a surface along any direction is proportional to the second derivative of the height function along that direction. Alternatively, the curvature at a point along any arbitrary direction can also be determined if the two *principal curvatures* and the directions of the associated *principal axes* are known [26]. The principal axes are defined as the directions along which the curvature of the surface is either a maximum or a minimum, and the principal curvatures are the curvatures of the surface along those directions. We denote the maximum principal axis by the unit vector $\hat{e}_{max}$ and the associated curvature as $C_{max}$, and the minimum principal axis by the unit vector $\hat{e}_{min}$ and the associated curvature as $C_{min}$. Although $\hat{e}_{max}$ and $\hat{e}_{min}$ will always be mutually orthogonal [26], for our purposes, it is convenient to separately represent them.

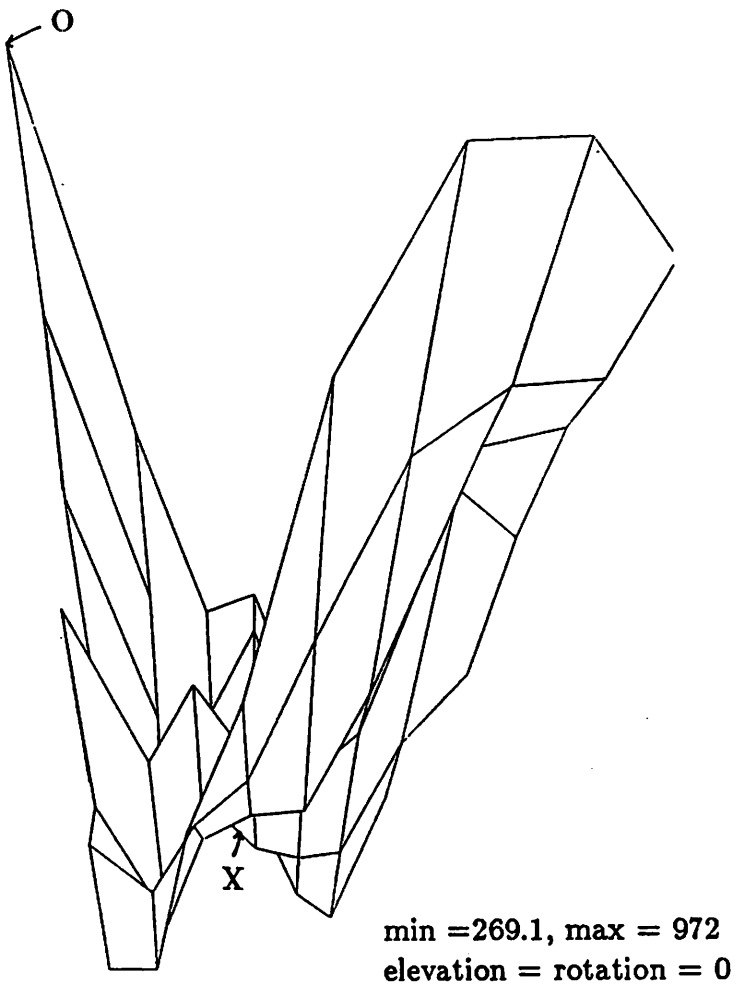



**Figure 32:** The cross-SSD surface at point *e*.

Our confidence measure consists of two magnitudes (called "confidences") $c_{max}$ and $c_{min}$, and two directions (or unit vectors) $\hat{e}_{max}$, and $\hat{e}_{min}$. As above, the unit vectors denote the principal axes of the SSD surface, while

$$c_{max} = \frac{C_{max}}{k_1 + k_2 S_{min} + k_3 C_{max}}$$

and

$$c_{min} = \frac{C_{min}}{k_1 + k_2 S_{min} + k_3 C_{max}}$$

where $k_1, k_2$, and $k_3$ are parameters that are normalization parameters, and $S_{min}$ is the SSD value corresponding to the best match.

The exact form of the normalization function was derived from the following considerations. The confidence measure is made proportional to the corresponding principal curvature according to the intuition that the reliability of the displacement should be proportional to the curvature of the SSD surface. Since a large value for $S_{min}$ indicates an unreliable match due to occlusion, noise, or deformation of the image area, the confidence is inversely proportional to $S_{min}$. The presence of the term containing the curvature in the denominator is useful to maintain the confidence value in the range $(0, 1/k_3)$. If it is desired not to restrict the range of the confidence, $k_3$ can be set to zero. Finally, the constant term $k_1$ is useful to maintain the value of the confidence measure to be finite when $S_{min}$ tends to zero. A similar type of reasoning applies to $c_{min}$.

The computation of the principal curvatures involves knowing the second partial derivatives of the SSD surface along the coordinate directions. Given that we have a discrete set of SSD values, the following approach is used for this purpose. First, the $3 \times 3$ set of SSD values around the best match location are computed. Since the best match location may be at the boundary of the $3 \times 3$ *search area* discussed in chapter III, it may be necessary to compute a few additional SSD values in order to obtain a $3 \times 3$ set of values around the best match. A uniform way of achieving this would be to expand the search area to be $5 \times 5$, but restrict

the candidates for the best match to the central $3 \times 3$ pixel area.

Given the $3 \times 3$ set of SSD values, a quadratic surface fit can be extracted by using the best-least-square method of Beaudet [15]. This method involves computing weighted sums of the $3 \times 3$ values to obtain the various first and second order derivatives of a surface. The principal curvatures can be expressed as non-linear combinations of the second derivatives. The mathematical principles underlying Beaudet's approach and the details of the various steps of our algorithm for computing the confidence measures are explained in Appendix B.

### IV.1.3 Discussion

Intuitively, the vectors $\hat{e}_{max}$ and $\hat{e}_{min}$ and the confidences $c_{max}$ and $c_{min}$ can be understood as follows: At a point along an edge in the image, the vector $\hat{e}_{max}$ will be approximately oriented in the direction normal to the edge, and $\hat{e}_{min}$ will be oriented parallel to edge. At such a point, $c_{max}$ will be large and $c_{min}$ will be small. On the other hand, in an area of the image with small intensity variations, both the measures will be small, whereas at a point along a contour with high curvature, both will be high.

The characteristics mentioned above are the same as the requirements on the confidence measure that were mentioned in chapter I. As it will be shown in the next chapter, in a number of experiments involving real images, the confidence measure usually performs as predicted here. However, a conclusive evaluation is made difficult by the absence of ground-truth information for most real images that are currently available.

An important issue that was somewhat sidestepped in this section was the sensitivity to occlusion. Although we have so far assumed that a large value $S_{min}$ indicates a "false match" (i.e., a match does not *exist*), in general $S_{min}$ can be large due to a variety of reasons. Some of these reasons are listed below:

1. The search area does not contain the true match. The search area may not

contain the true match either because of an incorrect initial displacement or because of occlusion.

2. The template window contains a discontinuity in image flow. This arises at points near depth or motion discontinuities. In this case, since the window straddles the boundary, the intensity structure within the window varies between frames.

3. The magnitudes of rotation and/or the translation in depth are large or the image area undergoes non-rigid motion. In this case, the template window undergoes an area deformation, which violates the assumption of locally translational motion.

4. The SNR (signal-to-noise ratio) is low. In this case, the intensity values of corresponding areas in the two images differ due to the presence of noise.

In all the cases listed above, $S_{min}$ will be large only if the local "spectral-energy" (i.e., the RMS value of the intensities in the template window) is large. A small value for the spectral energy suggests that the magnitudes of the intensity are small, i.e., the point is in a homogeneous intensity area; hence, all the values on the SSD surface will uniformly be small.

Finally, it should be noted that at a homogeneous occluded area, the $S_{min}$ will be small, even though the match estimate will usually be incorrect. As noted above, this arises due to the low spectral energy of the area. In these cases, the curvatures of the SSD surface may also be low, thereby making our confidence measures small. However, if the occluded homogeneous area straddles a textured area (where the spectral-energy is high), then the curvatures of the SSD surface may be large, because the displacements on one side of the best match will have large SSD values, since they are a result of comparing a homogeneous area with a textured area. Therefore, although no match for the occluded point exists in the second image, the confidence measure may be large for some (incorrect) match *in*

*the homogeneous area* of the second image. The estimated displacement is likely to equal the relative displacement between the textured area and the homogeneous area. Thus, there may be points in homogeneous areas that are occluded by textured areas, where our matching process may provide incorrect displacements with high values for the confidence measures. It should be noted that there are no solutions, or even approaches to this type of problem in the literature on the measurement of motion.

As indicated by the shapes of the surface computed for of our empirical study described in section IV.1.1 (see Figs 31 and 32), the shapes of the auto- and the cross-SSD surfaces will usually be different for most occluded areas. Therefore, an additional clue for the presence of occlusion may be obtained by comparing of the shape of these two surfaces. We expect to further develop these and other ideas for the detection of false matches during our own future research on the measurement of image motion.

## IV.2 A Smoothness Constraint

The problem of finding a smooth displacement field which approximates the displacement estimates computed at a discrete set of points by the local match process can been formulated as a minimization problem. That is, a vector field $\left\{\vec{U}\right\}$ is needed, which minimizes a quadratic functional $E\left(\left\{\vec{U}\right\}\right) = E_{sm}\left(\left\{\vec{U}\right\}\right) + E_{ap}\left(\left\{\vec{U}\right\}\right)$, where $E_{sm}$, which is called the smoothness error, measures the spatial variation of $\left\{\vec{U}\right\}$ and $E_{ap}$, which is called the approximation error, measures how well $\left\{\vec{U}\right\}$ approximates the field $\left\{\vec{D}\right\}$ provided by the matching process.

Intuitively, a displacement field can be considered "smooth" in an area of the image if its variation over the area is small. An example of a measure of the

spatial variation of a displacement field is

$$E_{sm}\left(\{\vec{U}\}\right) = \iint trace\left\{(\nabla U^T)(\nabla U^T)^T\right\}\, dx\, dy$$
$$= \iint \left(u_x^2 + u_y^2 + v_x^2 + v_y^2\right) dx\, dy \qquad \text{(IV.1)}$$

where $\{\vec{U}\}$ is the set of the displacement vectors $\vec{U}(x,y) = (u(x,y), v(x,y))$, and

$$\nabla U^T = \begin{bmatrix} \partial/\partial x \\ \partial/\partial y \end{bmatrix} \begin{bmatrix} u & v \end{bmatrix} = \begin{bmatrix} u_x & v_x \\ u_y & v_y \end{bmatrix}$$

The domain of integration is usually the whole image. For notational convenience, we have used $\vec{U}$ to mean the vector $\vec{U}(x,y)$.

The above formulation of a smoothness error is due to Horn and Schunck [51], who used this measure in a gradient-based approach for the computation of optical flow. Other examples of such measures will be discussed later in this section.

For the definition of the approximation error, the estimates $\vec{D}(x,y)$ provided by the match process are represented in the local orthogonal basis $(\hat{e}_{max}, \hat{e}_{min})$, which denote the the principal axes of the SSD surface. For a given displacement field $\left\{\vec{U}\right\}$, the approximation error is a weighted sum of the deviations of the components of the displacement vectors $\vec{U}(x,y)$ of the field along the basis directions from the corresponding components of the match estimates $\vec{D}(x,y)$. The weights are the confidences $c_{max}$ and $c_{min}$. Mathematically,

$$E_{ap}\left(\left\{\vec{U}\right\}\right) = \sum_{x,y} \Big[c_{max}(\vec{U}\cdot\hat{e}_{max} - \vec{D}\cdot\hat{e}_{max})^2$$
$$+ c_{min}(\vec{U}\cdot\hat{e}_{min} - \vec{D}\cdot\hat{e}_{min})^2\Big] \qquad \text{(IV.2)}$$

As noted in chapter I, the smoothness process is included at each level of our hierarchical algorithm. In particular, at each level, after the computation of

the initial displacement estimates, the smoothing algorithm is applied before the propagation of displacements to the next finer level.

The remainder of this section describes the motivations for our choice of the minimization problem, its precise mathematical relationship to gradient-based techniques, the approach used for solving the minimization problem, and an algorithm for its implementation. Finally, some possible alternate formulations of the smoothness error are examined.

### IV.2.1 The motivations for our formulation

The minimization problem posed above has its roots in other similar work in computer vision. As such, our formulation is similar to the gradient-based formulation of Horn and Schunck [51] and its hierarchical version used by Glazer [41], which were described in chapter II. Since as noted in chapter II, these gradient-based formulations are related to the formulations of Nagel [76,78], Enkelmann [31], and Hildreth [48] our formulation is also related to them. As explained below, all of these formulations can be regarded as a vector generalization of the surface reconstruction problem posed by Grimson [42] and Terzopoulos [104].

In all of the approaches mentioned above, as well as our approach, the purpose is to determine a function defined over the image plane, given some unreliable and/or partial information regarding its values at a discrete set of points. A global assumption regarding the "shape" of the function is used to obtain a dense set of values for the function. By formulating the problem as the minimization of a *quadratic* functional, we are guaranteed that it has a unique solution. Moreover, since a quadratic functional is also guaranteed to have no local minima other than the correct solution, the algorithm for its determination can be fairly simple, e.g., a type of gradient-descent approach can be used.

The simplest example involves a scalar function, as in the case of a depth map [42,104]. Both Grimson and Terzopoulos assume that a discrete set of depth

values $\{\delta_i\}$ are known with associated confidence measures $\{\beta_i\}$. Therefore, they define two errors $E_{sm}$ and $E_{de}$, which measure the spatial variation of a given depth function $d(x,y)$ and its deviation from the known data, respectively. Two different formulations of $E_{sm}$ are considered,

$$E_{sm:1} = \iint \left(d_x^2 + d_y^2\right) dx\, dy$$

and

$$E_{sm:2} = \iint \left(d_{xx}^2 + 2d_{xy}^2 + d_{yy}^2\right) dx\, dy$$

while $E_{de}$ is defined as

$$E_{de} = \sum_i = \beta_i^2 \left(d_i - \delta_i\right)^2$$

The differences between the two formulations of the smoothness errors can be explained as follows: The error $E_{sm:1}$ does not involve the second order derivatives of the depth function, whereas $E_{sm:2}$ will be finite only if the second order derivatives are finite. This means that while both the errors imply that the depth function must be continuous everywhere in the image ($C_0$ functions), the second order error restricts our attention to a smoother class of functions ($C_1$ functions). Based on available psychophysical evidence concerning the human visual system, Grimson and Terzopoulos[2] choose the second order function for their analysis and implementation.

Our problem of computing a dense displacement field $\vec{U}(x,y)$ involves the simultaneous determination of two functions $u(x,y)$ and $v(x,y)$. In this sense, it can be regarded as a vector generalization of the surface reconstruction problem. For a two dimensional vector field, the local data consists of two components with possibly different values for the associated confidences. The basis directions along

[2]It should be noted that Terzopoulos also includes a third term in his minimization problem which uses information regarding local surface orientation. However, since no such data is available for our problem of computing dense displacement fields, we have not discussed his use of this term.

which the vector is decomposed may vary across the image. Therefore, we have chosen the formulations of the two errors given in equations IV.1 and IV.2.

Note that our formulation of $E_{sm}$ generalizes the $C_0$ formulation $E_{sm:1}$ used by Grimson and Terzopoulos. The following alternate form can be obtained by generalizing the $C_1$ formulation $E_{sm:2}$,

$$E_{sm:2,2} = \iint \left( u_{xx}^2 + 2u_{xy} + u_{yy}^2 + v_{xx}^2 + 2v_{xy}^2 + v_{yy}^2 \right) dx\, dy \qquad \text{(IV.3)}$$

Although we have also implemented an algorithm based on this formulation (see [10]), most of experiments have been based on the simpler formulation given earlier. This is because results from early experiments with the two formulations did not show significant qualitative differences, while the solution to the second-order formulation given above requires more computational effort.

### IV.2.2 Relationship to gradient-based approaches

As mentioned above and explained in chapter II, the functional minimization formulation has been used in several gradient-based approaches. Of these, the formulation used by Horn and Schunck [51] and Glazer [41] is most similar to ours, since they have used the same mathematical expression for the smoothness error. The local data available in their technique is the estimate of normal-velocities given by the normal flow equation

$$V^{\perp} = -\frac{I_t}{|\nabla I|} \qquad \text{(IV.4)}$$

In their formulation, the equivalent to our approximation error is the intensity constraint $E_{int}$ which was defined as

$$E_{int} = \iint \left( |\nabla I| U^{\perp} + I_t \right)^2 dx\, dy$$

where $U$ denotes the image-velocity. The above expression can be rewritten as

$$E_{int} = \iint |\nabla I|^2 \left( U^{\perp} - V^{\perp} \right) dx\, dy \qquad \text{(IV.5)}$$

Since their smoothness error is given a weight of $\alpha^2$ relative to the intensity error, the Horn and Schunck formulation of $E_{int}$ is similar to our $E_{ap}$ where

$$c_{max} = \frac{|\nabla I|^2}{\alpha^2}, \; c_{min} = 0, \; \hat{e}_{max} = \hat{e}_{\nabla I}$$

The change from displacement to velocity is necessary because, in a strict sense the gradient-based approach is defined only for image velocities. Equivalently, we can also consider our displacements to be $U\delta t$, where $\delta t$ is the inter-frame time interval.

A geometric view of the relationship between our $E_{ap}$ and Horn and Schunck's $E_{int}$ is provided in Figure 33. These errors can be regarded as the heights of surfaces in the displacement space. As seen from the figures, the Horn and Schunck error can be represented by a valley like surface, whereas our approximation error has the general shape of an elliptic-paraboloid. The iso-error contours on the two surfaces are straight lines and ellipses (in the general case) respectively. When $c_{max} = c_{min}$, the contours on our $E_{ap}$ surface are circles, and when $c_{min} = 0$ they are straight-lines.

It was also noted in chapter II that Nagel modifies both the error functionals used by Horn and Schunck. His modification of the smoothness error involved using the weight matrix $W$ to redefine $E_{sm}$ as,

$$E_{sm} = \iint trace\left\{\left(\nabla U^T\right)^T W \left(\nabla U^T\right)\right\} dx\, dy$$

As noted in chapter II, the inclusion of $W$ by Nagel has the effect of restricting the smoothness assumption to be along image edges and contours. In that review, it was also noted that it would be more advantageous to propagate the smoothness constraint *isotropically* around each pixel, and use special mechanisms for restricting the propagation across known flow boundaries. Therefore, we have chosen the isotropic smoothness constraint for our formulation and the finite element method for solving the minimization problem so as to allow mechanisms for restricting the propagation across flow boundaries.

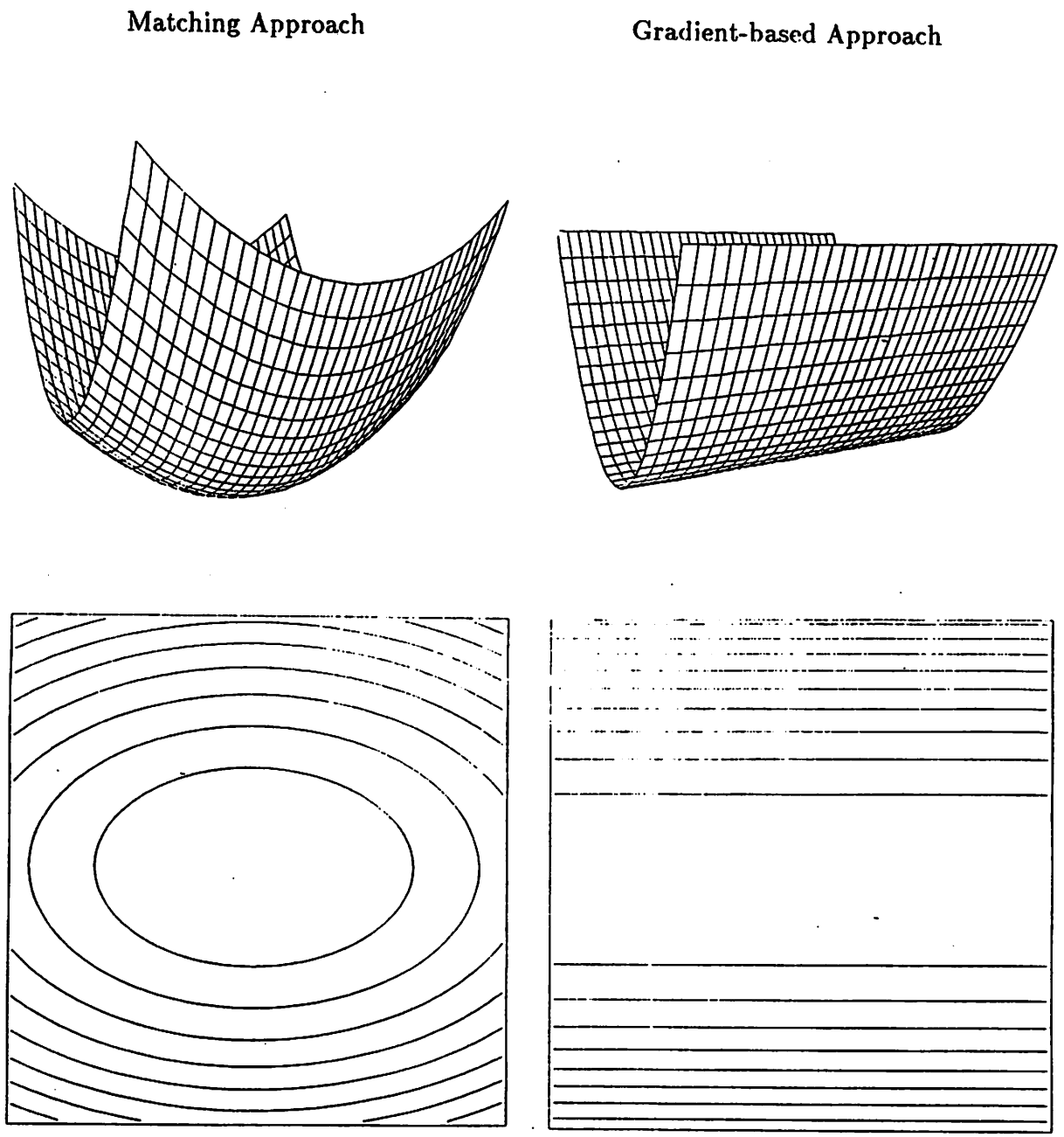



**Figure 33:** A geometrical interpretation of $E_{ap}$

Nagel's modification of $E_{int}$ is somewhat more interesting. As shown in Appendix C, Nagel's formulation of $E_{int}$ can be rewritten as

$$E_{int}\left(\{\vec{U}\}\right) = \iint (U - D)^T A (U - D)\, dx\, dy \qquad \text{(IV.6)}$$

where

$$A = (\nabla I)(\nabla I)^T + \bar{x^2}(\nabla\nabla I)(\nabla\nabla I)^T \qquad \text{(IV.7)}$$

and $\vec{D}$, which represents a local estimate of the image velocity at $(x, y)$, is any solution to the equation

$$AU = -I_t(\nabla I) - \bar{x^2}(\nabla\nabla I)(\nabla I_t) \qquad \text{(IV.8)}$$

We denote the right hand side of the above equation by the symbol $b$. Also recall that the $\nabla\nabla$ operator represents the matrix of second derivatives, e.g.,

$$\nabla\nabla I = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix},$$

and $\bar{x^2}$ denotes the size of a small image-window around a point $(x, y)$ which represents that point.

In Appendix C, we show that when the inter-frame time interval tends to zero, the minimization of the SSD measure is equivalent to solving $AU = b$, provided that the third and higher-order spatial derivatives of the intensity function are ignored, and $\vec{U}$ represents the image-velocity. In this limiting case, our confidence measures will be the eigenvalues of $A$, and our approximation error is the same as the $E_{int}$ defined in equation IV.6. Moreover, when the window-size denoted by $\bar{x}^2$ tends to zero, the equation $AU = b$ reduces to the normal-flow equation IV.4, and the second order formulation of $E_{int}$ used by Nagel reduces to the first order formulation used by Horn and Schunck given in equation IV.5. These relationships can be summarized as the following theorem:

**Theorem 1** *In the limit, when the inter-frame time interval tends to zero, the formulation of the approximation error for image displacements used in the discrete matching approach converges to the second-order formulation of $E_{int}$ for image-velocities used in the gradient-based approach, provided the third and higher order spatial intensity derivatives are ignored. Further, when the window function for correlation tends to a delta function (i.e., the template-window tends to a point), $E_{app}$ converges exactly to the first order gradient-based formulation of $E_{int}$.*

The proof of this theorem is given in Appendix C. The consequence of this relationship is that we can, for the first time, explicitly identify the confidence measures which have until now been implicitly used in the gradient-based approaches. This also explains how the approaches of Glazer [41] and Enkelmann [31], which are the respective hierarchical versions of the Horn and Schunck and Nagel approaches, are consistent with our framework. Although this fact was noted in chapter II, the clear identification of confidence measures and smoothness constraints given above is necessary for completing this demonstration.

### IV.2.3 Conditions for the existence of a solution

As mentioned in the previous section, the two functionals $E_{sm}$ and $E_{ap}$ have been chosen in such a manner that under certain weak conditions there will always exist a unique solution for our minimization problem. The proof of this is based on the following theorem:

*If $E(\{U\}) = B(\{U\},\{U\}) - f(\{U\})$, where $B$ is a symmetric, bilinear form on a Banach space and $f$ is a linear form, then $E$ has a unique minimum if $B$ is positive definite (in that case, $E$ is said to be elliptic).*

In order to apply the theorem, $E = E_{ap} + E_{sm}$ should be decomposed into quadratic and linear terms. According to our definitions given in equations IV.1 and IV.2, $E_{sm}$ is purely quadratic and $E_{approx}$ has both linear and quadratic terms,

so $B_{sm} = E_{sm}$ and

$$B_{ap}(\{U\}) = \sum c_{max}\left(\vec{U}\cdot\hat{e}_{max}\right)^2 + c_{min}\left(\vec{U}\cdot\hat{e}_{min}\right)^2 \qquad \text{(IV.9)}$$

Therefore,

$$\begin{aligned} f(\{U\}) &= B_{ap} - E_{ap} \\ &= \sum c_{max}\left[2(\vec{U}\cdot\hat{e}_{max})(\vec{D}\cdot\hat{e}_{max}) - (\vec{D}\cdot\hat{e}_{max})^2\right] + \\ &\quad + \sum c_{min}\left[2(\vec{U}\cdot\hat{e}_{min})(\vec{D}\cdot\hat{e}_{min}) - (\vec{D}\cdot\hat{e}_{min})^2\right] \qquad \text{(IV.10)} \end{aligned}$$

Now, $B$ is positive definite if $B(U,U) = 0$ implies that $U = 0$. Since $B_{sm}$ and $B_{ap}$ are both quadratic (and therefore non-negative), $B = B_{sm} + B_{ap} = 0$ implies that each of these two terms are zero. It is obvious from our definition of $E_{sm}$ that $E_{sm} = 0$ implies that all the first partial derivatives of $U$ are zero everywhere on the image plane, so $U$ must be constant, i.e. $U = (a,b)$. On the other hand, $B_{ap} = 0$ implies that each of the terms given in equation IV.9 must be zero. It is easy to verify that if there is at least one corner point in the image, i.e., both $c_{max}$ and $c_{min}$ are non-zero, and/or there are two points with different $\hat{e}_{max}$ vectors, then $B_{ap} = 0$ implies that $U = 0$. At least one of these conditions is almost always satisfied in most real images. Therefore, for almost all interesting situations, $E$ has a unique minimum.

We note briefly here that if the second order smoothness constraint defined in equation IV.3 is used, the conditions for the existence of a unique minimum are that there should be at least three non-collinear corner points, or six points with $c_{max} > 0$ which satisfy the following conditions:

1. $\hat{e}_{max}$ points in the same direction for no more than three points.

2. if $\hat{e}_{max}$ points in the same direction for any subset of three points then they are not collinear.

Since this second-order constraint has not been used on our experiments, the detailed derivation of these conditions is not included here. For such a derivation, the reader is referred to [10].

### IV.2.4 Solving the variational problem

In the previous section, it was shown that there is a unique smooth vector field which minimizes our variational measure. In this section the method used to obtain a discrete approximation to this solution is presented.

The most common approaches to solve the variational problem which has been formulated here are the finite difference method, a type of gradient-descent approach, and the finite-element method. For instance, Horn and Schunck, Glazer, Nagel, and Enkelmann have all used the finite-difference method. Hildreth and Grimson have both used the conjugate-gradient approach, while Terzopoulos has used the finite-element method. We have also chosen the the finite-element method because it has a well-developed theory for the inclusion of known discontinuities in the field which is being determined.

The basic idea behind the finite-element method is the tessellation of the image plane using a set of elements with pre-defined shapes, and representing the field using piecewise polynomials defined over these elements. The order of the polynomials are determined according to the order of the derivatives involved in the error functional. A key requirement is that the discrete solution should converge to the true minimum as the element sizes tend to zero.

Terzopoulos has developed the finite-element method algorithms for both the first and the second-order smoothness constraints for the surface-interpolation problem. Since our variational problem can be regarded as a vector generalization of his scalar formulation, his solution methods can also be adapted. For the first-order smoothness constraint, a triangular finite element as shown in Figure 34 was chosen. For the second-order constraint, a domain which consists of a set

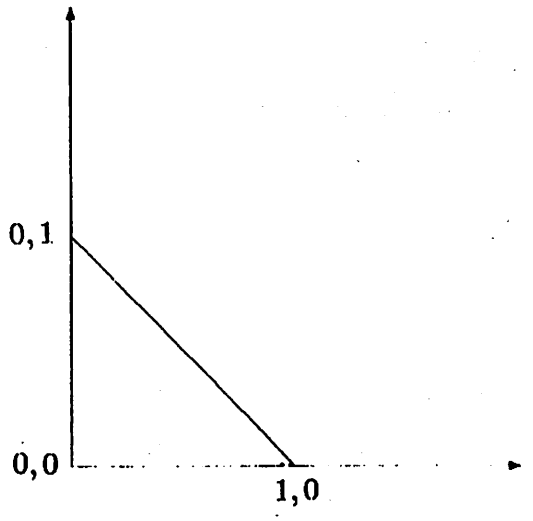



**Figure 34:** The finite-element domain for the first-order smoothness constraint.

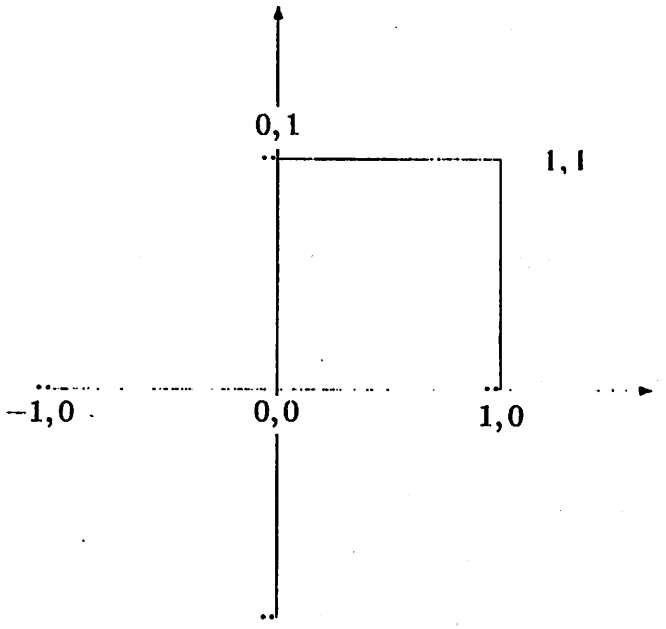



**Figure 35:** The finite-element domain for the second-order smoothness constraint.

of six points on a square grid as shown in Figure 35 was chosen. It should be noted that Terzopoulos' choice for the first-order problem is a standard one and is called a "conforming element", whereas his choice for the second-order problem is non-standard and non-conforming. The rate of convergence to the true solution is usually a criterion for the selection of an element. While the standard element used for the first-order problem is known to converge rapidly, the convergence properties of the second-order element used by Terzopoulos have not been analyzed in detail.

Since we have adapted Terzopoulos' approach for solving the variational problem, we do not discuss the mathematical details here. A clear description of such an analysis can be found in chapters 5 and 6 of [104]. Here, we simply describes the steps involved in our algorithmic implementation.

### Computation of masks

In order to solve the discrete minimization problem, linear equations in the values at the nodes of a square-grid (which is in registration with the image-array) are derived. These equations are used to update the values $\vec{U} = (u, v)$ at a point in terms of its neighbors. In particular, solving the discrete problem can be shown to be the same as solving the following system of coupled equations:

$$(U - \bar{U}) + c_{max}(U \cdot \hat{e}_{max} - D \cdot \hat{e}_{max})\hat{e}_{max} \qquad \text{(IV.11)}$$

$$+ c_{min}(U \cdot \hat{e}_{min} - D \cdot \hat{e}_{min})\hat{e}_{min} = \vec{0} \qquad \text{(IV.12)}$$

where for each point on the grid, $\overline{U}$ is a weighted average of the displacements of its neighbors. For the first-order smoothness constraint the weights are distributed as follows:

$$\frac{1}{4} \times \begin{matrix} & 1 & \\ 1 & 0 & 1 \\ & 1 & \end{matrix}$$

For the second-order constraint, the distribution of the weights is,

$$\frac{1}{20} \times \begin{array}{ccccccc} & & & -1 & & & \\ & & -2 & 8 & -2 & & \\ -1 & & 8 & 0 & 8 & & -1 \\ & & -2 & 8 & -2 & & \\ & & & -1 & & & \end{array}$$

### IV.2.5 Relaxation algorithm

There are a number of numerical methods for solving the system of coupled linear equations described above. One of the simplest methods is the Gauss-Seidel relaxation algorithm. This is an iterative process, where during each iteration the value of $U$ at each point in the image is solved in terms of the values of its neighbors.

The iterative update equation for the displacement field smoothing problem is,

$$U^{n+1} = \overline{U}^n + \frac{c_{max}}{c_{max}+1}((D-\overline{U}^n)\cdot\hat{e}_{max})\hat{e}_{max} \qquad \text{(IV.13)}$$

$$+ \frac{c_{min}}{c_{min}+1}((D-\overline{U}^n)\cdot\hat{e}_{min})\hat{e}_{min} \qquad \text{(IV.14)}$$

where the superscripts denote the number of the iteration.

From a geometric point of view, this updating scheme can be regarded as choosing a point in the $(u,v)$ space that is a combination $U^n$ and $D$. This idea is illustrated in figure 36. For convenience, the displacements in have been represented in a cartesian coordinate system with its axes parallel to $(\hat{e}_{max}, \hat{e}_{min})$. Since $c_{max} \geq c_{min}$, the location of $U^{n+1}$ will always be on or above the line joining $D$ and $U^n$ in figure 36. In particular, it can be seen that $U^{n+1}$ always will be within the triangle that is shown in that figure, moving towards the line joining $D$ and $\overline{U}^n$ as $c_{min}$ gets closer to $c_{max}$.

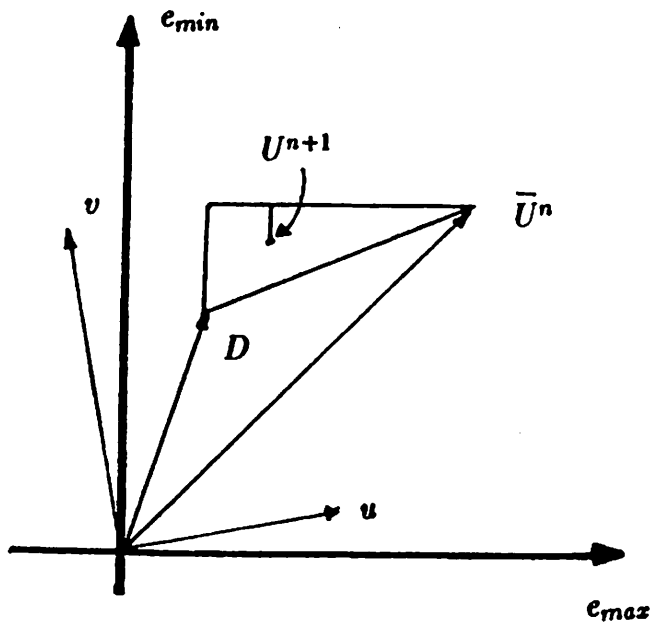



**Figure 36:** A geometric interpretation of the relaxation process

The two key parameters are $c_{max}/(1 + c_{max})$ and $c_{min}/(1 + c_{min})$, which vary between 0 and 1, as $c_{max}$ and $c_{min}$ vary between 0 and $\infty$, as shown in Figure 37. When $c/1+c$ (where $c = c_{max}$ or $c_{min}$ appropriately), is close to zero, the updated value is close to the average of the neighbors, whereas when it is close to 1, the updated value is close to the initial local displacement estimate. Since the function $c/1 + c$ rises rapidly and approaches its maximum value 1, even a small value of $c$ (e.g., 10) orients the updated value strongly $\left(\frac{c}{1+c} = 0.91\right)$ towards the initial local estimate. As it will be explained in chapter V, in our experiments, the choice of the normalization parameters for the confidence measures is based on this observation concerning the behavior of the updating algorithm.

Finally, note that during the projection of the displacements to the next finer-level, the values will be rounded to the nearest integer value; hence, it is not necessary to wait until the complete convergence of the smoothing algorithm.

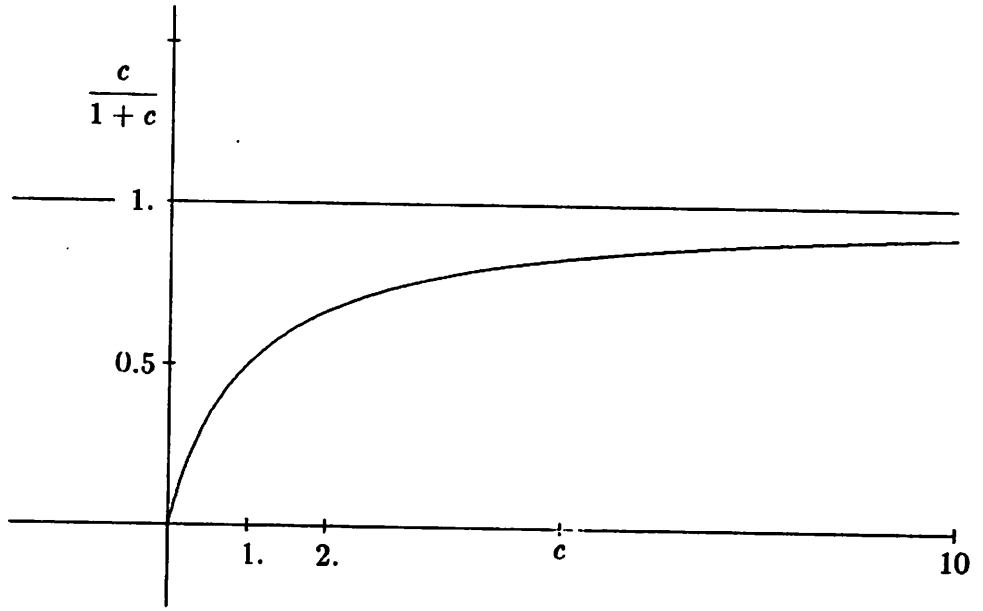



**Figure 37:** The variation of $c/1+c$.

The relaxation process can be stopped when the rounded-off values of the displacements do not change during an iteration. In practice, we found that usually 10 iterations were sufficient to achieve this condition.

### IV.2.6 Discussion

The functional minimization formulation for propagating reliable displacement information to their unreliable neighbors is sufficiently general that it appears to be a powerful formulation. Recent work by Morraquin [73] suggests that the functional minimization formulation can be interpreted as a method of combining probabilistic information. Hence, may be possible to relate this approach to the more recent work on stochastic relaxation [36].

Our choice of the smoothness constraint was based on the heuristic that the displacement field varies smoothly almost everywhere on the image plane. However, as noted earlier, when $E_{sm}$ exactly equals zero, then the displacement field is constant for the first-order smoothness constraint, and a linear function of the

image coordinates for the second-order constraint. On the other hand, it is well known that even for a planar surface patch under rigid motion, the displacement field is a second-order function of the image coordinates:

$$u = a_1 + a_2 x + a_3 y + a_7 x^2 + a_8 xy \qquad \text{(IV.15)}$$

$$v = a_4 + a_5 x + a_6 y + a_7 xy + a_8 y^2 \qquad \text{(IV.16)}$$

where $(a_1, \ldots, a_8)$ depend on the rotational and translational components of the motion as well as the three parameters $(k_1, k_2, k_3)$ of the plane expressed in the form $k_1 X + k_2 Y + k_3 Z = 1$. For more complex surface shapes, and non-rigid motions, the order of the function will be even higher. Hence, a more-realistic formulation of the minimization problem may involve higher order derivatives of the displacements. However, this is likely to lead to complex algorithms and larger convolution masks for the iterative process. Correspondingly, the computational efficiency of the smoothing process will be decreased.

Alternatively, it may be possible to use the following conditions (see [124]), which have been derived from the second-order equations for the displacement field of a planar patch.

$$u_{xx} = 2v_{xy}, u_{yy} = v_{xx} = 0, 2u_{xy} = v_{yy}$$

A direct use of these relationships leads to the formulation of $E_{sm}$ as,

$$E_{sm} = \iint (u_{xx} - 2v_{xy})^2 + u_{yy}^2 + v_{xx}^2 + (2u_{xy} - v_{yy})^2 \, dx \, dy$$

One problem with the constrained approach suggested above is that the planar surface approximation is not always appropriate. An alternative approach may be to assume that the depth varies smoothly over the image plane. In this case, the smoothness error would be applied to the depth map; the geometric relationship between the depth map and the displacement field will be used to reformulate the same constraint on the displacement field. One example of such an effort can be found in the recent work of Scott [97].

## IV.3 The Complete Algorithm

The preceding sections described the confidence measure and the smoothness constraint in isolation from the rest of the framework. While such isolation was convenient for the description of these processes, the complete hierarchical algorithm includes these processes at every level. To conclude this chapter, we give an overview of how all the components that have been described in Chapter III and in this chapter fit together.

Briefly, the hierarchical matching/smoothing algorithm is as follows:

- Each of the two images are processed with a set of band-pass filters and reduced in resolution. As noted earlier, the band-pass filters are isotropic difference-of-Gaussians filters, each of which is slightly wider than one octave and which are one octave apart from each other.

- The match process begins at a level where the maximum image-displacement along either coordinate direction is expected to be less than one pixel. If this information is not available, the process begins at sufficiently coarse level such that most of the template windows and the search areas are fully contained in the image. For example, for the $5 \times 5$ template windows, the $16 \times 16$ resolution is an appropriate starting level. Sum of squared differences is used as the match measure.

- After the matching at one level, confidence measures are computed, and the smoothness algorithm is employed.

- The smoothed vector field is projected to the next finer-level image, based on the overlapped pyramid projection scheme described in chapter III. These are used as initial values for the matching/smoothing process at this finer level.

- This process is repeated at all levels up to the resolution of the input image.

Note that the computation of the confidence measures can be performed in parallel across the image plane and requires only information from the $3 \times 3$ set of pixels around the best match. However, the best match itself can lie at the boundary of the $3 \times 3$ search area selected by our control strategy. Hence, for uniformity of computation, it is best to expand the "search" area to the set of $5 \times 5$ pixels around the coarse-level estimate, but restrict the selection of the best match to the central $3 \times 3$ values. The $3 \times 3$ set of values around the selected best-match can then be used for the computation of the confidence measures.

The relaxation algorithm involved in the implementation of the smoothness constraint can also be performed in parallel across the image plane. During at each iteration, only a small set of neighboring values are needed for computing the weighted average $U$. Thus, in our system, both these components of our framework (i.e., the confidence measure and the smoothness constraint), are implemented in a manner consistent with our computational considerations of parallelism, uniformity, and locality.

### The modifications for an MCC implementation

In chapter III, we described the modifications necessary for the implementation on an MCC of the algorithms for the spatial frequency decomposition, the match criterion, and the control strategy. Here, we briefly indicate the modifications needed for the MCC implementation of the algorithms for computing the confidence measure and enforcing the smoothness constraint.

**Confidence measure** - Since the confidence measure requires little additional computation besides what is already needed for the selection of the best match, the modifications are also simple. Basically, we need to expand the set of candidate matches to a $5 \times 5$ set, and store all the 25 SSD values. At the end of the spiral movement, after the best match location is determined, we simply use the 9 values (out of these 25) around the best match to compute the various derivatives of the SSD surface and the confidence measures, just as in the pyra-

mid algorithm. Note that only a fixed number of registers are required to store these values; this is consistent with the definition of the MCC given by Miller and Stout [71] and described in chapter III. The additional time required for these computation is also independent of the maximum displacement $\delta$ and the level number. Hence it does not influence the complexity of the algorithm.

**Smoothness constraint** The major step involved in each iteration of the smoothing process is the process of computing the weighted average of the neighbors. This process is the same as convolution of the current displacement field with the weight mask described in section IV.2.4. Hence an algorithm similar to CONVOLVE which was described in chapter III can be used. Also note that at level $l$, the distance between the neighbors is $2^{L-l}$ processors, where $L$ is the finest level. Therefore, just as in CONVOLVE, at level $l$, this process requires $O(2^{L-l})$ steps. Since as explained in section IV.2.5, the number of iterations are constant and independent of the level number, the complexity of the smoothing algorithm at level $l$ process is still $O(2^{L-l})$. It can be easily shown that for complexity analysis, the spiral movement is still the dominant process; hence, the complete algorithm on an MCC still requires $O(\delta^2 \log \delta)$ steps.

Finally, it should be noted that embedding the smoothness process in the hierarchical algorithm means that the displacement information is propagated rapidly across the image plane. If a completely single level algorithm is used, this process would be slow and require many more steps. A detailed discussion of the relative speeds of convergence of the single level and the multi-level smoothing processes is provided by Glazer in [41]. It appears that the improvement in the convergence rate is primarily due to the propagation of large-scale variations in the displacement field at a low-resolution, and the propagation of small-scale variations at the higher resolution. As noted in chapter I, this effect is automatically achieved in our strategy of separating the match process according to scale. Therefore, while Glazer's analysis does not directly apply to our approach, similar conclusions can be drawn regarding the improvement in the speed of the smoothing algorithm

due to the hierarchical approach.

# CHAPTER V

# EXPERIMENTAL EVALUATION

The last and perhaps the most crucial stage of the design and development of a system is its testing. This chapter describes a set of experiments conducted in order to evaluate our hierarchical matching algorithm with orientation-selective confidence measures and smoothing. A suitable environment for performing such experiments is provided by the VISIONS system at the University of Massachusetts [45], which simulates a hierarchical processing architecture and the associated pyramid data structures and runs on a Vax 11/780 (or on a Vax 11/750) under the VMS operating system. The design of the VISIONS system facilitates the writing of pixel-parallel programs, in addition to providing a suitable environment for writing various types of digital image processing algorithms, such as storing and retrieving images, their display, and performing various types of filtering, histogramming, and convolution operations.

Our experiments include images of synthetic motion as well as real motion. In both cases, real intensity images were used as input. The synthetic motion experiments were performed because ground-truth information is rarely available in real-image sequences, which prohibits a quantitative evaluation of the results. In order to illustrate the effect of the smoothness constraint, we performed all the tests both with and without the smoothing process.

The important parameters for our hierarchical algorithm are,

1. The shape of the weight function $W(x, y)$ (i.e., the template window) used in computing the sum of squared differences. For the reasons explained in

chapter III and in Appendix A, our primary choice is a Gaussian function. In particular, we used the convolution mask associated with Burt's Gaussian pyramid. For comparative purposes, however, we also included a few synthetic motion experiments using uniformly weighted windows (these are also called rectangular windows).

2. The size of the template window. For Gaussian windows, our primary choice was $5 \times 5$ pixel windows, which is the size of the convolution mask used at the finest-level of Burt's Gaussian pyramid. Secondarily, we also performed a few experiments using a $13 \times 13$ weighting function, which is the finest-level equivalent of the convolution mask at the first coarser-level of the Gaussian pyramid. For the rectangular windows, we considered $3 \times 3$, $5 \times 5$, and $9 \times 9$ pixel windows.

3. The number of levels for the hierarchical process. Based on an empirical observation that the maximum displacement along either coordinate direction over all our test cases was less than 15 pixels, we chose 4 levels for processing. For the sake of uniformity, this number was maintained for all our experiments, although in some cases the displacements were considerably smaller than 15 pixels.

4. The normalization parameters for the smoothness constraint. Recall that in chapter IV, our normalization function was defined as,

$$c_{max} = \frac{C_{max}}{k_1 + k_2 S_{min} + k_3 C_{max}}$$

and

$$c_{min} = \frac{C_{min}}{k_1 + k_2 S_{min} + k_3 C_{min}}$$

To maintain consistency and uniformity between our experiments, we chose $k_1 = 150, k_2 = 1$ and $k_3 = 0$ for all the experiments. As noted in chapter III, $k_1$ is an overall scaling factor, $k_2$ controls the influence of $S_{min}$, and $k_3$ is

useful to restrict the range of the confidence values. The choice of $k_3 = 0$ simply removes any restrictions on the range of the confidences. The choice of $k_2 = 1$ means that the influence of $S_{min}$ is of the same order as that of the curvatures. Our choice of $k_1 = 150$ was based on the empirical observation that the mean values of $C_{max}$ for our images usually varied between 100 and 200. Therefore, barring the effects of $S_{min}$, on the average $c_{max}$ will be approximately 1, and the factor $c_{max}/1 + c_{max}$ will be approximately 1/2. As explained in chapter IV (section 2.5), this means that on the average, the local displacement estimate and the weighted average of the neighbors' estimates have equal effect during each iteration of the relaxation process.

5. The number of iterations of the relaxation process. As noted in chapter IV, we found that usually the changes in the rounded-off values of the displacements were negligible after about 10 iterations. Therefore, for all our experiments, we performed 10 iterations of the relaxation process at all levels of the hierarchy.

The rest of this chapter contains a description of the synthetic motion experiments first, followed by a description of the real image tests, and then a summary of the evaluation process.

## V.1 Synthetic motion experiments

All the experiments described in this section were performed using the *Mandrill eye* image which is shown in Figure 38. This 128 × 128 pixel resolution image is actually a portion of a larger image of the face of a mandrill, which was provided by USC, and is currently available at the Visions Laboratory of the University of Massachusetts. In all cases, a second frame was generated by digital translation of this image 5 pixels upwards and 7 pixels to the right. Since the displacement vectors are the same for the entire image, a simple method of

examining the results is to study the histograms of the *error-field*, which is a vector field obtained by the pixel-wise subtraction of the correct displacement field from the one computed by our matching algorithm.

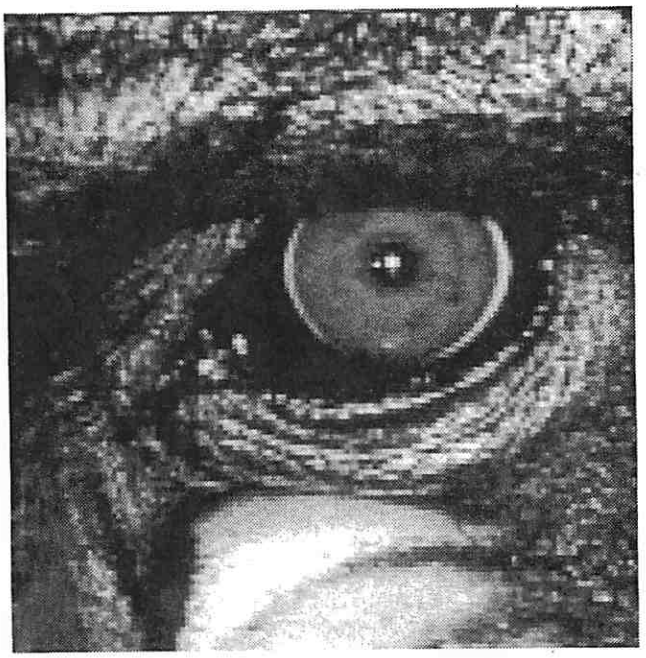


**Figure 38:** The input image for the synthetic motion experiment

Our first group of experiments involving the mandrill images demonstrate the effect of varying the shape and the size of the window function. For these experiments, no noise was added to either of the input images. The correct (i.e., ground-truth) displacement field is shown in Figure 39. There are 10 experiments in total, consisting of 3 sizes of rectangular windows, and 2 sizes of Gaussian windows, each used with and without the smoothness constraint. Table 2 indicates the percentage of image pixels where both components of the error in the displacement are within the $\pm 1/2$ pixel range and the percentage of pixels where the errors are within the $\pm 2.5$ pixel range.

As expected, the larger windows provide more accurate results, although good

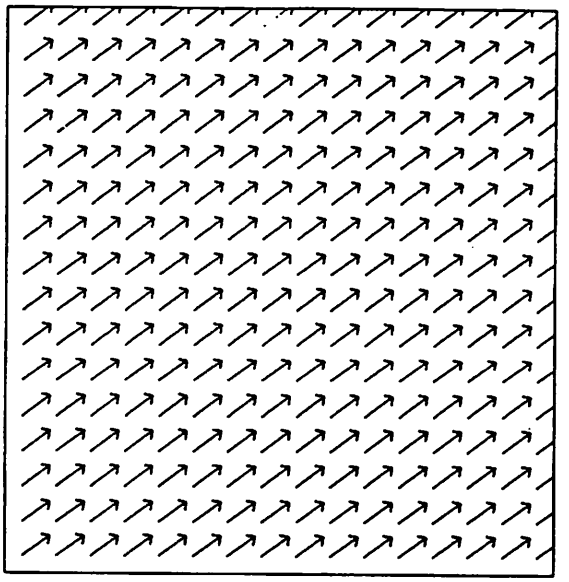


**Figure 39:** The correct displacement field for the synthetic motion experiment

| *window info* | *% within ±0.5 pix.* | | *% within ±2.5 pix.* | |
|---|---|---|---|---|
| | *unsmoothed* | *smoothed* | *unsmoothed* | *smoothed* |
| 3 × 3 rect. | 75.60 | 89.60 | 78.33 | 95.75 |
| 5 × 5 rect. | 83.65 | 92.88 | 86.00 | 98.55 |
| 9 × 9 rect. | 86.64 | 94.01 | 86.88 | 98.62 |
| 5 × 5 Gauss | 79.43 | 92.96 | 81.31 | 98.51 |
| 13 × 13 Gauss | 85.88 | 93.57 | 86.58 | 98.75 |

**Table 2:** Error statistics for various window sizes. For each window size, the entries in the first two columns indicate the percent of pixels where both components of the error in the displacement are within ±0.5 pixel range. The third and the fourth columns indicate the percent of pixels where the errors are within ±2.5 pixels.

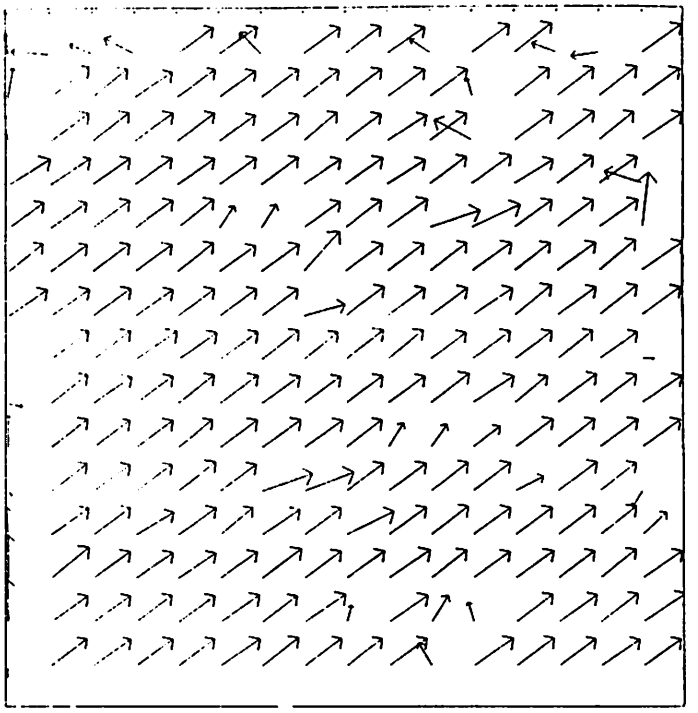
results are obtained even with the $5 \times 5$ windows. The smoothness constraint also definitely improves the results. In addition, when the smoothing process is included, less than 5 % of the pixels have an error outside the $\pm 2.5$ pixel range.

Since we have chosen the $5 \times 5$ Gaussian window for our real-image experiments, we have also included a more detailed display of the results for that case. Figure 40 shows the displacement field and an error histogram obtained with the $5 \times 5$ Gaussian window and *without* the smoothness constraint, while Figure 41 shows the results obtained with such a window and *with* the smoothness constraint.

The second group of experiments illustrates the effect of adding noise. Random noise with Gaussian distribution was added to the shifted image. The standard-deviation of the Gaussian distribution was set at 5, 10 and 25 percent of the intensity-range of the input image. Table 3 indicates the percentages of the pixels where the errors are within the ranges $\pm 0.5$ pixel, and $\pm 2.5$ pixels. In all cases, a $5 \times 5$ Gaussian window was used, and the experiments were performed both with and without the smoothness constraint. Finally, Figures 42 and 43 shows the displacement fields and the histograms for the case of 25 percent noise.

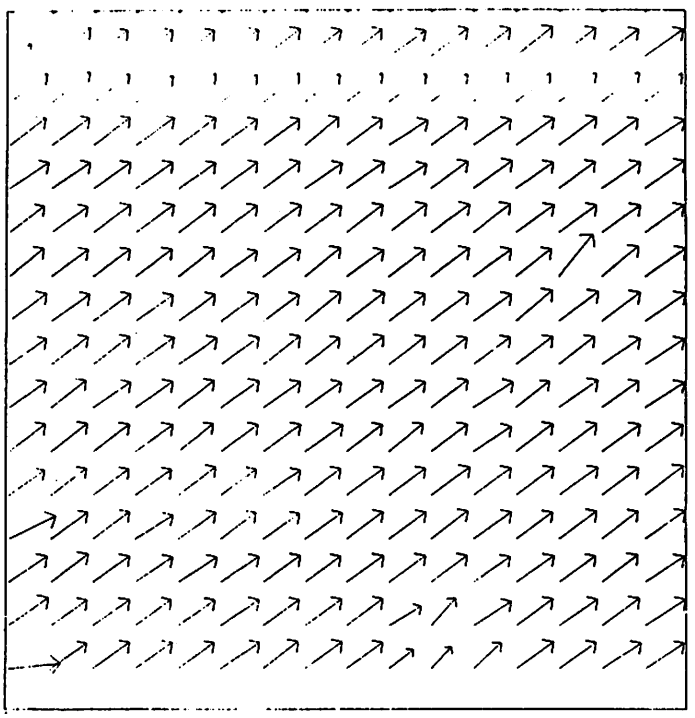
| *noise* | *% within* $\pm 0.5$ *pix.* | | *% within* $\pm 2.5$ *pix.* | |
|---|---|---|---|---|
| | *unsmoothed* | *smoothed* | *unsmoothed* | *unsmoothed* |
| 5 % | 66.45 | 88.40 | 77.31 | 96.77 |
| 10 % | 48.85 | 79.01 | 70.75 | 95.37 |
| 25 % | 17.11 | 45.11 | 48.55 | 88.71 |

**Table 3:** Error statistics for various amounts of noise. For each amount of noise, the entries in the first two columns indicate the percent of pixels where both components of the error in the displacement are within the $\pm 0.5$ pixel range. The third and the fourth columns indicate the percent of pixels where the errors are within $\pm 2.5$ pixels.



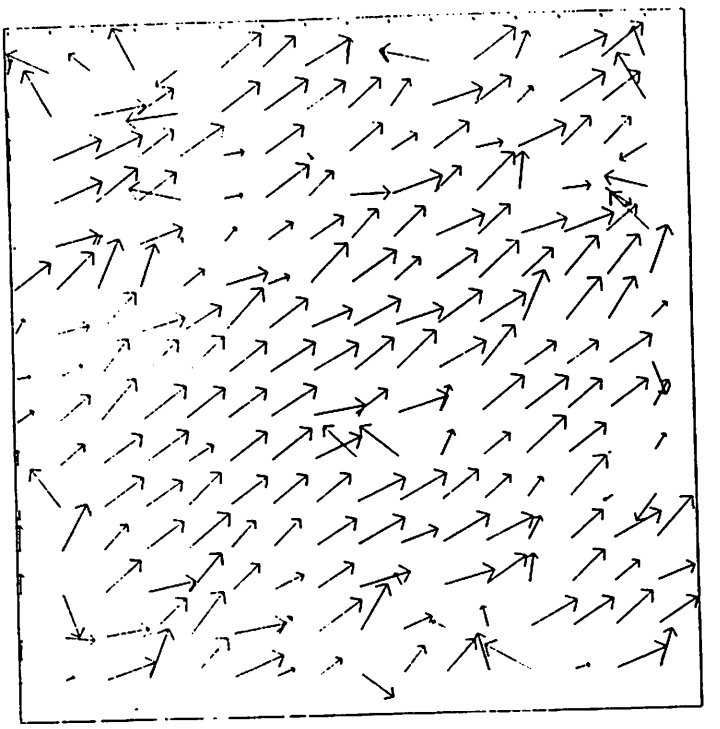
| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| -12 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -11 | 6 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -10 | 9 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 7 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 6 | 2 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| - 5 | 11 | 0 | 1 | 0 | 0 | 4 | 0 | 6 | 0 | 0 | 0 |
| - 4 | 5 | 4 | 2 | 3 | 2 | 9 | 6 | 2 | 0 | 1 | 0 |
| - 3 | 8 | 2 | 2 | 1 | 11 | 23 | 8 | 2 | 0 | 1 | 3 |
| - 2 | 3 | 6 | 8 | 2 | 4 | 19 | 4 | 0 | 2 | 0 | 0 |
| - 1 | 6 | 3 | 3 | 0 | 3 | 7 | 7 | 5 | 3 | 2 | 0 |
| 0 | 61 | 84 | 79 | 26 | 14 | 13015 | 57 | 44 | 57 | 20 | 19 |
| 1 | 28 | 50 | 37 | 15 | 21 | 10 | 23 | 18 | 15 | 6 | 17 |
| 2 | 21 | 18 | 5 | 3 | 7 | 13 | 5 | 2 | 1 | 7 | 2 |
| 3 | 58 | 11 | 10 | 6 | 5 | 19 | 3 | 1 | 0 | 1 | 1 |
| 4 | 13 | 7 | 9 | 3 | 2 | 5 | 8 | 0 | 0 | 0 | 0 |
| 5 | 23 | 14 | 8 | 5 | 5 | 1 | 0 | 2 | 1 | 0 | 0 |
| 6 | 12 | 10 | 7 | 8 | 0 | 6 | 0 | 2 | 2 | 0 | 0 |
| 7 | 7 | 3 | 5 | 2 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| 8 | 15 | 5 | 0 | 0 | 1 | 24 | 1 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | -5 | -4 | -3 | -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 |

**Figure 40:** The displacement field and the error histogram obtained for the synthetic motion experiment (no noise) with $5 \times 5$ Gaussian window and no smoothing.



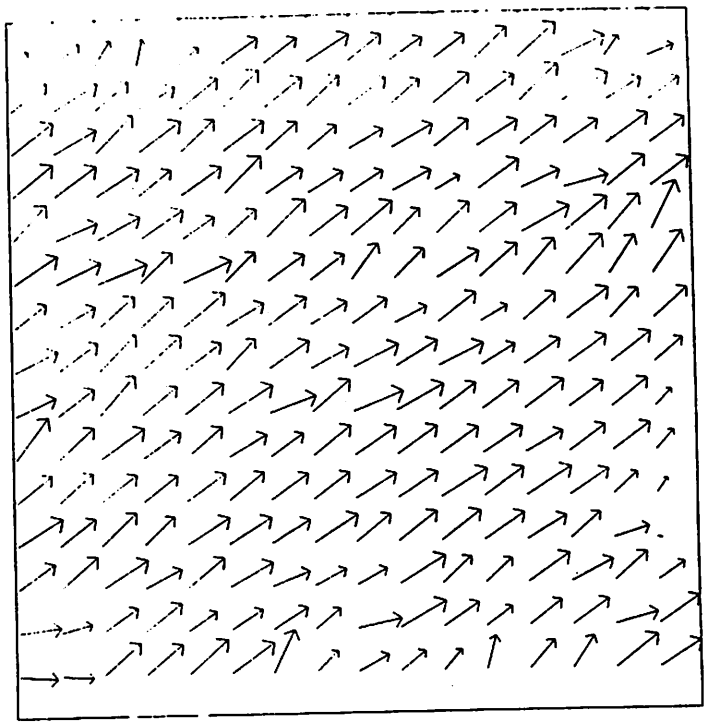
| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| -12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 2 | 0 | 0 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 0 | 0 |
| - 1 | 0 | 0 | 0 | 0 | 0 | 32 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 18 | 45 | 114 | 108 | 15231 | 16 | 10 | 8 | 0 | 0 |
| 1 | 0 | 7 | 73 | 146 | 164 | 35 | 10 | 5 | 0 | 0 | 0 |
| 2 | 0 | 1 | 61 | 219 | 2 | 3 | 7 | 17 | 4 | 0 | 0 |
| 3 | 0 | 0 | 2 | 2 | 1 | 0 | 3 | 7 | 16 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | -5 | -4 | -3 | -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 |

**Figure 41:** The displacement field and the error histogram obtained for the synthetic motion experiment (no noise) with $5 \times 5$ Gaussian window and smoothing.



| | -5 | -4 | -3 | -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| -12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| - 9 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 1 | 0 | 0 |
| - 8 | 0 | 0 | 1 | 1 | 4 | 6 | 1 | 0 | 1 | 0 | 0 |
| - 7 | 0 | 0 | 1 | 2 | 3 | 5 | 4 | 2 | 0 | 0 | 0 |
| - 6 | 0 | 1 | 4 | 2 | 4 | 17 | 11 | 4 | 3 | 1 | 1 |
| - 5 | 0 | 0 | 5 | 9 | 15 | 31 | 17 | 24 | 22 | 9 | 2 |
| - 4 | 2 | 1 | 5 | 21 | 42 | 76 | 33 | 39 | 36 | 18 | 8 |
| - 3 | 7 | 4 | 14 | 28 | 50 | 110 | 57 | 35 | 33 | 24 | 12 |
| - 2 | 25 | 21 | 18 | 51 | 82 | 224 | 81 | 68 | 34 | 24 | 13 |
| - 1 | 27 | 32 | 49 | 91 | 150 | 380 | 198 | 121 | 70 | 35 | 30 |
| 0 | 127 | 155 | 195 | 426 | 515 | 2804 | 546 | 311 | 191 | 116 | 72 |
| 1 | 53 | 65 | 79 | 188 | 163 | 486 | 250 | 138 | 62 | 65 | 49 |
| 2 | 73 | 95 | 103 | 132 | 182 | 326 | 122 | 99 | 44 | 34 | 28 |
| 3 | 76 | 71 | 52 | 49 | 75 | 155 | 85 | 50 | 34 | 16 | 13 |
| 4 | 49 | 56 | 76 | 40 | 46 | 110 | 39 | 29 | 14 | 17 | 4 |
| 5 | 41 | 24 | 19 | 29 | 21 | 33 | 16 | 11 | 4 | 2 | 0 |
| 6 | 45 | 31 | 19 | 10 | 9 | 27 | 11 | 10 | 5 | 3 | 1 |
| 7 | 25 | 16 | 11 | 4 | 7 | 11 | 6 | 8 | 6 | 12 | 0 |
| 8 | 56 | 58 | 45 | 51 | 36 | 23 | 17 | 11 | 9 | 10 | 2 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 42:** The displacement field and the error histogram obtained for the synthetic motion experiment (25 % noise) with $5 \times 5$ Gaussian window and no smoothing.



| | -5 | -4 | -3 | -2 | -1 | 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| -12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| -5 | 0 | 0 | 0 | 2 | 4 | 0 | 0 | 0 | 0 | 0 | 0 |
| -4 | 0 | 0 | 1 | 29 | 28 | 22 | 1 | 0 | 0 | 0 | 0 |
| -3 | 0 | 0 | 5 | 40 | 39 | 51 | 11 | 2 | 0 | 0 | 0 |
| -2 | 1 | 1 | 10 | 62 | 109 | 187 | 48 | 12 | 0 | 0 | 0 |
| -1 | 5 | 9 | 8 | 39 | 182 | 816 | 109 | 33 | 2 | 0 | 0 |
| 0 | 26 | 89 | 128 | 359 | 999 | 7390 | 1077 | 274 | 43 | 1 | 0 |
| 1 | 31 | 102 | 255 | 288 | 413 | 1037 | 278 | 110 | 30 | 0 | 0 |
| 2 | 25 | 119 | 166 | 160 | 176 | 341 | 56 | 34 | 5 | 0 | 0 |
| 3 | 17 | 8 | 16 | 55 | 48 | 46 | 9 | 4 | 0 | 0 | 0 |
| 4 | 5 | 5 | 8 | 22 | 25 | 22 | 6 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 8 | 17 | 26 | 9 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 11 | 2 | 0 | 0 | 0 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 5 | 1 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Figure 43:** The displacement field and the error histogram obtained for the synthetic motion experiment (25 % noise) with $5 \times 5$ Gaussian window and smoothing.

## V.2 Real image experiments

This section describes the results of applying our algorithm to real image pairs. Although our algorithm has been successfully applied to more than a dozen image pairs, we report the results of five experiments, which together seem to represent the types of situations that are commonly encountered in real images. These experiments are referred according to the scene contained in the images: (i) the optic fundus experiment, (ii) the dinosaur-image experiment, (iii) the road-scene experiment, (iv) the hallway-scene experiment, and (v) the office-scene experiment.

Recall that our standard choice of parameters are, (i) $5 \times 5$ Gaussian windows, (ii) 4 levels of hierarchical processing, (iii) the normalization of the confidence measures using $k_1 = 150, k_2 = 1$, and $k_3 = 0$, and (iv) 10 iterations of the relaxation process. As explained below, the only exception is the fundus image, for which "better" results were obtained by changing the window size. In this case, along with the "best results", the results of the standard parameter settings have also been shown.

In most cases, the results from four levels of the hierarchy are shown. Usually, we have followed the format of dividing a figure into four quadrants: if $L$ is the level of the input-image, the results from levels $L-3$ and $L-2$ are shown in the top left and right quadrants respectively, while the results from levels $L-1$ and $L$ have been shown in the bottom left and right quadrants respectively. In all cases, we have also included displays of the confidence measures computed at the four levels.

Since ground-truth displacement data is not available for any of the experiments reported here, the evaluation process involves making qualitative judgments about the performance of the algorithm as illustrated in the various displays. Therefore, all our displays are accompanied by brief write-ups which discuss the important characteristics of the images and the results.

In addition to the standard displays mentioned above, in some experiments, we have also included some additional illustrations which highlight specific properties of our results. Such illustrations are explained in the appropriate sections containing the experimental results.

### V.2.1 The optic-fundus experiment

The two input images used in the optic-fundus experiment are shown in Figure 44. The images are at $128 \times 128$ pixel resolution, and are part of a sequence of flourescein angiogram images of the optic-fundus of a human-subject obtained from Paul Nagin at the Tufts New England Medical image processing Lab. The problem is to register two images taken at the beginning and at the peak of dye filling. Areas which show very little change are recognized as regions where no filling of the dye occurs. This measurement can then be used in the prognosis of glaucoma.

The optic-fundus experiment is interesting because it contains two images where the image-motion is completely smooth, i.e., no depth or motion boundaries are present. The motion can be approximated by a rotation about an axis perpendicular to the image plane passing through a point left of the visible area in the first image. Also note that the overall brightness level and the contrast have both significantly changed between the two frames. This makes it challenging task to match these two images.

Figure 45 displays the results of the hierarchical matching algorithm without smoothing, and with the standard parameter settings noted above. Comparing these results with those obtained by the hierarchical matching algorithm with smoothing shown in Figure 46, it is also clear that the smoothing process contributes considerably towards their improvement.

While the improvement in results due to the smoothing process seems particularly significant at the bottom-right portion of the image, there still seem to be a

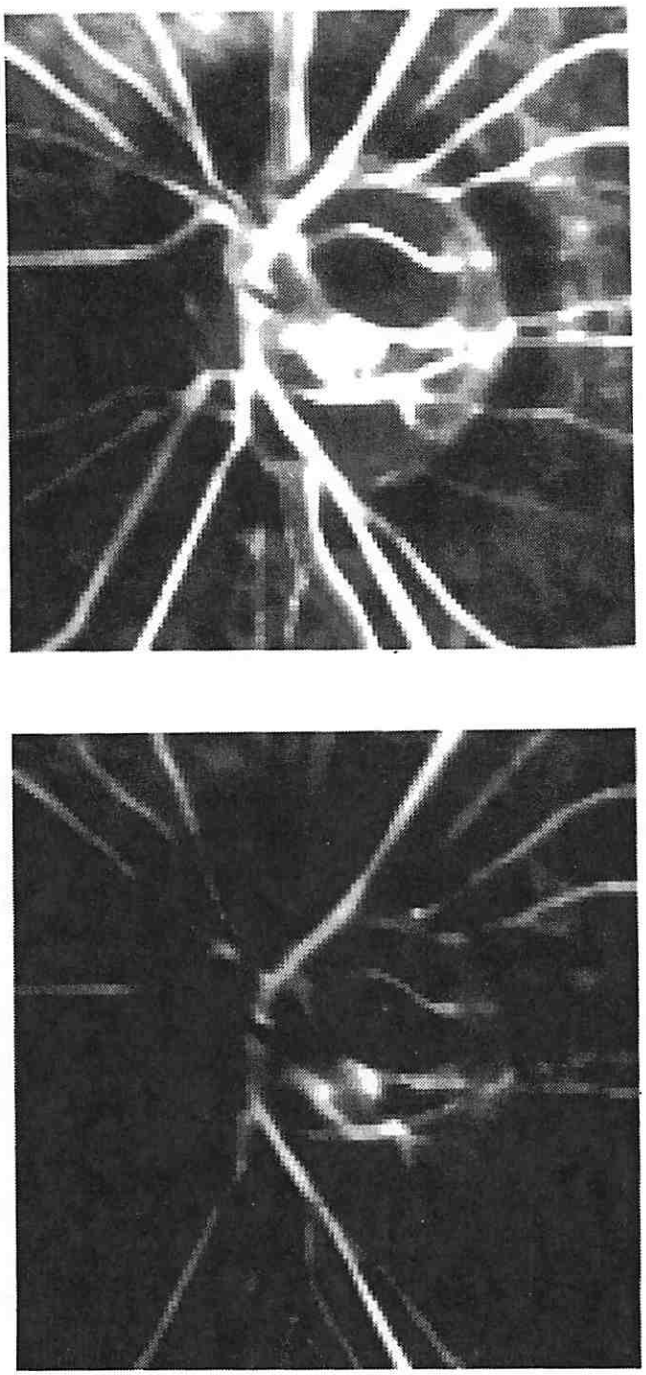


**Figure 44:** The optic-fundus experiment: input images.

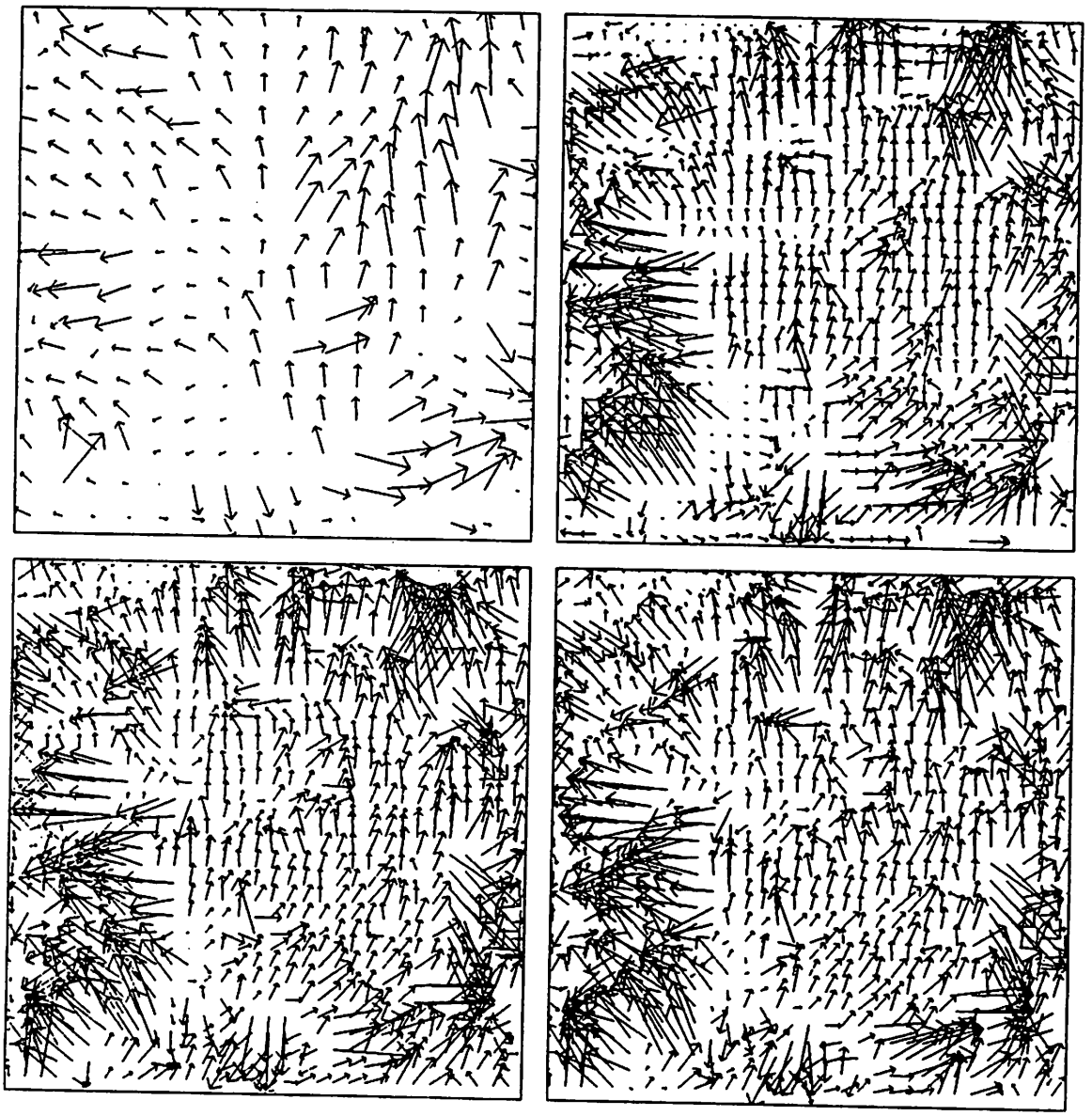


**Figure 45:** The results of the optic-fundus experiment without smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility, only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

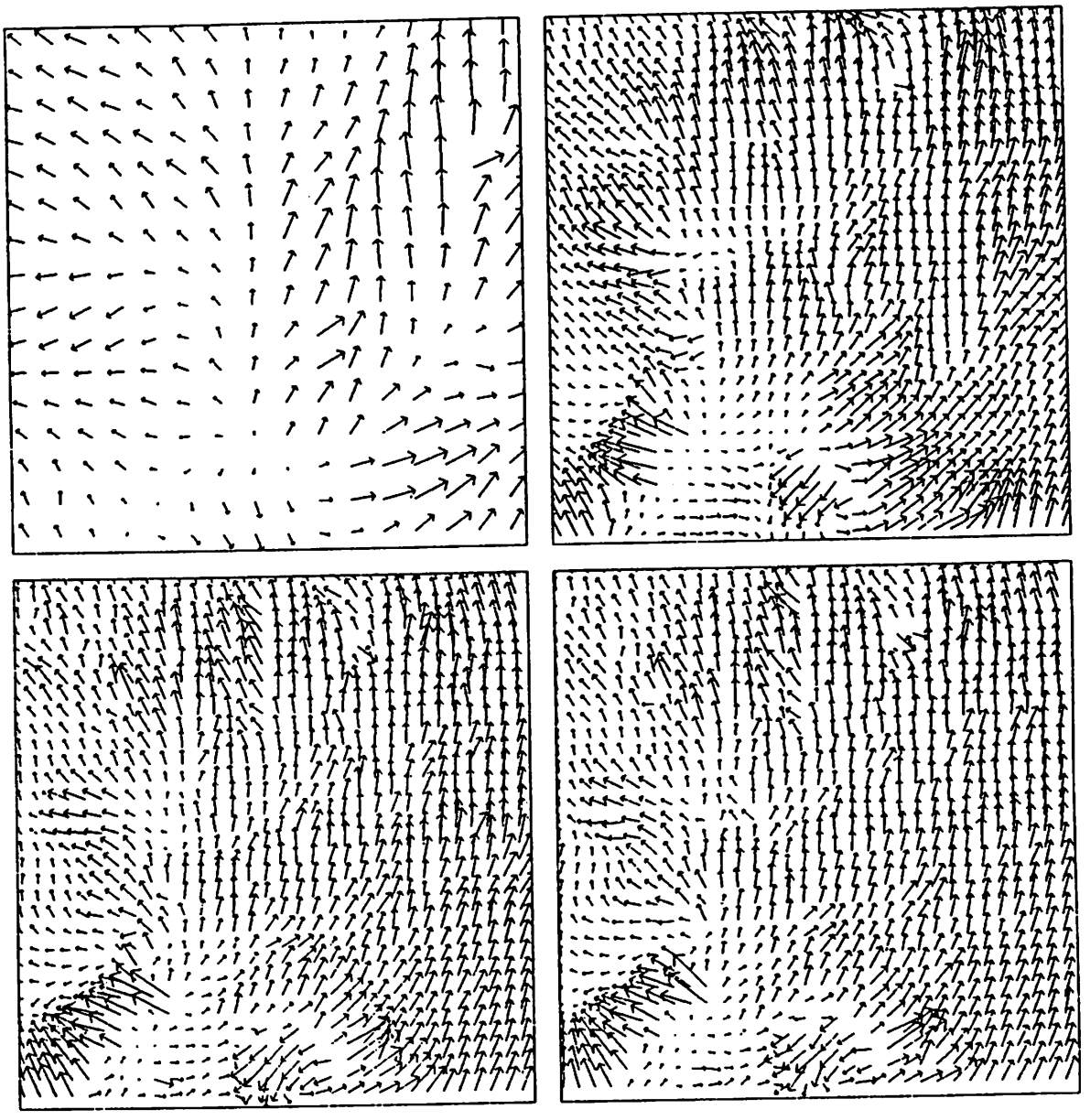


**Figure 46:** The results of the optic-fundus experiment with smoothing and with $5 \times 5$ windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

number of errors at the left-hand portion. This may due to the following reasons: First, the $5 \times 5$ windows may not be sufficiently large to eliminate the sensitivity of the SSD measure to the large differences in the mean-intensity between the two frames. Further, since the left-hand bottom portion appears darker, the local estimates are not reliable even at the coarse levels. This means that the smoothing process would have propagated the information from the more reliable information from the left-hand top portion, to the left-hand bottom portion of the image. As a result, the displacements at the top and the bottom portions point in the same direction.

Since the flow appears to smooth and continuous everywhere, a larger template window for matching may be useful to eliminate some of the errors due to changes in the mean and the contrast of the intensity function. Hence, the experiments were repeated with $13 \times 13$ Gaussian windows, which corresponds to the convolution at the first coarser level of Burt's Gaussian pyramid. The results of these experiments are shown in Figures 47 (without smoothing) and 48 (with smoothing). In Figure 49 we have also superimposed the finest level smoothed displacement field on the first input image. These results appear to be considerably closer to a rotational flow field about with the center of rotation to the left of the image. This is indicated by the gradual change in direction of the displacements as we go from the bottom to the top of the image, and also by the fact that the magnitudes decrease towards the left of the image.

Figure 50 displays the confidence $c_{max}$ with the direction vectors $\hat{e}_{max}$ superimposed and the confidence $c_{min}$ at the four levels. As expected, the confidence measure $c_{max}$ is small at homogeneous image-areas and large at linear structures and at corners, whereas $c_{min}$ is large only at corner-like points and textured-areas. The directions of $e_{max}$ are usually perpendicular to the lines (and edges) in the image. Since the input images contain distinct radial structure, the field of $\hat{e}_{max}$ vectors, which are perpendicular to the lines in the input images, appears circular.

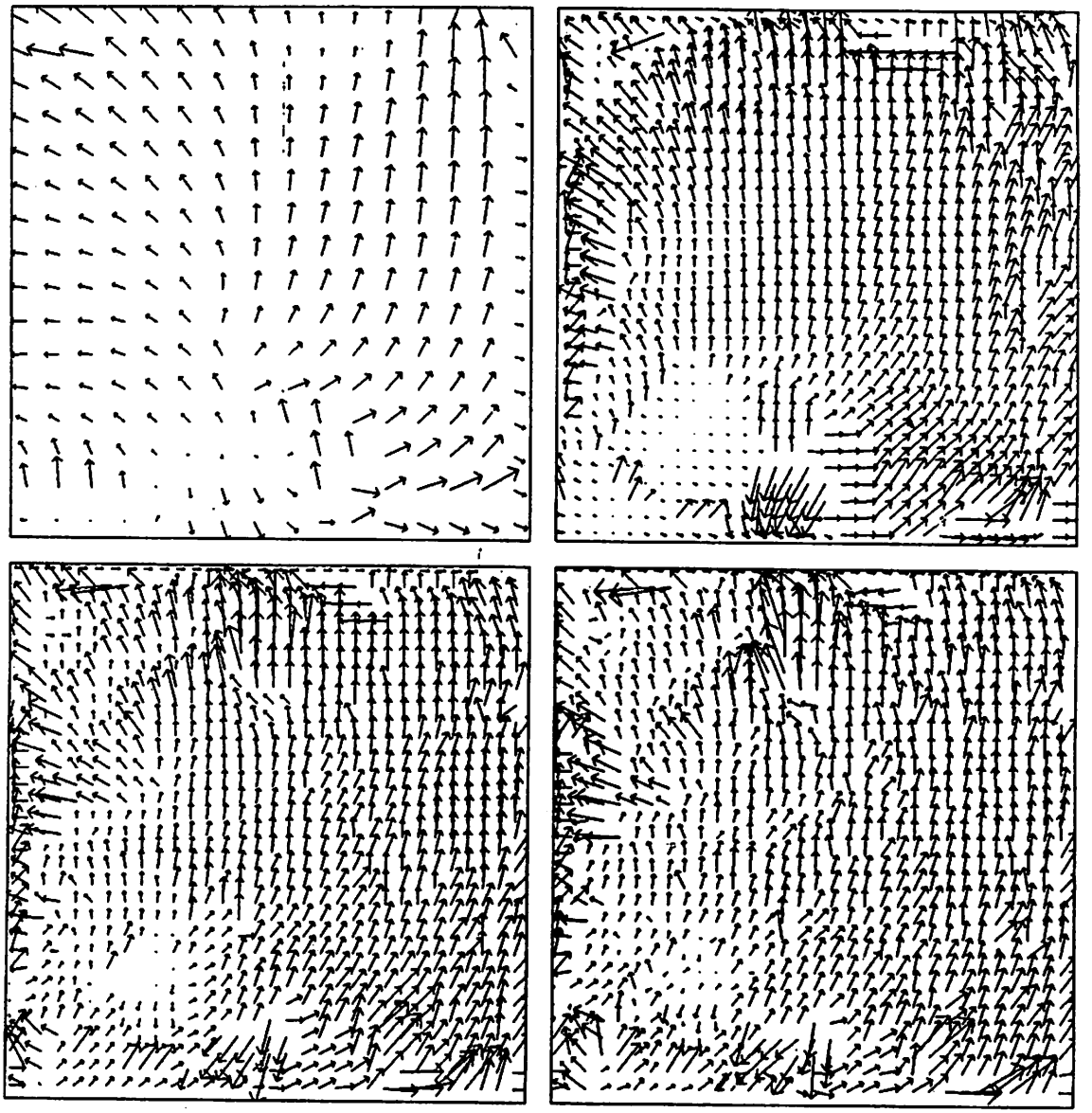


**Figure 47:** The results of the optic-fundus experiment without smoothing and with $13 \times 13$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

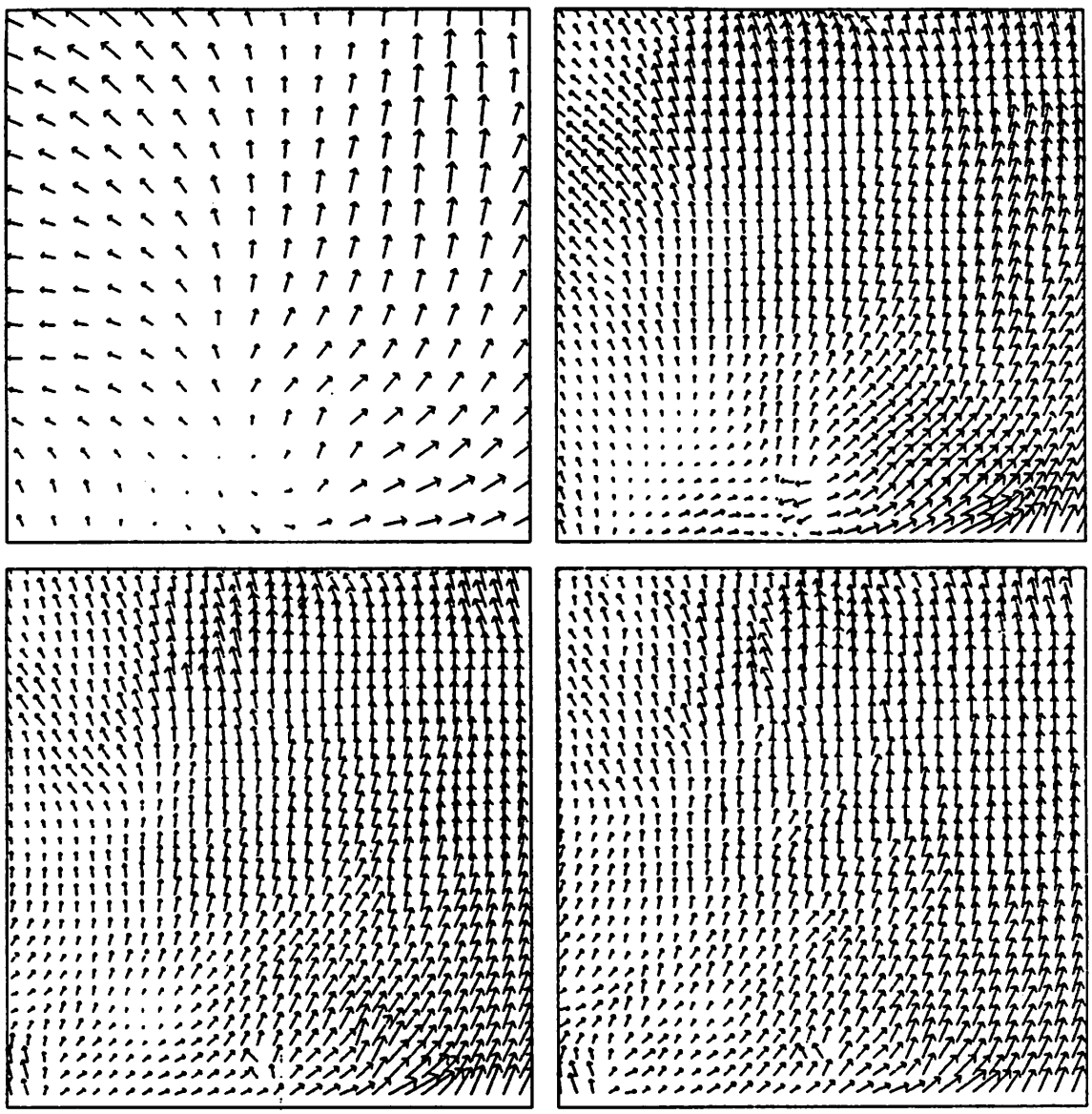


**Figure 48:** The results of the optic-fundus experiment with smoothing and with $13 \times 13$ Gaussian windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

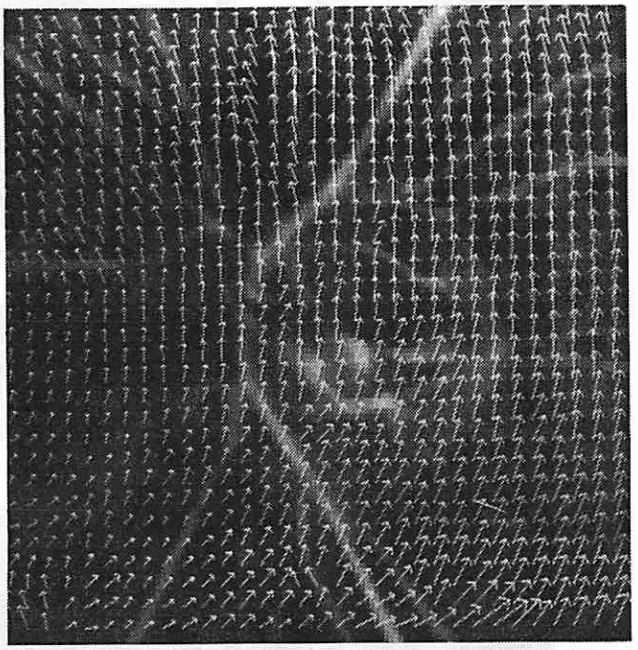


**Figure 49:** The smoothed displacement vector field, computed with the larger template window, at the finest level for the optic-fundus experiment. The field has been superimposed on the first input frame. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown.

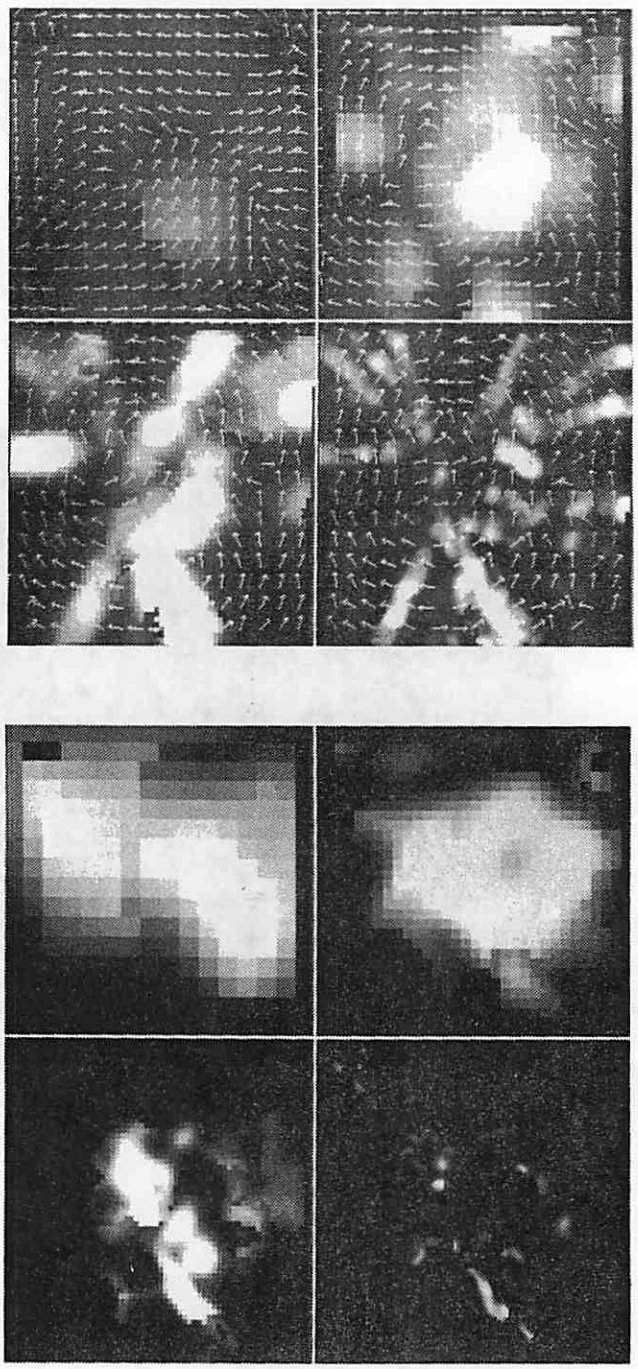


**Figure 50:** The confidence measures computed in the optic-fundus experiment with smoothing and with $13 \times 13$ windows. The confidence measures are shown at the four finest levels. The top figure shows $c_{max}$ and samples of $\hat{e}_{max}$, while the bottom figure displays $c_{min}$.

### V.2.2 The dinosaur image experiment

The input images for the dinosaur experiment are the two $128 \times 128$ resolution images shown in Figure 51. The scene consists of a toy-dinosaur, a toy-chicken in the background, and a tea-box in the foreground, all of which rest on a table-top which has a grid-pattern on it. The toy-chicken, which is somewhat hard to see in the images shown here, is behind the neck area of the toy-dinosaur. The 3-D motion between the two frames consists of a translation of the camera to the right along with a leftward rotation about the vertical axis (in order to bring the scene back into view), as well as an independent movement of the dinosaur. This scene is of interest for the obvious reason that it contains a distinct and prominent independently moving object besides a complex camera motion.

Figure 52 shows the displacement fields at the four levels of the pyramid computed by the hierarchical matching process *without* smoothing, while Figure 53 displays the displacement field at same four levels produced by the hierarchical algorithm *with* smoothing. Figure 54 displays the displacement field at the finest resolution superimposed on the first frame. Figure 55 displays the confidence measure $c_{max}$ at the four levels with the direction vectors $\hat{e}_{max}$ superimposed, and the confidence measure $c_{min}$.

It is evident from the displacement field shown in Figure 54 that the algorithm performs remarkably well in this real image containing complex motion. Note that while the toy-chicken and the tea-box undergo the same 3-D relative motion with respect to the camera, (i.e., a translation parallel to the image plane combined with a rotation about the vertical axis), the movement of their images appear to be in opposite directions. This is because, while the leftward rotation of the camera induces a rightward image-flow in both the regions, the effect of the compensatory rightward translation is greater on the image of the tea-box, since it is closer to the camera. Figure 54 shows that our algorithm has correctly determined the image-displacements of these two objects. It is also clear that the independent

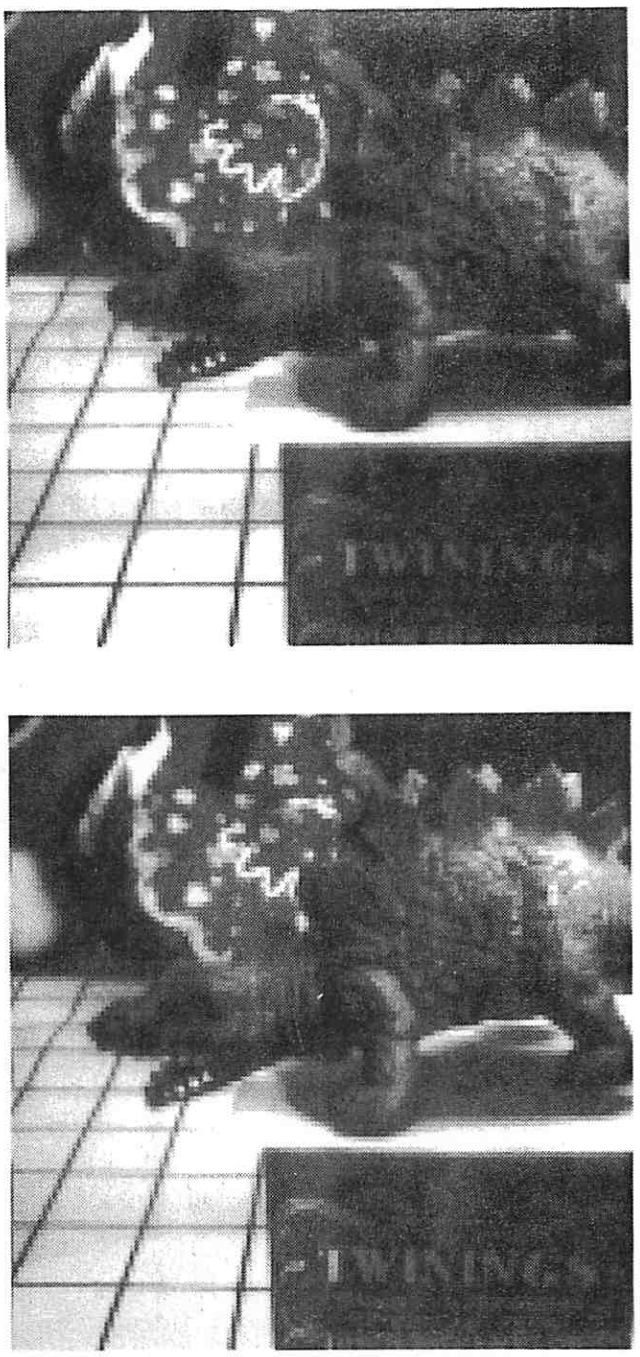


**Figure 51:** The dinosaur-image experiment: input images.

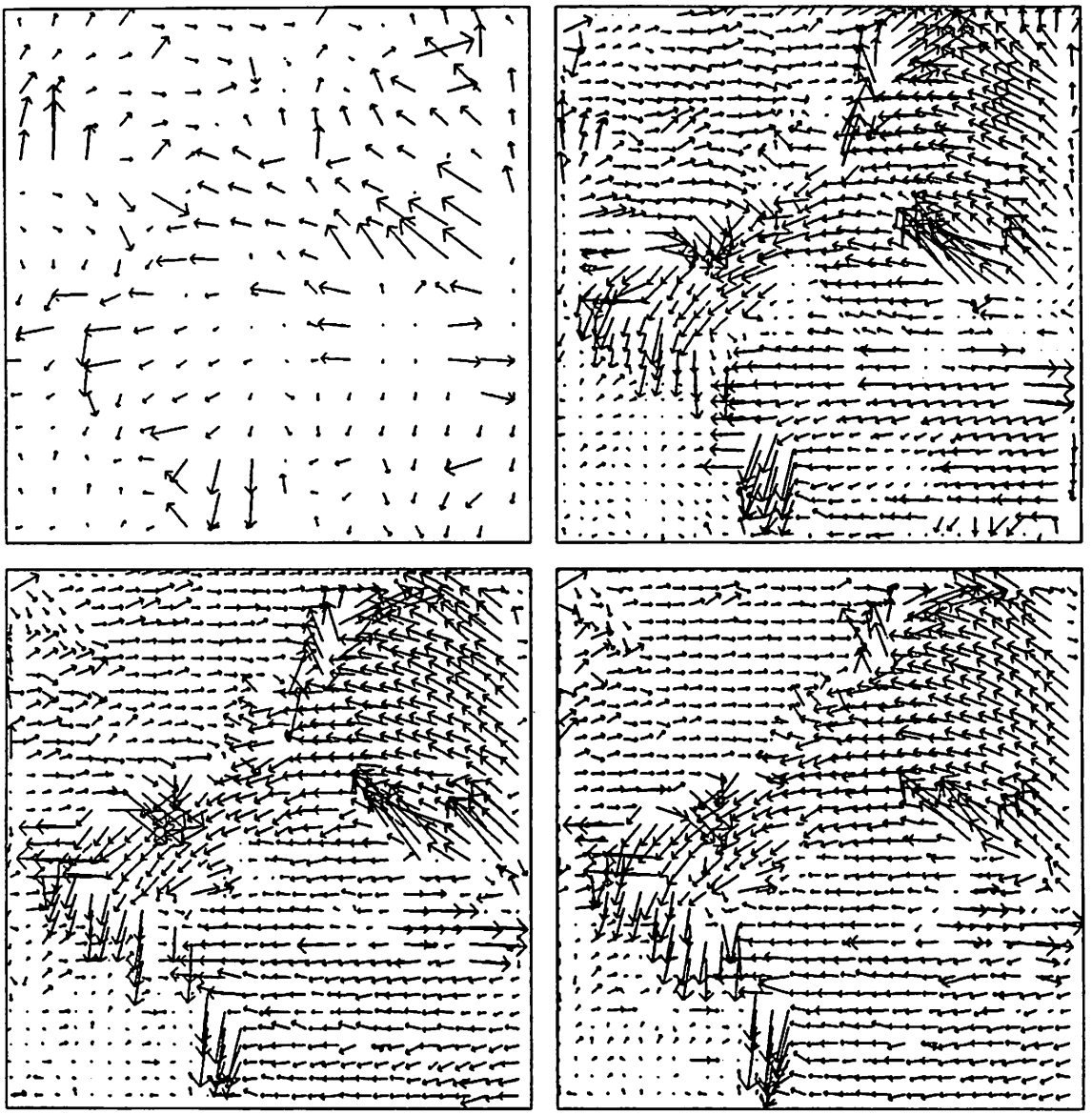


**Figure 52:** The results of the dinosaur-image experiment without smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

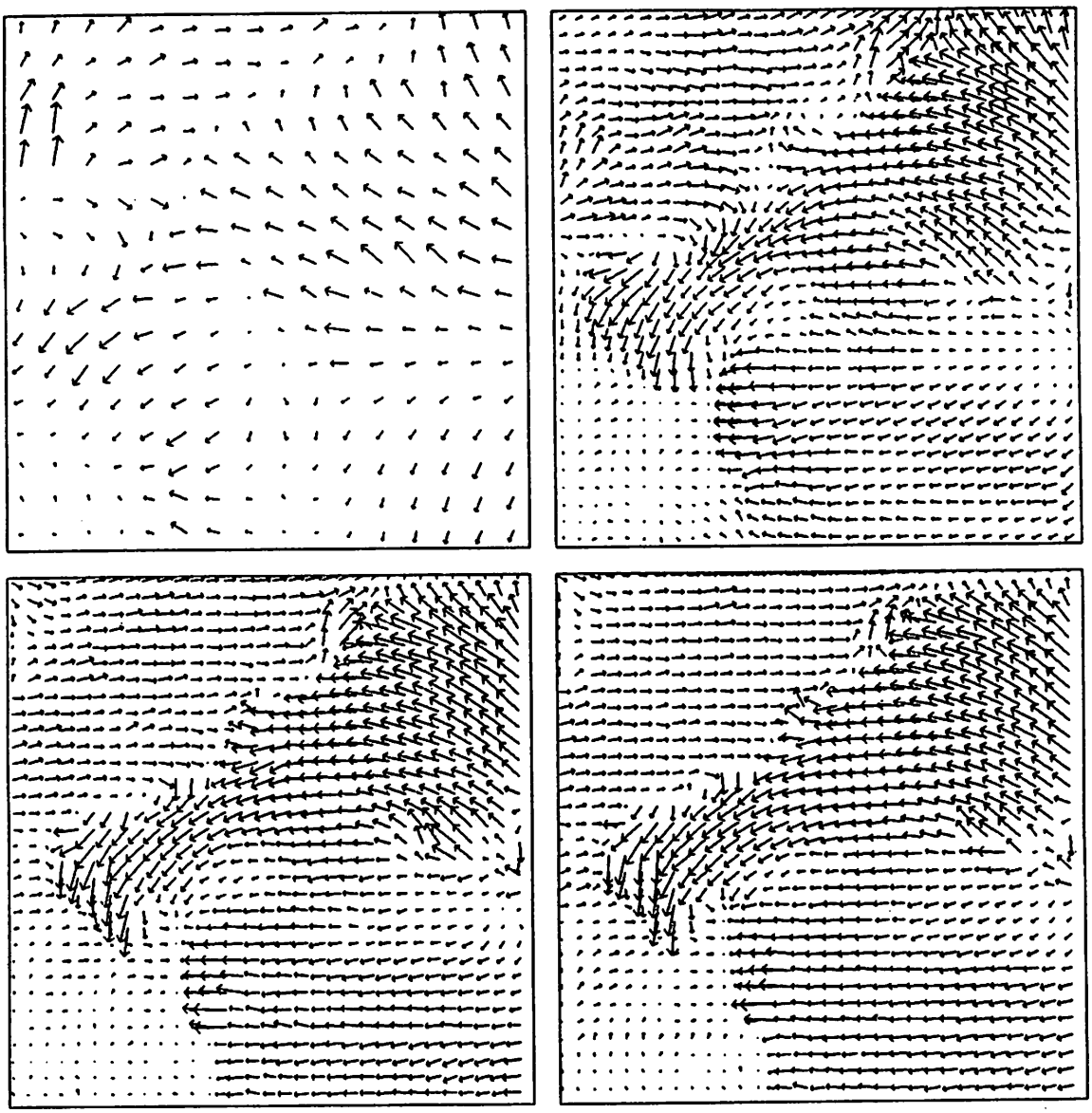


**Figure 53:** The results of the dinosaur-image experiment with smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

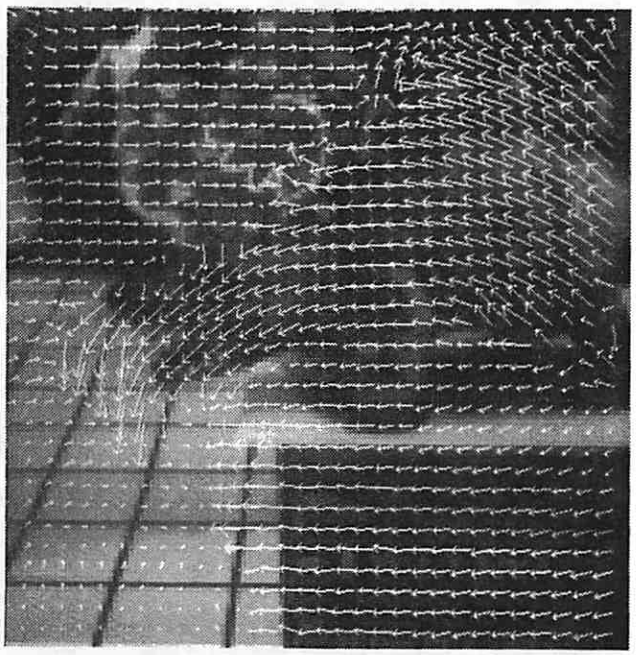


**Figure 54:** The smoothed displacement vector field at the finest level for the dinosaur-image experiment superimposed on the first input frame. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown.

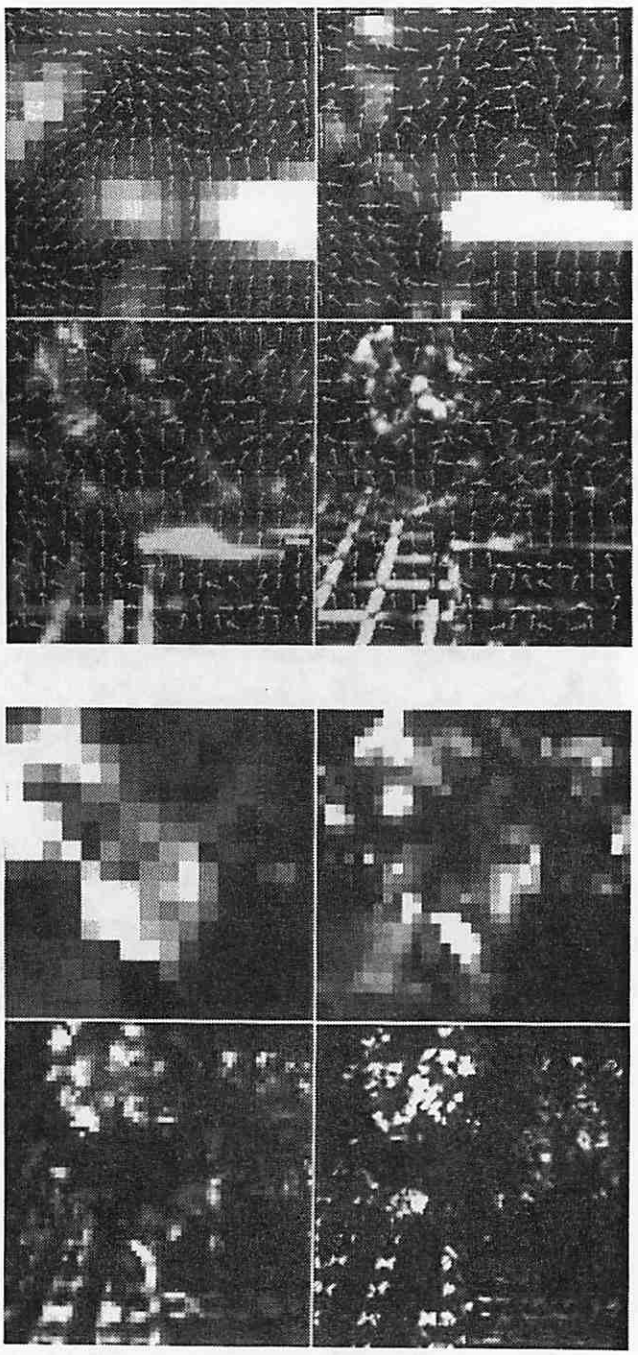


**Figure 55:** The confidence-measures computed in the dinosaur-image experiment with smoothing. The confidence measure are shown at the four finest levels. The top figure shows $c_{max}$ and samples of $\hat{e}_{max}$, while the bottom figure displays $c_{min}$.

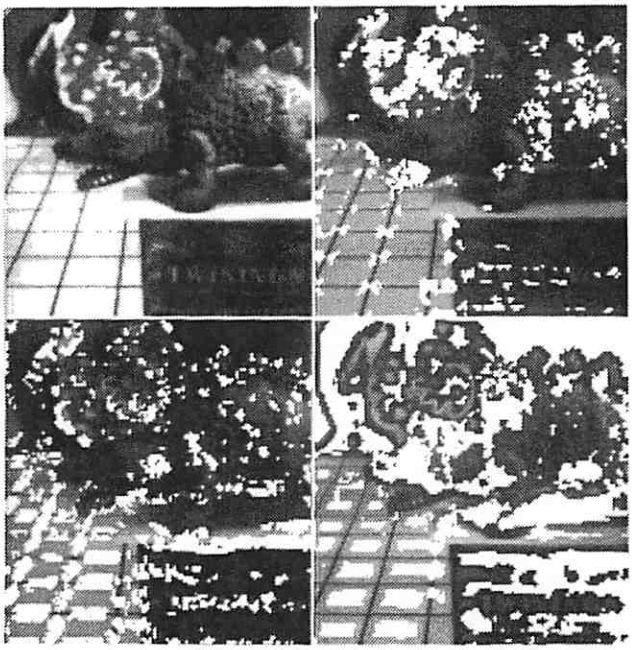


**Figure 56:** The dinosaur-image experiment with smoothing: The classification of pixels as corners, edges, or homogeneous areas. The top left quadrant contains the input image. In the images shown in the top-right, bottom-left, and bottom-right quadrants, the corners, edges, and the homogeneous areas have respectively been highlighted.

movement of the dinosaur has been successfully computed.

The expected behaviors of the confidence measures at corners, edges, and homogeneous areas are confirmed by the displays in Figure 55. In order to make these behaviors more explicit, we attempted to classify the image pixels as shown in Figure 56, as corners, edges, or homogeneous areas according to the following criteria:

- if $c_{min} > 0.5$ the pixel is classified as a corner,
- if $\frac{c_{max}}{c_{min}} > 100$ and $c_{max} > 1$, the pixel is classified as an edge, and
- if $c_{max} < 0.5$ and $\frac{c_{max}}{c_{min}} < 5$ the pixel is classified as a point in a homogeneous area.

The improvement obtained by the smoothing process is easily seen by comparing the results shown in Figures 52 and 53. Note that at the two coarsest resolutions, the displacement field has been smoothed across surface and object boundaries, whereas at the finer-levels there are sudden changes near such boundaries. This is due to the use of overlapped pyramid projection strategy, as well as the fact that the input images contain significant contrast at high-frequencies. Hence, the finer-level matching processes were able to correct some of the errors made at the coarser-levels. In particular, note that the boundary between the chicken and the dinosaur has been maintained during the finer-level smoothing processes.

Although the area on top of the dinosaur is part of the background, the vectors in that area seem to be influenced by the motion of the dinosaur. Similarly, the area of the floor just left of the tea-box has displacements that are obviously incorrect. This is because both these areas are somewhat homogeneous, and are adjacent to areas containing high-contrast information. In addition, parts of the floor have been occluded, or are near the occlusion boundary, and therefore do not

have reliable local estimates. Hence, the more reliable neighboring vectors have been propagated by the smoothing process to these areas with unreliable local estimates. As noted in chapter IV, such problems due to occlusion are pervasive in the field of motion analysis.

Note that near the top of the tea-box, one of the lines from the grid-pattern is visible in both the frames, although its displacement is incorrect. This error occurs because the vertical line on the floor is adjacent to the occlusion boundary, and the intensity structure of its neighborhood undergoes significant changes between the two frames. The problems here have been made more severe by the use of a coarse-to-fine control strategy, since as noted in chapter III, at low-frequencies the area affected by the occlusion increases; it appears that even the use of the overlapped pyramid projection strategy has not corrected these errors. However, as illustrated in Figures 55 and 56 the confidences associated with the incorrect displacements on the line are small. In particular, note that in Figure 56, this line has been classified as a homogeneous area. This is because our simple scheme for the classification does not discriminate between low-confidences due to homogeneous intensity structures and due to occlusion. As suggested in chapter IV, a comparison of the auto- and the cross- SSD surfaces would reveal that this area contains a linear structure, and is either occluded or is adjacent to an occlusion boundary.

The area of the floor just below the nose of the dinosaur also has incorrect displacements. In this case, however, the problem is not due to the smoothness constraint. Instead, it arises because the grid-pattern on the floor is periodic, and the difference in the displacement of the nose of the dinosaur and the floor is approximately equal to the period of the grid-pattern. Hence, at the coarse-levels of processing, the grid-pattern near the nose of the dinosaur appears to have moved one-period, whereas its actual image motion is much smaller (almost zero). This may be a harder problem to solve by a low-level two-frame matching technique, because all displacements which are multiples of the period of the grid-pattern

are equally valid. In order to resolve between them, either higher-level texture-based grouping processes, or constraints involving the temporal coherence of the movement may be necessary. The mechanisms for incorporating the temporal coherence assumption will be discussed in chapter VI.

### V.2.3 The road-scene experiment

This experiment illustrates the performance of the algorithm on an outdoor natural scene. The two $128 \times 128$ resolution input images for this experiment are shown in Figure 57. These are from the UMass Image Library, and were taken from a moving vehicle. The motion is primarily due to a pure translation of the camera towards a point slightly to the left of the visual field. Although there is a car in front which undergoes a slightly different motion, its independent motion appears negligible (perhaps due to its distance the camera).

Figure 58 shows the displacement field obtained by the hierarchical algorithm without using the smoothness constraint. Figure 59 shows the displacement fields obtained by the hierarchical matching algorithm (with smoothing) at the four levels, and Figure 60 shows the finest level results superimposed on the first input image. Figure 61 shows the maximum (with direction vectors superimposed) and minimum confidence measures at the four levels.

Once again, in order to study the behaviors of the confidence measure at corners, edges, and homogeneous, we attempted to classify the image pixels according to the following criteria:

- if $c_{min} > 0.05$ the pixel is classified as a corner,
- if $\frac{c_{max}}{c_{min}} > 20$ and $c_{max} > 0.1$, the pixel is classified as an edge, and
- if $c_{max} < 0.05$ and $\frac{c_{max}}{c_{min}} < 5$ the pixel is classified as a point in a homogeneous area.

**Figure 57:** The road-scene experiment: input images.

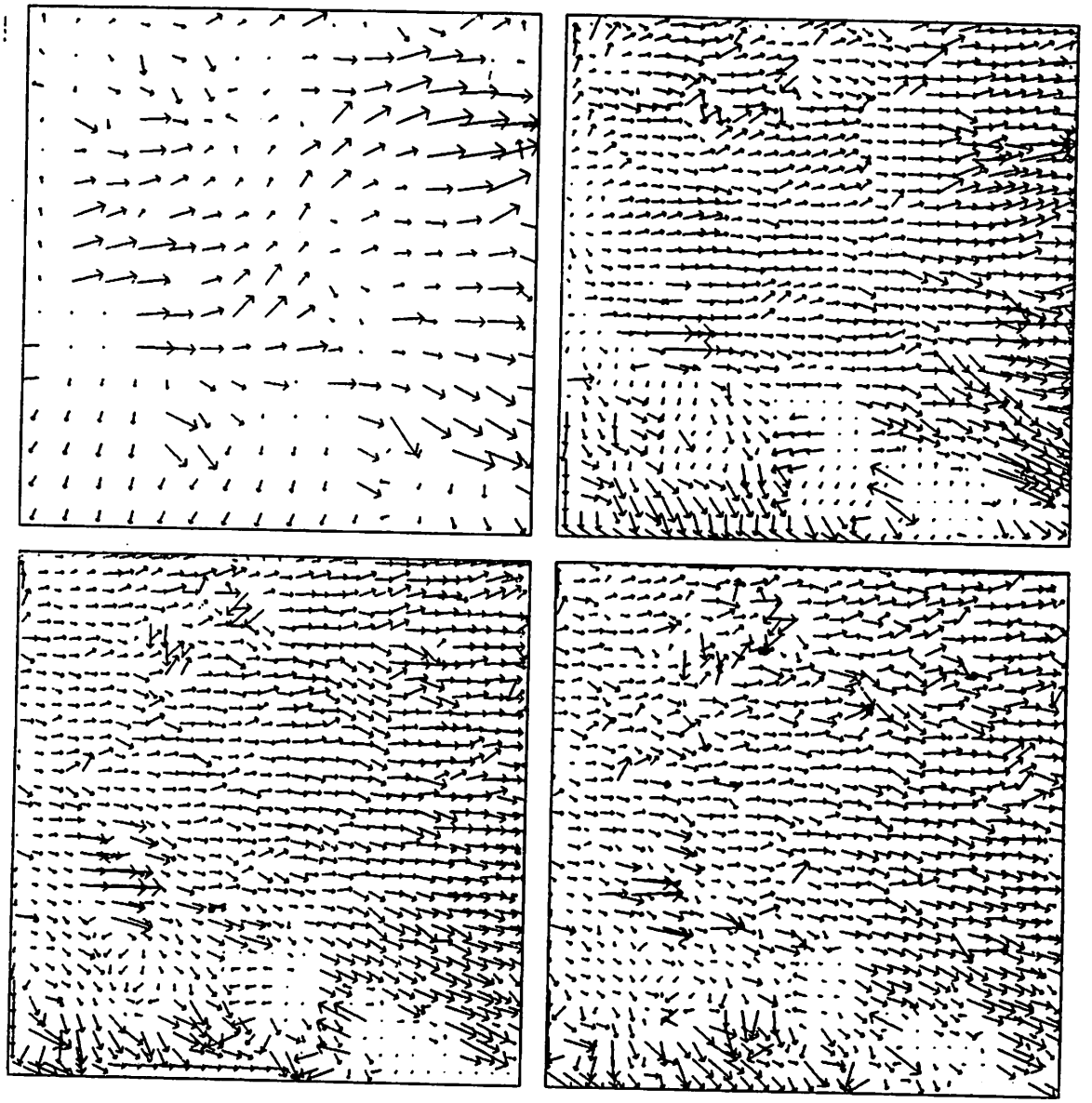


**Figure 58:** The results of the road-scene experiment without smoothing and with 5 × 5 Gaussian template windows. In order to enhance visibility only a 32 × 32 sample of the displacements have been shown at levels 6 and 7.

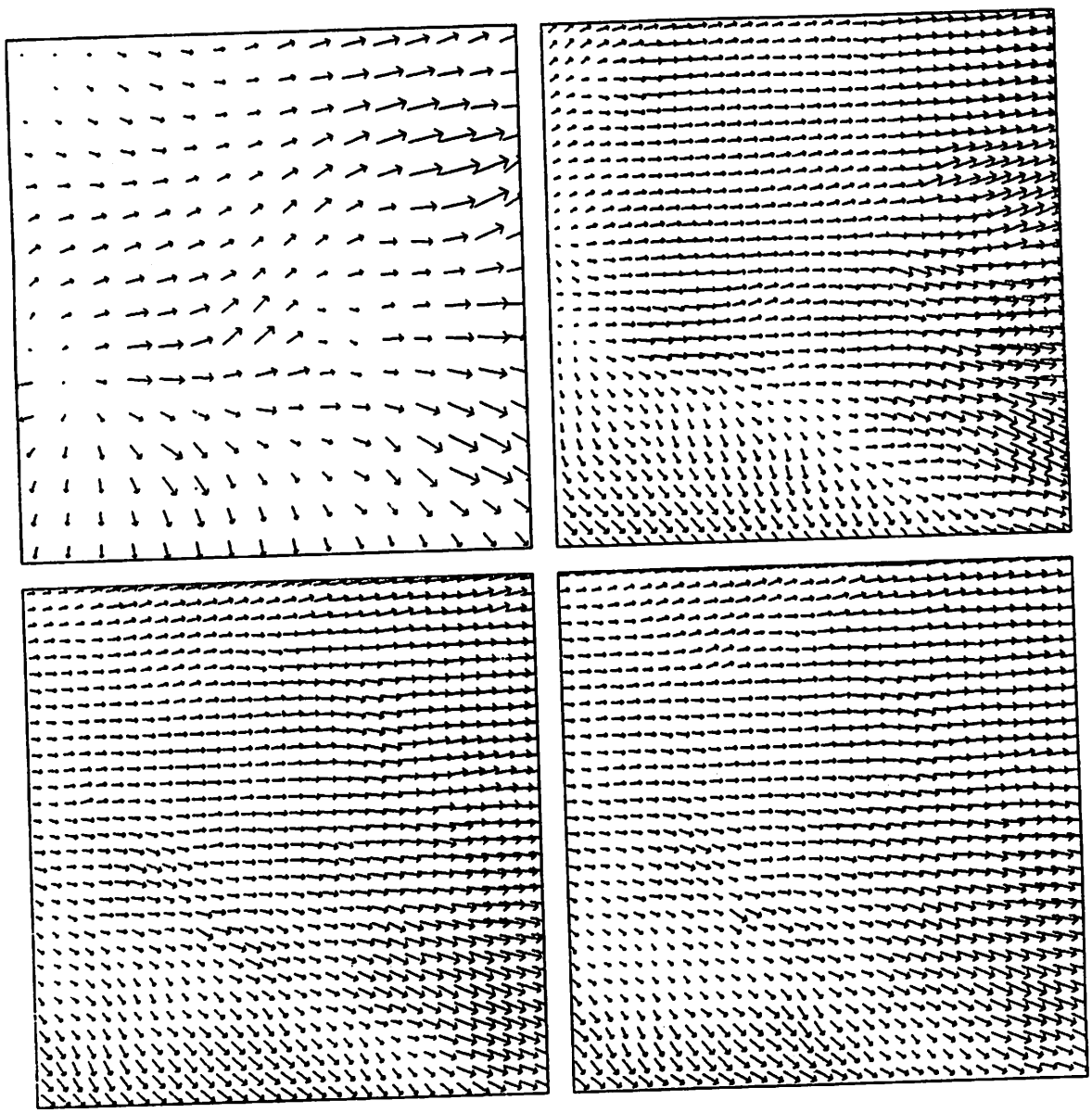


**Figure 59:** The results of the road-scene experiment with smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

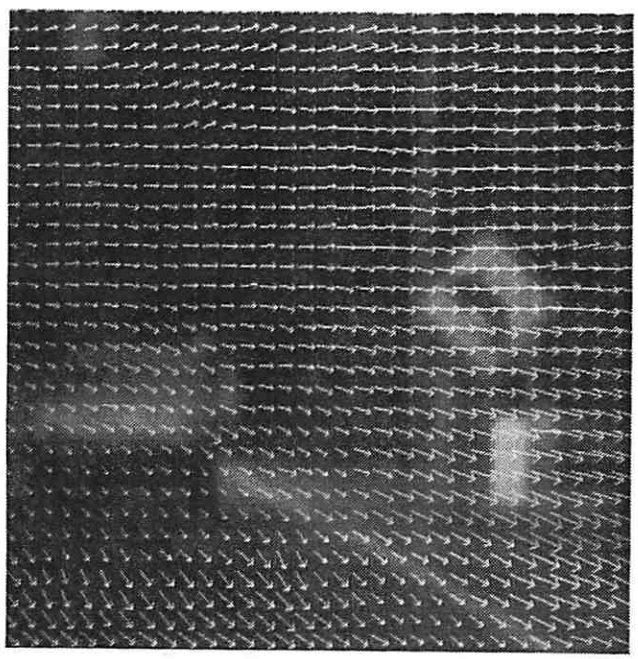


**Figure 60:** The smoothed displacement vector field at the finest level of the road-scene experiment superimposed on the first input frame. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown.

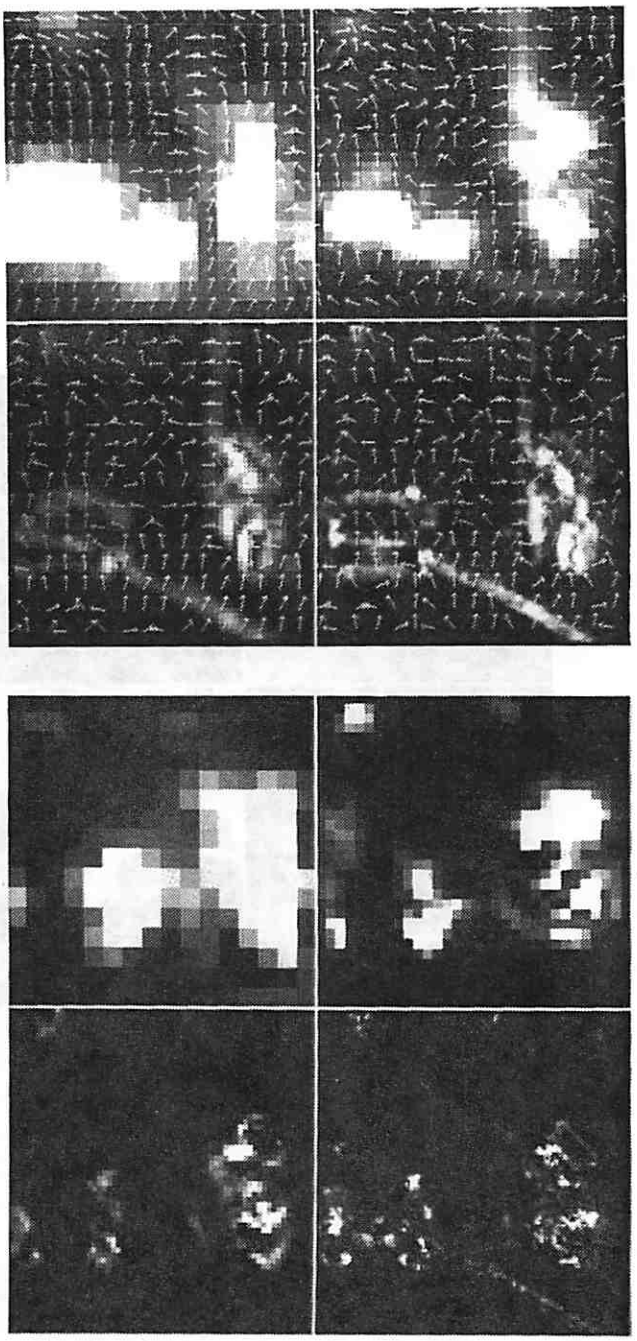


**Figure 61:** The confidence measures computed for the road-scene experiment with smoothing. The confidence measures are shown at the four finest levels. The top figure shows $c_{max}$ and samples of $\hat{e}_{max}$, while the bottom figure displays $c_{min}$.

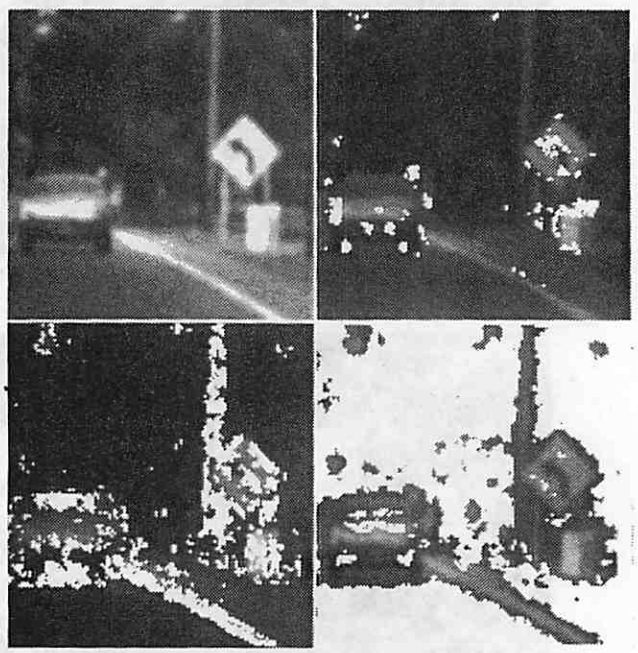


**Figure 62:** The road-scene experiment: The classification of pixels as corners, edges, or homogeneous areas. The top left quadrant contains the input image. In the images shown in the top-right, bottom-left, and bottom-right quadrants, the corners, edges, and the homogeneous areas have respectively been highlighted.

Figure 62 displays the results of this classification.

Note that the threshold values used for the classification of pixels are about a tenth of those used in the dinosaur-image experiment. Also, in Figure 61, we have used a dynamic range of $(0, 0.8)$ for the finest-level $c_{max}$ display, whereas for all the other levels we have used a range of $(0, 4.0)$. By comparing the respective input images, it can be seen that the road-scene images are somewhat more fuzzy. This indicates that these images have less energy at high-frequencies; hence, the overall confidence levels are lower than in the case of the dinosaur-images.

The displacement fields, the confidence measures, and the classifications of the pixels once again show the characteristics expected of them. The improvement in the results due to the use of the smoothness constraint is also evident. In particular, note the improvement in the homogeneous areas of the sky, the low-contrast areas of the trees, and the linear structure along the road.

Since the 3-D motion between the two frames used in this experiment is almost a pure translation, it is possible to use Lawton's translational algorithm [61,83], which is considerably more efficient. Based purely on a visual inspection of our results, we believe that the focus-of-expansion determined from the displacement field shown in Figure 60 will coincide with the results obtained by applying Lawton's algorithm.

### V.2.4 The hallway-scene experiment

The input images for this experiment are shown in Figure 63. These $256 \times 256$ pixel resolution images were produced at the UMass computer vision laboratory. Once again all image motion is due a camera undergoing translational motion. The reason for choosing this image-pair is the presence of the many long linear-structures in the image. Since the confidence measure separates such areas from corners and homogeneous areas, it is interesting to study its behavior in this experiment.

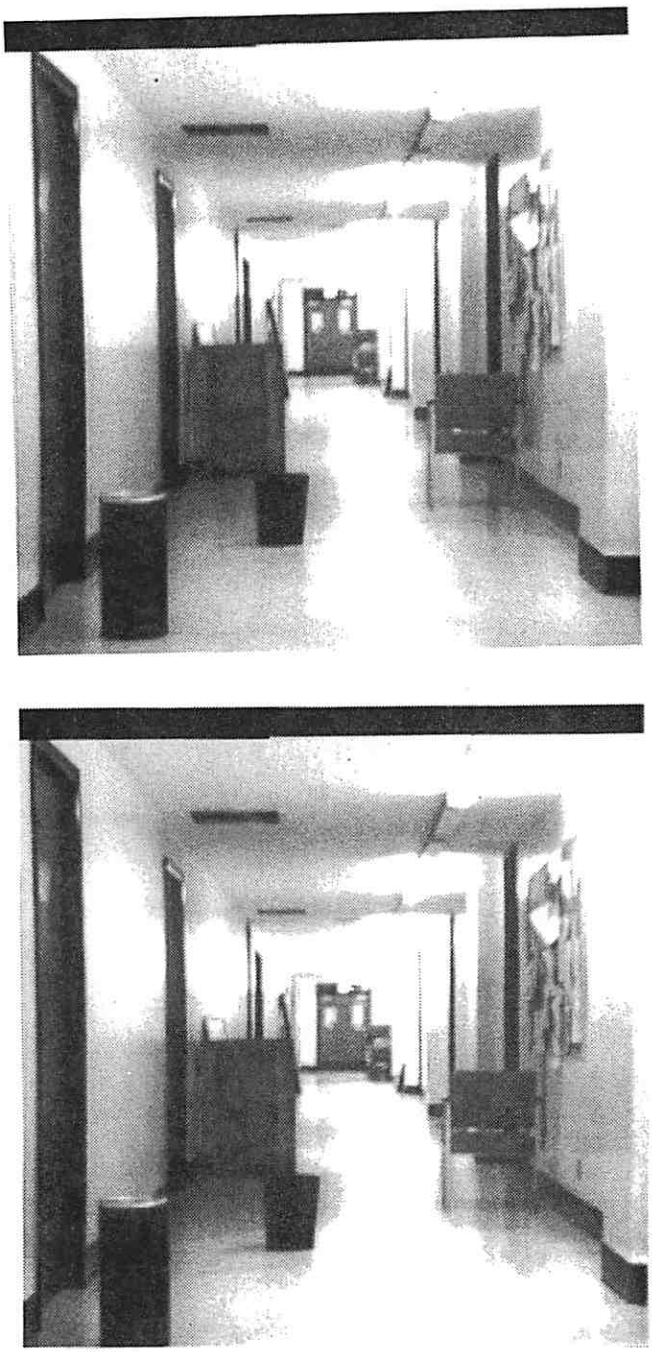


**Figure 63:** The hallway-scene experiment: input images.

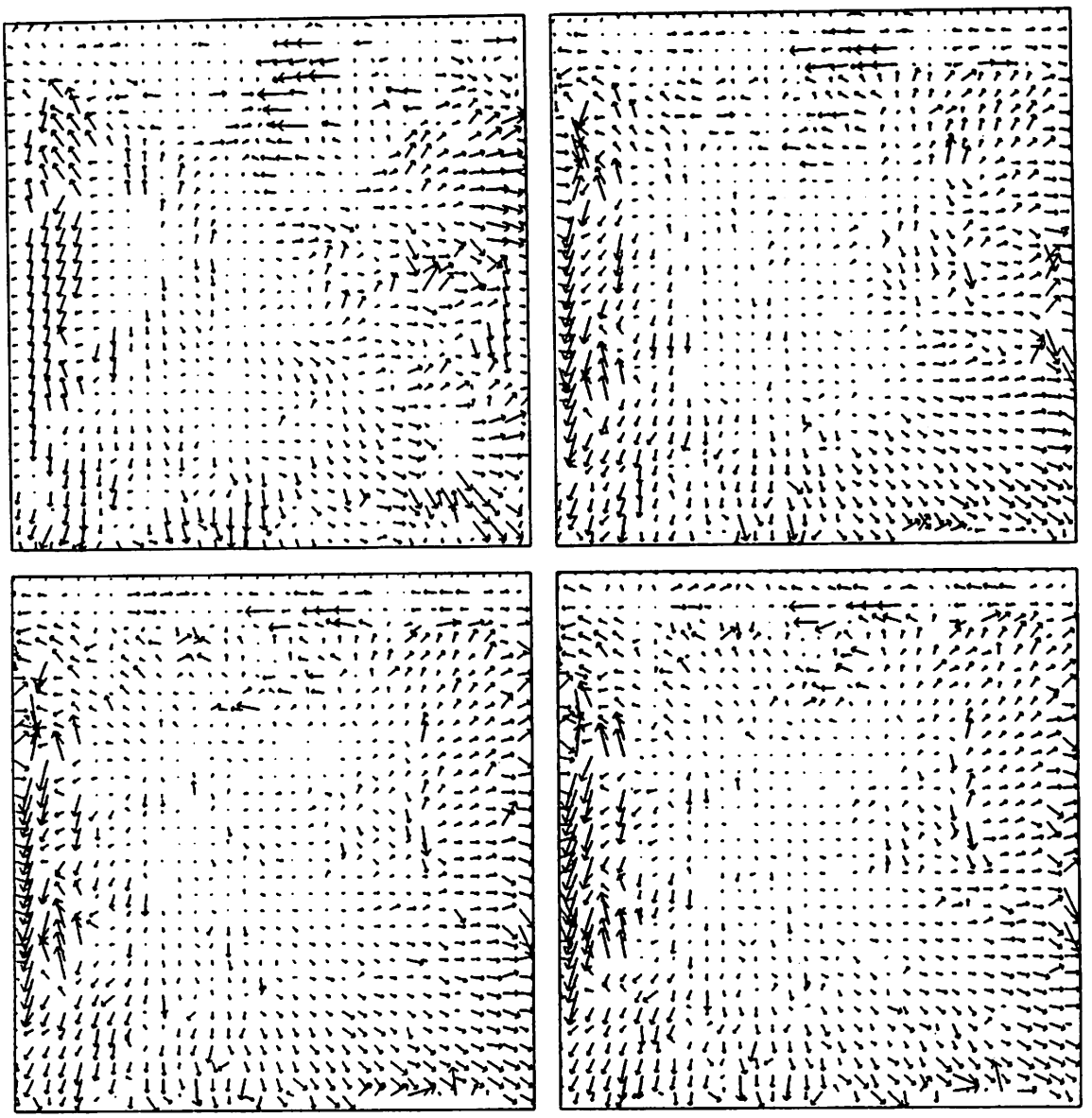


**Figure 64:** The results of the hallway-scene experiment without smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

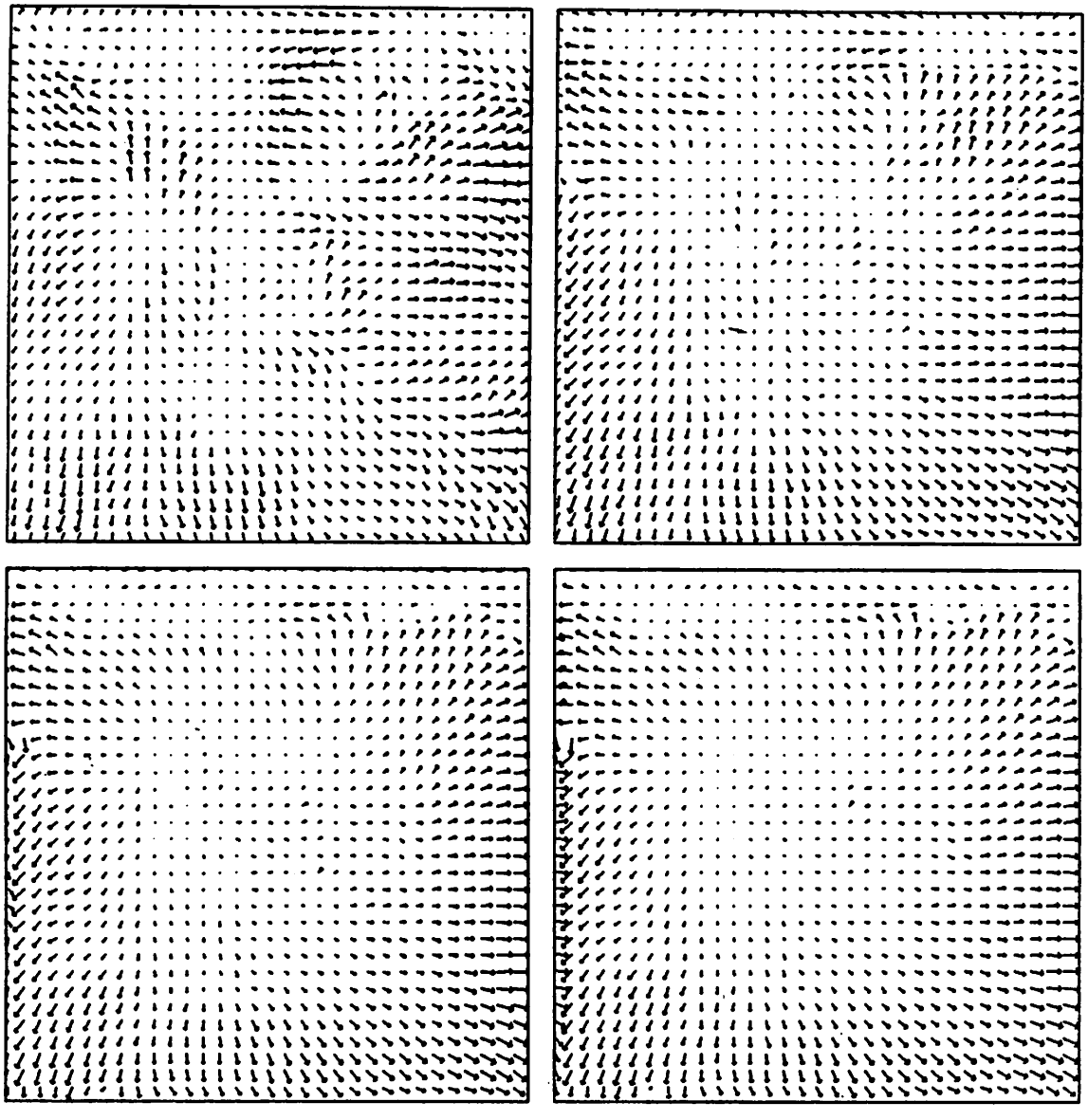


**Figure 65:** The results of the hallway-scene experiment with smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

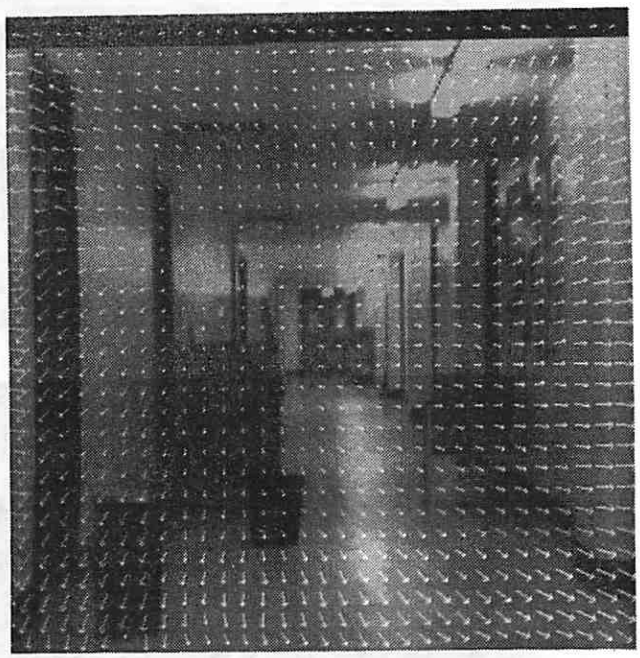


**Figure 66:** The finest-level smoothed displacements for the hallway-scene experiment superimposed on the first input frame. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown.

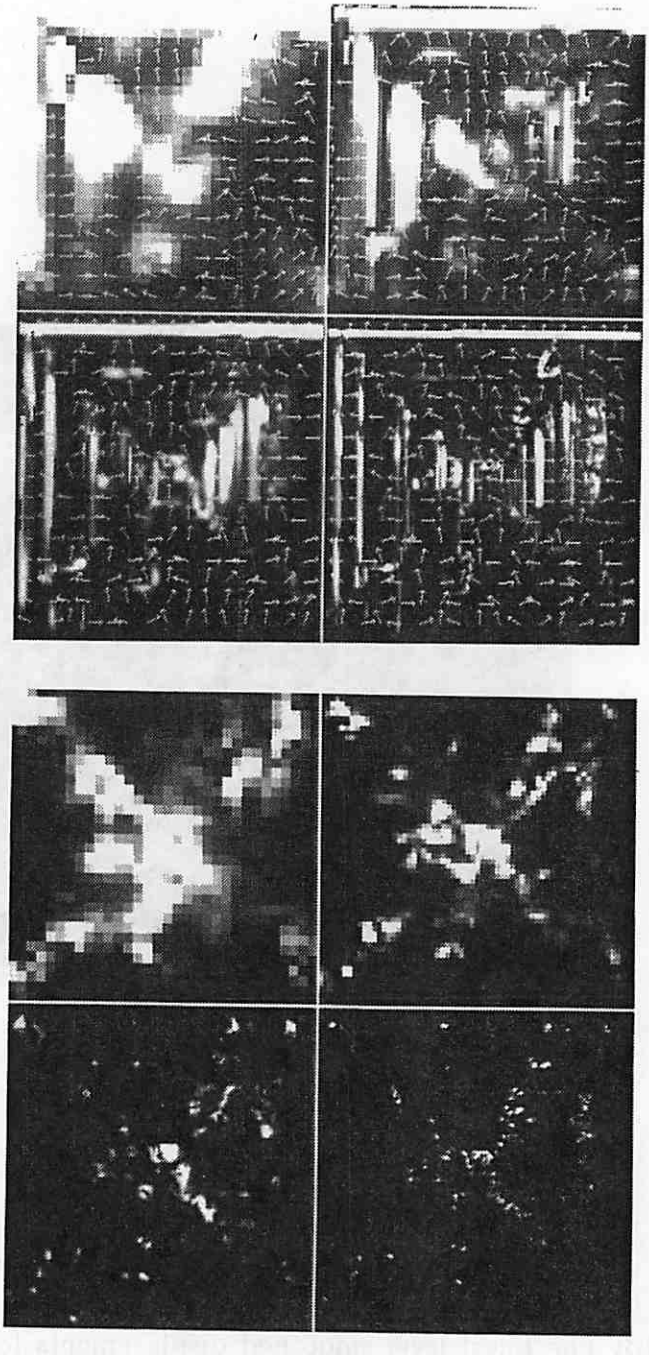


**Figure 67:** The confidence measures computed for the hallway-scene experiment with smoothing. The confidence measures are shown at the four finest levels. The top figure shows $c_{max}$ and samples of $\hat{e}_{max}$, while the bottom figure displays $c_{min}$.

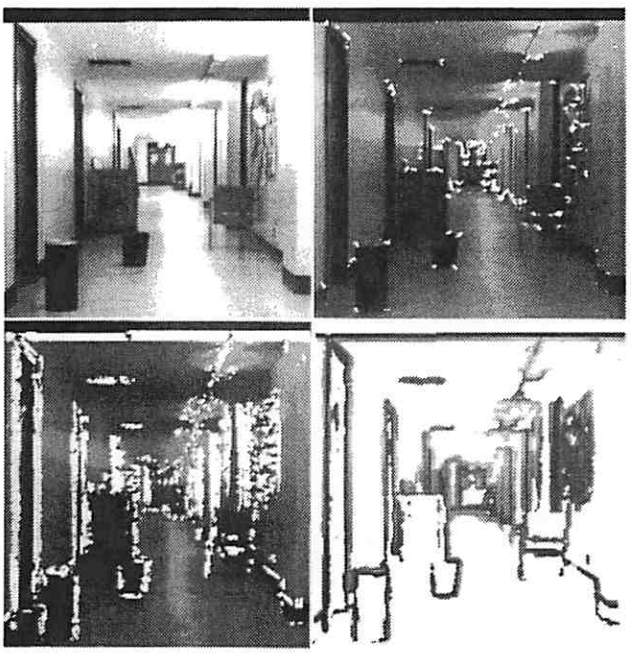


**Figure 68:** The hallway-scene experiment: The classification of pixels as corners, edges, or homogeneous areas. The top left quadrant contains the input image. In the images shown in the top-right, bottom-left, and bottom-right quadrants, the corners, edges, and the homogeneous areas have respectively been highlighted.

Figure 64 shows the displacement fields obtained at the four finest levels, when the smoothing process is not used. The results obtained by using the smoothing process are in Figure 65. The finest level smoothed-displacements have been shown superimposed on the first frame in Figure 66. Figure 67 displays the confidence measures $c_{max}$ with direction vectors $\hat{e}_{max}$ superimposed on them, and $c_{min}$ at the four levels.

Just as in the case of the dinosaur-image experiment and the road-scene experiment, we attempted to classify the pixels of the image as corners, edges, and homogeneous according to the following criteria:

- if $c_{min} > 0.5$ the pixel is classified as a corner,
- if $\frac{c_{max}}{c_{min}} > 100$ and $c_{max} > 1$, the pixel is classified as an edge, and
- if $c_{max} < 0.5$ and $\frac{c_{max}}{c_{min}} < 5$ the pixel is classified as a point in a homogeneous area.

Figure 68 displays the results of this classification.

As mentioned above, this experiment was chosen in order to study the difference in the behaviors of our algorithm at corners and edges. Therefore, in Figures 69 and 70, we display the the displacement-vectors computed at samples of the corner and the edge pixels, which were selected according to the classification method described above. In both cases, we have shown the smoothed and the unsmoothed vectors. Note that at corner points, the smoothed and the unsmoothed displacements are identical nearly everywhere (we verified this using a multi-colored display on a graphics terminal). At edge points, as expected, the normal-components of the smoothed and the unsmoothed displacements appear to be equal, whereas at many locations, the tangential components of the unsmoothed displacements are obviously incorrect.

In order to further illustrate the accuracy of our computations, and to confirm

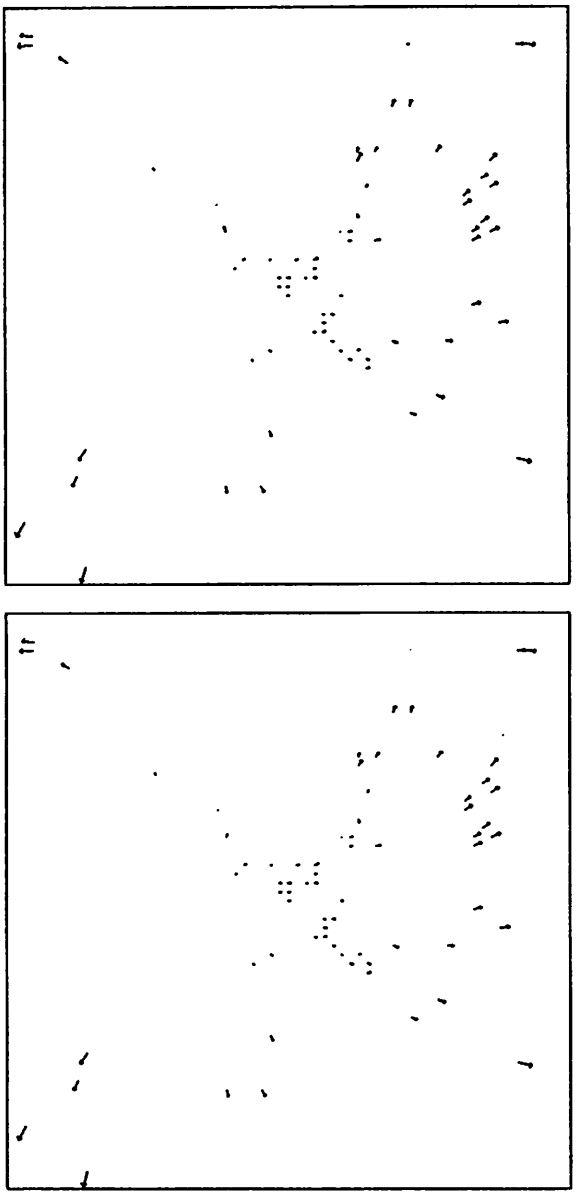


**Figure 69:** The hallway-scene experiment: The top figure shows the unsmoothed displacements at a sample of the "corner-like" pixels, while bottom figure shows the corresponding smoothed displacements.

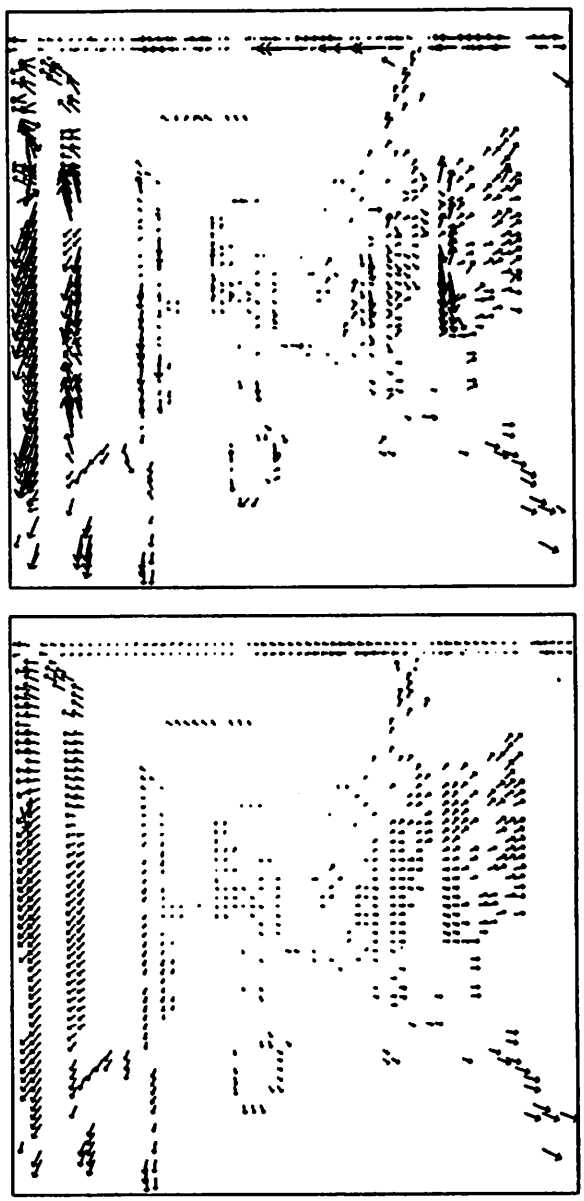


**Figure 70:** The hallway-scene experiment: The top figure shows the unsmoothed displacements at a sample of the "edge-like" pixels, while bottom figure shows the corresponding smoothed displacements.

**Figure 71:** The hallway-scene experiment: The lines extracted by Boldt's algorithm from the two input frames. The thick lines are those from the first frame, while the thin lines are from the second input frame. The area within the rectangular box in the upper left corner of the image will be more closely examined in the next two figures.

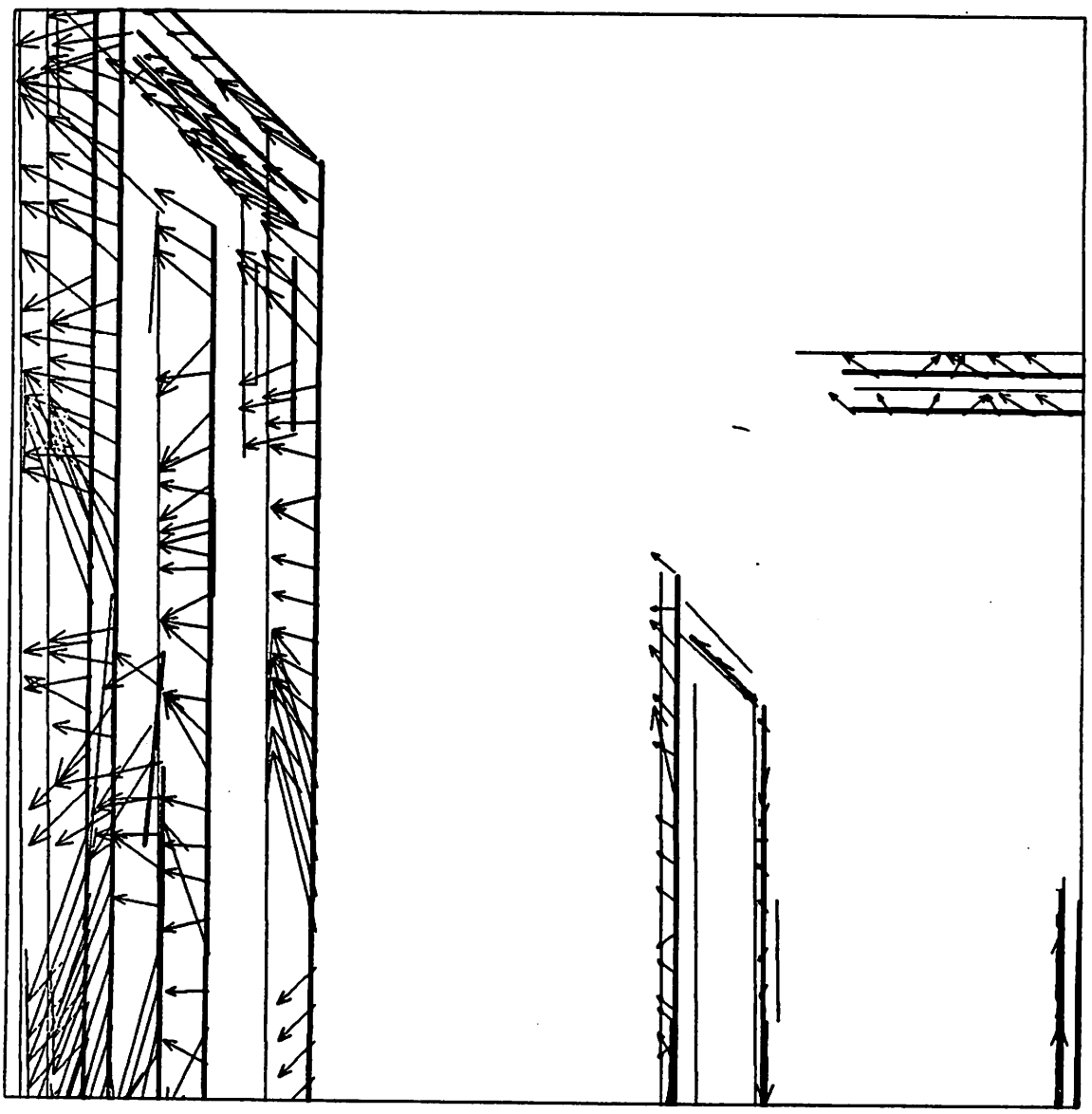


**Figure 72:** The hallway-scene experiment: The unsmoothed displacements superimposed on the lines extracted by Boldt's algorithm.

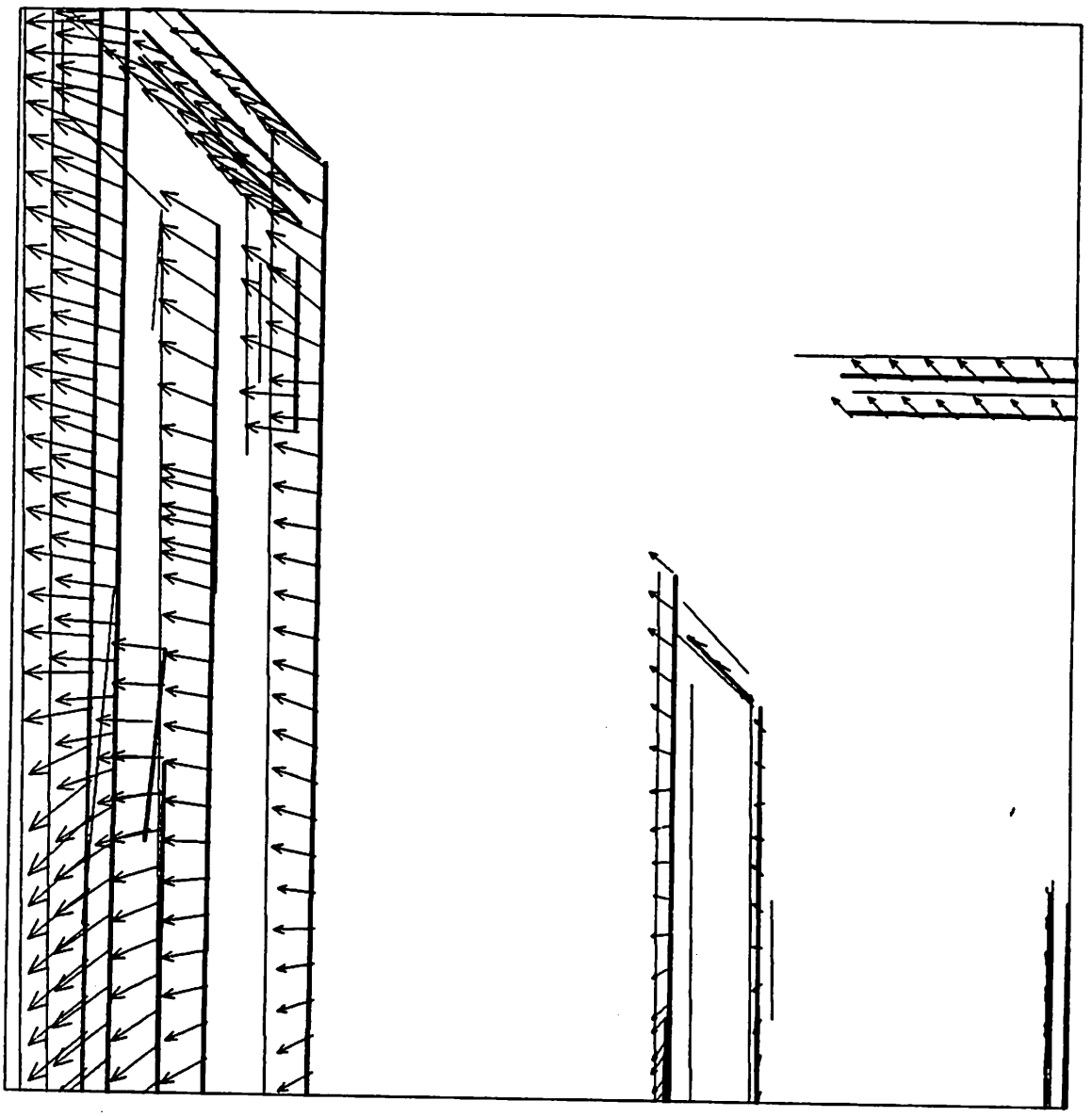


**Figure 73:** The hallway-scene experiment: The smoothed displacements superimposed on the lines extracted by Boldt's algorithm.

our statement regarding the normal and the tangential components, we superimposed our displacement vectors on the set of lines obtained from the two input images by a line-extraction and grouping algorithm developed by Boldt [17,119]. Figure 71 displays the lines extracted from the two images; the lines extracted from the first frame are shown as thick dark lines, while the thinner lines are those obtained from the second frame. Only lines whose associated contrast is greater than 15 grey-levels and which are longer than 7 pixels have been shown. In Figures 72 and 73, we have superimposed our displacement vectors on the sets of lines obtained from the two-frames in a $90 \times 90$ pixel area, which is marked in Figure 71. Figure 72 displays the *unsmoothed* displacement vectors for a sample of pixels which lie on the lines belonging to the first frame, while Figure 73 shows the *smoothed* displacement vectors for the same sample of pixels.

It is obvious from Figure 73, that the lines are correctly matched by our displacement vectors. From Figure 72, it is clear that the normal-components of the unsmoothed vectors are also correct, whereas their tangential components are often incorrect. Finally, the remarkable consistency of the results obtained by Boldt's line-extraction algorithm and our matching algorithm suggests that it may be possible to combine them to extract stable line tokens from image sequences. This idea is currently being pursued at the VISIONS research laboratory at the University of Massachusetts.

### V.2.5 The office-scene experiment

The last experiment was performed on the two input images shown in Figure 74. These $256 \times 256$ images were obtained from SRI International; the scene is that of an office with objects at various distances from the camera. All the image motion is due to a translation of the camera to the left between the two frames. Although the 3-D motion is rather simple, the image displacement field computations are complicated by the presence of a number of surfaces at various depths, and a number of objects of various sizes.

The smoothed displacements fields at the four levels of the pyramid are shown in Figure 75, while Figure 76 shows the finest level displacement field superimposed on the first frame. The confidence measures are shown in Figure 77.

A qualitative examination of the displacement field indicates that the vectors are predominantly oriented rightward with almost negligible vertical components; this is to be expected since the camera motion was simply a translation leftwards. The depth variations of the visible surfaces are reflected in the variations in the magnitudes of the displacements. The most striking errors are on the dark cabinet-door just behind the inclined plank in the right-hand portion of the visible scene. Less noticeable are some of the errors along the linear structures in the image where the displacement vectors show non-zero vertical components.

The errors on the door are due to the fact that the homogeneous background (i.e., the door) has insufficient local information for the accurate determination of the displacements. This is indicated by the low values for the confidence measures in this area of the image. The disoccluding edge of the plank seems to influence the computations in the homogeneous area. Just as in the case of the top-right portion of the dinosaur-image, we believe this is due to the use of the smoothness constraint across a discontinuity in image motion.

## V.3 Summary

The set of experiments described in this chapter confirm much of what has been said in the previous chapters about our algorithm. For the most part, the results are remarkably accurate and the algorithm seems to perform nearly as well as can be expected of a low-level measurement process. The major difficulties in the algorithm arise when there are discontinuities in the image flow, either due to discontinuities in depth of the surfaces projected on the image, or due to 3-D motion boundaries.

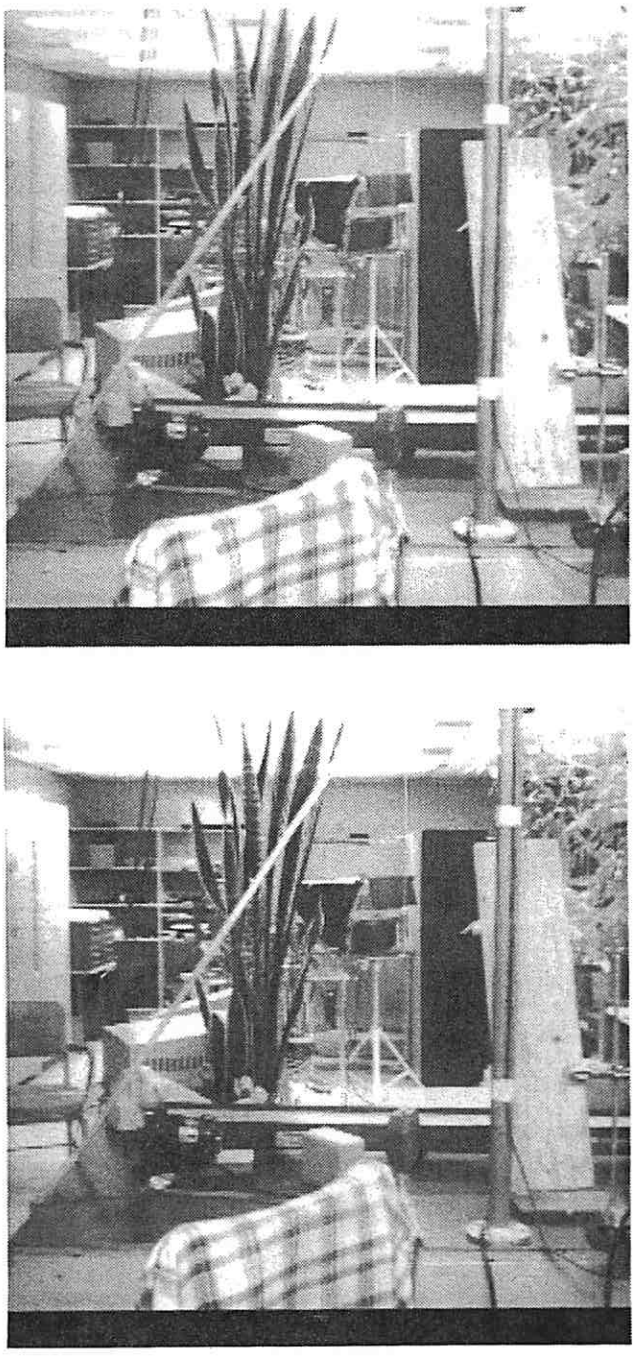


**Figure 74:** The office-scene experiment: input images.

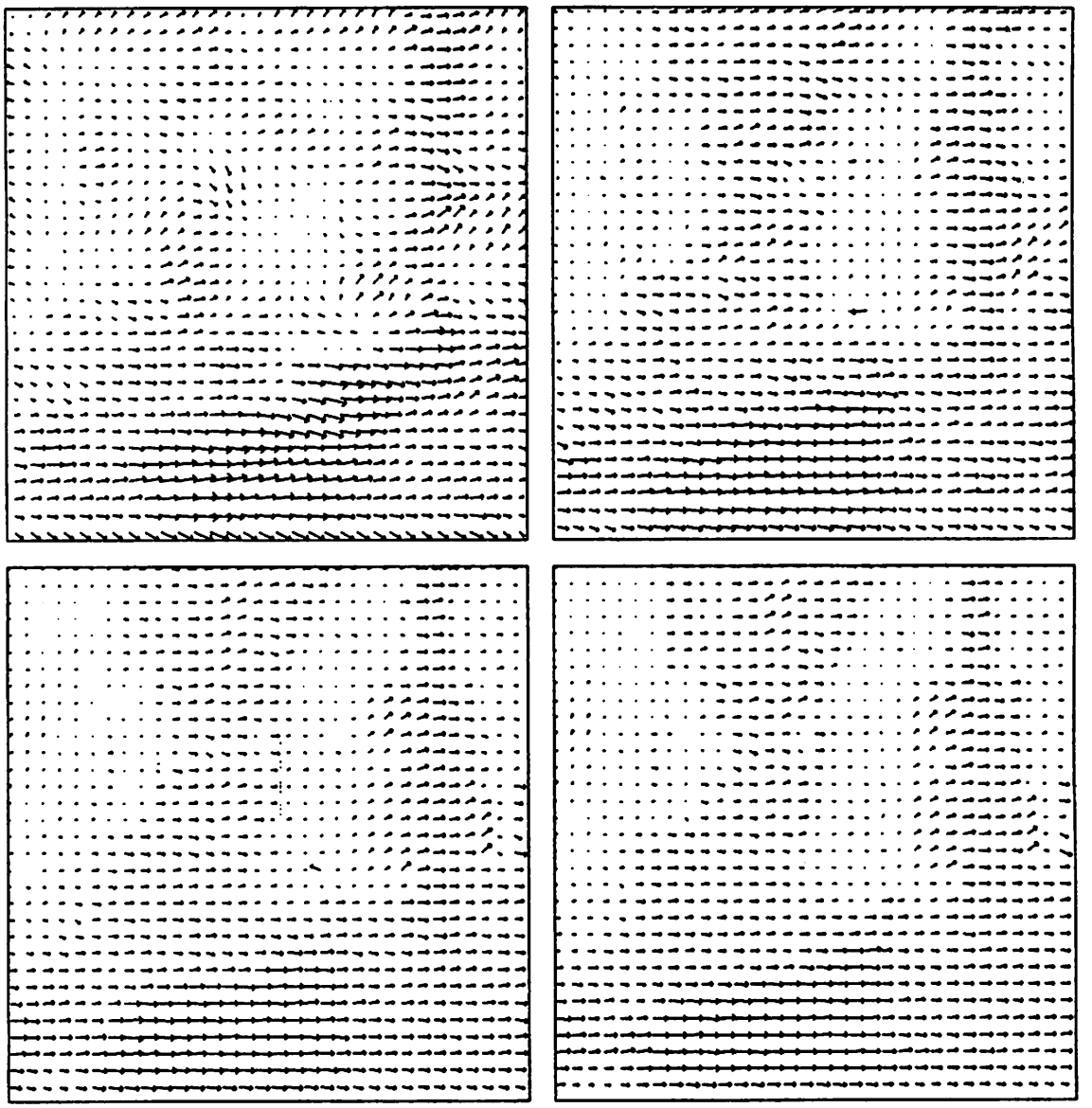


**Figure 75:** The results of the office-scene experiment with smoothing and with $5 \times 5$ Gaussian template windows. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown at levels 6 and 7.

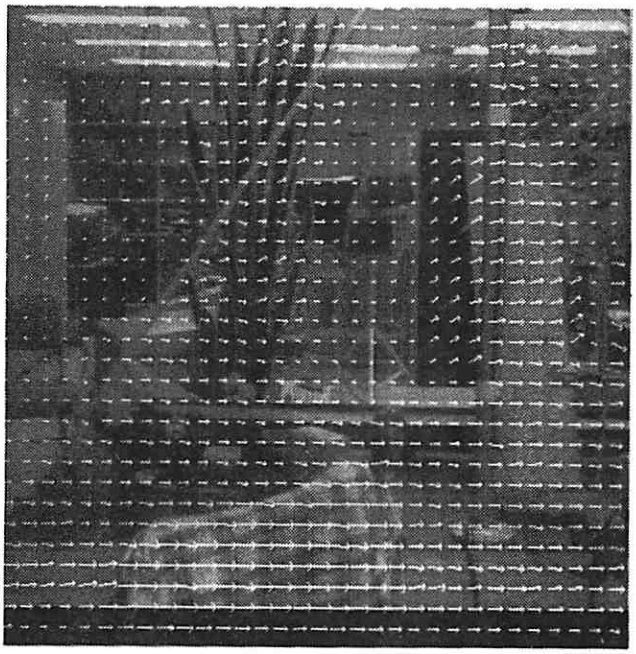


**Figure 76:** The finest-level smoothed displacement field for the office-scene experiment superimposed on the first input image. In order to enhance visibility only a $32 \times 32$ sample of the displacements have been shown.

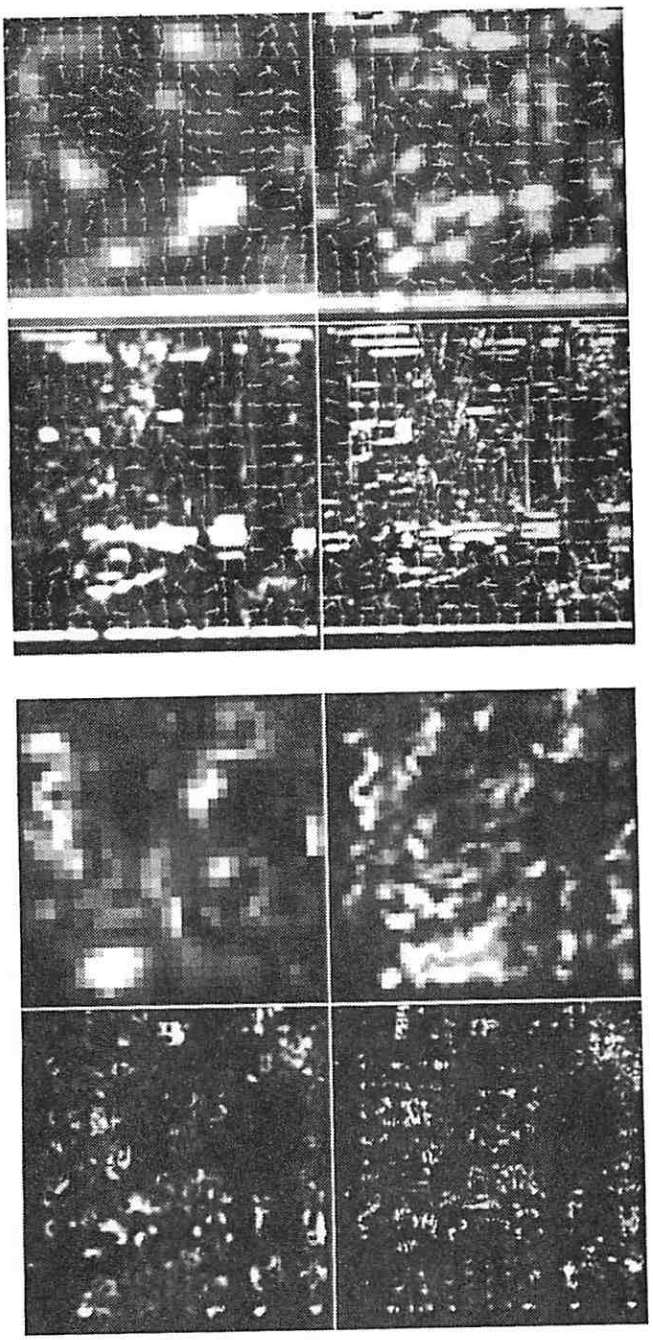


**Figure 77:** The confidence measures computed for the office-scene experiment with smoothing. The confidence measures are shown at the four finest levels. The top figure shows $c_{max}$ and samples of $\hat{e}_{max}$, while the bottom figure displays $c_{min}$.

It would be impossible and inappropriate for a low-level measurement process to completely determine areas that are occluded. The most that ought to be expected is that the measurement process should indicate that its results are unreliable. Although the confidence measures determine how uniquely a match is determined, as noted in chapter IV, the detection of discontinuities in image motion requires more sophisticated techniques. We believe that the development of such techniques is the most immediate direction for the further development of our computational framework.

# CHAPTER VI

# EXTENSIONS TO MULTIPLE FRAMES

While processing real images, it is rarely the case that we are concerned only with two frames. Since most situations of interest involve processing a continuous sequence of images (or a continuous image-stream), it is useful to consider how our framework can be extended to process multiple-frame image sequences. In this chapter, several approaches for such an extension are considered.

The type of extensions considered here can be broadly divided into two categories: what we call "direct extensions", which maintain our framework for the two-frame matching problem and consider how it may fit into an overall multiple-frame approach, and extensions that involve a slight modification of our framework using orientation-selective filters.

All our direct extensions involve the assumption that the image-velocity at a point varies continuously over time except when an image-motion boundary passes over that point. We formulate this assumption in the form of a "temporal continuity" constraint, and propose three distinct approaches for the measurement of motion from multiple-frame image sequences. The first approach is called the *prediction-refinement* approach (similar to the scheme used by Bharwani *et al.*, [16]), and involves determining the search areas for the matches between a given pair of frames based on the displacements estimates from the previous frames. The second approach is called the *variational approach* and involves adding a term involving the temporal derivatives of the displacement to our functional minimization problem formulated in chapter IV. The third approach is called the

*temporal coarse-to-fine control strategy* and involves using the match estimates for the pixels in the low-frequency channels for a given pair frames to determine the search areas for the corresponding pixels in the next higher frequency channel for the following pair of frames.

As explained before, the key idea underlying our current framework is the separation of the match process according to scale. Here, we consider the consequences of separating the match process also according to orientation. This consideration leads us to the decomposition of image-intensity variations according to orientation as well as scale. The range of magnitudes and directions of the components of image-displacements determined from these frequency/orientation selective channels are restricted. A new issue that arises as a consequence of this modification is the recombination of the measurements from the various orientation-tuned units. We propose a least-squares approach for the recombination problem, and outline an overall scheme for the determination of dense displacement fields.

We noted in chapter II that the application of the spatio-temporal energy models [2] for two-dimensional images requires the decomposition of the intensity function into a set of one-dimensional functions. In this chapter, we will explain that with this decomposition, the energy models can also be unified within our modified framework. The energy measurement units can be regarded as one of several possible ways of measuring the directional component of the image-motion at a point.

The multiple-frame extension of the modified framework using orientation-selective channels simply involves replacing the matching step of our algorithm with the measurement of spatio-temporal energy [2]. Our modified multiple-frame approach also suggests a novel view of visual motion analysis which does not involve the determination of dense displacement fields; instead, the localized energy measurement units can directly be used and controlled by higher level processes for navigation, spatial organization of the image, and incremental development

of the environmental structure. Based on this view, we suggest a redefinition of the "measurement" of motion as well as new directions for further research.

## VI.1 Direct Extensions of Our Framework

As noted in chapter I, the definition of the measurement of motion chosen for this dissertation is the field of inter-frame image displacements. Therefore, a natural way to extend our computational framework for processing multiple-frames is to determine dense displacement fields for every pair of successive frames of the multiple-frame sequence. An extreme approach would be to treat the frame-pairs as completely independent of each other. However, it is clear that to do so would be to ignore the fact that the movement of objects in the scene is usually continuous, and therefore, their 3-D velocities vary continuously most of the time. When combined with the spatial coherence assumption of image motion for opaque surfaces undergoing rigid or near-rigid motion, the temporal coherence of 3-D motion leads to the following "temporal continuity" constraint on the image flow. The image-motion at a point is continuous over time, except at the images of surface or object boundaries, and when there is a temporal discontinuity in the 3-D motion.

As mentioned earlier, this section contains the outlines of three proposals for using the temporal continuity constraint: the prediction-refinement method, the variational approach, and the temporal coarse-to-fine control strategy. Just as in the case of the spatial smoothness constraint, any approach for using the temporal continuity constraint should also contain mechanisms for detecting and processing the violations of that constraint.

An important source of additional information in multiple-frame sequences is the temporal coherence of the behavior of flow-discontinuities. For example, when a surface moves in front of another, the area that is occluded moves with the frontal surface. Usually the future locations and the movement of flow-

discontinuity boundaries can be predicted from past flow information. This idea should also be incorporated in any scheme for multiple-frame analysis.

### VI.1.1 The prediction-refinement approach

Let the frames be numbered $(0, 1, \ldots)$ and the process $p_i$ be the matching process between frames $i$ and $(i+1)$. The prediction-refinement approach involves using the displacement estimates obtained from some of the previous frames (say, $p_{i-1}, \ldots, p_{i-n}$) to predict the match location and constrain the search area for the matches in process $p_i$. Such an approach would be similar to the technique used for the temporal refinement of depth maps from a translational-image sequence described by Bharwani, *et al.*. [16].

The simplest case involves only the immediate process before for the prediction, i.e., the process $p_i (i \neq 0)$ is influenced by the results at $p_{i-1}$ and by no other previous processes. For a discrete spatial image, the displacement vector $(dx, dy)$ assigned by process $p_{i-1}$ for pixel $(x, y)$ will be an initial estimate of the displacement of the corresponding pixel $(x + dx, y + dy)$[1] in process $p_i$. For our multiple-resolution approach, the displacement field obtained at a particular level can be directly propagated to the next process (i.e. the next frame) at the same level.

Instead of directly using the results of the match-process $p_{i-1}$, we can apply an "interpretation" process (e.g., see [4]) to the displacement field provided by $p_{i-1}$, and determine the motion parameters and segment the image. These parameters can be used to recompute a dense displacement field at multiple-resolutions. The recomputed field can then be propagated to $p_i$.

The communication between successive processes simply involves the transfer of the displacement vector from pixel $(x, y)$ in process $p_{i-1}$ to the corresponding

[1] If displacement estimates are obtained to subpixel precision, $dx$ and $dy$ will not be integers. Then the assignment is communicated to the pixel nearest to the tip of the displacement vector in process $p_i$.

pixel in process $p_i$. However, the pixel $(x, y)$ may not have a reliable displacement assigned to it due to occlusion or other difficulties in matching, it may move out of the field of view, or there may be multiple initial values for a single pixel in $p_i$. If a pixel has no previous estimate, then the temporal continuity constraint simply does not provide an additional advantage. On the other hand, if there are multiple estimates from the previous process, the search can be conducted around each of these estimates, just as in our overlapped pyramid algorithm. Note that similar problems are encountered and addressed by Bharwani *et al.* in [16], when they consider the temporal refinement of depth information from a translational image sequence.

### VI.1.2 The variational approach

The temporal continuity constraint can also be used in the form of a variational problem. While allowing the local displacement computation to proceed using the coarse-to-fine control strategy combined with a local match criterion, an additional term is included in the minimization problem described in chapter IV. For example, the first order smoothness constraint can be modified as follows:

$$E_{smooth} = \iiint (u_x^2 + u_y^2 + \frac{1}{\beta^2} u_t^2 + v_x^2 + v_y^2 + \frac{1}{\beta^2} v_t^2)\, dx\, dy\, dt$$

The difference between the version here and the one used in chapter IV is the presence of the $u_t$ and the $v_t$ terms in the above equation. The scale factor $\beta$ is necessary because the spatial and temporal derivatives of flow have different units of measurement.

A detailed version of the above formulation must take into account a number of issues. First, whereas the spatial integration is limited to the image area, the limits of the temporal integration have not been specified. This would involve detection of temporal discontinuities in motion and restricting the smoothness constraint to be "piecewise", i.e., valid within a bounded 3-D region in the space-time domain. Second, it is not obvious how the factor $\beta$ is determined. The third

and the most important issue is the asymmetry of temporal influence, i.e., the fact that computation at any instant of time cannot be affected by data from the future. This must be taken into account while employing a solution method for the variational problem that has been formulated. Nevertheless, the idea is clear and simple. The temporal smoothness assumption suggests that the reliable displacements be propagated not only to its neighbors on the image plane, but to future measurements of image displacements as well.

### VI.1.3 Temporal coarse-to-fine control strategy

A simple method of processing multiple-frames within the hierarchical framework is to spread the coarse-to-fine control strategy over time. In this approach, the displacement computations between a particular pair of frames in a specific spatial-frequency channel influences the computations in the next higher-frequency channel between the *next* pair of frames (see Figure 78). In this way, the temporal continuity constraint is automatically introduced as a part of the control strategy.

This formulation is attractive because it is simple and inherently dynamic. The computation at each pixel within each channel for each frame is the same. At each time-step, the local match process uses the previous time displacement estimate from its ancestor in the adjacent lower-frequency channel as the initial match value. The uni-directionality of the temporal continuity constraint is automatically included in the formulation.

Finally, note that the temporal coarse-to-fine control strategy can also be regarded as a model for tracking. When a new object comes into the field of view, the low-frequency channels provide a quick but rough estimate of its movement. This information is communicated to the higher-frequency channels, which then "track" the area and provide increasingly precise estimates of its motion, and perhaps information regarding its 3-D shape.

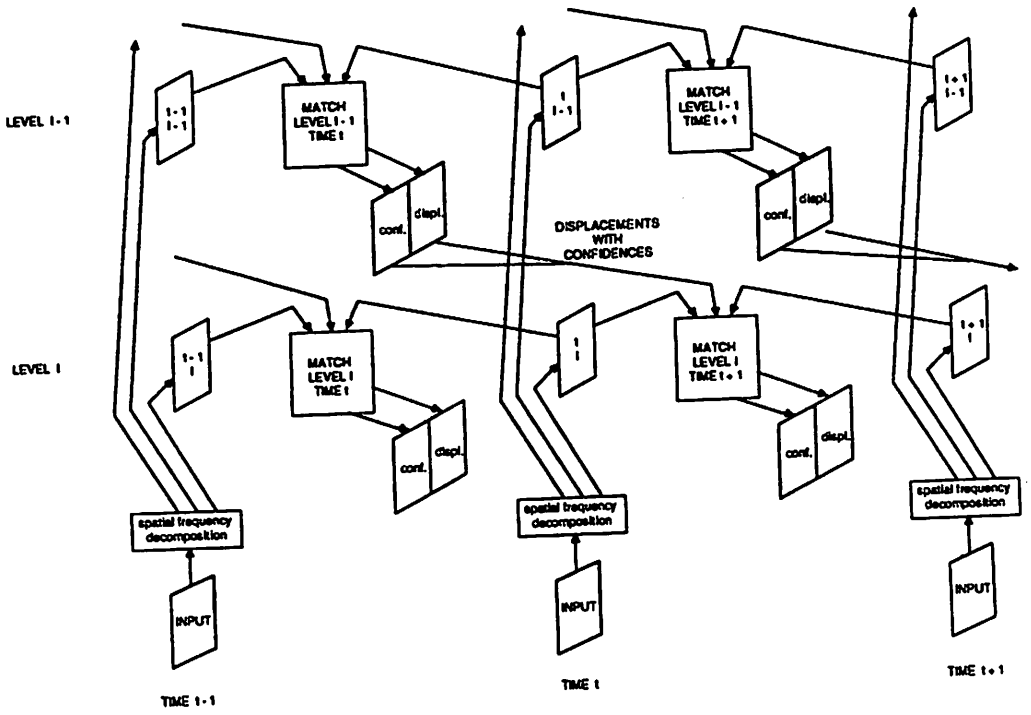



**Figure 78:** Spreading coarse-to-fine control over time – a schematic view

## VI.2 The Use of Orientation-Selective Filters

As noted in chapter I, our computational framework is founded on the principle of separating the matching process according to scale. Its effectiveness was due to the use of resolution vs. range tradeoff, the coarse-to-fine control strategy, the orientation sensitive confidence measure, and the smoothness constraint. Although all of these are important to completely specify the framework, as we have noted before, scale-based separation of computation is the primary idea, since it underlies everything else.

In this section, we consider the separation of the image intensities and the matching process according to orientation. The primary motivation for this is the observation that intensity variations along a particular direction can only provide information regarding the component of motion along that direction. In a sense, the *aperture problem* mentioned in chapter II and the behavior of the SSD surface at edge-like points discussed in chapter IV can both be considered as special cases of this observation, since at such points the image-intensity varies only along the direction perpendicular to the edge.

In general, since the image intensity is a function defined on a two-dimensional region, there are two parameters involved in the decomposition process, e.g., scale and orientation. Thus far, the decomposition according to scale has been included in our framework. The filters used have been circularly symmetric (or *isotropic*) in the space and the spatial frequency domain. This means that the intensity variations at the same scale, but along different orientations have not been separated. If maximal decomposition is desired, then it is logical to consider the design of filters that are also tuned to specific orientations.

It has been noted by Burt and others [22,23] that Burt's Laplacian-pyramid transform can be regarded as the determination of the coefficients for a set of basis functions whose shapes are Gaussian and which differ according to scale. On the other hand, based on a two-dimensional version of an "uncertainty relationship"

for signal representation which was derived by Gabor [35], Daugman [30] suggests the use of a basis set consisting of orientation-selective 2-D functions. These functions have the general form of a sinusoid (or a co-sinusoid) enveloped by a Gaussian. Daugman's consideration of these filters is also motivated by recent psychophysical evidence for the presence of cortical cells that are tuned to specific ranges of orientation and spatial-frequency (see [59,30]). Although it is not our intention to consider this evidence in detail, the design of a system based on our modified framework may benefit from a closer examination of the data reported in these studies of the human visual system.

A second source of motivation for our consideration of orientation-selective filters arises from of our interest in unifying the spatio-temporal energy models along with the matching and the gradient approaches into a single framework. As noted in our review of the energy models in chapter II, using the energy models for two-dimensional signals requires preprocessing the signals using orientation-selective filters. To this end, the description of our modified framework given below identifies how the energy models fit into it.

The rest of this section outlines a computational framework for the determination of displacement fields using orientation-selective filters. The organization of this outline is as follows: first, a quick overview of the new computational framework is provided. Following this are the descriptions of each of the major components of the framework. These descriptions are primarily in the form of the characteristics that will be required of the components in any implementation. While this section focuses on the determination of displacement fields from two images, the use of this approach for multiple-frame analysis is discussed in the next section.

### VI.2.1 An overview of the modified framework

At this stage, the primary goal of our computational process is to determine dense displacement fields with confidence measures from a pair of frames. The

framework consists of three major components – the decomposition of the image using orientation/frequency selective channels, the matching process within each channel, and the combination of match-results from the different channels. The matching process within each channel also includes the computation of a confidence measure, while the combination process incorporates a spatial-smoothness constraint as well as the spectral continuity constraint. As is evident, the description of this framework is organized slightly differently from that of our original framework in chapter I. At this stage, this is primarily a matter of convenience and clarity. However, it will be seen later in this chapter that this new organization provides a new perspective on the word "measurement" of motion.

Briefly, the various components of the framework are organized as shown in Figure 79. Each input is processed by a set of filters, each of which is tuned to a specific range of spatial frequencies and a specific range of orientations of intensity variations. This is the decomposition stage, where each channel can be regarded as measuring different components of the input intensity image. Following this is the matching stage, which operates separately within each channel, i.e., the outputs of applying a particular channel to each input image are matched with each other. As before, the the range and the resolution of the displacements will correspond to the frequency range of a particular channel. In addition, a channel tuned to intensity variations along a specific direction on the image plane only provides information about the "component" of displacement along that direction. Once again each local measurement within each frequency/orientation-tuned channel is accompanied by a confidence measure. The last stage is the combination of the match results from the different orientation and frequency selective channels. During this process, a smoothness constraint, a displacement consistency constraint and a spectral continuity constraint are all applied. Hence, the control strategy for the invocation of the matching process within different channels is essentially a part of this process of combination.

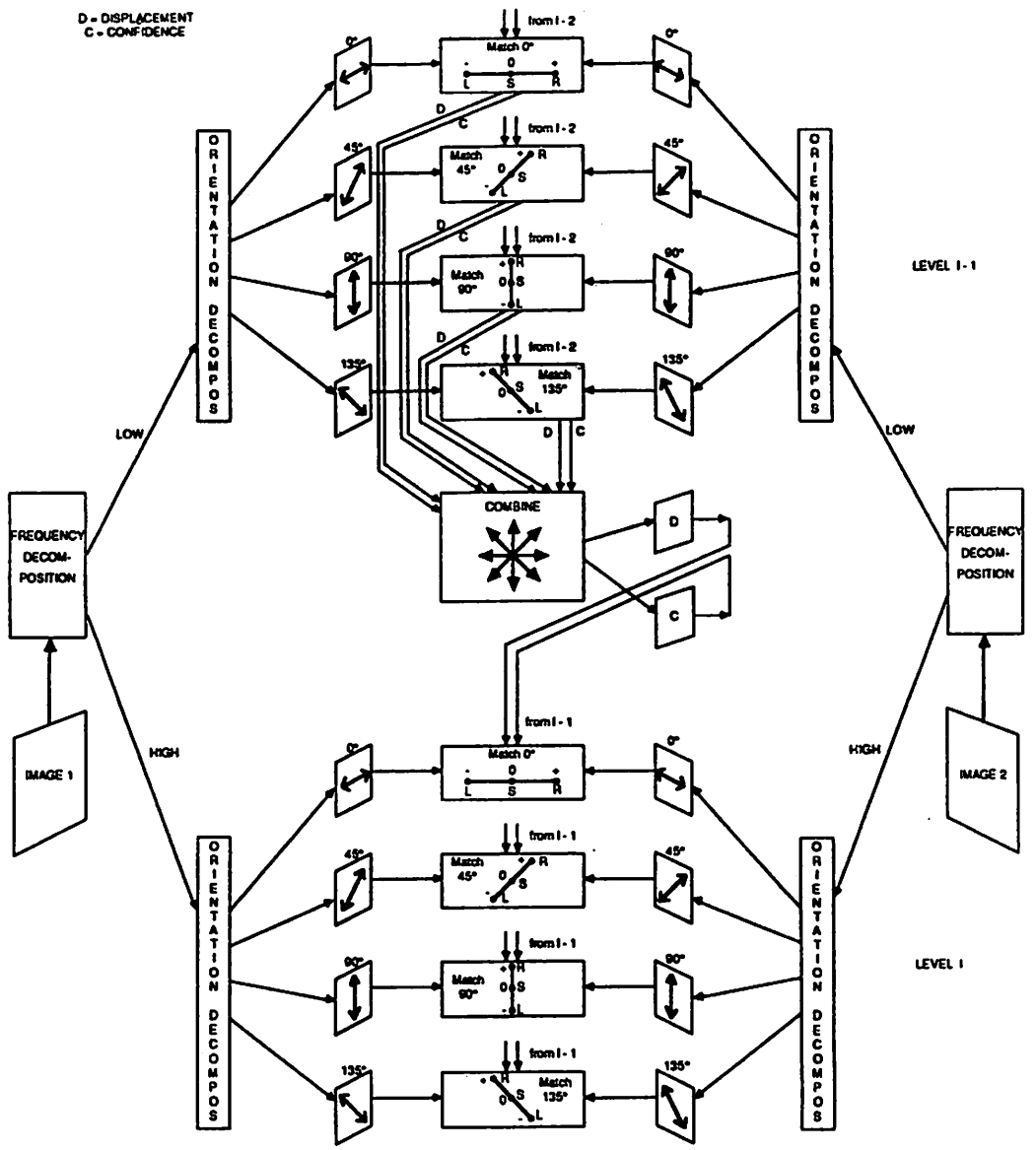



**Figure 79:** The hierarchical orientation-selective framework a schematic view

### VI.2.2 The decomposition stage

This section describes the type of orientation-selective filters that may be used for the decomposition process. The major characteristics desired of such filters are outlined. For illustrative purposes, a brief outline of a possible approach for their implementation is also included.

As mentioned before, the filters proposed here are both frequency and orientation selective. Linear filters are proposed, which can implemented as convolutions of the input images. The characteristics that are required of the set of filters are (i) the spatial convolution masks should be local, i.e., most of their contribution should come from within a small image area, (ii) in the frequency domain, these filters should cover a narrow range of frequencies both in terms of their range and their orientations, and (iii) the spatial convolution masks should be scaled and rotated versions of each other.

Although the Gabor filters discussed above are minimal in the sense of the joint area occupied by them in the space and the frequency domain, they do not have the scale-invariant shape properties. This is because the Gaussian envelop of the Gabor filters has the same width for all the frequencies. An alternative set of filters proposed by Daugman [29] is based on the directional derivatives of the Gaussian. The following is a description of a possible approach for the implementation of a set of orientation/frequency selective filters based on such directional derivatives.

First, a Gaussian pyramid is constructed from each input image according to the scheme discussed in section IV.2. Following this, four $3 \times 3$ convolution masks are applied to each level of the Gaussian pyramid to create an oriented-Laplacian pyramid, each level of which contains four images. The four convolution masks are discrete approximations to the second directional derivatives along four directions which are 45 degrees apart from each other. The convolution masks for the four directional derivatives are shown in Figure 80. For illustrative purpose,

in Figure 81 we have also included surface plots of the power-spectra of the 0 and 135 degree filters at the two finest levels. Note that at the finest-level the filters show a high-pass behavior, while at the next coarser-level they show band-pass behavior.

$$\frac{1}{3}\begin{bmatrix} 1 & -2 & 1 \\ 1 & -2 & 1 \\ 1 & -2 & 1 \end{bmatrix} \qquad \frac{1}{12}\begin{bmatrix} 1 & -2 & 7 \\ -2 & -8 & -2 \\ 7 & -2 & 1 \end{bmatrix}$$

$0°$ $45°$

$$\frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ -2 & -2 & -2 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{12}\begin{bmatrix} 7 & -2 & 1 \\ -2 & -8 & -2 \\ 1 & -2 & 7 \end{bmatrix}$$

$90°$ $135°$

**Figure 80:** The convolution masks for the four oriented-filters.

Note that these filters are oriented high-pass filters. The band-pass filtering is achieved via the cascading of these oriented high pass filters with the isotropic Gaussians.

### VI.2.3 The matching stage

In the current scheme, since four orientation channels are used, they will measure the displacement along the four corresponding directions. Following the same logic as in the previous hierarchical framework, at any level of the pyramid processing scheme, the displacement can be expected to be within a $3 \times 3$ area around the estimate projected from the previous coarser level. In this case, as illustrated in Figure 82, the four orientation channels each select

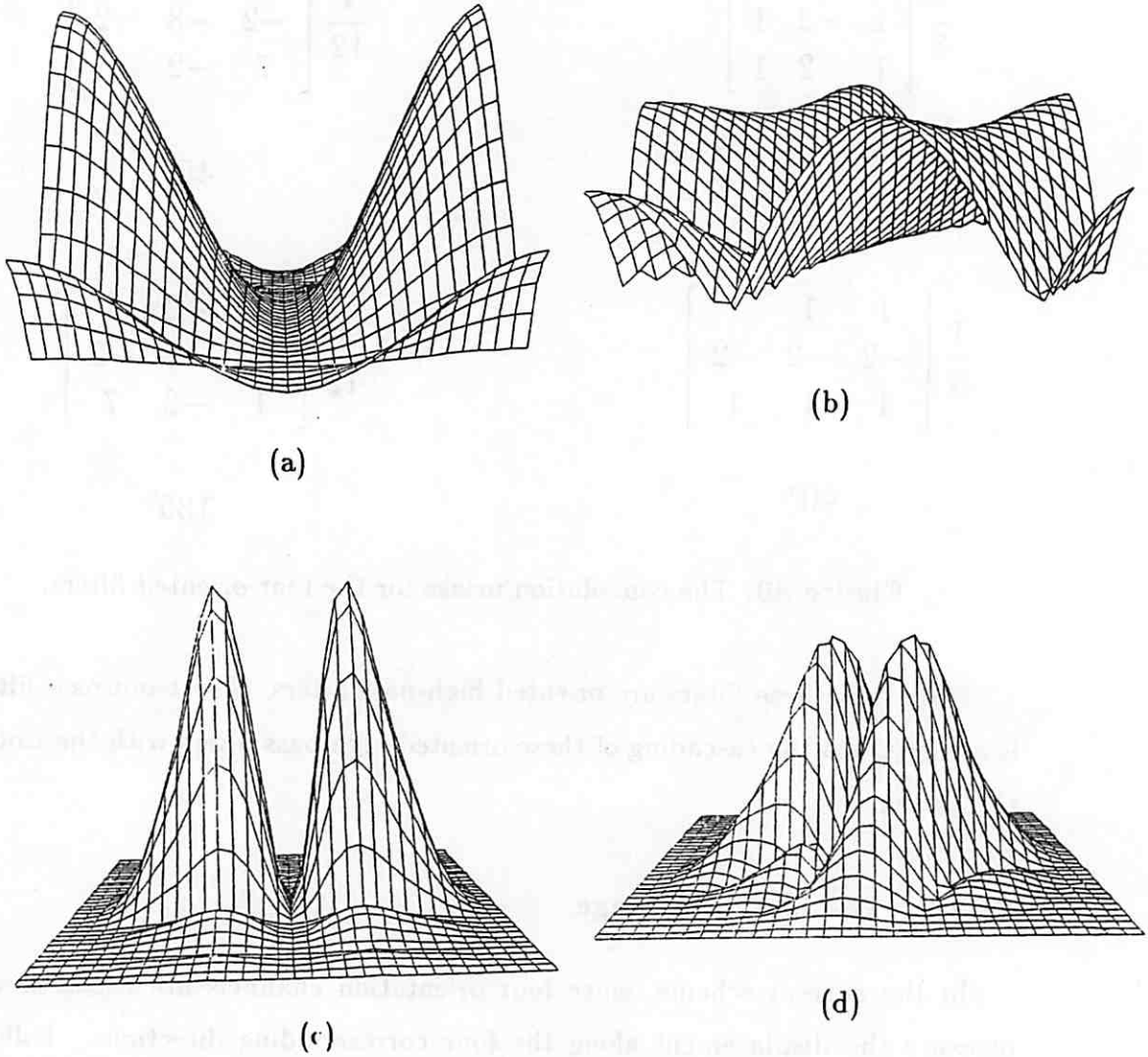



**Figure 81:** The power-spectra of the oriented band-pass filters. (a) the finest-level 0 degree filter, (b) the finest-level 135 degree filter (c) the 0 degree filter at the next coarser-level, (d) the 135 degree filter at the next coarser-level.

one of three possible displacement magnitudes $(-1, 0, 1)$, along the appropriate directions. Note that these three magnitudes correspond to left, stationary, and

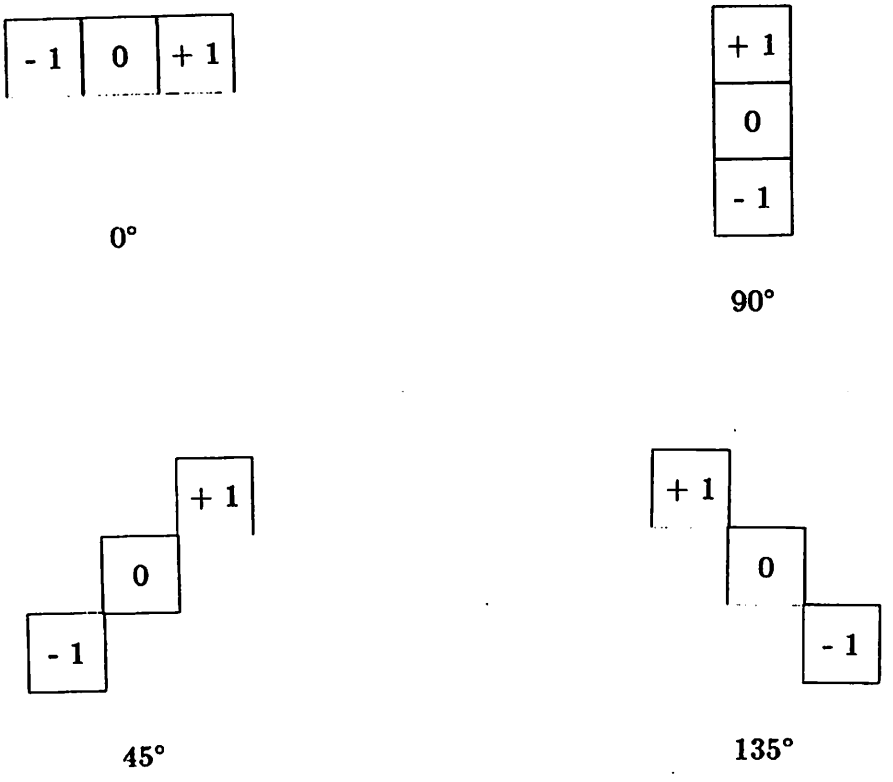



**Figure 82:** The displacement ranges of the four orientation-selective matching units

rightward motions. We call this match process as the determination of the "sign" of motion.

Since we have considered the input to be a pair of images, the most natural approach for the computation of the directional components of the displacement is a matching approach. As in our earlier approach, a type of correlation (e.g., the minimization of SSD) matching approach can be used. A confidence measure should also be computed for each match; for instance, if the minimization of the SSD measure is used as the match-criterion, the confidence measure will be based on the curvature of the *one-dimensional* SSD function.

Given that the scale and the direction of the intensity changes are predefined, it may be possible to use a one-dimensional version of the gradient-based approach

according to equation $V_s = -I_t/I_s$, where $I_t$ is the temporal derivative of the intensity function, $I_s$ is the spatial derivative along the direction of the channel, and $V_s$ is the component of the image velocity along the same direction. We can also directly used one of the energy models [2] for the determination of the "sign" of motion.

As it will be noted in the next section, the energy models are easily adaptable for multiple-frame analysis. In Appendix D, we show that for the two-frame matching problem, the minimization of the SSD measure can be cast as a spatio-temporal filter, in a manner similar to the energy models.

### VI.2.4 Recombination of the measurements of motion

Each frequency/orientation-tuned channel provides a displacement vector for each pixel, whose direction is fixed by the orientation of the channel, whose magnitude is determined by the scale of the channel, and whose "sign" is determined by the match process. In addition, an associated confidence measure is provided. Since the displacement vector provided at a pixel from a particular channel measures the component of the true 2-D displacement along a specific direction, all displacement vectors along a line in the displacement space are supported by this vector. As shown in Figure 83, there are four such displacement constraint lines provided at a pixel, one from each orientation-selective unit. A simple rule for combining the information from the four channels is to find the best intersection of the four lines by minimizing an error measure $E(\vec{U})$ associated with a displacement vector $\vec{U}$:

$$E(\vec{U}) = \sum_{i=0}^{3} C_i(\vec{U} \cdot \vec{e_i} - d_i)^2$$

where $(\vec{e_i}, d_i, C_i)$ are respectively the unit-vector along the direction of channel $i$, the displacement estimate from that channel, and the associated confidence

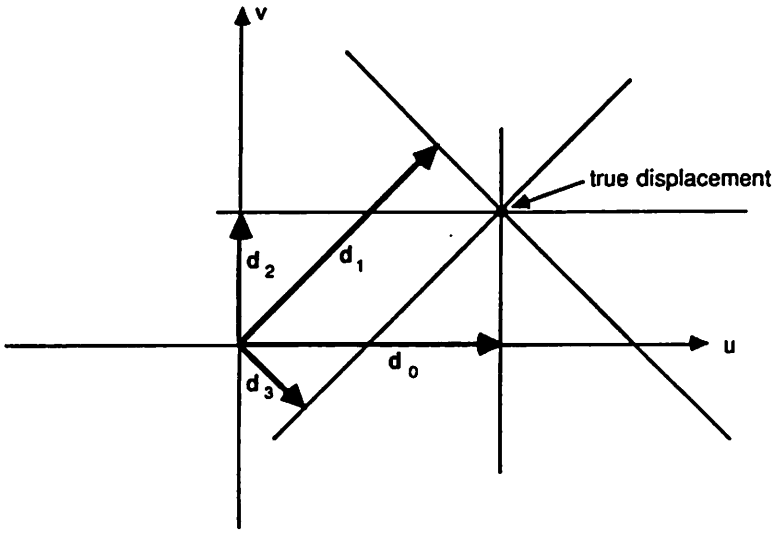



**Figure 83:** The displacement constraint lines of the orientation-selective units

measure. In a more general sense, this error measure can take the place of the approximation error in the smoothness constraint discussed in chapter IV.

It should be noted that while this measure has an intuitive appeal and is simple, it has its disadvantages. Along a linear structure in the image, it would be appropriate not to determine a unique displacement vector but simply determine its normal component. However, minimizing the above measure will provide a unique vector even in that case. Therefore, a more sophisticated measure will be necessary for any practical situation.

The recombination of the information from the units tuned to different spatial-frequencies can be achieved through the use of a "spectral continuity" constraint, which can be implemented as a coarse-to-fine control strategy similar to the one described in chapter IV.

## VI.3 Extending the Hierarchical Orientation-Selective Framework for Multiple Frames

Since a major focus of this chapter is the consideration of multiple-frames, it is natural to examine the approach based on orientation-selective units with respect to multiple-frame processing. As it might be expected, there is a direct extension of this approach for multiple-frames, wherein each frequency/orientation tuned unit uses the average "sign" of motion over a small number of frames. The average sign of motion can be obtained either by combining the results of the match process over the few frames, or by directly using the one of the energy models [2,114,117]. An advantage of using the energy models is that the extension is valid even for the limiting case of a continuous image-stream.

The temporal continuity constraint on image flow is implicit in the use of a time-average of the measurement of motion and in the use of the spatio-temporal energy models. Although an alternative approach such as the prediction and refinement approach suggested earlier in this chapter is also possible, such an approach will not be discussed at this preliminary stage of this effort.

The motivation for the computation of image flow (i.e., dense displacement fields) arises out of the numerous analyses that relate the structure of the image flow field to that of the environmental surfaces and the parameters of motion. However, it is tempting to consider if there are methods that do not involve the computation of such image velocity or displacement fields, but directly use the outputs of the spatio-temporal energy units instead.

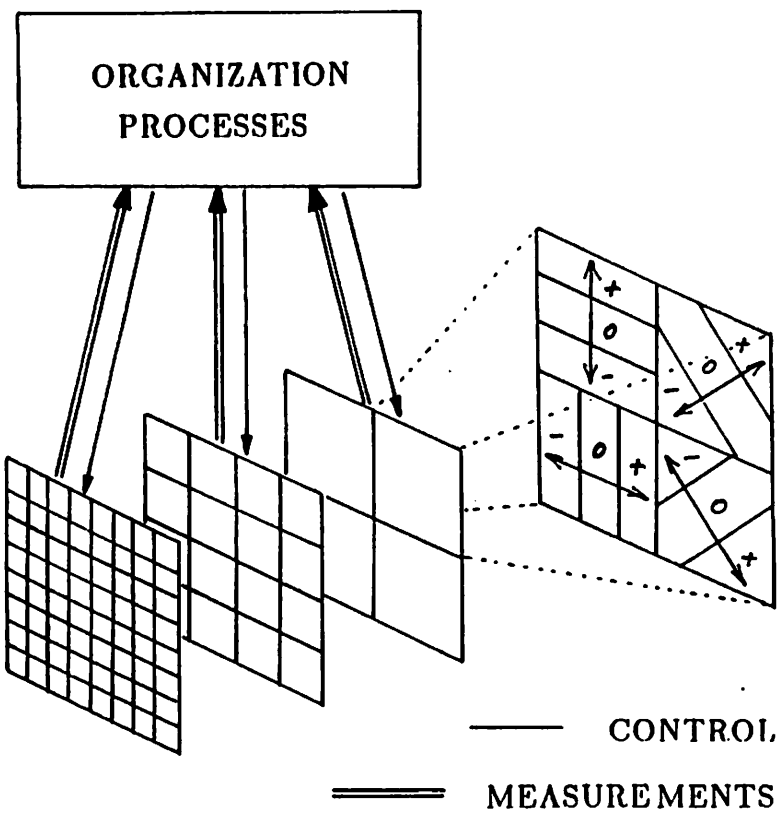
It was noted in chapter I that a statement such as "the intensity at a point is changing at a particular rate" is not a valid measurement of motion, since it did not refer explicitly to movement. While the energy measurement units appear to be making similar statements, they are different because they are *tuned* to (i) specific locations on the image-plane, (ii) specific ranges of spatial-frequencies and associated ranges of speeds, (iii) specific ranges of image-orientations and

associated range of directions of movement. This tuning property allows them to be regarded as performing a measurement of motion.

In general, the outputs of the localized energy units can feed directly to various higher-level processes for a diverse range of goals, e.g., navigation, spatial-organization, determination of 3-D structure (see Figure 84). In this sense, the measurements can be treated as "feature" values with an associated probability measure. Equivalently, the output of each of these units can be regarded as a piece of evidence for the presence of an intensity structure at a specific scale, orientation, and velocity with an associated uncertainty measure. There are a number of schemes currently under investigation for the combination of such probabilistic "features" or evidences (for a survey, see [100]); most of these approaches appear suited for a connectionist model of computation.

We can also allow the measurement processes to be actively "controlled" by the high-level processes. Although the units themselves are simple and perform simple convolutions and maximum selection operations, by controlling their input they can be used to achieve the more complex goals of the higher-level processes. The coarse-to-fine strategy is one example of such a control mechanism.

For a slightly more general example of the use of the energy measurement units, consider the spatial organization process. First, a rough separation of rapidly moving areas from stationary areas can be obtained from the low-frequency tuned units. The low-frequency units near the boundaries of motion will have low-confidence information, and should therefore be ignored. The coarse-to-fine strategy can be applied selectively to the units in the moving area to obtain high-frequency measurements of movement, while simultaneously refining the segmentation. As the refinement of the segmentation progresses, closer attention can be paid to the boundary units, whose search areas may be redefined according to the motion of the area containing them. As much as possible, the process of spatial-organization (or "grouping") should progresses without a 3-D interpretation, thereby avoiding the various numerical instabilities encountered in




**gure 84:** The overall schematic of an orientation-selective hierarchical framework for multiple-frame analysis

the interpretation process. Once the grouping and segmentation is finished, the high resolution processes can provide highly accurate measurements of motion, which can then be used for the determination of the 3-D structure and motion.

Any scheme for the use of these energy measurement units should also include an understanding of how these models behave under a "tracking" situation, i.e., when the camera (or the eye) is rotated to fixate the image of a particular environmental point on the image-plane (or the retina). Here, it is not sufficient to consider the output of the measurement-units under tracking. A comprehensive analysis must also address methods of using the measurements to trigger the tracking mechanisms and schemes for the incremental development of a 3-D model of the environment.

## VI.4 Summary

In this chapter, some methods for the measurement of motion from an image sequence (containing more than two frames) or a continuous image-stream were examined. We considered direct extensions of the framework, as well as extensions of a modified form of our framework using orientation-selective filters. It was also noted that the matching approach can simply be regarded as one way of measuring the spatio-temporal energy. Therefore, it now seems apparent that these seemingly distinct methods have some strong similarities.

The shift towards the energy models opens up new possibilities not only regarding the type of measurements, but also with respect to the use of such measurements. Such a comprehensive examination of alternate frameworks for motion analysis is a significant research effort in itself and can perhaps form the basis of future work in this area of Computer Vision.

# CHAPTER VII

# SUMMARY

This chapter contains a summary of our major research contributions, a review of some of the unsolved problems, and the possible avenues for future research.

## VII.1 Major Contributions

The most significant contribution of our research described in this dissertation is the development of a robust algorithm based on a theoretical computational framework for the determination of dense displacement fields from a pair of images. Our framework includes spatial-frequency channels, an efficient multi-resolution control strategy, an orientation sensitive confidence measure, and a rigorous mathematical formulation of a smoothness constraint. The framework appears sufficiently powerful that it can incorporate either of the the two traditional low-level approaches for the measurement of image motion, viz., the gradient-based and matching techniques.

All the computations in our algorithm are entirely suitable for low-level pixel-parallel processing, and involve only information from a small neighborhood around a pixel. Our algorithm is naturally suited for a pyramid architecture and can be performed in $O(\log \delta)$ time on such a machine. Given that no pyramid machines are commercially available at present, we also provide a version of our algorithm suitable for a mesh-connected computer (MCC), which operates

at $O(\delta^2 \log \delta)$ time. Our algorithm has been shown to provide some of the best published results for several real image pairs.

An important original contribution of our research is the confidence measure associated with every displacement estimate. The confidence measure is orientation sensitive, thereby associating different degrees of reliability for different components of the displacements. In particular, the confidence measure distinguishes between three distinct and qualitatively significant types of intensity structures – homogeneous areas, linear intensity structures and points of high-curvatures along visible image contours. The confidence measure is computed at each level of our multi-resolution algorithm; it requires little additional effort, since it is based on the shape of the SSD surface, which is already computed during the matching stage.

We have also provided a rigorous formulation of a smoothness constraint; this constraint is in the form of a functional-minimization problem, which is traditionally used in gradient-based approaches. We have employed the finite-element method for solving this minimization problem, which lead to an iterative relaxation process. Since this relaxation process is embedded within our hierarchical computational framework, only a few iterations are necessary at each level; moreover, some of the usual convergence problems associated with such processes have been avoided.

The formulation of the smoothness constraint as a minimization problem also enabled us to identify the mathematical relationship between our matching approach and the gradient-based approaches of Horn and Schunck and Nagel. More specifically, it was shown that the our formulation of the minimization problem converges to a similar formulation (by Nagel) for image-velocities which uses second-order intensity gradients when the inter-frame time interval tends to zero. Further, when the size of the template-window used in the computation of the SSD measure tends to zero, our formulation converges to the well-known Horn and Schunck's formulation. This relationship has also enabled us to explicitly identify,

for the first time, the confidence measures implicitly used in the gradient-based approaches.

Finally, we have proposed several methods of extending our approach for multiple-frame analysis. While the direct extensions of our framework would involve embedding it "as is" in an overall scheme for processing multiple-frames, we have also considered novel approaches for the extension based on a modification of the framework using orientation-selective filters; with such a modification, the matching process can be regarded as a type of spatio-temporal energy detector. The spatial-frequency-tuning and the reduction in displacement range with increasing frequency are important in establishing this connection.

The link between the energy models and our modified framework also opens new possibilities for the measurement of motion, and what appears to be a biologically feasible approach for motion-analysis. Some possible directions for further research along this direction were indicated in chapter VI.

## VII.2 Major Unsolved Issues

Perhaps the single most important issue that has not been addressed fully in the current framework is the processing of images with discontinuities in image-motion. As noted earlier, this may arise either due to a discontinuity in depth or due to a discontinuity in the 3-D velocity (or displacement) field. Although some of the problems in processing such situations have been alleviated by our use of the overlapped pyramid projection strategy, and our choice of the finite-element method, which allows the prevention of smoothing across the known locations of discontinuities, we have not provided a complete solution. In this section, we briefly examine the types of problems caused by discontinuities in image-motion and suggest possible approaches for dealing with them.

The spatial-frequency decomposition process is insensitive to such boundaries;

hence, within a low-frequency channel, the information around a boundary of discontinuity is not derived from a single surface, and therefore cannot be expected to behave coherently over time. Therefore, our match-criterion of minimizing the SSD measure is not valid. In fact, since the intensity constancy assumption is not valid for such locations, any intensity-based match criterion would be inappropriate.

As noted earlier, the problem at the boundaries is made severe by the coarse-to-fine control strategy, since the area of the image corrupted by smoothing over the boundaries increases at lower-resolutions. Since each displacement at coarse level determines the search area for its descendants at the finer level, a mistake at the coarse-level due to areas whose intensity structures are incoherent over time adversely affects the computations at all the descendants. An example of the difficulties at image-boundaries due to the coarse-to-fine strategy was noted in the discussion of the results of the dinosaur-image experiment in chapter V.

The smoothing process also compounds the difficulties in processing scenes with image motion discontinuities. If a textured area is adjacent to the boundary, the smoothing process is not likely to modify the local match estimates in that area, because they are likely to have high confidences. On the other hand, when a homogeneous area is occluded, the smoothness process is likely to propagate the displacements from the occlu*ding* in the front to the occlu*ded* area.

The processing of boundaries of image-motion involves the detection of the presence of such boundaries, as well as the measurement of image flow for the pixels near the boundaries. As noted in chapter IV, the presence of a boundary of image-motion is often indicated by a large value for the minimum of the SSD surface, as well as the difference in the shapes between the auto- and the cross-SSD surfaces. If, in addition, the local spatial-derivatives of image-flow are large, we may be able to conclude with certainty that a boundary of image-motion is present.

Once the presence of image-motion boundaries have been detected, the search

areas for the boundary-pixels should be defined according to the displacements of the reliable neighbors in the region to which they belong. The template-windows should be modified so as to include only the pixels in that region. The displacements can now be recomputed by minimizing the match measure over the new search areas using the modified template-windows.

Note that the idea suggested above for the detection of image-motion boundaries and the recomputation of the displacements at the boundary-pixels can be easily extended to each level of our hierarchical approach. If multiple-frames are considered, the image-flow information from previous frames and the temporal-coherence of discontinuities may be useful as additional sources of information for this process.

## VII.3 Directions for Further Research

The most obvious direction for further research is the pursuit of the suggestions given in the previous section for processing discontinuities in image-motion. In addition, the extension of the current framework and the algorithm for multiple-frame analysis, and the use of orientation-selective channels are also important avenues for further research. Since both these have been discussed in detail in chapter VI, we simply mention them here. Of these, the use of orientation selective channels is perhaps the one of immediate interest, both because of its psychophysical support and because it has not yet been developed and tested even for the two-frame matching problem.

Finally, as we have already noted, the modified framework using orientation-selective filters and the energy-measurement units suggests novel connectionist views of motion-analysis. We expect that our own future efforts will also involve pursuing such views towards the development of more comprehensive systems which use these energy-measurement units for the analysis of image-sequences.

# APPENDIX A

# CORRELATION MEASURES

This appendix contains a brief mathematical review of the different types of correlation measures used in matching, their relationship to each other, and an ideal-case analysis of the effect of window shapes and sizes on matching results.

## A.1 Definitions of the Correlation Measures

### A.1.1 Preliminaries

- $F(i,j)$ and $G(i,j)$ are the two intensity images that are matched. $(i,j)$ are pixel indices and hence can take only integer values. The origin $(0,0)$ is assumed to correspond to the center of the template window (i.e., the pixel of interest) in the first image.

- Each correlation measure is a function of the displacement $(\Delta i, \Delta j)$.

- The template window is assumed to be a square of width $(2n + 1)$ pixels. The weighting function $W(i,j)$ is normalized, i.e., $\frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) = 1.$

### A.1.2 Definitions

The *direct correlation* measure $C$ is defined as:

$$C(\Delta i, \Delta j) = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) F(i,j) G(i+\Delta i, j+\Delta j) \quad . \qquad \text{(A.1)}$$

The *mean-normalized correlation* $M$ is defined as:

$$M(\Delta i, \Delta j) = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) \left(F(i,j) - \overline{F}\right) \left(G(i+\Delta i, j+\Delta j) - \overline{G}\right) \qquad \text{(A.2)}$$

where $\overline{F}$ and $\overline{G}$ are the weighted averages of the intensity values in the template and the candidate match windows respectively:

$$\overline{F} = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) F(i,j) \qquad \text{(A.3)}$$

$$\overline{G} = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) G(i+\Delta i, j+\Delta j) \qquad \text{(A.4)}$$

The *variance-normalized correlation* measure $V$ is defined as,

$$V(\Delta i, \Delta j) = \frac{C(\Delta i, \Delta j)}{\sqrt{\overline{F^2}\,\overline{G^2}}} \qquad \text{(A.5)}$$

where $\overline{F^2}$ and $\overline{G^2}$ are the weighted second moments of the two windows:

$$\overline{F^2} = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) F(i,j)^2 \qquad \text{(A.6)}$$

$$\overline{G^2} = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j) G(i+\Delta i, j+\Delta j)^2 \qquad \text{(A.7)}$$

The second-moments are equal to the variances only when the corresponding means are zero; hence, the word *variance*-normalized is a slight misnomer.

The *mean-and-variance-normalized correlation* measure $MV$ is defined as,

$$MV(\Delta i, \Delta j) = \frac{M(\Delta i, \Delta j)}{\sqrt{\left(\overline{F^2} - \overline{F}\,\overline{F}\right)\left(\overline{G^2} - \overline{G}\,\overline{G}\right)}} \tag{A.8}$$

In this case, the denominator is exactly the variance of the two windows.

The *sum of squared-differences* measure is defined as,

$$S(\Delta i, \Delta j) = \frac{1}{(2n+1)^2} \sum_{i=-n}^{n} \sum_{j=-n}^{n} W(i,j)\,(F(i,j) - G(i+\Delta i, j+\Delta j))^2 \tag{A.9}$$

## A.2 Relationships Between the Different Correlation Measures

In this section, we examine the relationship between the different correlation measures, by considering their mathematical definitions given above. We show that for our hierarchical matching approach, the effect of using these measures can be expected to be similar to each other.

The relationship between $C$, $M$, $V$, and $MV$ can be understood by examining the effect of the first and the second moments of $F$ and $G$ in these measures. If $\bar{F}$ and $\bar{G}$ are small, then it is easy to see that $M \simeq C$ and $MV \simeq V$. Further, since $\overline{F^2}$ is independent of the displacement $(\Delta i, \Delta j)$, and $\overline{G^2}$ also does not vary significantly over a small area, we can assume that *if the search area is small*, the denominators of $V$ and $MV$ are more-or-less constant over the entire search area. Hence, maximizing $V$ or $MV$ is equivalent to maximizing their numerators. Since according to our definitions given in the previous section, the numerator of $V$ is $C$ and the numerator of $MV$ is $M$, we can conclude the following: *If the search area is small and the means of $F$ and $G$ are small, the maxima of the four measures $C$, $M$, $V$, and $MV$ will nearly coincide.*

The expansion of the sum of squared-differences measure shows that

$$S(\Delta i, \Delta j) = \overline{F^2} + \overline{G^2} - 2C(\Delta i, \Delta j) \tag{A.10}$$

Once again for small search areas, the effect of the second moments can be ignored. Since $S$ is proportional to $-C$, minimizing $S$ is roughly equivalent to maximizing $C$.

Thus, under the assumptions that the first moments are small and the search area is small enough that the variations in the second moment of $G$ can be ignored, all the correlation measures behave approximately alike. Both these conditions are achieved in our hierarchical matching process based on spatial frequency channels. The band-pass filtering process helps maintain small means within each channel, and the coarse-to-fine control strategy allows the search area within each level to be small. Therefore, we conclude that for our hierarchical matching approach, any of the correlation measures defined above can be chosen as the match measure. As we noted in chapter III, we have chosen the sum of squared-difference measure, because it is easy to perform a mathematical analysis of its behavior, and because it is guaranteed to be non-negative.

## A.3 Choosing the Size of the Template-Window

An issue that recurs during the design of any correlation matching algorithm is the choice of the size and "shape" (i.e., the weighting function) of the correlation window. A mathematical analysis is made difficult by the fact that it is difficult to characterize the type of variations of the input intensity functions found in real images. Preliminary efforts in this regard can be found in the recent work of Kass [55]. An empirical approach is also difficult, because it is hard to specify a set of test cases that are representative of the important situations. One of the few examples of a carefully conducted empirical study is due to Burt, Yen, and Xu [19]. As noted in chapter III, the conclusions of Burt *et al.* have influenced our own choice of parameters.

Given that the range of frequencies allowed within each of our spatial-frequency tuned channels is restricted and predetermined by the filters we have

used, a qualitative analysis of the effect of the window-function and the window-size on the match results seems possible. In this section, we outline a preliminary analysis which lends support to our choice of $5 \times 5$ Gaussian windows.

The analysis performed here is for an ideal one-dimensional version of the problem. The SSD measure has been chosen for analysis, since it has been used in our algorithm. The analysis is for the "continuous case", i.e., the displacement is allowed to take all real values and the summation in the definition of $S$ given in equation A.9 is replaced by an integral.

Assume that the input function $F = \sin 2\pi x/\lambda$ and $G$ is simply a shifted version of this, i.e. $G = \sin 2\pi(x - \delta)/\lambda$. The continous definition of the SSD measure is,

$$S(\Delta x) = \int_{-\infty}^{\infty} W(x)\,(F(x) - G(x + \Delta x))^2\,dx$$

Although in the above definition the range of the integration is $(-\infty, \infty)$, the weighting function may have non-zero values for a finite range; if so, the infinite integral can be replaced with a suitably finite sized integration. Moreover, even if the weighting function is chosen to have infinite support, it can be chosen as something which asymptotically tends to zero (e.g., a Gaussian). In this case, an approximate estimate can still be obtained by limiting the integration to a finite range where the values of the weighting function are significant.

If the weighting function is assumed to be even, i.e., $W(x) = W(-x)$, and normalized unity, then after some derivation, it can be shown that for the inputs given above,

$$S(\Delta x) = (1 - k)(1 - cos\frac{2\pi(\Delta x - \delta)}{\lambda}) + \frac{1}{2}k(1 - cos\frac{4\pi(\Delta x - \delta)}{\lambda})$$

where $k$ is the real part of the Fourier component of $W$ at the frequency $\frac{2}{\lambda}$, which is twice frequency of the input sinusoid,

$$k = \int W(x) cos\frac{4\pi x}{\lambda}\,dx$$

Figure 85 graphically displays the two terms in the above expression as a function of the displacement $\Delta x$, as well $S$, which is their sum.

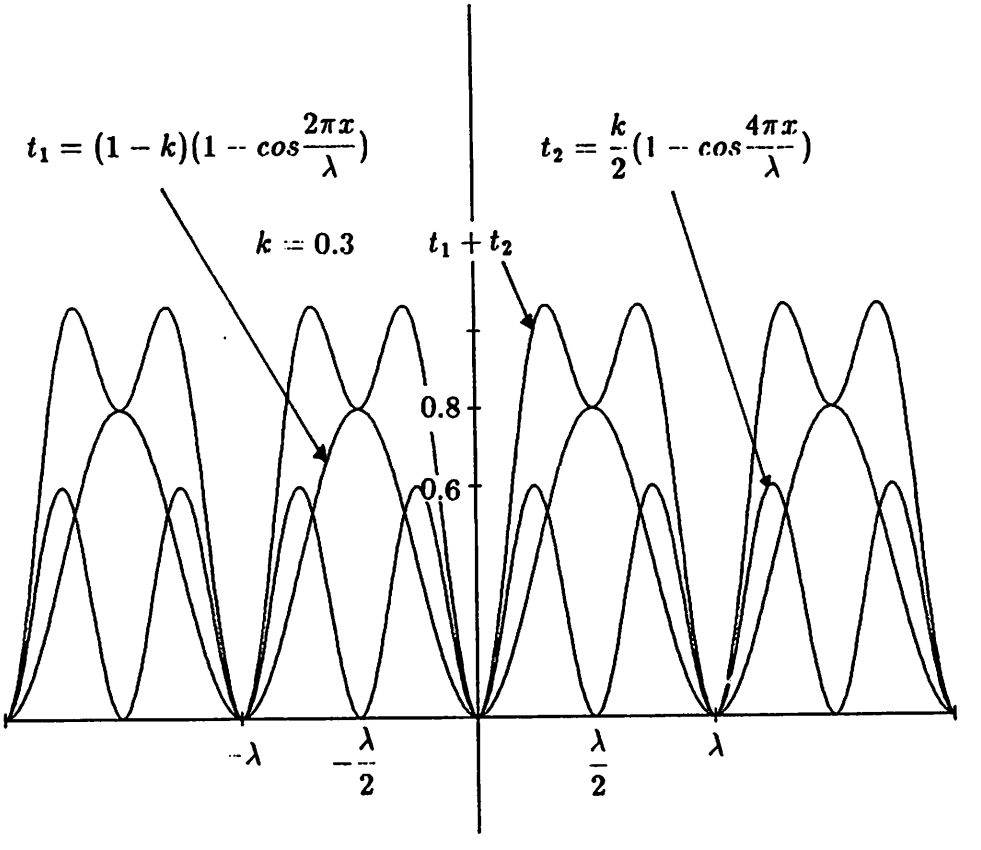



**Figure 85:** The two terms that compose the $SSD$ measure

The graphs and the equations indicate that the SSD function is periodic with the same frequency as the input sinusoid. This implies that all displacements of the form $(n\lambda+\delta), n = \cdots, -1, 0, 1, \cdots$ will give the same $SSD$ value. This means that the effective search area width is $\lambda$.

The key factor in the equation for $S$ is $k$. It is clear that as the ratio $\frac{k}{2(1-k)}$ increases, the second-harmonic term in the SSD function becomes dominant. Hence, there will be local minima in $S$ at the mid-point (i.e., at a distance $\lambda/2$)

between the true minima. In the ideal case, where $G$ is a purely shifted version of $F$, and neither is corrupted by noise or any signal distortion, this is not a problem. The true minima can be recognized as the one where $S$ attains the value zero. In practice, however, the true minima will rarely equal zero due to noise and other such factors. The local minima can even be smaller than the true minima, which leads to incorrect matches. As seen from the figure above, the effective range of displacements may be reduced to $\lambda/2$.

The type of reasoning given above sugggests that the effect of the second harmonic should be minimized. This can be achieved by reducing the value of $k$. Thus, our choice of the window-function and window-size can be made by comparing the factor $k$ for different candidates.

Consider two weighting functions that are typically used in correlation matching algorithms: (i) a uniform rectangular window which has a constant weight within an interval $(-w, w)$ and zero elsewhere, and (ii) a Gaussian window, wherein the weighting function is a Gaussian distribution of standard-deviation $\sigma$.

The rectangular window has a Fourier transform which is a sinc function, which does not monotonically decrease to zero. This means that as the window-size $w$ increases $k$ may actually increase for some range. It is preferable to have a weighting function that shows a more monotonic behavior. The Gaussian window affords such a choice. For a given input frequency, $k$ monotonically decreases with increasing values for the standard-deviation of the Gaussian function. Thus, the standard-deviaton $\sigma$ can be chosen so as to make $k$ smaller than any desired value. The size of the Gaussian window is determined by $\sigma$, since the values of the Gaussian weighting function are insignificant outside the range $(-4\sigma, 4\sigma)$.

Instead of a sinusoidal signal of single frequency, let us consider a more general signal with significant energies at all frequencies within a range $(f_l, f_u)$. Although, in a strict sense, the signals at the different frequencies cannot be treated independently of each other, a qualitative idea concerning the window-size can be

obtained by such a treatment. Since the power-spectrum of the Gaussian window monotonically decreases with increasing frequencies, for a given window-size, the value of $k$ will be the largest for the smallest frequency in that range. This means that the window-size can be determined according to the smallest frequency $f_l$.

The band-pass filters which constitute Burt's Laplacian pyramid are difference-of-Gaussian filters. The filter at the finest level is simply a high-pass filter, and for the one dimensional case it can be approximated by the following function,

$$H(s) = 1 - e^{-2\pi\sigma^2 s^2}$$

where $s$ is the spatial-frequency, and $\sigma$ is the standard deviation of the Gaussian convolution function. We noted earlier that the standard deviation of the $5 \times 5$ mask used in Burt's algorithm, is approximately 0.56 pixels.

At the frequency of 1/4 cycles per pixel, the value of $H$ is approximately 0.32. This means that the frequencies lower than 1/4 cycles per pixel will be attenuated even more, and can be neglected. Thus, the lowest-frequency which has significant energy is 1/4 cycles per pixel, which corresponds to a wavelength of 4 pixels. Recall that for our algorithm, we used $5 \times 5$ template windows, and that the window function was the same as the Gaussian convolution mask of Burt's pyramid, i.e., its standard deviation is 0.56 pixels. It is easy to verify that the value of $k$ corresponding to a sinusoid of frequency 1/4 cycles per pixel is approximately 0.21 and the corresponding value of $\frac{k}{2 \times (1-k)}$ is 0.13. This means that the effect of the second harmonic term is about a tenth of the fundamental signal, and can therefore be ignored.

While it is clear that larger windows would be more effective in reducing the effect of the second harmonic term, as explained in chapter III, the windows should be small in order to reduce the computational load as well to reduce the overlap across image-motion boundaries. It is on this basis that we have chosen to use the $5 \times 5$ Gaussian windows for most of our experiments. Given that

our hierachical algorithm is uniform across the scale-space, this analysis easily generalizes to all levels of the pyramid process.

Our analysis given above is for a one-dimensional signal. Although its extension to a general two-dimensional image is more complex, we believe that the qualitative results obtained here would easily extend to the 2-D case as well.

# APPENDIX B

# ALGORITHM FOR COMPUTING THE CONFIDENCE MEASURE

This appendix contains a brief description of the procedure used for determining the principal curvatures of the SSD surface and the confidence measures. Our approach is based on using a second-order polynomial approximation for $S$ within a $3 \times 3$, area around the displacement estimate provided by the matching process. The coefficients of this polynomial are computed by a numerical method described below. Once these coefficients are known, it is a simple matter to determine the location of the minimum of $S$ to subpixel accuracy and the two principal curvatures and the orientations of the associated principal axes. The principal curvatures are scaled according to the crieteria discussed in chapter IV to obtain the confidence measures.

## B.1 Computing the derivatives – Beaudet's masks

In his brief paper titled "Rotationally invariant image operators", [15] Beaudet described a set of masks for computing the various derivatives (at a pixel) of a two-dimensional discrete image based on the values within an $n \times n$ window around the point. Beaudet's paper contains the masks for a range of values of $n$ – the smallest of these, viz., $3 \times 3$ masks are the ones of interest here.

Beaudet's approach for determining these masks was the following: If the

image-function is approximated by its Taylor series expansion of order $n$ – i.e.,

$$S(\Delta x, \Delta y) = \sum_{k,l=0}^{n} \frac{1}{k!l!} S_{kl} \Delta x^k \Delta y^l,$$

where the derivative

$$S_{kl} = \frac{\partial^{k+l} S}{\partial \Delta x^k \partial \Delta y^l},$$

evaluated at the origin, then the values in the $n \times n$ window $W$ around the origin can be regarded as noisy estimates of this function evaluated at the pixel locations within the window. Let these values be denoted by the set $\{F(\Delta x_i, \Delta y_j), (\Delta x_i, \Delta y_j) \in W\}$. Given a set of values for the coefficients $C = \{S_{kl};\ k, l = 0, \ldots, n\}$, an error measure can be defined as

$$E(C) = \sum_{(\Delta x_i, \Delta y_j) \in W} (S(\Delta x_i, \Delta y_j; C) - F(\Delta x_i, \Delta y_j))^2$$

In Beaudet's method, the coefficent set $C$ is chosen so as to minimize the error $E(C)$.

Starting from this minimization criterion, Beaudet was able to express each of the coefficients in the set $C$ as a weighted sum of the values of the function within the $n \times n$ window. The weight set for each coefficient can then be regarded as a mask, and specifying the mask completely specifies the method of computing these coefficients. In figure 86 the masks for the various partial derivatives upto the second order are shown for a $3 \times 3$ window.

Once the image derivatives are computed according to this procedure, the remaining process involves three steps: first, the principal curvatures and the orientation of the principal axes are determined by diagonalizing the $2 \times 2$ matrix containing the second partial derivatives of $S$. Following this a rotation of the image coordinate system is performed such that the two coordinate axes are parallel to the principal axes, and the minimum of $S$ is obtained to sub-pixel precision. This is taken as a sub-pixel estimate of the displacement. Finally, the

$$S_x = S_y^T = \frac{1}{6}\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$S_{xy} = \frac{1}{4}\begin{bmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

$$S_{xx} = S_{yy}^T = \frac{1}{3}\begin{bmatrix} 1 & -2 & 1 \\ 1 & -2 & 1 \\ 1 & -2 & 1 \end{bmatrix}$$

**Figure 86:** Beadet's masks for a $3 \times 3$ window

curvatures are scaled to obtain the confidence measures. Since this last process has been discussed at length in chapter V, the remainder of this appendix simply describes the first two steps.

## B.2 Determining the Principal Curvatures

Since the surface $S$ is defined by the second order polynomial function $S$, the second derivatives of this surface will be constant throughout its domain of definition. Hence, these derivatives can be evaluated at any location on the surface, and in particular, they can be evaluated at the origin (which, as noted above, is the displacement estimate given by the matching process), instead of the true minimum of $S$ determined to sub-pixel precision. This means that the second-derivatives computed above according the Beaudet's masks can be used.

It is a well known fact in the study of the differential geometry of surfaces [26] that when the gradient vector is small, the second fundamental form is approximated by the Hessian, and the the principal curvatures and the principal axes are the eigenvalues and the corresponding eigenvectors of the the matrix of the second derivatives $H$ (usually called the *Hessian*), which is defined as follows,

$$H = \begin{pmatrix} S_{xx} & S_{xy} \\ S_{xy} & S_{yy} \end{pmatrix}$$

Note that the above definition incorporates a change of variable names from $(\Delta x, \Delta y)$ to $(x, y)$. This has been done simply for notational convenience. The process of diagonalization of this matrix reveals that the eigenvalues are,

$$C_{max} = \frac{1}{2}(S_{xx} + S_{yy}) + \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}^2} \qquad \text{(B.1)}$$

$$C_{min} = \frac{1}{2}(S_{xx} + S_{yy}) - \sqrt{(S_{xx} - S_{yy})^2 + 4S_{xy}^2} \qquad \text{(B.2)}$$

and the angle $\theta_{min}$ measured clockwise from the eigenvector $\hat{e}_{min}$ to the $x$ axis is,

$$\theta_{min} = \arctan(C_{min} - S_{xx}, S_{xy})$$

The eigenvector $\hat{e}_{max}$ corresponding to $C_{max}$ is perpendicular to $\hat{e}_{min}$, since the matrix $H$ is real and symmetric.

## B.3 Determining the Minimum of $S$ to Sub-Pixel Precision

The two eigen vectors of the Hessian matrix of $S$ form a new basis for the $(x, y)$ plane. This is equivalent to performing a counter-clockwise rotation of the coordinate system by an angle $\theta_{min}$. If the new coordinates of a point are denoted by $\alpha$ (corresponding to $\hat{e}_{min}$) and $\beta$ (corresponding to $\hat{e}_{max}$), then the function $S$ can be rewritten as

$$S(\alpha, \beta) = S_0 + S_\alpha \alpha + S_\beta \beta + \frac{1}{2} C_{min} \alpha^2 + \frac{1}{2} C_{max} \beta^2$$

where $S_\alpha = S_x \cos\theta_{min} + S_y \sin\theta_{min}$, and $S_\beta = -S_x \sin\theta_{min} + S_y \cos\theta_{min}$ are the directional derivatives of $S$ along $\hat{e}_{min}$ and $\hat{e}_{max}$ respectively.

The exact location $(\alpha_0, \beta_0)$ of the minimum of $S$ can be obtained by setting the partial derivatives of $S$ simultaneously to zero:

$$\alpha_0 = -S_\alpha / S_{\alpha\alpha} \tag{B.3}$$

$$\beta_0 = -S_\beta / S_{\beta\beta} \tag{B.4}$$

However, an extremum so determined need not be a minimum. It may be a minimum in one direction, but a maximum along another; or it may be minimum (or a maximum) along all directions. The exact behaviour of $S$ at the extremum is indicated by the sign of the second derivatives, which in this case are $C_{max}$ and $C_{min}$.

First, note that $C_{max} \geq C_{min}$. If both are negative, then the extremum is a maximum of $S$. If only $C_{min}$ is negative, then it is called a *saddle point* and indicates a minimum along $e_{max}$ and a maximum along $e_{min}$. If, on the other hand, both are positive, then the extremum is a minimum of $S$. These observations are used in the procedure for sub-pixel determination of the displacement given below.

- If $|\alpha_0| > 1$ or $C_{min} \leq 0$, then $\alpha_0$ is set to 0, and $C_{min}$ is set to 0.
- If $|\beta_0| > 1$ or $C_{max} \leq 0$, then $\beta_0$ is set to 0, and $C_{max}$ is set to 0.

The reason for the restriction on the magnitude of $\alpha_0$ and $\beta_0$ arises from the fact that the matching process has already provided the displacement within a pixel accuracy. In this case the larger factional displacement indicates that the second order approximation for the SSD surface is invalid for the estimation of the corresponding component of the displacement of that pixel. Hence, the assciated confidence measure is set to 0, and the fractional displacement is simply ignored.

# APPENDIX C

## RELATIONSHIP BETWEEN MATCHING AND GRADIENT APPROACHES - A MATHEMATICAL VIEW

Traditionally, the gradient based approaches for velocity computations and the correlation-based approaches for displacement computations have been regarded as being significantly different from each other. In an obvious sense they are indeed different, since they compute different quantities. However, in this appendix we will show that there are strong connections between the two types of approaches. In particular, we will show that in the limit, when the inter-frame time interval tends to zero, the results obtained by minimizing the sum of squared-differences (SSD) measure converges to those obtained from the gradient-based velocity estimation process.

When the inter-frame time interval tends to zero, the image displacements will also tend to zero. However, the corresponding "average" velocities $\vec{U} = \frac{\vec{D}}{\delta t}$ may still be finite. The average velocities obtained from the matching approach correspond to the instantaneous velocities obtained from the gradient-based approach.

In correlation-based approaches, the match measure is computed using the intensity values in a finite-sized area around the point of interest, whereas the most commonly used first order normal-flow constraint (see chapter II) uses quantities computed over infinitesmal areas. Therefore, for comparison with the first order gradient-based approach, the area of the template-window should also tend to zero. Note, however, that the second-order constraint used by Nagel considers

finite-sized windows. As it will be shown here, the matching approach converges to the second-order gradient-based approach even when the window sizes are finite.

Our approach for determining the relationship between the matching and the gradient-based approaches is based on comparing the minimization problems formulated in this thesis to those of Nagel (using the second order intensity constraint) and Horn and Schunck (using the first order intensity constraintt). In particular, we will show that under the appropriate limiting conditions, these minimization problems are equivalent to one another. We formulate our conclusions in the form of the following theorem:

**Theorem 1** *In the limit, when the inter-frame time interval tends to zero, the formulation of the approximation error for image displacements used in the discrete matching approach converges to the second-order formulation of* $E_{int}$ *for image-velocities used in the gradient-based approach, provided the third and higher order spatial intensity derivatives are ignored. Further, when the window function for correlation tends to a delta function (i.e., the template-window tends to a point),* $E_{app}$ *converges exactly to the first order gradient-based formulation of* $E_{int}$.

The remainder of this appendix consists of the proof of the theorem given above and a discussion of its implications. First, we repeat Nagel's derivations (see [75]) of the mathematical condition for the minimum of the SSD measure. This condition is in the form of a vector equation $AU = -b$, where $A$ is a $2 \times 2$ matrix and $b$ is a $2 \times 1$ vector, both involving the derivatives of the intensity function, and $U$ is the image-velocity corresponding to the displacement determined by minimizing the SSD measure. Following this, we show that there is at least one solution to this equation. Finally, we prove Theorem 1 above by relating the various minimization problems formulated in the matching and the gradient-based approaches to the equation $AU = -b$.

## C.1 Nagel's Derivation of the Second Order Intensity Constraint

Without loss of generality, let the point of interest be the origin $(0,0)$ and the two intensity functions being matched $I$ and $J$. Then, according to the definitions given in Appendix A[1]

$$S(\delta x, \delta y) = \iint W(x,y)\,|I(x,y) - J(x+\delta x, y+\delta y)|^2\,dx\,dy \qquad \text{(C.1)}$$

where $S$ is the SSD measure, $W(x,y)$ is an even symmetric, normalized weighting function, i.e.,

$$\iint W(x,y)\,dx\,dy = 1$$

$$\iint x^i y^j W(x,y)\,dx\,dy = 0 \quad \text{for } i \text{ or } j \text{ odd}$$

For future use, we also define $\sigma_x$ and $\sigma_y$ to be the standard deviations of the weight function.

$$\sigma_x^2 = \iint x^2 W(x,y)\,dx\,dy$$

$$\sigma_y^2 = \iint y^2 W(x,y)\,dx\,dy$$

Usually, circularly symmetric weight functions are used, hence, $\sigma_x = \sigma_y = \sigma$.

In [75], Nagel derives the expressions for the first-order partial derivatives of the SSD surface in terms of the image intensity-derivatives. In this section, his analysis is extended to the second-order partial derivatives as well, since those derivatives are of interest for the determination of the confidence measures.

[1]Since the discrete correlation process can be regarded as an approximation of the correlation matching scheme on continuous spatial images, we have replaced the summation involved in the definition of the SSD measure by an integration. The limits of all of the above integrals are $(-\infty, \infty)$, although most of the support for the weight function will usually be within a small area. For mathematical convenience, this continuous version will be used in remainder of this appendix.

In his approach Nagel assumes that (i) the displacement of a point is "small", and (ii) locally, the intensity functions of the two images can be described by their second-order Taylor series approximations. The same assumptions will be used here; they are valid for the hierarchical matching framework, because within a spatial-frequency tuned channel, the displacements and the search areas are usually small compared to the scale of significant intensity variations.

$$J(x,y) = J_0 + J_x x + J_y y + \frac{1}{2}J_{xx}x^2 + J_{xy}xy + \frac{1}{2}J_{yy}y^2 \qquad \text{(C.2)}$$

The SSD measure defined in equation C.1 can then be rewritten as,

$$\begin{aligned} S(\delta x, \delta y) \;=\; & \iint W \Big[ I - J_0 - J_x(x+\delta x) - J_y(y+\delta y) - \frac{1}{2}J_{xx}(x+\delta x)^2 \\ & - J_{xy}(x+\delta x)(y+\delta y) - \frac{1}{2}J_{yy}(y+\delta y)^2 \Big]^2 \, dx\, dy \end{aligned} \qquad \text{(C.3)}$$

Note that in the equation given above, $W$ and $I$ are functions of $(x,y)$, while the terms involving $J$ are constant coefficients. The partial derivatives of the SSD measure can be obtained by differentating equation C.3,

$$\begin{aligned} S_{\delta x} \;=\; & -2 \iint W(x,y) \left[ J_x + J_{xx}(x+\delta x) + J_{xy}(y+\delta y) \right] \\ & \left[ I(x,y) - J_0 - J_x(x+\delta x) - J_y(y+\delta y) - \cdots \right] dx\, dy \end{aligned} \qquad \text{(C.4)}$$

$$\begin{aligned} S_{\delta y} \;=\; & -2 \iint W(x,y) \left[ J_y + J_{xy}(x+\delta x) + J_{yy}(y+\delta y) \right] \\ & \left[ I(x,y) - J_0 - J_x(x+\delta x) - J_y(y+\delta y) - \cdots \right] dx\, dy \end{aligned} \qquad \text{(C.5)}$$

Using the fact that the weighting function is even-symmetric and normalized, we obtain

$$\begin{aligned} S_{\delta x} &= -2\left[J_x + J_{xx}\delta x + J_{xy}\delta y\right]\left[\tilde{I} - \tilde{J} - \frac{1}{2}(J_x\delta x + J_y\delta y)\right. \\ &\quad \left. -\frac{1}{2}\delta x(J_x + J_{xx}\delta x + J_{xy}\delta y) - \frac{1}{2}\delta y(J_y + J_{xy}\delta x + J_{yy}\delta y)\right] \\ &\quad -2J_{xx}\sigma^2\left[I_x - (J_x + J_{xx}\delta x + J_{xy}\delta y)\right] \\ &\quad -2J_{xy}\sigma^2\left[I_y - (J_y + J_{xy}\delta x + J_{yy}\delta y)\right] \qquad \text{(C.6)} \\ S_{\delta x\delta x} &= 2J_{xx}\left[\tilde{I} - \tilde{J} - \frac{1}{2}(J_x\delta x + J_y\delta y) - \right. \\ &\quad \left. \frac{1}{2}\delta x(J_x + J_{xx}\delta x + J_{xy}\delta y) - \frac{1}{2}\delta y(J_y + J_{xy}\delta x + J_{yy}\delta y)\right] \\ &\quad +2\left[J_x + J_{xx}\delta x + J_{xy}\delta y\right]^2 + 2\left(J_{xx}\right)^2\sigma^2 + 2\left(J_{xy}\right)^2\sigma^2 \qquad \text{(C.7)} \end{aligned}$$

where

$$\tilde{I} = \iint I(x,y)W(x,y)\,dx\,dy$$

and

$$\tilde{J} = \iint J(x,y)W(x,y)\,dx\,dy$$

The formulas for the other derivatives of $S$ can be easily written down by symmetry, and have been omitted here for the sake of brevity.

If the inter-frame time interval is $\delta t$, we can replace $\delta x$ by $u\delta t$ and $\delta y$ by $v\delta t$, where $u$ and $v$ are the components of the average image-velocities. When $\delta t$ tends to zero, the above expression for $S_{\delta x}$ trivially tends to zero; however, the limiting values for $u$ and $v$ can still be obtained by equating $S_u = S_{\delta x}/\delta t$ to zero and then taking the limit. Let $S_u$ and $S_v$ denote $S_{\delta x}/\delta t$ and $S_{\delta y}/\delta t$ respectively. Thus, we obtain

$$\begin{aligned} \frac{1}{2}S_u &= I_x(\tilde{I}_t - I_x u - I_y v) + I_{xx}\sigma^2(I_{xt} - I_{xx}u - I_{xy}v) \\ &\quad I_{xy}\sigma^2(I_{yt} - I_{xy}u - I_{yy}v) \qquad \text{(C.8)} \end{aligned}$$

$$\frac{1}{2}S_v = I_y(\tilde{I}_t - I_x u - I_y v) + I_{xy}\sigma^2(I_{xt} - I_{xx}u - I_{xy}v)$$
$$I_{yy}\sigma^2(I_{yt} - I_{xy}u - I_{yy}v) \qquad \text{(C.9)}$$

In order to determine the minimum of $S$, we set $S_u$ and $S_v$ to zero, and obtain the equation

$$AU = -b \qquad \text{(C.10)}$$

where $U = (u, v)^T$,

$$A = \begin{pmatrix} I_x^2 + I_{xx}^2\sigma^2 + I_{xy}^2\sigma^2 & I_xI_y + I_{xx}I_{xy}\sigma^2 + I_{yy}I_{xy}\sigma^2 \\ I_xI_y + I_{xx}I_{xy}\sigma^2 + I_{yy}I_{xy}\sigma^2 & I_y^2 + I_{xy}^2\sigma^2 + I_{yy}^2\sigma^2 \end{pmatrix}$$

and

$$b = \begin{pmatrix} I_x\tilde{I}_t + I_{xx}I_{xt}\sigma^2 + I_{xy}I_{yt}\sigma^2 \\ I_y\tilde{I}_t + I_{xy}I_{xt}\sigma^2 + I_{yy}I_{yt}\sigma^2 \end{pmatrix}$$

The matrix $A$ and the vector $b$ can also be written in the following compact forms:

$$A = (\nabla I)(\nabla I)^T + \sigma^2(\nabla\nabla I)(\nabla\nabla I)^T \qquad \text{(C.11)}$$

and

$$b = I_t(\nabla I) + \sigma^2(\nabla\nabla I)(\nabla I_t) \qquad \text{(C.12)}$$

where the $\nabla\nabla$ operator represents the matrix of second derivatives, i.e.,

$$\nabla\nabla I = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix},$$

The above equations indicate that even when the second order intensity variations are not ignored, the constraint on the velocity vector is linear. When $\sigma = 0$, the following equation is obtained:

$$(\nabla I)(\nabla I)^T U = -I_t(\nabla I)$$

which reduces to the equation

$$\nabla I\, U = -I_t$$

which is the same as the normal-flow equations traditionally used in the first order gradient-based approaches (see chapter II).

Since our confidence measures depend on the second-order derivatives of $S$, it is useful to derive the expressions for those derivatives. When inter-frame time interval tends to zero, the components $\delta x$ and $\delta y$ of the image-displacements tend to zero, and $J \rightarrow I$. Therefore, from equation C.7 we obtain that

$$\begin{aligned} S_{\delta x \delta x} &= 2\left(I_x^2 + I_{xx}^2\sigma^2 + I_{xy}^2\sigma^2\right) \\ S_{\delta x \delta y} &= 2\left(I_x I_y + I_{xx} I_{xy}\sigma^2 + I_{xy} I_{yy}\sigma^2\right) \\ S_{\delta x \delta x} &= 2\left(I_y^2 + I_{xy}^2\sigma^2 + I_{yy}^2\sigma^2\right) \end{aligned} \tag{C.13}$$

Note that these derivatives are proportional to the terms composing the matrix $A$. This means that the Hessian matrix is identical to $A$, and that the confidence measures are proportional to the eigenvalues of $A$.

For windows that are small but of non-zero size, the intensity surface around a point along an edge in a band-pass filtered image can be approximated by a plane. If so, the second-order terms vanish, and it is easy to show that one of the eigenvalues of $A$ is zero; hence the corresponding confidence is also zero. However, at a corner point such a planar approximation is invalid (see [75]), and hence the second-order terms do not vanish. Hence, both the eigenvalues and the corresponding confidences have finite values, and depending on the precise values of the second-order derivatives, they can both be large. Finally, at a homogeneous area all the derivatives vanish and so do both the confidence measures.

The above line of reasoning, although brief, supports the empirical observations concerning the SSD surface, and hence the properties of the confidence measure described in chapter IV. Moreover, the availability of these equations allows for a detailed and precise analysis of the SSD surface under a wide-range of conditions, provided such conditions can be stated in terms of the properties of the image intensity functions.

## C.2 The Existence of a Solution to the Equation $AU = -b$

The analysis in the previous section showed that in the limit, when the inter-frame time interval tends to zero, the image-velocity $\vec{U}$ corresponding to the displacement estimate determined by minimizing the SSD measure are given by the equation $AU = -b$. In this section, we prove that there is at least one solution to this equation.

Since $A$ is real and symmetric, it can be diagonalized by a unitary transformation (see theorem 9.2 of [81]), or equivalently by a rotation $R$ of the coordinate system. Applying this transformation to the equation $AU = -b$, we obtain the equation $A'U' = -b'$, where $A' = RAR^T$, $U' = RU$, and $b' = Rb$. For convenience, we can drop the "primes" and continue to use $A$, $U$, and $b$, with the understanding that the derivatives of the intensity function which compose $A$, and the components $(u, v)$ of $U$ are all expressed with respect to the rotated coordinate system $(\alpha, \beta)$. Since $A$ is diagonal, we have two independent equations,

$$\left(I_\alpha^2 + I_{\alpha\alpha}^2\sigma^2 + I_{\alpha\beta}^2\sigma^2\right) u = -\left(I_\alpha I_t + I_{\alpha\alpha}I_{\alpha t}\sigma^2 + I_{\alpha\beta}I_{\beta t}\sigma^2\right) \qquad \text{(C.14)}$$

$$\left(I_\beta^2 + I_{\alpha\beta}^2\sigma^2 + I_{\beta\beta}^2\sigma^2\right) v = -\left(I_\beta I_t + I_{\alpha\beta}I_{\alpha t}\sigma^2 + I_{\beta\beta}I_{\beta t}\sigma^2\right) \qquad \text{(C.15)}$$

Note that the multipliers of $u$ and $v$ in these equations are the eigenvalues of the original matrix $A$.

Obviously, a solution exists if the left-hand sides of the two equations given above are non-zero. However, if the LHS of either equation is zero (e.g., $I_\alpha^2 + I_{\alpha\alpha}^2\sigma^2 + I_{\alpha\beta}^2\sigma^2 = 0$), then each of its component terms is zero (i.e., $I_\alpha = I_{\alpha\alpha} = I_{\alpha\beta} = 0$), and hence the corresponding right-hand side is also zero, (i.e., $I_\alpha I_t + I_{\alpha\alpha}I_{\alpha t}\sigma^2 + I_{\alpha\beta}I_{\beta t}\sigma^2 = 0$). Therefore, the corresponding component of $U$ can be arbitrarily set to zero. Thus, a solution to the equation $AU = -b$, exists, independent of whether or not the eigenvalues of $A$ are zero.

## C.3 Relating the Minimization Problems to the Equation $AU = -b$

Let $\vec{D}$ be any vector that solves the equation $AU = -b$, i.e., $AD = -b$. Consider the functional

$$E(U) = \iint (U - D)^T A(U - D)\, dx\, dy + \iint trace\left\{(\nabla U)\, W\, (\nabla U)\right\}\, dx\, dy \quad \text{(C.16)}$$

In this section, we will show that the first term of this functional is closely related to the our $E_{app}$ which was defined in chapter IV, and the definitions of $E_{int}$ used in the gradient based approaches of Horn and Schucnk and Nagel.

### The SSD minimization approach

Ignoring the normalization factors involved, since the confidences $C_{max}$ and $C_{min}$ computed in our matching approach are the eigenvalues of $A$, and $\hat{e}_{max}$ and $\hat{e}_{min}$ are the corresponding unit eigenvectors, i.e.,

$$\begin{aligned} Ae_{max} &= C_{max}e_{max} \\ Ae_{min} &= C_{min}e_{min} \end{aligned}$$

Further, since $A$ is real and symmetric, its eigenvectors are orthogonal to each other; hence, $e_{max}^T e_{min} = 0$.

By rewriting $(\vec{U} - \vec{D})$ as,

$$\vec{U} - \vec{D} = \left[\left(\vec{U} - \vec{D}\right) \cdot \hat{e}_{max}\right] \hat{e}_{max} + \left[\left(\vec{U} - \vec{D}\right) \cdot \hat{e}_{min}\right] \hat{e}_{min}$$

and using the relationships given above, it is easy to show that

$$(U - D)^T A(U - D) = C_{max}\left[(U - D)^T e_{max}\right]^2 + C_{min}\left[(U - D)^T e_{min}\right]^2$$

Since the right hand side of this equation is the same as the integrand in our $E_{app}$ defined in chapter IV, it is clear that the first term in the definition of $E$ given

in equation C.16 is the same as our $E_{app}$ defined in chapter IV. If, in addition, $W$ is chosen to be the identity matrix, the second term is the same as $E_{sm}$ defined in chapter IV, and $E(U)$ given above is the same as the functional minimized in our approach.

### The first-order gradient-based approach

It is easy to show that when $\sigma = 0$, $C_{max} = |\nabla I|$, $C_{min} = 0$, $\hat{e}_{max} = \hat{e}_{\nabla I}$, and $\vec{D} \cdot \hat{e}_{\nabla I} = -I_t/|\nabla I|$. Hence,

$$(U - D)^T A(U - D) = -\left((\nabla I)^T U + I_t\right)^2$$

From this, it can be seen that the first term of $E$ given in equation C.16 reduces to the $E_{int}$ used in the first-order gradient-based approach of Horn and Schunck [51]. Once again by choosing $W$ to be the identity matrix, the definition of $E$ given above can be reduced exactly to the functional that is minimized by Horn and Schunck.

### The second-order gradient-based approach

By using the Euler-Lagrange equation, it is easy to show that the minimization of the functional $E(U)$ is equivalent to solving the following system of differential equations:

$$AU + b - \begin{bmatrix} trace\{W\nabla\nabla u\} \\ trace\{W\nabla\nabla v\} \end{bmatrix} = 0$$

These are exactly the equations derived by Nagel and Enkelmann [31,78] in their second-order gradient-based approach for the measurement of image motion.

Although Nagel and Enkelmann do not separate the terms $E_{int}$ and $E_{sm}$, the analysis given above suggests that such a separation can be made. The term $E_{int}$ which reflects a local constraint on image-motion is the first term in our definition of $E(U)$. The term $E_{sm}$, which incorporates a smoothness assumption

is the second term. Nagel's approach differs from our matching approach in the choice of $W$; while we have chosen $W$ to be the identity matrix, Nagel and Enkelmann have chosen it to be a matrix which involves the spatial derivatives of the intensity function.

## C.4 Summary

In summary, we have shown that the matching approach and the gradient based approaches are closely related by considering the limiting case of our matching approach. This relationship also allows us, for the first time, to explicitly identify the confidences that have been thus far implicitly used in the gradient-based approaches; these confidences are proportional to the eigenvalues of the matrix $A$, which was defined earlier in this appendix.

# APPENDIX D

## THE SPATIO-TEMPORAL ENERGY VIEW OF THE HIERARCHICAL MATCHING APPROACH

In this appendix, we explain the relationship between the energy models and the matching approach based on the minimization of the SSD within the frequency/orientation-tuned channels. Since the energy models have been concerned with a one-dimensional spatial signal, it is appropriate to restrict attention to a one-dimensional version of the hierarchical matching process.

Consider the following extreme version of the hierarchical scheme. Let the input filters be ideal band-pass filters that select a single frequency – i.e., they have a pair of unit impulses at $-\frac{1}{\lambda}$ and $\frac{1}{\lambda}$ as their frequency spectrum. The analysis provided in Appendix A indicates that given an inter-frame displacement $\delta$, the SSD function varies as a function of the candidate displacement $\Delta x$ as follows:

$$SSD(\Delta x) = (1-k)(1-\cos\frac{2\pi(\Delta x-\delta)}{\lambda}) + \frac{1}{2}k(1-\cos\frac{4\pi(\Delta x-\delta)}{\lambda})$$

Another way of looking at the above expression is that if a detector is "tuned" to a displacement of $\delta$, its response to any other displacement will be defined by the equation given above.

An important conclusion from the analysis provided in Appendix A is was that the effective range of displacements that can be processed without ambiguity is $(-\lambda/2, \lambda/2)$, where $\lambda$ is the wavelength of the input sinusoid. Assume that the window sizes have been so arranged as to render the effect of the second-harmonic term in the above equation to be negligible. In this case, if three motion-detectors

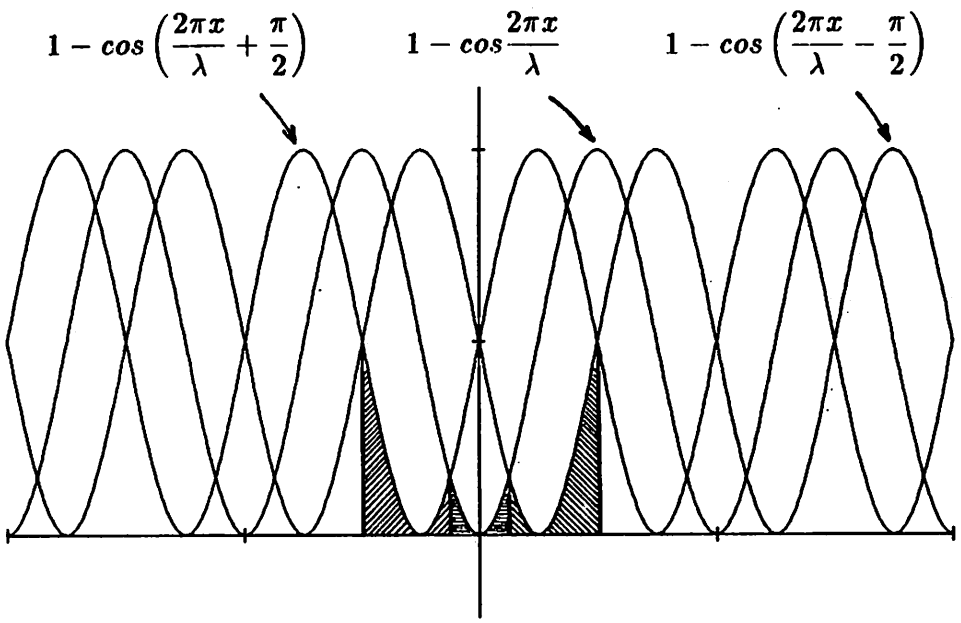



**Figure 87:** The response curves of the three detectors placed $\frac{\lambda}{4}$ apart from each other.

were arranged so as to cover the range of unambiguous motion given above, it is best to place them so that they are most sensitive to displacements of $(-\frac{\lambda}{4}, 0, \frac{\lambda}{4})$, as shown in figure 87. In this figure, the three curves indicate the response of the three detectors.

Except at the points where the response curves mutually intersect, for any displacement within the given range, one of the three detector will have a response smaller than the other two. Hence, a simple decision process that corresponds to minimizing the SSD measure would be to select its range as the measurement of motion (in discrete terms, the three detectors can be regarded as indicating whether the motion is to the left, right, or if the pattern remains stationary). Note, however, that due to the periodic nature of the response curves, one of these detectors will be selected as being suitable for displacements even beyond the $(-\lambda/2, \lambda/2)$ range. If the three detectors are named **L** , **S**, and **R**, indicating

the direction of motion then the horizontal bars marked in figure 87 shows their "receptive" fields along the "displacement" dimension.

There is an alternative way of describing the detection ranges of the three detectors described above. Let the inter-frame time interval be $\Delta t$. Then, a displacement of $\Delta x$ corresponds to an average velocity $\frac{\Delta x}{\Delta t}$. Using the relationship $f_t = f_s v$, where $f_s$ and $f_t$ are the spatial and the temporal frequency of the sinusoid and $v$ is its velocity, for a sinusoid of wavelength $\lambda$, the "average" temporal frequency associated with any displacement $\Delta x$ in time $\Delta t$ is the following:

$$\tilde{f}_t = \frac{\Delta x}{\lambda \Delta t}$$

It is convenient to measure the displacements in terms of $\lambda$, since the ranges of the various detectors are all proportional to it. Let $\Delta x = k\lambda$. The "average" temporal-frequency corresponding to a displacement of $k\lambda$ in time $\Delta t$ is $\frac{k}{\Delta t}$. From this, it is clear that the spatio-temporal frequency range supported by each of the three motion detectors (**L**, **S**, and **R**), which are all tuned to a spatial-frequency $\frac{1}{\lambda}$ will be as shown in figure 88. Two facts are brought to light by figure 88 as well as the analysis given above. First, it is clear that the temporal-frequency ranges of any of the detectors is independent of the spatial-frequency of the input signal. Second, the temporal aliasing that arises as the velocity increases is evident from the fact that a single detector is sensitive to several ranges of temporal frequencies.

Although the temporal-frequency range over which a motion detector has maximal response is independent of the spatial-frequency, the associated velocity range increases when the spatial-frequency selected by that detector is decreased. This is obvious from the inverse relationship between velocity and spatial frequency. This means that as the spatial frequency is decreased, it would take a larger image-velocity to create the temporal aliasing effect. Figure 88 also shows the velocity ranges supported by each motion-detector. The velocities are indicated by lines through the origin of the spatio-temporal frequency plane and the ranges supported by two different sets of motion-detectors are shown; the two

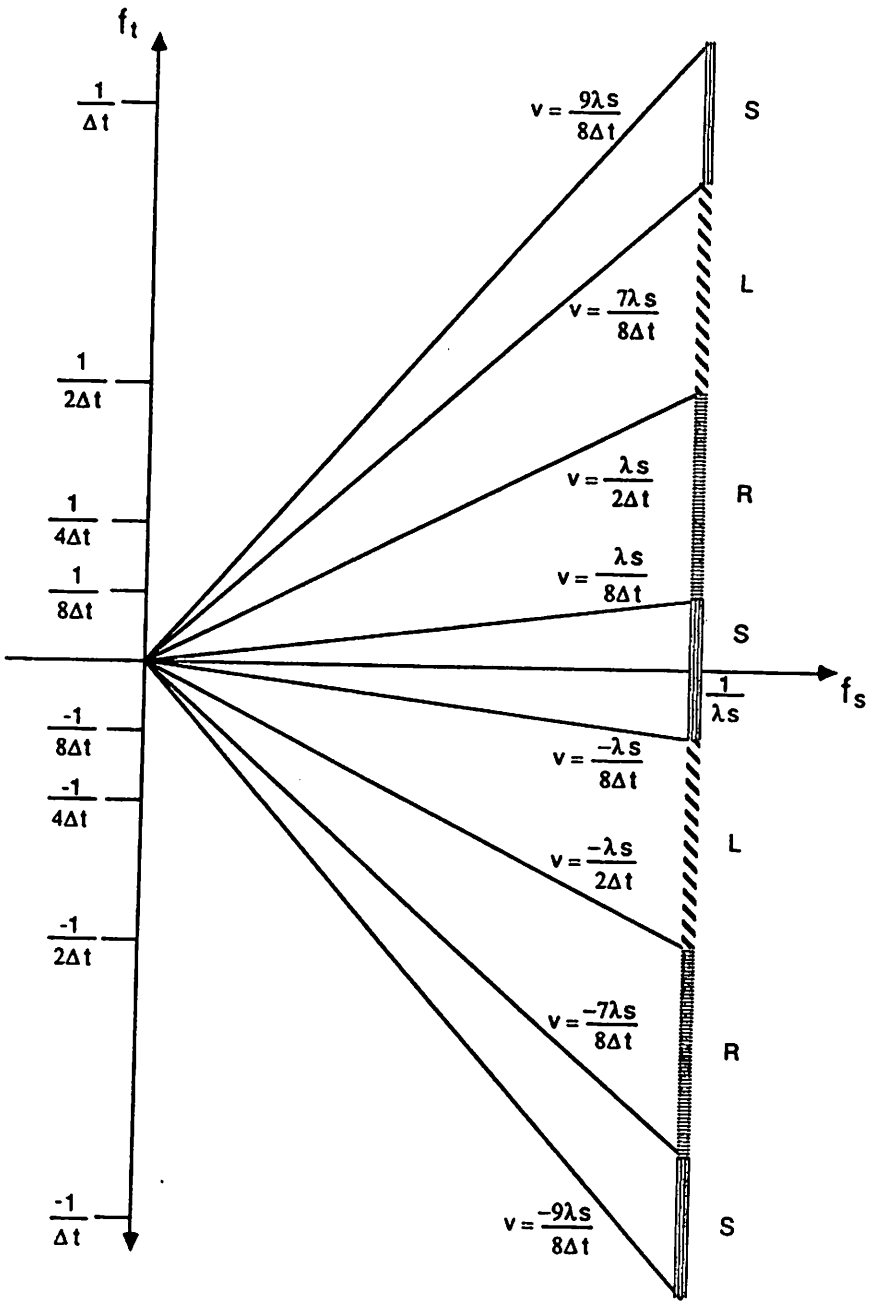



**Figure 88:** The temporal-frequency range of the three detectors shown in the spatio-temporal frequency domain

sets are tuned to different spatial-frequencies.

In the spatio-temporal energy view of the matching process, The coarse-to-fine strategy can be simply regarded as locating the proper velocity "line" in the spatio-temporal frequency plane in the following manner. First, the low spatial-frequency tuned detector selects one of the three velocity ranges. As seen from the figure 89, when considered by itself, the activity in the **L** detector tuned to a higher spatial-frequency cannot uniquely determine the correct temporal-frequency (and hence the velocity) range present. However, the velocity range selected by the low-spatial-frequency tuned unit can disambiguate between the multiple-choices for velocity. As the reader may recall, this was in fact the essence of the "spectral-continuity" constraint. Hence, the spatio-temporal-frequency view can be regarded as an alternative explanation of the same principle.

Finally, it must also be noted that there is a close similarity between the type of one-dimensional matching based on minimizing the sum of squared differences that was presented above and the spatio-temporal energy model proposed by Watson and Ahmuda [117]. The primary distinction between the two models is that Watson and Ahumda's units perform an integration over time (after squaring the differences) whereas the hierarchical framework presented here performs an integration over an area of the image. This difference, in turn, arises because of the difference in the paradigms assumed by the two approaches [54]. Watson and Ahmuda's scheme fits into a *discrete-space continuous-time* view, whereas the viewpoint of the area based inter-frame matching approach fits into a *continuous-space discrete time* view.

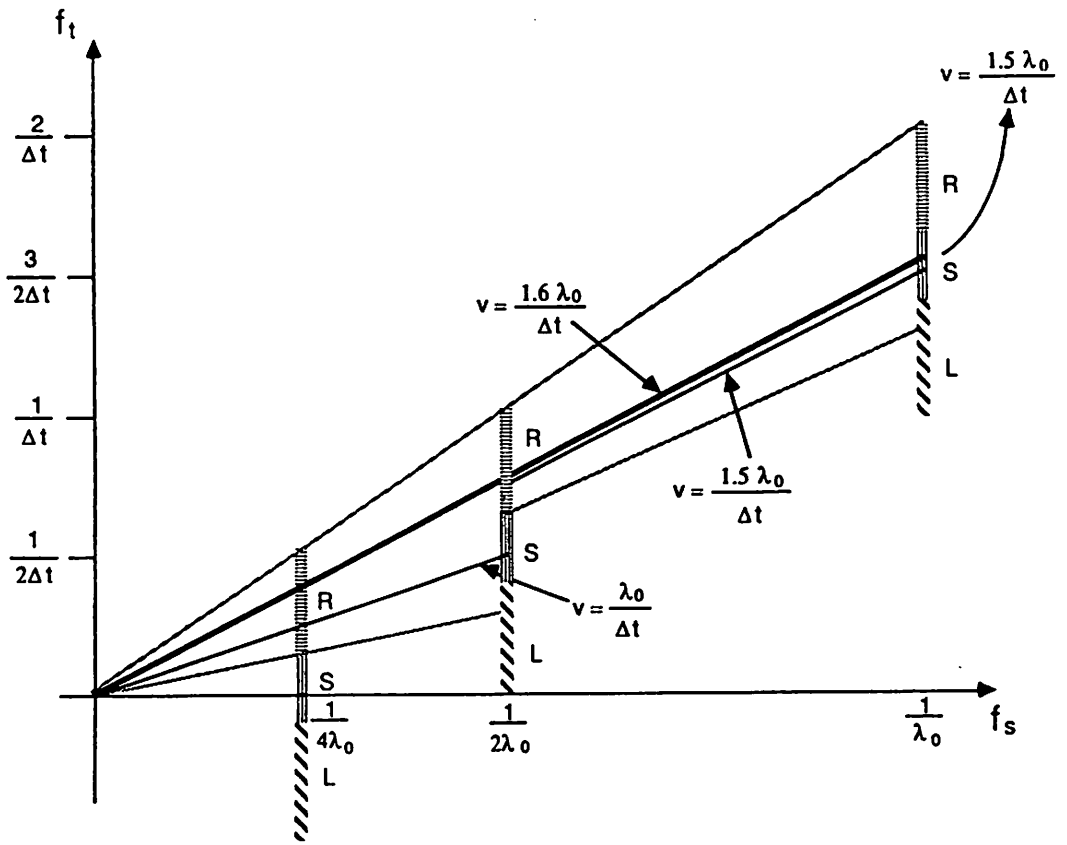



**Figure 89:** The coarse-to-fine strategy illustrated in the spatio-temporal frequency domain

## BIBLIOGRAPHY


[1] Adelson E. H. and Movshon J. A. The perception of coherent motion in two-dimensional patterns *ACM Workshop on Motion*, Toronto, Canada, pp. 11-16, 1983.

[2] Adelson E. H. and Bergen J. R. Spatiotemproal energy models for the perception of motion, *Journal of Optical Society of America A*, Vol. 2, No. 2, pp. 284-299, 1985.

[3] Adiv G, Determining 3-d motion and structure from optical flows generated by several moving objects, *IEEE T-PAMI*, 7 (4), pp. 384-401, 1985.

[4] Adiv G., Interpreting Optical Flow, *Ph. D. dissertation*, COINS Technical Report no. 85-35, 1985.

[5] Adiv G., Inherent ambiguities in recovering 3-d motion and structure from a noisy flow field, *Proc. CVPR*, pp. 70-77, 1985.

[6] Aggarwal J. K., Davis L. S., and Martin W. N., Correspondence processes in dynamic scene analysis, *Proc. IEEE*, Vol. 69, No. 5, pp. 562-572, 1981.

[7] Aggarwal J. K., Structure and motion from images, *DARPA IU Workshop Proc.*, pp. 89-95, 1985.

[8] Aggarwal J. K., Structure and motion from images: fact and fiction, *Proc. of the third Workshop on Computer Vision: Representation and Control*, pp. 127-128, 1985.

[9] Anandan P., Computing dense displacement fields with confidence measures in scenes containing occlusion, *SPIE Intelligent Robots and Computer Vision Conference*, Vol. 521, pp 184-194, 1984, also *COINS Technical Report 84-32*, University of Massachusetts, December 1984.

[10] Anandan P. and Weiss R., Introducing a smoothness constraint in a matching approach for the computation of displacement fields, *DARPA IU Workshop Proc.*, pp. 186-196, 1985.

[11] Baker H. H., Depth from edge and intensity based stereo, *Report no. STAN-CS-82-930*, Department of Computer Science, Stanford University, California, September 1982.

[12] Barnard S. T. and Thompson W. B., Disparity analysis of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-2, Number 4, July 1980, pp. 333-340.

[13] Barron J. A survey of approaches for determining optic flow, environmental layout, and egomotion, *University of Toronto Tech. Rep. No. RBCV-TR-84-5*, 1984.

[14] Batcher K. E., Bit serial parallel processing systems, *IEEE Tran. Comp.*, vol. C-31, no. 5, May 1982, pp. 377-384.

[15] Beaudet, P. Rotationally invariant image operators, *Proc. International Conference on Pattern Recognition*, 1978, pp. 579-583.

[16] Bharwani S., Riseman E. M., and Hanson A., Refinement of environmental depth maps over multiple frames, *Proc. DARPA IU Workshop*, 1985, pp. 413-420.

[17] Boldt M., Token based image abstraction, *COINS Tech. Report*, University of Massachusetts, *in preparation*.

[18] Bouknight, W. J., *et al.*, The Illiac IV system, *Proc. of the IEEE*, vol. 60, no. 4, April 1972, pp. 369-382.

[19] Burt P. J., Yen C. and Xu X., Local correlation measures for motion analysis: A comparative study, *IEEE Proc. PRIP*, 269-274, 1982.

[20] Burt P. J., Yen C. and Xu X., Multi-resolution flow-through motion analysis, *IEEE CVPR Conference Proceedings*, pp. 246-252, 1983.

[21] Burt P. J., Hong T. H and Rosenfeld A., Image segmentation and region property computation by cooperative hierarchical computation, *IEEE Trans. Systems, Man, Cybernetics* 11, 1981, pp. 802-809.

[22] Burt P. J., Fast filter transforms for image processing, *Computer graphics and image processing*, 16, pp. 20-51, 1981.

[23] Burt P. J. and Adelson E., The Laplacian pyramid as a compact image code, *IEEE Transactions on Communication*, vol. COM-31, pp. 532-540, 1983.

[24] Buxton B.F. and Buxton H., Monocular depth perception from optical flow by space time signal processing, *Proc. of the Royal Society, London* vol. B-218, pp. 27-47, 1983.

[25] Ciarlet P. G., *Numerical analysis of the finite element method*, Seminare de Mathematiques Supreieures, Univerisity of Montreal Press, 1976.

[26] do Carmo M. P., *Differential geometry of curves and surfaces*, Prentice-Hall, New Jersey, 1976.

[27] Cornelius N. and Kanade T., Adapting optical flow to measure object motion in reflectance and X-ray image sequences, *Proc. ACM Siggraph/Sigart Interdisciplinary workshop on motion*, Toronto, Canada, pp. 50-58, 1983.

[28] Crowley J. L. and Stern R. M., Fast computations for the difference of low-pass transform, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 2, pp. 156-169, 1984.

[29] Daugman J. D., Six formal properties of two-dimensional anisotropic visual filters, structural properties and frequency/orientation selectivity, *IEEE Trans. Systems, Man, and Cybernetics*, vol. SMC-13, no. 5, pp. 882-887, 1983.

[30] Daugman J. D., Spatial visual channels in the fourier plane, *Vision Research*, Vol. 24, No. 9, pp. 891-910, 1984.

[31] Enkelman W., Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences, *Workshop on motion: Representation and analysis*, S.C., pp. 81-87, 1986.

[32] Fang J. and Huang T. S., Some experiments on estimating the 3-d motion parameters of a rigid body from two consecutive Image Frames, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, NO. 5, pp 545-554, September, 1984.

[33] Fennema C. L. and Thompson W. B. Velocity determination in scenes containing several moving objects, *Computer Graphics and Image Processing*, 9, pp 301-315, 1979.

[34] Fleet D. J. and Jepson A. D., A cascaded filter approach to the construction of velocity selective mechanisms, *Technical report, University of Toronto, RBCV-TR-84-6*, December, 1984.

[35] Gabor D., Theory of communication, *Journal of IEE*, 93, pp. 429-457, 1946.

[36] Geman S. and Geman D., Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-6, no. 6, pp. 721-741, 1984.

[37] Gennery D., Modeling the environment of an exploring vehicle by means of stereo vision, *Ph. D. thesis, Stanford Artificial Intelligence Laboratory, AIM-339*, June 1980.

[38] Gibson J. J., *The perception of the visual world*, Cambridge, Mass, Riverside, 1950.

[39] Glazer F., Computing Optic Flow, *IJCAI-7*, Vancouver B. C., Canada, Aug. 1981, pp. 644-647.

[40] Glazer F., Reynolds G. and Anandan P., Scene matching by hierarchical correlation, *IEEE CVPR conference*, June 1983, pp. 432-441.

[41] Glazer F., Hierarchical motion detection, *Ph. D. dissertation*, COINS Department, University of Massachusetts, Amherst, Ma., January 1987.

[42] Grimson W. E. L., *From images to surfaces: A computational study of the human early visual system*, Cambridge, Mass, MIT Press, 1981.

[43] Grimson W. E. L., Computational experiments with a feature based stereo algorithm, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No. 1, pp. 17-34, 1985.

[44] Hannah M. J., Computer matching of areas in stereo images, *Stanford A.I. memo*, 239, 1974.

[45] Hanson A. R. and Riseman E. M., *Computer Vision Systems*, Academic Press, New York, 1978.

[46] Hanson A.. R and Riseman E. M., Processing cones: A computational struture for image analysis, in: *Structured computer vision*, Tanimoto S. and Klinger A. (Eds.), Academic Press, New York, 1980.

[47] Heeger D., Depth and flow from motion energy, *AAAI-86 Proc.*, pp. 657-663, 1986.

[48] Hildreth E. C., The measurement of visual motion, *Ph. D. dissertation*, Dept. of Electrical Engineering and Computer Science, MIT, Cambridge, Ma., 1983.

[49] Hildreth E. C. and Grzywacz N. M., The incremental recovery of structure from motion: position vs. velocity based formulations, *Proc. Workshop on motion: representation and analysis*, S.C., pp. 137-144, 1986.

[50] Hillis, W. D., The connection machine, *MIT AI Lab., Cambridge*, MA, Memo 646, 1981.

[51] Horn B. K. P., and Schunck B. G., Determining Optical Flow, *Artificial Intelligence*, vol. 17, pp. 185-203.

[52] Ibrahim, H. A. H., Kender, J. R., and Shaw, D. E., The Analysis and Performance of Two Middle-Level Vision Tasks on a Fine-Grained SIMD Tree Machine, *Proc. CVPR conference*, pp. 248-256, 1985.

[53] Jacobus, C. J., Chien, R. T. and Selander, J. M., Motion Detection and Analysis by Matching Graphs of Intermediate Level Primitives, *IEEE transactions on pattern analysis and machine intelligence*, Vol. PAMI-2, Number 6, Nov. 1980, pp. 495-510.

[54] Kahn P., Local determination of a moving contrast edge, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No. 4, pp. 402-409, 1985.

[55] Kass M., Computing visual correspondence, in *From pixels to predicates: Recent advances in computational and robotic vision*, Pentland A (ed.), Ablex, New Jersey, pp. 78-92.

[56] Kitchen L. and Rosenfeld A., Grey-level corner detection, Tech. report 887, Computer vision lab., Computer Science Center, University of Maryland, 1980.

[57] Klinger A. and Dyer R. D., Experiments on picture representation using regular decomposition, *Computer graphics and image processing*, 5(1), pp. 68-105, 1976.

[58] Koenderink, J. J. and van Doorn A., Local structure of movement parallax of the plane, *Journal of the Optical Soc. America*, 66, pp. 717-723, 1976.

[59] Kronauer R. E., and Zeevi Y. Z., Reorganization and Diversification of Signals in Vision, *IEEE Trans. on Systems, Man, and Cybernetics*, 15, 1985, pp. 91-101.

[60] Lawton D. T., Processing translational motion sequences, *Computer Graphics and Image Processing*, Vol. 22, pp. 116-144, 1982.

[61] Lawton D. T., Processing dynamic image sequences from a moving sensor, *Ph. D. dissertation*, COINS TR 84-05, University of Massachusetts, Amherst, Mass, 1984.

[62] Limb J. O. and Murphy J. A., Estimating velocity of moving images in television signals, *Computer Graphics and Image Processing*, Vol. 4, pp. 311-327, 1975.

[63] Longuet-Higgins H. C., and Prazdny K., The interpretation of a moving retinal image, *Proc. Roy. Soc. London, B.*, vol. 208, pp. 385-397, 1980.

[64] Lucas B. D., and Kanade T., An iterative image registration technique with an application to stereo vision, *Proc. 7th IJCAI*, Vancouver, B. C., Canada, pp. 674-679, 1981.

[65] Marr D. and Poggio T., A computational theory of human stereo vision, *Proc. Roy. Soc. London, Ser. B*, 204, pp 301-308, 1979.

[66] Marr D. and Hildreth E. C., Theory of edge detection, *Proc. Royal Society of London*, B207, pp. 187-217, 1980.

[67] Marr D. and Ullman S., Directional selectivity and its use in early visual processing, *Proc. Royal Soc. London B*, 211, pp. 151-180, 1981

[68] Marr D., *Vision*, Freeman Prss, 1982.

[69] Mayhew J. E. W. and Frisby J. P., Psychophysical and computational studies towards a theory of human stereopsis, *Artificial Intelligence*, Vol. 17, pp. 349-385, 1981.

[70] Huertas, A. and Medioni, G. Edge Detection with Subpixel Precision, *Proc. of the third workshop on computer vision*, pp. 63-71, 1985.

[71] Miller, R. and Q. F. Stout, Geometric algorithms for digitized pictures on a mesh-connected computer. *IEEE T-PAMI*, vol. PAMI-7, pp. 216-228, 1985.

[72] Moravec H. *Robot rover visual navigation*, UMI Research press, Ann Arbor, Michigan, 1981.

[73] Morraquin J., Mitter S., and Poggio T.,, Probabilistic solution for ill-posed problems in computational vision, *Proc. DARPA IU Workshop*, Miami Beach, Fl., pp. 293-309, 1986.

[74] Mutch K. and Thompson W., Analysis of accretion and deletion at boundaries in dynamic scenes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-7, no. 2, pp. 133-138, 1985.

[75] Nagel H. H., Displacement vectors derived from second order intensity variations in image sequences, *CVGIP*, vol. 21, pp. 85-117, 1983.

[76] Nagel H. H. On the estimation of dense displacement vector fields from image sequences, *ACM Motion workshop proc.*, Toronto, Canada, pp. 59-65, 1983.

[77] Nagel H. H., Constraints for the Estimation of Displacement Vector Fields from Image Sequences, *IJCAI-83*, Karlsruhe, W. Germany, pp 945-951, 1983.

[78] Nagel H. H. and Enkelmann W., An investigation of smoothness constraints for the estimation of displacement vector fields from image sequnces, *IEEE transactions on PAMI*, vol. PAMI-8, pp. 565-593, 1986.

[79] Nagel H. H., Image sequences - Ten (Octal) years – from phenomenology towards a theoretical foundation, *Proc. of Eighth ICPR*, Paris, France, 1986.

[80] Nishihara, H. K. PRISM: a practical real-time imaging stereo matcher. *MIT AI Lab.*, Cambridge, MA, Memo 780., 1984.

[81] Noble B. and Daniel J. W., *Applied Linear Algebra*, Prentice Hall, New Jersey, 1977.

[82] Ohta Y. and Kanade T., Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming, *IEEE PAMI*, vol. 7, no. 2, 1985, pp. 139-154.

[83] Pavlin I., Riseman E., and Hanson A., Analysis of an algorithm for detection of translational motion, *Proc. DARPA IU Workshop*, Miami Beach, Florida, pp. 388-398, 1985.

[84] Poggio T. and Torre V., Ill-posed problems and regularization analysis in early vision, *MIT AI Memo 773*, April, 1984.

[85] Prager J. M. and Arbib M. A. Computing the Optic Flow: The MATCH Algorithm and Prediction, *Computer Vision, Graphics, and Image Processing*, 24, pp 271-304, 1983.

[86] Price K., I have seen your demo; so what?, *Proc. of the Third Workshop on Computer Vision*, Bellaire, Michigan, pp. 122-124, 1985.

[87] Quam L. H., Hierarchical Warp Stereo, *Proceedings of DARPA IU Workshop*, Louisiana, October 1984, pp. 149-156.

[88] Radig B., Kraasch R., and Zack W., Matching symbolic descriptors for 3-d reconstruction of simple moving objects, *Proc. of IEEE ICPR*, Miami, Florida, pp. 1081-1084, 1980.

[89] Ranade S. and Rosenfeld A., Point pattern matching by relaxation, *Pattern Recognition*, Vol. 12, pp. 269-275, 1980.

[90] Rieger J. H. and Lawton D. T., Determining the instantaneous axis of translation from optic flow generated by arbitrary sensor motion, *Proc. ACM Siggraph/Sigart Interdisciplinary Workshop on Motion*, Toronto, pp. 33-41, 1983.

[91] Roach J. W. and Aggarwal J. K., Determining the movement of objects from a sequence of images, *IEEE transactions on Pattern Analysis and Machine Intelligence*, Vol. 2, No. 6, pp. 554-562, 1980.

[92] Rosenfeld A. and Kak A. C., *Digital picture processing*, Academic Press, New York, 1976.

[93] Sawheny H., The implementation of a coarse-to-fine cotrol strategy for motion on a mesh connected computer,

[94] Schunck B.G., Motion segmentation and estimation, *Ph. D. dissertation*, Massachusetts Institute of Technology, Department of EE & CS, 1983.

[95] Schunck B., Image flow: Fundamentals and future research, *Proc. IEEE CVPR conference*, San Francisco, Ca., pp. 560-573, 1985.

[96] Schunck B., Image flow continuity equations for motion and density, *Proc. Workshop on motion: representation and control*, S. C., pp. 89-94, 1986.

[97] Scott G. L. Smoothing the optic flow field under perspective projection, *Proc. IEEE CVPR*, Miami Beach, Florida, pp. 504-509, 1986.

[98] Shaw, D. E., The NON-VON supercomputer, *Dept. of Computer Science, Columbia University Technical Report*, 1982.

[99] Smith G., A fast surface interpolation technique, *Proc. DARPA IU workshop*, pp. 211-215, 1984.

[100] Strahman D. M., Structure in evidence and beliefs, *M. S. thesis*, COINS Department, University of Massachusetts, Amherst, Mass., *in preparation*.

[101] Tanimato S., A pyramidal approach to parallel processing, *Proc. 10th int. symp. Comp. Arch.* , Stockholm, 1983, pp. 372-378.

[102] Tanimato S. and Pavlidis T., A hierarchical data structure for picture processing, *Computer Graphics and Image Processing*, vol. 4, no. 2, 104-119, 1975.

[103] Tanimato S. and Uhr L., A pyramid data-structure for picture processing, *Computer Graphics and Image Processing*, Vol. 4, No. 2, pp. 104-119, 1975.

[104] Terzopoulos D., Mutiresolution Computation of Visible-Surface Representations, *Phd Dissertation*, Massachusetts Institute of Technology, Jan. 1984.

[105] Terzopolous D., Image analysis using multi-resolution methods, *IEEE transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 2, pp. 129-139, 1986.

[106] Tsai R. Y., and Huang T. S., Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects With Curved Surfaces, *IEEE T-PAMI*, 6, 1984.

[107] Tsuji S., Osada M. and Yachida M., Tracking and segmentation of moving objects in dynamic line images, *IEEE T-PAMI*, 2 (6), pp. 517-522, 1980.

[108] Uhr L., Layered "recognition cone" networks that preprocess, classify, and divide, *IEEE transactions on computers*, vol 21, no. 7, 758-768, 1972.

[109] Uhr L., "Recognition cones", and some test results: The imminent arrival of well-structured parallel-serial computers; positions, and positions on positions: in *Computer vision systems*, Hanson A. and Riseman E.M., (eds.), Academic Press, New York, 1978.

[110] Ullman S., Analysis of visual motion by biological and computer systems, *IEEE computer*, pp. 57-69, 1981.

[111] Ullman S. *The Interpretation of Visual Motion*, The MIT Press, Cambridge, Ma., 1979.

[112] Fountain T. J., The development of the CLIP7 image processing system, *Pattern Recognition Letters*, 1983, pp. 331-339.

[113] van der Wal G. S. and Sinniger J. O., "Real time pyramid transform architecture", *Proc. SPIE Intelligent Robots and Computer Vision Conference*, vol. 579, 1985.

[114] van Saanten J. P. H. and Sperling G., Elaborated Reichardt detectors, *Journal of Optical Society of America A*, Vol. 2, No. 2, pp. 300-321, 1985.

[115] Wallach, H., and O'Connell, D. N., The kinetic depth effect, *Journal of Experimental Psychology*, (45) 4, pp. 205-217.

[116] Waxman A. and Wohn K., Contour Evaluation, Neighbourhood Deformation and Global Image Flow: Planar Surfaces in Motion, *CS-TR-1394* University of Maryland, April 1984.

[117] Watson A B. and Ahmuda, Model of human visual-motion sensing, *Journal of Opt. Soc. America*, vol. 2, no. 2, pp. 322-342, 1985.

[118] Weems, C. C., Image processing on a content addressable array parallel processor., *Phd dissertation and COINS Technical Report 84-14*, Dept. of Comp. and Info. Science, University of Massachusetts, Amherst, Ma., 1984.

[119] Weiss R. and Boldt M., Geometric grouping applied to straight lines, *Proc. IEEE CVPR*, Miami Beach, Florida, pp. 489-493, 1986.

[120] Williams, L. R., Spectral Continuity and Eye Vergence Movement, *Proc. of the ninth IJCAI*, pp. 985-987, 1985.

[121] Williams L. R. and Anandan P., A coarse-to-fine control strategy for stereo and motion on a mesh-connected computer, *to appear in IEEE Computer Vision and Pattern Recognition conference*, Miami Beach, Florida, 1986.

[122] Wilson H. R., and Bergen J., A four mechanism model for threshold spatial vision, *Vision Research*, 19, pp. 19-33, 1979.

[123] Wilson H. R. Psychophysical evidence for spatial channels, in *Physical and biological processing of images*, Braddick O. J. and Sleigh A. C., (eds.), Springer-Verlag, Berlin, pp. 88-99, 1982.

[124] Wohn K., A contour-based approach to image flow, *Ph. D. dissertation*, Depr. of Computer Science, University of Maryland, Nov., 1984.

[125] Wong R. Y. and Hall E. L., Sequential hierarchical scene matching, *IEEE transactions on Computers*, Vol. 27, No. 4, pp. 359-366, 1978.