

INTERPRETING OPTICAL FLOW

Gilad Adiv

COINS Technical Report 85-35

September 1985

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

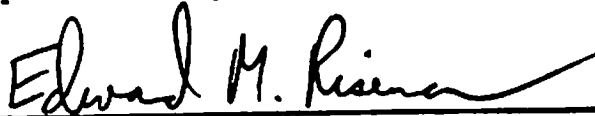
INTERPRETING OPTICAL FLOW

A Dissertation Presented

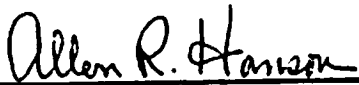
By

GILAD ADIV

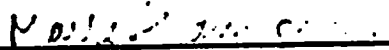
Approved as to style and content by:



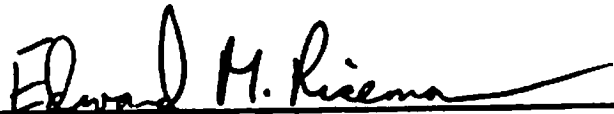
Edward M. Riseman, Chairperson of Committee



Allen R. Hanson, Member



Mark Snyder, Member



**Edward M. Riseman, Chairman
Department of Computer and Information Science**

ACKNOWLEDGEMENTS

I would like to thank my advisors, Ed Riseman and Al Hanson, for their continuous support. Their suggestions and comments significantly improved the quality of this dissertation. I would also like to thank Mark Snyder, the outside member of my committee, for his effort in reading this work.

I am indebted to many members of the computer vision group at the University of Massachusetts for useful discussions and for their generous help. I would especially like to thank P. Anandan, Daryl Lawton, Brian Burns, Terry Weymouth, Joey Griffith, and Lenny Wesley. The images used in some of the experiments were taken by Randy Ellis in the UMass robotics lab.

Finally, I am grateful to my wife, Aviva, for her support, encouragement, and patience.

ABSTRACT

Interpreting Optical Flow

September 1985

Gilad Adiv

B.S., M.S., The Hebrew University of Jerusalem, Israel

M.S., Ph.D., University of Massachusetts at Amherst

Directed by: Professor Edward M. Riseman

A new approach for the interpretation of optical flow fields is presented. The flow field, which can be produced by a sensor moving through an environment with several, independently moving, rigid objects, is allowed to be sparse, noisy and partially incorrect. The approach is based on two main stages. In the first stage the flow field is partitioned into connected segments of flow vectors, where each segment is consistent with a rigid motion of a roughly planar surface. Such a segment is assumed to correspond to a part of only one rigid object. This initial organization of the data is utilized in the second stage without the assumption of planar surfaces, and segments are now grouped under the hypothesis that they are induced by a single rigidly moving object and/or by the camera motion. Each hypothesis is tested by searching for 3-D motion parameters which are compatible with all the segments in the corresponding group. Once the motion parameters are recovered, the relative environmental depth can be estimated as well. Experiments based on real and simulated data are presented.

Two inherent ambiguities, which may arise due to the presence of noise in the flow field, are analyzed and demonstrated. First, motion parameters of the sensor or a moving object may be extremely difficult to estimate because there may exist

a large set of significantly incorrect solutions which induce flow fields similar to the correct one. Second, the decomposition of the flow field into sets corresponding to independently moving objects may be ambiguous because two such objects may induce optical flows which are compatible with the same motion parameters. These ambiguity analyses are general in the sense that they are algorithm-independent. Constraints and parameters which can be recovered even in ambiguous situations are presented.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iv
ABSTRACT	v
TABLE OF CONTENTS	vii
LIST OF ILLUSTRATIONS	ix
Chapter	
I. INTRODUCTION	1
1. Thesis Outline	2
II. LITERATURE REVIEW	4
III. THE MODEL AND THE TASK	
— A MATHEMATICAL FORMULATION	9
1. Basic Model and Equations	9
2. The Task — Inputs and Outputs	13
IV. SEGMENTATION	15
1. Ψ Transformations — A Segmentation Constraint	15
2. Adjacency and Connectivity in Flow Fields	18
3. Segmentation Algorithm	20
1. First stage — grouping into components	23
1. A modified version of the generalized Hough technique	23
2. Implementation of a multipass approach	27
2. Second stage — merging of components	31
1. Computing an optimal Ψ transformation	31
2. Criteria for a merging decision	32
3. The merging procedure	34
3. Third stage — assimilating ungrouped flow vectors	35
V. TESTING OBJECT HYPOTHESES AND	
RECOVERING 3-D MOTION AND STRUCTURE	36
1. Estimating Motion Parameters and Depth Information	
of a Rigid Object	36
1. Optimization constraint	36
2. Algorithm	40
1. Reducing the search space	41
2. Search strategy	47
3. Recovering environmental structure	51
3. Comments on inherent ambiguity and instability	51
2. Testing Object Hypotheses	53

VI. INHERENT AMBIGUITIES IN RECOVERING 3-D INFORMATION FROM A NOISY FLOW FIELD 58
1. Ambiguity in Determining Motion Parameters of a Rigid Object 58
1. Introduction 58
2. Mathematical analysis 62
1. A planar surface 62
2. The general case 71
3. Examples 72
4. Constraints and parameters which can be determined 81
2. Ambiguity in Grouping Flow Vectors into Sets Corresponding to Rigidly Moving Objects 83
1. Introductory discussion 83
2. Analysis of a specific case 84
3. An example 87
4. Concluding remarks on the rigidity assumption 88
VII. EXPERIMENTS 91
1. Experiment 1 91
2. Experiment 2 100
3. Experiment 3 111
4. Experiment 4 117
5. Experiment 5 124
VIII. SUMMARY 138
REFERENCES 143

LIST OF ILLUSTRATIONS

3.1	Coordinate Systems	10
4.1	Illustration of the Separation Concept	19
4.2	The Adjacency Relations in a Sparse Set of Points	21
5.1	A Geometrical Interpretation of the Algorithm	42
5.2	The Error Function and its Lower and Upper Bounds	46
5.3	A Sampling Pattern for $fov = 30^\circ$	52
6.1	Purely Translational and Purely Rotational Flow Fields	61
6.2	The Error Function in Example 1	74
6.3	The Error Function in Example 2	75
6.4	The Error Function in Example 3	76
6.5	The Error Function in Example 4	77
6.6	The Error Function in Example 5	78
6.7	The Error Function in Example 6	79
6.8	The Error Function in Example 7	80
6.9	The Optical Flows Induced by the Translation of Two Objects	87
6.10	Graphical Representation of the Optimal Solution	89
7.1	Experiment 1	93
7.2	Experiment 2	102
7.3	Experiment 3	112
7.4	Experiment 4	118
7.5	Experiment 5	127

CHAPTER I

INTRODUCTION

Dynamic visual information can be produced by a sensor moving through the environment and/or by independently moving objects in the visual field. The interpretation of such information consists of forming object hypotheses, recovering the motion parameters of the sensor and each moving object, and structure determination. The results of this interpretation can be used to control behaviour, as in robotics or navigation. They can also be integrated, as an additional knowledge source, into an image understanding system, such as the VISIONS system [HAN78].

The most common approach for the analysis of visual motion is based on two phases: computation of an optical flow field and interpretation of this field. In the present discussion, the term 'optical flow field' refers to both a 'velocity field', composed of vectors describing the instantaneous velocity of image elements, and a 'displacement field', composed of vectors representing the displacement of image elements from one frame to the next. In the latter case we will assume small values of motion parameters.

The second phase, i.e., the interpretation of the optical flow field, is the main concern of this thesis. The information in only one flow field, as opposed to a time sequence of such fields, is utilized. Hence, this work is relevant also to stereoscopic vision, when the relative position and orientation of each camera are not accurately known.

A major problem which has emerged in research on optical flow interpretation is sensitivity to noise. Flow fields generated by existing techniques are noisy and partially incorrect, especially near occlusion or motion boundaries (see the discussion in [ULL81]). Many of the algorithms for interpreting these fields fail under such conditions. Global approaches, which utilize all the available information, can be

expected to be relatively robust. Still, an inadequate choice of an optimisation criterion often limits the performance of these techniques. Furthermore, the presence of independently moving objects usually makes such global techniques impractical.

These two issues, the presence of noise and the presence of independently moving objects, are addressed in this thesis. A new two-stage scheme is proposed. In the first stage the flow field, which is allowed to be either dense or sparse, is partitioned into connected segments of flow vectors, where each segment is consistent with a rigid motion of a roughly planar surface and, therefore, is likely to be associated with only one rigid object. In the second stage segments are grouped under the hypothesis that they are induced by a single rigidly moving object. Each hypothesis is tested by searching for 3-D motion parameters which are compatible with all the segments in the corresponding group. Once the motion parameters are recovered, the relative environmental depth can be estimated as well.

This technique, of segmenting the flow field and then combining segments to form object hypotheses, makes it possible to deal with independently moving objects while employing all the available information associated with each object. In addition, the search for 3-D motion parameters is based on a least-squares technique which minimizes the deviation between the given flow field and that predicted from the computed parameters. Thus, the proposed scheme is relatively insensitive to noise. There are, however, inherent ambiguities in the interpretation of noisy flow fields. These ambiguities are analysed and demonstrated in the thesis.

I.1 Thesis Outline

Chapters II and III complement the introductory part of the thesis. In Chapter II techniques existing in the literature for visual motion interpretation are reviewed. The ability of these techniques to deal with noisy flow fields and independently moving objects is examined. In the third chapter the mathematical relation between

the flow field and 3-D motion and structure is presented. In addition, the inputs and outputs of the interpretation process are mathematically defined.

A segmentation constraint, based on compatibility of sets of flow vectors with rigid motions of planar patches, is introduced in Chapter IV. The algorithm for achieving such a segmentation is based on a modification of the generalised Hough technique. This technique is employed to group flow vectors into components consistent with affine transformations of the image. If appropriate, components are then merged together to create segments.

An optimization criterion for determining 3-D motion and structure from a set of flow vectors generated by a rigid motion is presented in Chapter V. An algorithm for minimizing an error function derived from this criterion is also described. In the second part of Chapter V, this algorithm is combined with the segmentation results to form and test object hypotheses and to recover the 3-D information associated with each object.

Two inherent ambiguities, which may arise in the presence of noise in the flow field, are analyzed and demonstrated in Chapter VI. The first ambiguity is in the recovery of the motion parameters of a rigid object, and the second one is in the decomposition of the flow field into sets corresponding to independently moving objects. Constraints and parameters which can be recovered even in ambiguous situations are presented.

Experiments, based on real and simulated data, are described in Chapter VII. These experiments demonstrate our proposed scheme for the interpretation of optical flow fields, as well as the difficulties in ambiguous situations. Chapter VIII summarizes our approach and the major results. In addition, possible extensions and future research are discussed.

CHAPTER II

LITERATURE REVIEW

In this chapter we examine existing methods for interpreting optical flow fields. Most of these methods basically rely on the information contained in one flow field and are restricted to rigid motion. In addition, it is usually assumed that the scene contains only one object, or, equivalently, that the sensor is moving but the environment is stationary (e.g., [LAW82, BRU81, TSA84]). In this case the interpretation consists of recovering the motion parameters and the structure of the environment up to a scaling factor. Only a few researchers (e.g., [ULL79, NEU80]) explicitly consider scenes with several rigidly moving objects, and investigate techniques for decomposing the flow field into sets corresponding to these objects, while simultaneously recovering the 3-D information associated with each of them.

Many of the algorithms described in the literature for interpreting flow fields can not successfully deal with a realistic level of noise. A few algorithms are less sensitive and may work reasonably well in restricted real world situations. This issue, as well as the adequacy of the various techniques in the presence of independently moving objects, will be emphasised in the review. Let us now start with a discussion of techniques which assume only one rigid motion.

Several researchers [THO59, PRA80, NAG81a,b, FAN83a,b] present sets of nonlinear equations with motion parameters as unknowns. Methods for solving such equations are usually iterative and require good initial guesses of the unknowns. Sensitivity to noise is indicated by experiments reported in [PRA80, FAN83a,b]. It is shown there that the results can be improved by using a large number of flow vectors and by increasing the size of the region containing these vectors. Therefore, assuming no independently moving objects, the best results can be achieved when these algorithms are applied to the whole image.

Longuet-Higgins [LON81] and Tsai and Huang [TSA84] develop computationally simple techniques based on solving a set of linear equations. Furthermore, conditions for the uniqueness of the solutions are formulated. However, difficulties in the presence of noise are reported again [TSA84].

The sensitivity to noise, indicated by experiments in [PRA80, FAN83a,b, TSA84], may partly be due to inadequate choices of an optimization criterion. A more appropriate approach, in the presence of noise, is probably taken by Roach and Aggarwal [ROA80] and Bruss and Horn [BRU81]. A least squares criterion is employed in order to minimize the deviation between the measured data and the corresponding values predicted from the computed 3-D motion and structure. This approach leads to a system of nonlinear equations from which the motion parameters and the depth values can be numerically computed. Difficulties with noisy data, which are still reported in [ROA80], may indicate that in certain situations recovering 3-D information from flow fields is inherently unstable.

As a response to this instability, Thompson, Mutch and Berzins [THO84] argue for a more qualitative approach. In restricted but noisy situations, this approach is shown, for example, to be able to classify the object motion into four classes and to determine boundaries between independently moving objects. The results are unsatisfactory, though, when more quantitative information is required. In addition, the performance of this scheme has not been demonstrated yet in scenes containing complex structures.

Assuming a purely translational motion, all the flow vectors are oriented towards or from a single point in the image plane. Determining this point, called the focus of expansion (FOE), yields the direction of the translation. A few techniques, reviewed below, are based on this observation.

Early results based on real images are reported in [WIL81]. However, only sensor motion restricted to translation is allowed and the environment is assumed

to contain only planar surfaces at one of two given orientations. Thus, the algorithm can be based on a search for the FOE and the distances to the surfaces in the scene. Lawton [LAW82] describes a robust algorithm which has been applied to real world images from several different task domains. This algorithm makes no assumptions about the shape of environmental objects, but is still restricted to translational motion. It is based on a global sampling of an error measure corresponding to possible positions of the FOE, followed by a local search to determine the exact location of the minimum value. Results for other restricted cases of motion are presented in [LAW84].

Prazdny [PRA81] describes a method which relies on decomposition of the velocity field into rotational and translational components. For a hypothesized rotational component, the FOE of the corresponding translational field and a related error measure are computed. Thus, an error function of the three rotation parameters is obtained and the solution can be determined by minimizing this function. Jerian and Jain [JER83] report on difficulties with applying a similar approach to noisy data.

Rieger and Lawton [RIE83] develop a relatively robust and simple technique for computing the motion parameters of a camera moving in a stationary environment with significant depth discontinuities. The algorithm is based on the observation that the differences between vectors at the corresponding discontinuities in the flow field are oriented towards or from the FOE of the translational field [LON80]. This technique depends on the ability to compute reasonably accurate values of the flow vectors near occlusion boundaries.

A number of methods presented in the literature are based on a local analysis of the flow field. Ullman [ULL79] allows the presence of independently moving objects and examines small sets of adjacent vectors. If there exists a unique rigid interpretation consistent with all the vectors in a given set, then this interpretation

is assumed to be correct and the vectors in the set are grouped together. Because of its local nature, this approach seems to be very sensitive to errors in the flow field.

Longuet-Higgins and Prasadny [LON80] and Waxman et al. [WAX83, WAX84a,b] introduce equations for computing the motion parameters and the local structure at a given point in the environment from the flow field and its first and second spatial derivatives at the corresponding point in the image. If the scene consists of several objects in relative motion, then a separate computation can be carried out on each one. Local estimates of the second derivatives of the optic flow seem to be inaccurate in the presence of noise, and no algorithm has been presented for reliably computing such derivatives. Good results when determining 3-D motion and structure of planar surfaces are reported, however, in [WAX84b].

Global approaches, which still take into account independently moving objects, have been proposed by a few researchers. Neumann [NEU80] proposes an elegant hypothesize-and-test scheme: for any rotation hypothesis, the translation component may be decomposed such that motion compatibility of many flow vectors can be easily tested. This technique heavily relies on the assumption of orthographic projection.

The generalized Hough technique [BAL81a] is another global approach for recovering motion parameters from a given flow field [BAL81b, ADI83a,b]. In this technique the motion parameters are represented by a discrete multi-dimensional parameter space, where each dimension corresponds to one of the parameters. Each point in this space uniquely characterizes a motion transformation defined by the corresponding parameter values. A flow vector 'votes' for a point in the space if the related transformation is consistent with this vector. The points receiving the most votes are likely to represent the motion parameters of different objects.

The Hough technique is relatively insensitive to noise and partially incorrect

data. It can also be applied, using a multipass approach [FEN79, ADI83a,b], to scenes containing independently moving objects. However, this technique may be very expensive, since high dimensionality and fine resolution in the parameter space require large amounts of memory and computation time. Therefore, in [BAL81b] the depth information is assumed to be known, thus making the task much easier, and in [ADI83a,b] the approach is demonstrated only for 2-D motion with 4 parameters.

This review demonstrates typical restrictions and difficulties of algorithms reported in the literature for interpretation of optical flow fields. Most of these techniques are sensitive to noise and are difficult to apply in the presence of independently moving objects, unless severe restrictions are assumed. As will be shown in Chapter VI, in some situations there exists inherent ambiguity in recovering 3-D information from noisy flow fields. Yet, in order to improve the performance as much as possible, it is necessary to efficiently utilize all the available information and to employ an adequate optimization criterion. These obvious conclusions constitute the basis of the approach developed in this thesis.

CHAPTER III

THE MODEL AND THE TASK — A MATHEMATICAL FORMULATION

III.1 Basic Model and Equations

In this section we present a notation for describing the motion of a camera through an environment containing independently moving objects. We also review the equations describing the relation between the 3-D motion and structure and the corresponding optical flow, assuming a perspective projection. The equations are developed both for velocity fields and displacement fields.

Let (X, Y, Z) represent a cartesian coordinate system which is fixed with respect to the camera (see Figure 3.1), and let (x, y) represent a corresponding coordinate system of a planar image. The focal length, from the nodal point O to the image, is assumed to be known. It can be normalized to 1, without loss of generality. Thus, the perspective projection (x, y) on the image of a point (X, Y, Z) in the environment is:

$$x = X/Z, \quad y = Y/Z. \quad (3.1a,b)$$

The motion of a rigid object in the scene, relative to the camera, can be decomposed into two components: translation $\mathcal{T} = (T_X, T_Y, T_Z)$ and rotation $\Omega = (\Omega_X, \Omega_Y, \Omega_Z)$. In the equations corresponding to velocity fields, these symbols represent instantaneous spatial velocities, and, in the equations corresponding to displacement fields, they represent differences in position and orientation between two time instances.

In the velocity-based scheme, let (X, Y, Z) be the instantaneous camera coordinates of a point P on the object. Then, the velocity of P is

$$\dot{P} = \Omega \times P + \mathcal{T}, \quad (3.2)$$

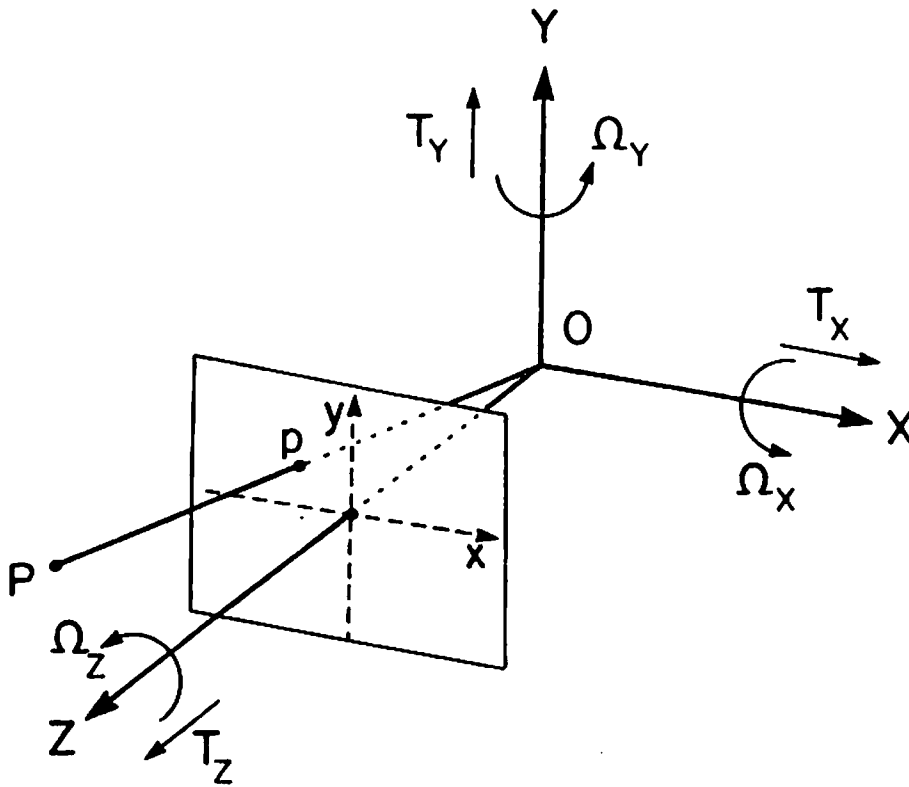


Figure 3.1 (redrawn from [LON80]): A coordinate system (X, Y, Z) attached to the camera, and the corresponding image coordinates (x, y) . The image position p is the perspective projection of the point P in the environment. $\mathbf{T} = (T_X, T_Y, T_Z)$ and $\mathbf{\Omega} = (\Omega_X, \Omega_Y, \Omega_Z)$ represent the relative translation and rotation of a given object in the scene.

that is,

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{pmatrix} = \begin{pmatrix} \Omega_Y Z - \Omega_Z Y + T_X \\ \Omega_Z X - \Omega_X Z + T_Y \\ \Omega_X Y - \Omega_Y X + T_Z \end{pmatrix}. \quad (3.3)$$

The corresponding projection (x, y) on the image moves with a velocity (α, β) ,

where, using (3.1):

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} \stackrel{\text{def}}{=} \begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \frac{1}{Z^2} \begin{pmatrix} \dot{X}Z - \dot{Z}X \\ \dot{Y}Z - \dot{Z}Y \end{pmatrix}. \quad (3.4)$$

Hence,

$$\alpha = -\Omega_X xy + \Omega_Y(1 + x^2) - \Omega_Z y + (T_X - T_Z x)/Z \quad (3.5a)$$

and

$$\beta = -\Omega_X(1 + y^2) + \Omega_Y xy + \Omega_Z x + (T_Y - T_Z y)/Z, \quad (3.5b)$$

as has already been established elsewhere (e.g., [LON80]). Notice that (α, β) can be represented as the sum

$$(\alpha, \beta) = (\alpha_R, \beta_R) + (\alpha_T, \beta_T), \quad (3.6)$$

where (α_R, β_R) and (α_T, β_T) are, respectively, the rotational and translational components of the velocity field:

$$\alpha_R = -\Omega_X xy + \Omega_Y(1 + x^2) - \Omega_Z y, \quad \alpha_T = (T_X - T_Z x)/Z, \quad (3.7a,b)$$

$$\beta_R = -\Omega_X(1 + y^2) + \Omega_Y xy + \Omega_Z x, \quad \beta_T = (T_Y - T_Z y)/Z. \quad (3.7c,d)$$

In the displacement-based scheme, let (X, Y, Z) be the camera coordinates at time t of a point on the object and let (X', Y', Z') be the corresponding coordinates at time t' . If the rotation Ω is followed by the translation \mathcal{T} , then

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \mathcal{T}, \quad (3.8)$$

where the rotation matrix R can be approximated, assuming small values of the

rotation parameters, by (see [FAN83a]):

$$R = \begin{pmatrix} 1 & -\Omega_Z & \Omega_Y \\ \Omega_Z & 1 & -\Omega_X \\ -\Omega_Y & \Omega_X & 1 \end{pmatrix}. \quad (3.9)$$

If (x, y) and (x', y') are the image coordinates corresponding to the points (X, Y, Z) and (X', Y', Z') , respectively, then:

$$x' = \frac{X'}{Z'} = \frac{x - \Omega_Z y + \Omega_Y + T_X/Z}{-\Omega_Y x + \Omega_X y + 1 + T_Z/Z} \quad (3.10a)$$

and

$$y' = \frac{Y'}{Z'} = \frac{\Omega_Z x + y - \Omega_X + T_Y/Z}{-\Omega_Y x + \Omega_X y + 1 + T_Z/Z}. \quad (3.10b)$$

Now, let (α, β) be, in this case, the displacement vector $(x' - x, y' - y)$. Then from (3.10) we get:

$$\alpha = \frac{-\Omega_X x y + \Omega_Y (1 + x^2) - \Omega_Z y + (T_X - T_Z x)/Z}{1 + \Omega_X y - \Omega_Y x + T_Z/Z} \quad (3.11a)$$

and

$$\beta = \frac{-\Omega_X (1 + y^2) + \Omega_Y x y + \Omega_Z x + (T_Y - T_Z y)/Z}{1 + \Omega_X y - \Omega_Y x + T_Z/Z}. \quad (3.11b)$$

If $|T_Z/Z| \ll 1$ and the field of view of the camera, i.e., the visual angle corresponding to the whole image, is not very large, then (employing also the assumption that the rotation parameters are small) we can approximate the displacement vector (α, β) by equations (3.5).

To conclude: equations (3.5) hold not only for velocity fields, but also for displacement fields, given that the following conditions are satisfied: (a) the rotation parameters are small, (b) the Z -component of the translation is small relative to the distance of the object from the camera, and (c) the field of view is not very large. The conditions (a) and (b) are reasonable if the time interval between the

two image frames is short enough or if the motion is slow. For example, if a video camera is moving along the line of sight at speed of 50km/hr and the frame sampling rate is 30/sec, then $T_z \approx 0.5\text{m}$ and $T_z/Z < 0.01$ for objects which are more than 50m away from the image plane. In the following chapters we restrict ourselves to conditions which allow us to employ equations (3.5) as the basis of our analysis.

III.2 The Task — Inputs and Outputs

The input utilized by our scheme for interpreting motion information is a flow field described by $\{(\alpha(x, y), \beta(x, y), W(x, y))\}$, where $(\alpha(x, y), \beta(x, y))$ is the flow vector at the (x, y) pixel in the image and $W(x, y)$ is a corresponding weight between 0 and 1. High reliability of the flow vector is represented by a weight close to 1 and low reliability by a weight close to 0. The flow field can be either dense, thus defined at most of the pixels, or sparse, thus defined only on a sparse subset of the image pixels. If the flow field is undefined at a pixel (x, y) , then $W(x, y)$ is determined to be 0. A rough estimate of the noise level in the flow field is assumed to be known.

The interpretation process should result in three outputs: object masks, motion parameters and depth. We wish to partition the set $\{(x, y) : W(x, y) > 0\}$ into disjoint sets of pixels, where each set corresponds to a different rigid object. The pixels corresponding to the stationary environment, where the optical flow is induced only by the camera motion, should be grouped together.

The five recoverable motion parameters of each rigid object, relative to the camera, should be estimated. These parameters include the rotation parameters $(\Omega_X, \Omega_Y, \Omega_Z)$ and the direction of the translation vector defined by the unit vector $\underline{U} = \underline{T}/r$, where r is the length of the translation vector \underline{T} . Once the motion parameters are recovered, it is also possible to estimate the relative depth, $Z(x, y)/r$, corresponding to each pixel (x, y) where a flow vector is defined, unless $r = 0$ or

the location of the vector is exactly at the FOE.

CHAPTER IV

SEGMENTATION

In this chapter we develop a method for segmenting the flow field into connected sets of flow vectors, where each set is consistent with a rigid motion of a roughly planar patch. A segment satisfying this constraint is very likely to be associated with only one rigid object. Thus, the data is organized into coherent units which constitute the basis for the formation of object hypotheses in the second phase. This organization makes it possible to deal with independently moving objects, while employing all the information associated with each object and preventing the suppression of valuable data in distinct, but possibly small, surfaces. Another purpose of the segmentation is exclusion of incorrect flow vectors which are inconsistent with their neighbors.

IV.1 Ψ Transformations — A Segmentation Constraint

In order to achieve a useful segmentation, we employ a few simple observations about the structure of optical flow fields. First, we examine the flow field induced by a rigid motion of a planar surface. Excluding the degenerate case in which the same plane contains both the surface and the nodal point (and, therefore, the corresponding region in the image is a straight line), the surface can be represented by the equation

$$k_X X + k_Y Y + k_Z Z = 1. \quad (4.1)$$

The coefficients k_X , k_Y and k_Z can be any real numbers, except the case in which all of them are zero. Using (3.1), we obtain:

$$1/Z = k_X x + k_Y y + k_Z. \quad (4.2)$$

Substituting (4.2) in (3.5), we realize that, given a relative motion $\{\mathcal{T}, \Omega\}$, the flow

field is:

$$\alpha = a_1 + a_2x + a_3y + a_7x^2 + a_8xy, \quad (4.3a)$$

$$\beta = a_4 + a_5x + a_6y + a_7xy + a_8y^2, \quad (4.3b)$$

where:

$$a_1 = \Omega_Y + k_Z T_X, \quad (4.4a)$$

$$a_2 = k_X T_X - k_Z T_Z, \quad (4.4b)$$

$$a_3 = -\Omega_Z + k_Y T_X, \quad (4.4c)$$

$$a_4 = -\Omega_X + k_Z T_Y, \quad (4.4d)$$

$$a_5 = \Omega_Z + k_X T_Y, \quad (4.4e)$$

$$a_6 = k_Y T_Y - k_Z T_Z, \quad (4.4f)$$

$$a_7 = \Omega_Y - k_X T_Z \quad (4.4g)$$

and

$$a_8 = -\Omega_X - k_Y T_Z. \quad (4.4h)$$

Equations (4.3) represent what we shall call a Ψ transformation. They describe a 2-D motion in the image plane, represented by the 8 parameters a_1, \dots, a_8 . Note that a similar representation of the optical flow produced by a moving planar surface is introduced in [WAX83].

We proceed now with another observation, related to arbitrary surfaces in the environment. Given such a surface, it can be described as a function $Z = Z(x, y)$ defined on the image region \mathcal{R} which corresponds to the projection of this surface. Let $Z' = Z'(x, y)$ be a *relative approximation* to the surface Z , that is,

$$|\Delta Z(x, y)| \stackrel{\text{def}}{=} |Z(x, y) - Z'(x, y)| \ll Z(x, y) \quad \text{for any } (x, y) \in \mathcal{R}. \quad (4.5)$$

If (α_T, β_T) and (α'_T, β'_T) are the translational components of the flow fields induced

by the same motion of the surfaces Z and Z' , respectively, then

$$\begin{aligned}\alpha'_T &= \frac{T_X - T_{Zx}}{Z'} = \frac{T_X - T_{Zx}}{Z - \Delta Z} \\ &\approx \frac{T_X - T_{Zx}}{Z} \left(1 + \frac{\Delta Z}{Z}\right) = \alpha_T \left(1 + \frac{\Delta Z}{Z}\right) \approx \alpha_T\end{aligned}\quad (4.6a)$$

and

$$\begin{aligned}\beta'_T &= \frac{T_Y - T_{Zy}}{Z'} = \frac{T_Y - T_{Zy}}{Z - \Delta Z} \\ &\approx \frac{T_Y - T_{Zy}}{Z} \left(1 + \frac{\Delta Z}{Z}\right) = \beta_T \left(1 + \frac{\Delta Z}{Z}\right) \approx \beta_T.\end{aligned}\quad (4.6b)$$

The rotational component of the flow field is independent of the structure of the environment. Hence, given (4.5), the flow field induced by the relatively approximating surface Z' is very similar to the real flow in the region \mathcal{R} . As a conclusion, if Z' is a planar surface which satisfies equation (4.5), then the flow field in \mathcal{R} can be approximated by a Ψ transformation.

In a real world environment the surface can be usually approximated by a piecewise planar surface, containing only a few planar patches, for which the distance between the real surface and the approximating one is small relative to the distance from the sensor to the surface. If this is the case, then the flow field can be approximated, reasonably well, by a piecewise Ψ transformation. This suggests that a *useful segmentation of the flow field can be based on finding connected sets of flow vectors, where the vectors in each set approximately satisfy the same Ψ transformation*. Thus, each segment is consistent with a rigid motion of a roughly planar surface and can be assumed to be induced by the relative motion of only one rigid object.

Notice that different segmentation constraints could have been employed. For example, it is possible to use a stronger constraint, based on consistency with affine

transformations (see equations (4.7)). However, in such a case, an excessive over-segmentation is possible, as demonstrated by the results of the first step of the segmentation algorithm in experiment 1 (Section VII.1). This would make the formation of object hypotheses in the next phase of the interpretation much more difficult.

Another option is to use a weaker constraint based, for example, on a transformation which includes all the 12 coefficients of the second-order Taylor expansion of the flow field (a representation of the optical flow, based on these coefficients, is proposed in [WAX83,84a]). In this case, however, the space of all the possible transformations strictly contains the space of all the Ψ transformations; therefore, the probability of grouping together flow vectors induced by different rigid motions is increased. In Section IV.3 we will describe an algorithm for achieving a segmentation based on the Ψ transformation.

IV.2 Adjacency and Connectivity in Flow Fields

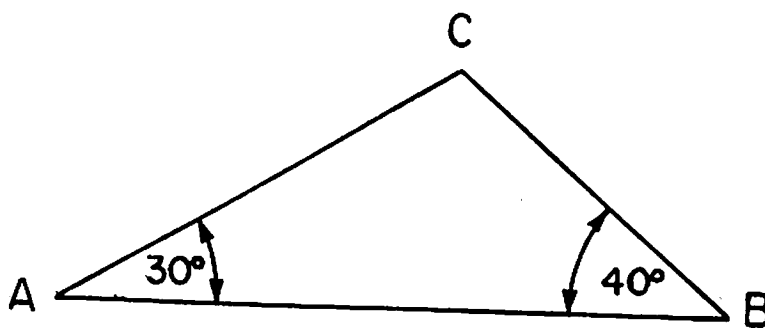
The concept of connectivity in optical flow fields, which is employed in the previous section for characterizing segments, is, as a matter of fact, solely a property of the definition domain $\{(x, y) : W(x, y) > 0\}$ of the flow field, and it does not depend on the flow values. To define this concept, let us start by defining *adjacency* between two points in the image. If the flow field is dense, then the adjacency concept is simple: any pair of points (x_1, y_1) and (x_2, y_2) , such that the differences $|x_1 - x_2|$ and $|y_1 - y_2|$ are both not larger than one pixel, are considered to be adjacent.

The definition of adjacency in *sparse* flow fields is more complicated. In this case we first introduce another concept:

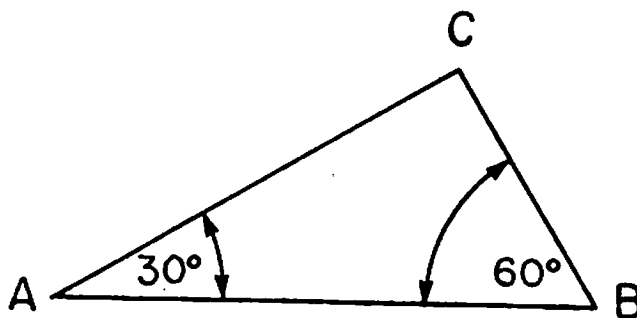
Definition 4.1: Consider the triangle created by three distinct points, denoted by

A , B and C . The point C separates A and B if the triangle angles at A and B are both smaller than a given threshold.

In our experiments we have chosen the threshold in the last definition to be 45° . Thus, in Figure 4.1a the points A and B are separated by C , while in Figure 4.1b C does not separate A and B . We can now proceed with the following definition of adjacency:



(a)



(b)

Figure 4.1: (a) The point C separates the points A and B .
 (b) The point C does not separate A and B .

Definition 4.2: Let S be a set of points in a plane. The points A and B in this set are *adjacent* with respect to S if no point in the set separates them.

The adjacency relation in sparse sets of points is demonstrated in Figure 4.2, where adjacent points are connected by straight line segments. The points in this figure correspond to the flow vectors in experiment 3 (Section VII.3).

In this chapter we will also employ the concept of adjacent sets:

Definition 4.3: Let S_1 and S_2 be two subsets of points in S . S_1 and S_2 are *adjacent sets* if there exist points $A \in S_1$ and $B \in S_2$ such that A and B are adjacent with respect to S .

Employing the concept of adjacency between points, we can also define connected subsets of the definition domain, denoted by D , of a flow field:

Definition 4.4: A subset D' of D is *connected* if for each pair of points A and B in this subset, there exists a sequence $\{A_1, \dots, A_n\}$ of points in D' such that $A_1 = A$, $A_n = B$ and the points A_i and A_{i+1} are adjacent with respect to D for each $1 \leq i \leq n - 1$.

Note that if D' is a connected subset of the definition domain of a given flow field, then we also refer to the corresponding subset of flow vectors as being connected. In the next section we will describe an algorithm for grouping the flow vectors into connected sets consistent with rigid motions of planar surfaces.

IV.3 Segmentation Algorithm

The generalized Hough transform technique [BAL81a] is a useful tool for grouping together flow vectors which satisfy the same 2-D parameterized transformation [ADI83a,b]. In this technique the set of relevant transformations is represented by a discrete multi-dimensional parameter space, where each dimension corresponds to one of the transformation parameters. Each point in this space uniquely characterizes a transformation, defined by the corresponding parameter values. A flow

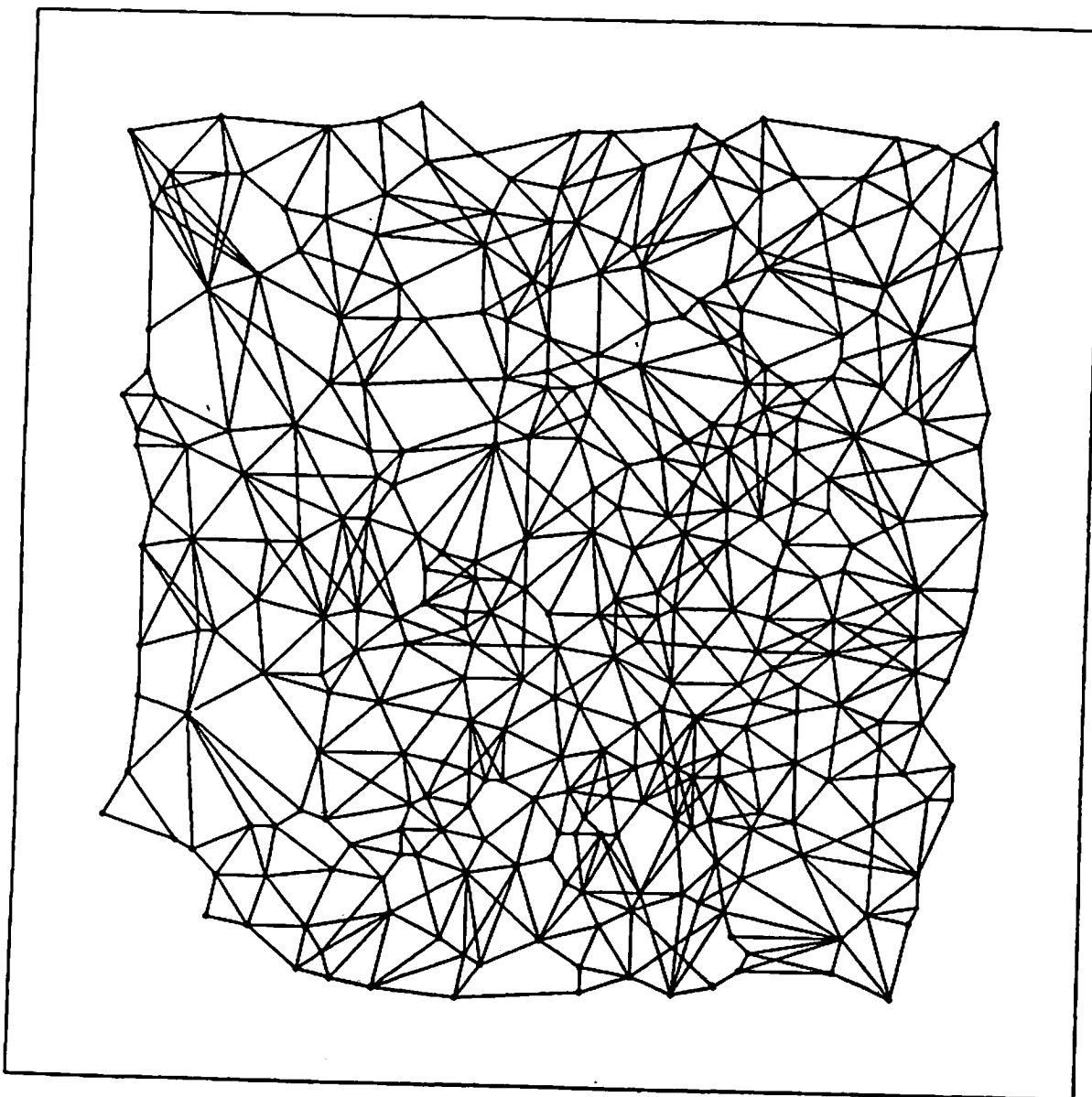


Figure 4.2: The adjacency relations in a sparse set of points.

vector 'votes' for each point with an associated transformation consistent with this vector. The points receiving the most votes are likely to represent transformations corresponding to large segments in the flow field.

As a global technique, the Hough transform is relatively insensitive to noise and partially incorrect or occluded data. However, high dimensionality of the parameter space requires large amounts of memory and computation time. In our case, the segmentation constraint is based on the 8-parameter Ψ transformations (equations (4.3)). The Hough technique can, in principle, be employed, but the computational cost required for such a number of parameters is very high. Therefore, a three-stage algorithm is proposed.

The first stage is based on grouping together adjacent flow vectors into *components* consistent with *affine transformations*. The affine transformations, represented by

$$\alpha = a_1 + a_2x + a_3y \quad (4.7a)$$

and

$$\beta = a_4 + a_5x + a_6y, \quad (4.7b)$$

are a sub-class of the Ψ transformations, parameterized by only 6 parameters. Furthermore, these parameters can be partitioned into two disjoint sets of 3 parameters each, corresponding to equations (4.7a) and (4.7b). Thus, the grouping problem in the first stage can be basically solved by applying the Hough technique to 3-dimensional parameter spaces, as will be shown in Section IV.3.1.

In the second stage, components which are consistent with the same Ψ transformation are merged into *segments*. Given a set of adjacent components, optimal parameters are computed, using the least-squares technique. Related error measures, associated with each component in the set, can be thus obtained. If these error values are not high (in a sense defined in Section IV.3.2), then the components are merged. Note that this hypothesize-and-test technique could not be employed

without the initial grouping into components, which drastically reduces the number of hypotheses to be tested.

Sometimes an over-fragmentation may occur in the first stage of the segmentation, that is, a segment is partitioned into a large number of small components. This is demonstrated in experiment 1 in Chapter VII (see Figure 7.1c), where the flow field in one of the segments contains large second-order terms. In order to reduce the computational cost of the first and second segmentation stages, the grouping of vectors belonging to small connected sets may be postponed, in such a case, to the third stage. In this stage, flow vectors which are not contained in any of the segments are merged into neighboring segments, if they are consistent with the corresponding Ψ transformations. If, after the third stage, some of these small sets are still not merged into the existing segments, then the first and second stages of the segmentation may be repeated, focused only on these sets, thus possibly creating new segments. In the following sections the three stages of the segmentation are fully described.

IV.3.1 First stage — grouping into components

The grouping of flow vectors into components consistent with affine transformations is based on a multipass Hough approach [FEN79, ADI83a,b] where in each iteration a modification of the generalized Hough technique is employed. This modified Hough technique will be presented in the next section, while the details of the multipass scheme will be described in Section IV.3.1.2.

IV.3.1.1 A modified version of the generalized Hough technique.

The affine transformations can be represented by a 6-dimensional parameter space where each dimension corresponds to one of the parameters a_1, \dots, a_6 in equations (4.7). For computational reasons the parameter space must contain only a finite number of points. Therefore, minimal and maximal values are determined for each

parameter such that the values associated with the actual transformations can be reasonably assumed to be between these values, and the corresponding interval is quantized. The parameter space is the cartesian product of the discrete sets so obtained.

A flow vector $(\alpha(x, y), \beta(x, y))$ votes for a transformation (a_1, \dots, a_6) if it approximately satisfies the constraint equations (4.7), that is, if

$$\delta \stackrel{\text{def}}{=} \sqrt{\delta_x^2 + \delta_y^2} \leq \epsilon, \quad (4.8a)$$

where

$$\delta_x = |\alpha - a_1 - a_2x - a_3y| \quad (4.8b)$$

and

$$\delta_y = |\beta - a_4 - a_5x - a_6y|. \quad (4.8c)$$

The threshold ϵ is taken to be the maximal value of the resolution p in the parameter space (see below) and the estimated noise level in the flow field; however, to compensate for the fact that the image transformations can not be expected to be exactly affine, it is never less than a given bound, typically one pixel. The amount of support is determined by the function

$$V(a_1, a_2, a_3, a_4, a_5, a_6, x, y) = 1 - 0.75\delta/\epsilon \quad (4.9)$$

which allows the support to range from 1 down to 0.25 for those flow vectors at the limit of the acceptable error range. The total amount of support given to each transformation (a_1, \dots, a_6) by a set \mathcal{G} of flow vectors is the weighted sum

$$S(a_1, a_2, a_3, a_4, a_5, a_6) = \sum_{(x,y) \in D} W(x, y) V(a_1, a_2, a_3, a_4, a_5, a_6, x, y), \quad (4.10)$$

where D is the definition domain of \mathcal{G} and $W(x, y)$ is the weight of the flow vector at the pixel (x, y) .

The optimal affine transformation, among those represented in the parameter space, is the one which is maximally supported by the set \mathcal{G} of flow vectors. However, it is sufficient to find a highly supported transformation, which is not necessarily maximally supported, since the goal in the current segmentation stage is only to group vectors into components as a pre-processing step for the next stage of creating segments. By loosening the maximality constraint, we will be able to drastically reduce the required memory and computation time, as will be shown below.

Let us first discuss considerations for determining the resolution of the quantization in the parameter space. Suppose that \mathcal{G}' is a subset of \mathcal{G} which satisfies the constraints (4.7) for some affine transformation. It is desirable that at least one of the affine transformations represented in the parameter space generates a flow field similar to \mathcal{G}' , when applied to the corresponding definition domain \mathcal{D}' . Otherwise, in order to guarantee the full support of \mathcal{G}' in an adequate affine transformation, it may be necessary to use a large value of ϵ and, then, flow vectors which are not included in \mathcal{G}' may be mistakenly grouped into \mathcal{G}' . In addition, the probability of creating false peaks in the parameter space is increased when ϵ is large. As a conclusion (ignoring for the moment computational considerations which will be referred to later), it is desirable to use a fine resolution in the parameter space.

Let p be the quantization resolution in the axes corresponding to the additive parameters a_1 and a_4 . Moving along one of these axes from one quantization point to an adjacent point, a uniform change of p is induced in the flow field predicted by the affine transformation. On the other hand, moving similarly along one of the axes corresponding to the multiplicative parameters a_2 , a_3 , a_5 and a_6 induces a change which depends on the values of x or y . If these values are large, then the change in the flow field will be correspondingly large. In order to decrease this change, the origin of the image coordinate system is shifted to the center of gravity of the set \mathcal{D} , thus usually reducing the absolute values of x and y . In

addition, to ensure that at least one of the predicted flow fields will be similar to the set \mathcal{G}' defined above, we choose the resolution in the parameters a_2 and a_5 to be $p/\max_D|x|$, and the resolution in a_3 and a_6 to be $p/\max_D|y|$. Thus, the resolution in the multiplicative parameters is inversely proportional to the size of the region containing the set \mathcal{G} of flow vectors.

Basically, we have to compute the support, according to equation (4.9), given by each vector to any of the transformations represented in the parameter space. A serious computational problem may arise if the number of points in the parameter space is very high. For example, in the experiments described in Chapter VII the minimal and maximal possible values of the parameters a_1 and a_4 are taken to be -64 pixels and $+64$ pixels, respectively, and the desired accuracy in determining these parameters is 0.25 pixels. Thus, 512 quantization points are apparently needed for each of them. Using an equal number of quantization points also for the other parameters, the parameter space should contain $512^6 \approx 18 \times 10^{16}$ points, and a straightforward Hough technique is computationally impractical.

This problem is alleviated by using two techniques. First, a multi-resolution scheme in the parameter space is employed. The Hough technique is iteratively used, where in each iteration the parameter space is quantised around the values estimated in the previous iteration, using a finer resolution. Thus, utilizing a limited memory size, accurate parameter values can still be found. In the experiments, for example, 17 quantization points are used for each parameter and 3 iterations are required for achieving the desired accuracy. Notice that using a coarser resolution in the first iteration would increase, more than necessary, the risk of creating false peaks in the parameter space. Other methods for improving the efficiency of the Hough technique are presented in [ORO81, SLO81].

The second technique is based on a decomposition of the parameter set into two disjoint subsets, $\{a_1, a_2, a_3\}$ and $\{a_4, a_5, a_6\}$. The Hough technique is separately

applied to the corresponding 3-dimensional parameter spaces, using the relevant constraint, (4.7a) or (4.7b). Sets of highly supported parameter triples, $A_\alpha = \{(a_{1i}, a_{2i}, a_{3i}) : i = 1, \dots, N\}$ and $A_\beta = \{(a_{4i}, a_{5i}, a_{6i}) : i = 1, \dots, N\}$, are thus found, where N was experimentally determined to be 10. As a result, a set of N^2 hypothesized affine transformations,

$$A_{\alpha\beta} = A_\alpha \times A_\beta = \{(a_{1i}, a_{2i}, a_{3i}, a_{4j}, a_{5j}, a_{6j}) : i = 1, \dots, N; j = 1, \dots, N\}, \quad (4.11)$$

is obtained. The support function (4.10) can be then directly applied to the set $A_{\alpha\beta}$, thus determining the maximally supported transformation T^* in this set. T^* is not necessarily the maximally supported transformation in the 6-dimensional parameter space. However, large components in the flow field, corresponding to maxima points in the 6-dimensional space, can be expected to produce maxima points also in each of the 3-dimensional parameter spaces. Therefore T^* is hoped to be at least a near optimal transformation, as can also be concluded from the experimental results. The decomposition technique is employed in each iteration of the multi-resolution scheme and, therefore, only $17^3 \approx 5000$ elements are required in the parameter space. Thus, the combination of these techniques creates a very efficient algorithm.

IV.3.1.2 Implementation of a multipass approach. The multipass Hough technique is an iterative approach, where in each iteration the motivating goal is to find the largest component of the flow vectors which do not belong to any of the components created in previous iterations. Let us specify this more precisely. We consider the set S of all the vectors except those which fall into one of three categories: (a) they are assigned a zero weight; (b) they are included in one of the already extracted components; (c) they were excluded from further consideration, because of detected incoherency, as will be described below. Initially, S is the set of all the vectors with non-zero weight. We attempt to find a large component in S , where a component is a connected set of vectors which support the same affine

transformation. Note that the new component may also include vectors which do not belong to S , as will be described below.

Flow fields are typically noisy and partially incorrect. Furthermore, in a dense flow field the noise values in adjacent vectors can be expected to be highly correlated and, as a result, false components may be created. Therefore, the weight of each component (the *weight of a set* is defined as the sum of weights of all the flow vectors in this set) is required to exceed a given threshold H , which should be high enough to prevent such components from being created. However, using a relatively large value of H may prevent detection of small components and, eventually, segments corresponding to distinct surfaces and/or independently moving objects. The selection of H should reflect a compromise between these two types of risk, and it heavily depends on the type of algorithm used for computing the flow field. If, for example, connected groups of compatible but errorful vectors are expected to be occasionally present in the flow field, then a selection of a large value of H may be appropriate. Note that H should be at least 3, since almost any set of 3 or less flow vectors is compatible with some affine transformation.

Sometimes an over-fragmentation is unavoidable, and a segment is partitioned into a large number of small components. In order to prevent an excessive number of iterations in such a case, the threshold H is increased if the number of components already determined is higher than a given number, typically 10. The grouping of vectors in small sets is thus postponed to the third stage.

Before describing in detail the multipass technique, let us provide a brief overview of a typical iteration in this process. In the set S , defined above, we first find the largest connected subset, S' (the term *largest subset* refers here, as well as elsewhere in this section, to the subset with maximal weight). An affine transformation, highly supported by vectors selected from S' , is then computed using the Hough technique. In the next step, each vector, in the entire flow field, is examined to

determine its support of the affine transformation. In the set S'' of those vectors which do support this transformation, a connected subset C , with a maximal sum of weights of vectors from S' , is found (note that S'' and C are not necessarily subsets of S'). If the weight of $C \cap S'$ is higher than H , then C is added to the list of components and a new iteration begins. Otherwise, another affine transformation, which is also highly supported by vectors in S' , is selected and a related component is found and checked. If after a few such trials no component is accepted, then the set S' is excluded from further consideration and a new iteration begins.

Let us now describe in more detail the steps composing each iteration of the multipass technique:

1) In the set S , defined above, find the largest connected subset S' . In the current iteration we will focus our attention to S' , attempting to extract an affine transformation supported by a large component of this set. If the weight of S' is not above the threshold H , then there is no point in continuing the search for new components, and the merging stage begins.

2) Select K flow vectors from S' , where K is a given number, typically 64. The Hough technique will be applied only to these vectors, and not to the whole set S' , in order to reduce the computation time. We wish to select vectors which are assigned a high weight and are uniformly distributed over the region corresponding to S' . To achieve this goal, we determine in the image K non-overlapping windows with their union containing the set S' , and then we select from each window a vector with maximal weight. Optimally, these windows should be squares, but for implementation reasons they are rectangles in which the ratio between the long side and the short side is not above 2. In addition, we want the sum of weights of flow vectors belonging to S' to be roughly the same in each window. Thus, the following iterative procedure is employed: in the first iteration there is only one window which is the whole image; then, in each subsequent iteration, the window

with the largest subset of vectors from S' is partitioned into two equal sub-windows across its long axis. Note that if the number of vectors in S' does not exceed K , then, of course, the Hough technique will be applied to all of them.

3) Use the modified Hough technique, described in Section IV.3.1.1, to find the affine transformation which receives the maximal support from the flow vectors selected in the previous step.

4) Find the set, denoted by S'' , of all the vectors which support the new affine transformation. More specifically, for each vector in the entire flow field compute the error measure δ (equations (4.8)) and compare it to the threshold ϵ (Section IV.3.1.1). If δ is smaller than ϵ and the corresponding vector does not belong yet to any component, then include the vector in S'' . In addition, even if the vector already belongs to a component, but the new value of the error measure δ is smaller than the old value corresponding to the existing component, then include the vector in S'' . If the sum of weights in the set $S' \cap S''$ does not exceed the threshold H , then, assuming that other affine transformations will not have significantly more support in the set S' , exclude S' from further consideration and start a new iteration. Otherwise, continue with step (5).

5) Find in the set S'' a connected subset C with a maximal sum of weights of vectors from S' . We consider the weight of the set $C \cap S'$, as opposed to the set C , in order to avoid creation of a new component by basically splitting an already formed component. If this weight is above H , then C is accepted as a new component. Otherwise, we wish to find another affine transformation, also highly supported by vectors in S' , which may be supported by a larger component. Note that the transformation computed in step (3) may be supported by a disconnected set of vectors; thus, although a different transformation is expected to have less total support by vectors in S' , it may still be supported by a larger *connected* subset of vectors. The goal of selecting a new transformation is achieved by decreasing the

weights of all the vectors in S' which support the current affine transformation (in the experiments they are divided by two) and going back to step (2). The weights are decreased only temporarily, while executing again steps (2) and (3) in the current iteration. If after a few such cycles (three, in the experiments) no sufficiently large component is found, then the set S' is excluded from further consideration and a new iteration begins.

6) If the weight of the set $C \cap S'$ exceeds H , then C is added to the list of components. Note that the new component may contain vectors which previously belonged to other components, but were changed because they produce a smaller error measure with the new affine transformation. These components must be examined to see whether they are still connected sets. If not, then the corresponding component is reduced to its largest connected sub-component, and the other vectors are among the vectors to be included in the set S in the next iteration. Note that even if the weight of a reduced component is smaller than H , it is not deleted from the list of components because it can still be assumed not to be a result of noise. Besides, such deletions may interfere with the convergence of the multipass process, and even result in an infinite loop.

IV.3.2 Second stage — merging of components

Components created in the first stage of the segmentation are atomic units which, if consistent with the same Ψ transformation, should be merged together to create a segment. Consistency with a Ψ transformation is detected by computing, using a least-squares technique, optimal parameters and related error values for sets of adjacent components. The merging decision is primarily based on these error values.

IV.3.2.1 Computing an optimal Ψ transformation. Given a set of n flow vectors, we wish to compute the optimal Ψ transformation corresponding to

this set, by employing a least-squares criterion. The error function to be minimized is

$$E(a_1, \dots, a_8) = \sum_{i=1}^n W_i \left[(\alpha_i - a_1 - a_2 x_i - a_3 y_i - a_7 x_i^2 - a_8 x_i y_i)^2 + (\beta_i - a_4 - a_5 x_i - a_6 y_i - a_7 x_i y_i - a_8 y_i^2)^2 \right], \quad (4.12)$$

where, for each i between 1 and n , $(\alpha_i, \beta_i) = (\alpha(x_i, y_i), \beta(x_i, y_i))$ is a flow vector and W_i is the corresponding weight. Taking partial derivatives with respect to a_1, \dots, a_8 and equating to 0, a set of 8 linear equations is obtained. In certain geometric configurations of the set of flow vectors, these equations are linearly dependent; we will refer to this case in Section IV.3.2.2. Usually, however, these equations are independent, and their solution, denoted by a_1^*, \dots, a_8^* , represents the optimal Ψ transformation. Substituting this solution in (4.12) and using the normalization equation

$$\sigma = \sqrt{E(a_1^*, \dots, a_8^*) / \sum_{i=1}^n W_i}, \quad (4.13)$$

an error value, corresponding to the given set of flow vectors, is obtained. σ is an estimate of the standard deviation of the actual flow values from those predicted by the optimal Ψ transformation.

IV.3.2.2 Criteria for a merging decision. In the second stage of the segmentation, groups of adjacent components are examined to see whether they can be merged into one segment, that is, a connected set of flow vectors which support the same Ψ transformation. In Section IV.3.2.3 we will describe a procedure for sequentially selecting groups of components to be examined, but first let us present the criteria for a merging decision.

Let \mathcal{F} denote a group of adjacent components $\{C_j : j = 1, \dots, m\}$. The decision whether to merge these components into one segment is based on the

degree of their consistency with the same Ψ transformation. To determine such consistency, an optimal Ψ transformation and a related error measure, denoted, respectively, by Ψ_j and σ_j , are separately computed for each component C_j , $j = 1, \dots, m$. In addition, an optimal Ψ transformation, denoted by $\Psi_{\mathcal{F}}$, is computed for the entire set of vectors $\bigcup_{j=1}^m C_j$. Substituting the coefficients of $\Psi_{\mathcal{F}}$ and the flow values of the vectors contained in C_j , $1 \leq j \leq m$, in equations (4.12) and (4.13), a new error value, σ'_j , is obtained for each component C_j .

The ratios $\{\sigma'_j/\sigma_j\}$ play a major role in the merging decision. The values of these ratios are at least 1, since Ψ_j can be adjusted to the local surface and noise associated with C_j , and, therefore, σ'_j is never less than σ_j . If the ratios are close to 1, then a decision to merge the components seems to be justified. However, the allowed level of σ'_j/σ_j , $1 \leq j \leq m$, should be a function of the ratio, denoted by p_j , of the sum of vector weights in C_j to the total sum of weights in the set \mathcal{F} . If p_j is close to 1 and the components are really parts of one segment, then σ'_j can be expected to be very close to σ_j . On the other hand, if p_j is close to 0, then $\Psi_{\mathcal{F}}$ can not be expected to be adjusted to C_j , and therefore σ'_j should be allowed to be significantly higher than σ_j , up to a given upper bound l_j (see below). Therefore, a merging decision is accepted if and only if, for each component C_j in \mathcal{F} ,

$$\sigma'_j \leq p_j \sigma_j + (1 - p_j) l_j. \quad (4.14)$$

Thus, the allowed level of σ'_j ranges from σ_j to l_j as p_j varies from 1 to 0.

The upper bound l_j , $1 \leq j \leq m$, is determined to be

$$l_j = \max\{l_a \sigma_j, l_b\}, \quad (4.15)$$

where l_a and l_b are given thresholds. The value of l_a has typically been set to 1.5, while l_b represents a reasonable upper bound on the noise level. If, for example, the most significant noise is round-off error induced by using integer flow values, then the noise is uniformly distributed between -0.5 pixels and $+0.5$ pixels, and l_b will

be $\sqrt{2} \times 0.5$. (The deviation between the measured and predicted flow values is measured as a distance in the 2-D image plane, therefore 0.5 is multiplied by $\sqrt{2}$.)

Sometimes, an error value σ_j , $1 \leq j \leq m$, can not be computed because the corresponding linear equations, derived from (4.12), are linearly dependent. In such a rare situation, the value of σ_j is arbitrarily set to 0, thus making a merging decision more difficult. In addition, if the equations corresponding to all the vectors in the group \mathcal{F} are linearly dependent, then the components are not merged. This 'conservative' policy is adopted to prevent under-segmentation, which may interfere with forming correct object hypotheses in the next phase of the interpretation. Over-segmentation, on the other hand, may increase the computational cost of the next phase, but does not prevent the formation of correct hypotheses.

IV.3.2.3 The merging procedure. The algorithm for finding groups of components to be merged starts by computing an optimal Ψ transformation and a related error measure for each component. After this initial step, segments are sequentially created by testing possible mergings of components. In each cycle of the algorithm a given segment is formed and only components which are not yet assigned to any of the already created segments are considered as candidates for merging into this segment. Eventually, all the components are contained in one of the segments.

The process for creating a given segment starts by detecting the largest component, denoted by C^* , among the candidates for forming this segment. Then, other candidates are sequentially tested, in the order of their associated weights, for merging with C^* . In general, given a set of components which are already assigned to the segment, neighboring candidates, which have not been examined yet, are sequentially tested for merging with this set. Once a merging decision is made, additional components are tested for merging with the *new* set. This process continues until all the candidates for forming the segment are examined.

Results of the merging stage are demonstrated in Figure 7.1d. Note that such a sequential merging process might produce different segments, if another ordering strategy would be chosen.

IV.3.3 Third stage — assimilating ungrouped flow vectors

The purpose of the third stage of the segmentation is examination of flow vectors which were assigned positive weights, but were not grouped into any of the components in the first stage of the segmentation and, therefore, do not belong to any of the segments. Such vectors, called 0-vectors, which are neighbors of one of the segments, are tested for consistency with the Ψ transformation corresponding to this segment and, if consistent, are merged into it. Then, 0-vectors, neighbors of the just segmented vectors, are examined in their turn. This process is iteratively executed until no new vector is merged into one of the segments. Results of this stage of the segmentation are demonstrated in Figure 7.1e.

It is possible that, as a result of over-fragmentation in the first segmentation stage, the threshold H was increased (Section IV.3.1.2) and, therefore, small sets could not be detected as components. Hence, after the third stage we look for connected sets of 0-vectors, which were not excluded from further consideration in the first segmentation stage, and were not merged into one of the existing segments in the third stage. If such sets exist, then the first and the second stages of the segmentation are executed again, focused only on these sets, thus possibly creating new segments.

CHAPTER V

TESTING OBJECT HYPOTHESES AND RECOVERING 3-D MOTION AND STRUCTURE

In the first phase of the interpretation process, described in the preceding chapter, the flow field is segmented into connected sets of flow vectors, where each set is consistent with a rigid motion of a roughly planar surface. Such a segment is assumed to correspond to a portion of only one rigid object. The next task is to group segments which are consistent with the same 3-D motion parameters. Employing a rigidity assumption similar to the one in [ULL79], such a group can be assumed to be induced by one rigidly moving object (or by the camera motion). All the segments corresponding to the stationary environment will be grouped together to create one rigid object. It should be noted, however, that the segmentation results may also provide a useful decomposition of the environment into roughly planar surfaces.

In Section V.1 we describe a general algorithm for computing the motion parameters, relative to the camera, from a set of flow vectors generated by a rigid motion. In Section V.2 we combine this algorithm with the segmentation results to test object hypotheses and to estimate the corresponding 3-D motion and structure.

V.1 Estimating Motion Parameters and Depth Information of a Rigid Object

V.1.1 Optimization constraint

Given a set of flow vectors, assumed to be induced by a rigidly moving object, we wish to find the 3-D motion parameters and depth values which are maximally consistent with this data. Following the general approach proposed in [BRU81],

we employ a least-squares technique which minimizes the deviation between the measured flow field and that predicted from the estimated motion and structure. This approach is adopted because of its relative robustness in the presence of noise. Based on (3.5), the error function to be minimized is

$$\sum_{i=1}^n W_i \left[\left(\alpha_i + \Omega_X x_i y_i - \Omega_Y (1 + x_i^2) + \Omega_Z y_i - (T_X - T_Z x) / Z_i \right)^2 + \left(\beta_i + \Omega_X (1 + y_i^2) - \Omega_Y x_i y_i - \Omega_Z x_i - (T_Y - T_Z y) / Z_i \right)^2 \right], \quad (5.1)$$

where $\underline{T} = (T_X, T_Y, T_Z)$ and $\underline{\Omega} = (\Omega_X, \Omega_Y, \Omega_Z)$ are the translation and rotation vectors, respectively, and, for each i between 1 and n , (α_i, β_i) is the flow vector computed at the pixel (x_i, y_i) , W_i is its weight and Z_i is the spatial depth of the corresponding point in the environment. The task is to determine \underline{T} , $\underline{\Omega}$ and $\{Z_i\}$ which minimize this function. Using the decomposition of the flow field into its rotational and translational components, denoted by (α_R, β_R) and (α_T, β_T) (see equations (3.6) and (3.7)), the error function can be more concisely represented by

$$\sum_{i=1}^n W_i [(\alpha_i - \alpha_{Ri} - \alpha_{Ti})^2 + (\beta_i - \beta_{Ri} - \beta_{Ti})^2]. \quad (5.2)$$

As can easily be seen, it is actually impossible to determine the absolute values of (T_X, T_Y, T_Z) and $\{Z_i : i = 1, \dots, n\}$. However, if the magnitude, denoted by r , of the translation is non-zero, then it is possible to estimate the direction of the 3-D translation, represented by the unit vector

$$(U_X, U_Y, U_Z) = (T_X, T_Y, T_Z) / r, \quad (5.3)$$

and the relative depth values, represented by

$$\tilde{Z}_i = r / Z_i, \quad i = 1, \dots, n. \quad (5.4)$$

We employ this representation of the depth values, rather than the more 'natural'

representation Z_i/r , in order to avoid the possibility of infinite values in cases where \bar{Z}_i , as given by equation (5.10), is 0.

Introducing the abbreviations

$$\alpha_U = U_X - U_Z x = \alpha_T / \bar{Z} \quad (5.5a)$$

and

$$\beta_U = U_Y - U_Z y = \beta_T / \bar{Z}, \quad (5.5b)$$

(5.2) can be rewritten as

$$\sum_{i=1}^n W_i \left[(\alpha_i - \alpha_{R_i} - \alpha_U \bar{Z}_i)^2 + (\beta_i - \beta_{R_i} - \beta_U \bar{Z}_i)^2 \right]. \quad (5.6)$$

Thus, the task can be reformulated as one of finding the values of (U_X, U_Y, U_Z) , $(\Omega_X, \Omega_Y, \Omega_Z)$ and $\{\bar{Z}_i : i = 1, \dots, n\}$ which minimize this expression. In addition, the depth constraints

$$\bar{Z}_i \geq 0, \quad i = 1, \dots, n, \quad (5.7)$$

should be satisfied. Note that this error measure is different from the one actually employed in [BRU81] where the contribution of each flow vector is multiplied by $\alpha_U^2 + \beta_U^2$ in order to simplify the error function.

For any given i , $1 \leq i \leq n$, we can find the optimal value of \bar{Z}_i , as a function of the motion parameters, by examining the first derivative of (5.6) with respect to \bar{Z}_i . This derivative is given by

$$2W_i \left[-(\alpha_i - \alpha_{R_i})\alpha_U - (\beta_i - \beta_{R_i})\beta_U + (\alpha_U^2 + \beta_U^2)\bar{Z}_i \right]. \quad (5.8)$$

Setting it equal to 0 yields

$$\bar{Z}_i = \left((\alpha_i - \alpha_{R_i})\alpha_U + (\beta_i - \beta_{R_i})\beta_U \right) / (\alpha_U^2 + \beta_U^2), \quad (5.9)$$

unless $\alpha_{U_i}^2 + \beta_{U_i}^2 = 0$, in which case \bar{Z}_i can be assigned any non-negative value. If the expression in (5.9) is negative, then the corresponding depth constraint in (5.7) is unsatisfied. In such a case, to minimize the error function (5.6), \bar{Z}_i should be set to 0, because the derivative (5.8) is non-negative for non-negative values of \bar{Z}_i and, therefore, the error function is monotonically non-decreasing for these values. To summarize, the optimal value of \bar{Z}_i is given by:

$$\bar{Z}_i = \left[\left((\alpha_i - \alpha_{R_i})\alpha_{U_i} + (\beta_i - \beta_{R_i})\beta_{U_i} \right) / (\alpha_{U_i}^2 + \beta_{U_i}^2) \right]^+. \quad (5.10)$$

Substituting (5.10), for any $1 \leq i \leq n$, into (5.6) and expanding the resulting expression yields the following representation of the error, as a function of the motion parameters:

$$E(\underline{U}, \underline{\Omega}) = \sum_{i=1}^n W_i E_i, \quad (5.11a)$$

where

$$E_i = \begin{cases} \frac{[(\alpha_i - \alpha_{R_i})\beta_{U_i} - (\beta_i - \beta_{R_i})\alpha_{U_i}]^2}{\alpha_{U_i}^2 + \beta_{U_i}^2} & \text{if } (\alpha_i - \alpha_{R_i})\alpha_{U_i} + (\beta_i - \beta_{R_i})\beta_{U_i} > 0; \\ (\alpha_i - \alpha_{R_i})^2 + (\beta_i - \beta_{R_i})^2 & \text{otherwise.} \end{cases} \quad (5.11b)$$

A normalized version of this error function, defined by

$$\sigma(\underline{U}, \underline{\Omega}) = \sqrt{E(\underline{U}, \underline{\Omega}) / \sum_{i=1}^n W_i}, \quad (5.12)$$

will also be utilized. σ is an estimate of the standard deviation of the measured flow values from those predicted by the motion parameters and the corresponding depth values.

Note that the expression (5.11) for the error function was obtained by assuming a non-zero translation. In the case of a purely rotational motion, the appropriate

error function to be minimized is:

$$E_R(\Omega) = \sum_{i=1}^n W_i \left((\alpha_i - \alpha_{R_i})^2 + (\beta_i - \beta_{R_i})^2 \right). \quad (5.13)$$

The minimal value of this function is never less than the minimal value of $E(\underline{U}, \Omega)$, since the latter function is derived from (5.6) where the values of \tilde{Z}_i , $i = 1, \dots, n$, can be chosen to be 0, thus predicting a purely rotational flow field. If, however, the minimal value of $E_R(\Omega)$ is close to the minimal value of $E(\underline{U}, \Omega)$, then the 3-D motion is, possibly, purely rotational. Hence, both values should be computed and compared to each other.

The task of finding the three rotation parameters which minimize the function $E_R(\Omega)$ can easily be accomplished by taking the partial derivatives of $E_R(\Omega)$ with respect to the rotation parameters, setting them equal to 0, and solving the linear equations so obtained [BRU81]. In the next section we concentrate on the much more difficult task of finding values of \underline{U} and Ω which minimize the error function $E(\underline{U}, \Omega)$ (or, equivalently, the function $\sigma(\underline{U}, \Omega)$), where \underline{U} can be any unit vector and Ω is unconstrained.

V.1.2 Algorithm

The algorithm for recovering the motion parameters employs an error measure, derived from (5.12), corresponding to possible locations of the FOE in the image plane. For each hypothesized FOE, the optimal rotation parameters and a related error value are computed. A minimum value of the resulting error function is determined, using a multi-resolution sampling technique. Notice the difference of this approach from the one employed in [PRA81], where for each hypothesized *rotational* component, the FOE of the corresponding translational field and a related error measure are computed. We believe that our approach is more efficient, since the search space is only 2-dimensional, as opposed to 3-dimensional in [PRA81], and the error function which we employ is easier to compute. Before we proceed with a

mathematical description of the algorithm, we refer the reader to the geometrical interpretation given in Figure 5.1.

V.1.2.1 Reducing the search space. In this section we show how to derive a new error function from $E(\underline{U}, \underline{\Omega})$. This function will be defined only on the unit hemisphere which is isomorphic to the image plane, thus reducing the dimensionality of the search space from five to two. Let us start the derivation of the new function with the observation that if the depth constraints (5.7) are ignored, then, for any hypothesized direction of translation, the optimal rotation parameters can easily be extracted by solving a set of three linear equations. To see that, notice that the error function (5.11) can be reduced in this case to the function

$$E'(\underline{U}, \underline{\Omega}) = \sum_{i=1}^n W_i \left[\left((\alpha_i - \alpha_{R_i}) \beta_{U_i} - (\beta_i - \beta_{R_i}) \alpha_{U_i} \right)^2 / (\alpha_{U_i}^2 + \beta_{U_i}^2) \right]. \quad (5.14)$$

Differentiating $E'(\underline{U}, \underline{\Omega})$ with respect to the rotation parameters $\Omega_X, \Omega_Y, \Omega_Z$ and setting the derivatives equal to 0 yields three linear equations with the rotation parameters as unknowns. Thus, ignoring the depth constraints (5.7), the search space can be limited to the unit sphere $\{ \underline{U} : |\underline{U}| = 1 \}$.

Moreover, changing the sign of any unit vector \underline{U} has no effect on the value of $E'(\underline{U}, \underline{\Omega})$ since it only affects the sign of α_U and β_U . Therefore, the search space can be further restricted to the hemisphere

$$\mathcal{H} = \{ \underline{U} : |\underline{U}| = 1 \text{ and } U_Z \geq 0 \}. \quad (5.15)$$

The depth constraints or, equivalently, the equations (5.11b) must still be incorporated in the algorithm for recovering the motion parameters, at least for determining the sign of the optimal vector \underline{U} found in \mathcal{H} . Bruss and Horn [BRU81], for example, select the sign which gives $\tilde{Z}_i \geq 0$ for most indices i . We propose an alternative and more rigorous way for incorporating these constraints, in which, for

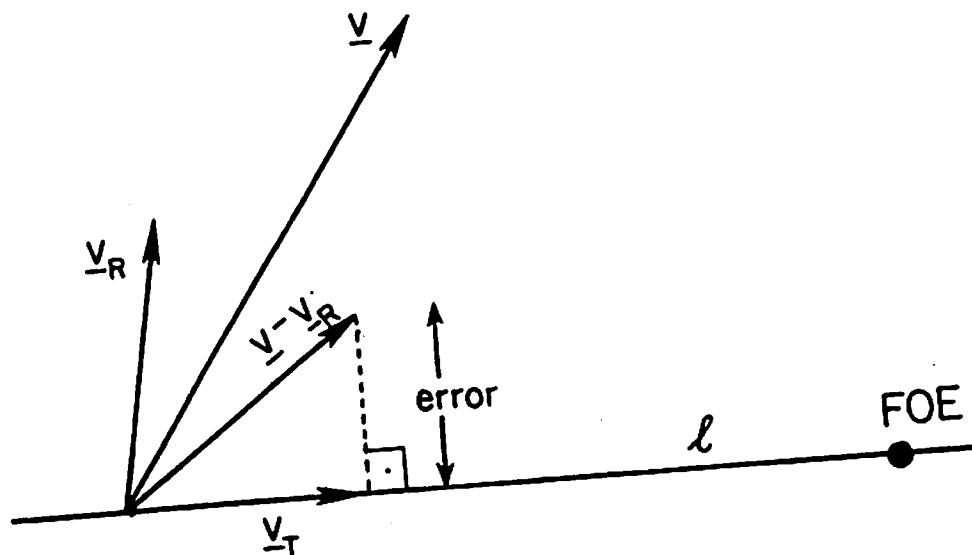


Figure 5.1: A geometrical interpretation of the algorithm. l is the line connecting the initial point of a flow vector \underline{v} to a hypothesized location of the FOE. We wish to find rotation parameters such that the difference $\underline{v} - \underline{v}_R$, where \underline{v}_R is the corresponding rotational component, will be oriented towards or away from the FOE. The projection of $\underline{v} - \underline{v}_R$ on l is \underline{v}_T , the estimated translational component of the flow vector. The related depth value \bar{Z} is determined as the one which yields such a translational component. The error measure is the distance between the end points of $\underline{v} - \underline{v}_R$ and \underline{v}_T . Note that the depth constraint means that if the Z -component of the object translation is positive (i.e., forward object motion), then \underline{v}_T should be oriented *towards* the FOE, while if the Z -component is negative (i.e., backward object motion), then \underline{v}_T should be oriented *from* the FOE. Therefore, the translational components should all be oriented towards, or all be oriented from, the FOE. If the depth constraint is unsatisfied, we choose \underline{v}_T to be $(0, 0)$.

each \underline{U} in \mathcal{N} , we define the error measure

$$\sigma_{\mathcal{N}}(\underline{U}) = \min_{s, \underline{\Omega}} \sigma(s\underline{U}, \underline{\Omega}), \quad (5.16)$$

where s can have the values $+1$ or -1 . The goal is to find a vector \underline{U} in \mathcal{N} that minimizes the function $\sigma_{\mathcal{N}}$. The associated values of s and $\underline{\Omega}$ are, respectively, the determined sign of the translation vector and the estimated rotation parameters.

The function $\sigma_{\mathcal{N}}$ is, however, difficult to compute, because the depth constraints are involved in the computation. Therefore, in the proposed algorithm we compute an approximation to $\sigma_{\mathcal{N}}$ which in the experiments was usually found to be very accurate. A few main steps can be distinguished in the procedure for computing this approximation:

1) Given a vector \underline{U} in \mathcal{N} , estimate the optimal rotation vector $\underline{\Omega}^*$ by minimizing $E'(\underline{U}, \underline{\Omega})$ with respect to $\underline{\Omega}$, and compute the corresponding normalized error measure $\sigma'(\underline{U}, \underline{\Omega}^*)$. This error value, denoted by $\sigma_{\mathcal{N}}^l(\underline{U})$, is a lower bound of $\sigma_{\mathcal{N}}(\underline{U})$ since it minimizes the error function $\sigma(\underline{U}, \underline{\Omega})$, with respect to $\underline{\Omega}$ and the sign of \underline{U} , without considering the depth constraints (5.7). Note that it may be impossible to determine the optimal rotation parameters, if the corresponding linear equations are singular. Still, even in such a situation (which never happened in our experiments), the error measure $\sigma'(\underline{U}, \underline{\Omega})$ can be minimized with respect to $\underline{\Omega}$ by employing, for example, a Hough technique. It may also be possible to change slightly the sampling point \underline{U} (see Section V.1.2.2 for a description of the sampling procedure), in order to avoid the singular situation.

2) Compute $\sigma(\underline{U}, \underline{\Omega}^*)$ and $\sigma(-\underline{U}, \underline{\Omega}^*)$. The minimum of these two error values determines the preferred sign of \underline{U} , denoted by s^* . The value $\sigma(s^*\underline{U}, \underline{\Omega}^*)$, denoted by $\sigma_{\mathcal{N}}^u(\underline{U})$, is an upper bound of $\sigma_{\mathcal{N}}(\underline{U})$, because it gives the actual error measure for some values of s and $\underline{\Omega}$ in equation (5.16).

3) Compute an approximation to $\sigma_{\mathcal{N}}(\underline{U})$ by averaging its lower and upper

bounds:

$$\delta_N(\underline{U}) = \left(\sigma_N^{\downarrow}(\underline{U}) + \sigma_N^{\uparrow}(\underline{U}) \right) / 2. \quad (5.17)$$

This approximation is utilized because it minimizes the maximal possible error. The relative deviation of $\sigma_N(\underline{U})$ from $\delta_N(\underline{U})$ is bounded by

$$\left(\sigma_N^{\downarrow}(\underline{U}) - \sigma_N^{\uparrow}(\underline{U}) \right) / (2\delta_N(\underline{U})). \quad (5.18)$$

In the experiments this value is usually very small, typically much less than 0.01.

The following proposition shows that the solution \underline{U}^* obtained by minimizing δ_N is guaranteed to be optimal or near optimal, if the difference d between the minimal values of the upper and lower bounds of σ_N is very small. Note that the value of \underline{U} for which each of these bounds is minimized will in general be different.

Now let us prove that

$$\sigma_N(\underline{U}^*) \leq \min_{\underline{U}} \sigma_N(\underline{U}) + 2d, \quad (5.19)$$

where

$$d = \min_{\underline{U}} \sigma_N^{\downarrow}(\underline{U}) - \min_{\underline{U}} \sigma_N^{\uparrow}(\underline{U}). \quad (5.20)$$

First, it should be noted that

$$\frac{1}{2} \left(\sigma_N^{\downarrow}(\underline{U}^*) + \sigma_N^{\uparrow}(\underline{U}^*) \right) = \delta_N(\underline{U}^*) = \min_{\underline{U}} \delta_N(\underline{U}) \leq \min_{\underline{U}} \sigma_N^{\downarrow}(\underline{U}). \quad (5.21)$$

Therefore,

$$\sigma_N^{\downarrow}(\underline{U}^*) \leq 2 \min_{\underline{U}} \sigma_N^{\downarrow}(\underline{U}) - \sigma_N^{\uparrow}(\underline{U}^*) \leq 2 \min_{\underline{U}} \sigma_N^{\downarrow}(\underline{U}) - \min_{\underline{U}} \sigma_N^{\uparrow}(\underline{U}) = \min_{\underline{U}} \sigma_N^{\downarrow}(\underline{U}) + d. \quad (5.22)$$

Finally, employing also the inequalities

$$\min_{\underline{U}} \sigma_N^{\downarrow}(\underline{U}) \leq \min_{\underline{U}} \sigma_N(\underline{U}) \leq \min_{\underline{U}} \sigma_N^{\uparrow}(\underline{U}), \quad (5.23)$$

it follows that

$$\sigma_N(\underline{U}^*) \leq \sigma_N^{\dagger}(\underline{U}^*) \leq \min_{\underline{U}} \sigma_N^{\dagger}(\underline{U}) + d \leq \min_{\underline{U}} \sigma_N(\underline{U}) + 2d. \quad (5.24)$$

Even in rare situations (which never occurred in our experiments), where the difference d is relatively large, the bounds σ_N^{\dagger} and σ_N will usually be useful. Based on these bounds, vectors \underline{U} can be removed from further consideration, while still guaranteeing the near optimality of the final solution. To show how this can be done, let us first denote by $\tilde{\underline{U}}$ the vector which minimizes the upper bound function σ_N^{\dagger} , and let ϵ denote a small error value, typically 0.01 pixels (see Figure 5.2). Given a vector $\underline{U} \neq \tilde{\underline{U}}$ for which the lower bound $\sigma_N^{\dagger}(\underline{U})$ is larger than $\min \sigma_N^{\dagger} - \epsilon$, then this lower bound is larger or only slightly smaller than the upper bound of $\sigma_N(\tilde{\underline{U}})$. Therefore, \underline{U} can not produce an error value which is much lower than the one produced by $\tilde{\underline{U}}$. Thus, \underline{U} can be eliminated as a possible solution without affecting the near optimality of the final result.

On the other hand, if there exist vectors \underline{U} for which the lower bound $\sigma_N^{\dagger}(\underline{U})$ is much smaller than $\min \sigma_N^{\dagger}$, then it is usually necessary to more accurately estimate the corresponding error values $\sigma_N(\underline{U})$. Given such a vector \underline{U} , this can be done by searching for optimal rotation parameters using, for example, a sampling technique, and then substituting the results in equation (5.12). Note that if many error values $\sigma_N(\underline{U})$ must be re-evaluated, then we basically face again a problem of search in a 5-dimensional parameter space. Finally, note that a vector \underline{U} for which $\sigma_N^{\dagger}(\underline{U})$ is significantly lower than $\min \sigma_N^{\dagger}$ can still be accepted as a solution without further examination, if two conditions are simultaneously satisfied: (a) $\sigma_N^{\dagger}(\underline{U})$ is at most slightly larger than $\min \sigma_N^{\dagger}$; and (b) no other vector \underline{U}' produces lower bound $\sigma_N^{\dagger}(\underline{U}')$ which is much smaller than $\min \sigma_N^{\dagger}$.

To summarize, the algorithm can be efficiently applied if the minimal values of the lower and upper bounds are close to each other, or if not many vectors \underline{U} pro-

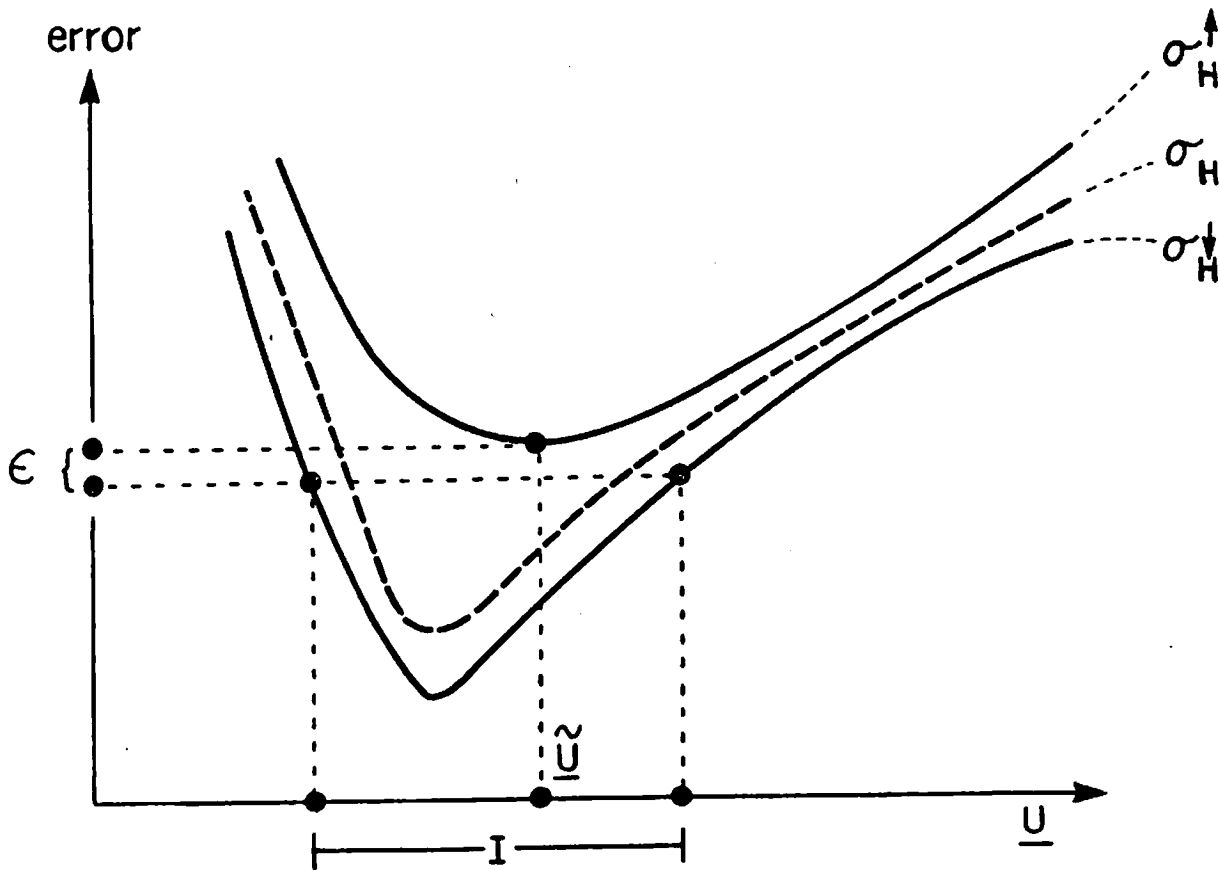


Figure 5.2: The error function $\sigma_{\mathcal{N}}$ and its upper and lower bounds, $\sigma_{\mathcal{N}}^{\uparrow}$ and $\sigma_{\mathcal{N}}^{\downarrow}$, are displayed as functions of the translation axis \underline{U} . To simplify the illustration, the 2-dimensional definition domain, which is the unit hemisphere \mathcal{N} , is plotted as a 1-dimensional space. The vector $\tilde{\underline{U}}$ is the vector which minimizes the function $\sigma_{\mathcal{N}}^{\downarrow}$. The interval I contains all the vectors \underline{U} for which the lower bound $\sigma_{\mathcal{N}}^{\downarrow}(\underline{U})$ is smaller than $\min \sigma_{\mathcal{N}}^{\downarrow} - \epsilon$. I is the domain where the correct error measure $\sigma_{\mathcal{N}}$ must be more accurately estimated. Vectors \underline{U} which are not included in I can not produce error values $\sigma_{\mathcal{N}}(\underline{U})$ which are significantly smaller than $\sigma_{\mathcal{N}}(\tilde{\underline{U}})$.

duce lower bounds $\sigma_N^l(\underline{U})$ much smaller than $\min \sigma_N^l$. Based on our experimental results, we conjecture that these conditions hold in almost any practical situation. More work should be done, however, in order to characterize situations where these conditions may be not satisfied.

V.1.2.2 Search strategy. The search for an optimal vector in \mathcal{N} consists of a sampling of the error measure δ_N . A multi-resolution scheme is employed, where in the first iteration the set \mathcal{N} is coarsely sampled (similarly to [LAW82,84]) and in each additional iteration only the neighborhood of the vector giving a minimum value in the previous iteration is sampled, using a finer resolution. Note that solutions near the boundary of \mathcal{N} require a vector \underline{U}' to be defined as a 'neighbor' of a vector \underline{U} if either \underline{U}' or $-\underline{U}'$ is close to \underline{U} . Another way to obtain the same effect while using the normal definition of a neighborhood is to extend the domain of definition of the function δ_N to the whole unit sphere, employing exactly the same definition used for the domain \mathcal{N} . In this case, $\delta_N(-\underline{U}) = \delta_N(\underline{U})$ for each unit vector \underline{U} , and hence, computationally, it makes no difference which domain of definition is used.

In [LAW82,84] the unit sphere is uniformly sampled in the first step of the search process. We propose a different approach, where the density of the sampling in a given region of \mathcal{N} is proportional to the rate of change of the flow field, as a function of \underline{U} , when \underline{U} varies in this region. Using this approach, the relative number of samples is decreased in areas of the hemisphere where the error function is expected to be flat, but it is increased in areas where this function is expected to vary rapidly. Thus, for a given number of samples, the probability of missing the correct solution, because of a sampling that is too sparse around it, becomes smaller.

In order to formulate this sampling constraint more concretely, suppose that the translation vector \underline{U} is changed by $\Delta\underline{U}$. This induces a deviation of $(\Delta\alpha_T(x, y),$

$\Delta\beta_T(x, y)$) in the translational component of the flow field (α_T, β_T) at the pixel (x, y) . The total deviation, given by the weighted sum

$$\sum_{x,y} W(x, y) (\Delta^2\alpha_T(x, y) + \Delta^2\beta_T(x, y)), \quad (5.25)$$

will be employed as a measure of the global sensitivity of the flow field to changes in \underline{U} . Our goal is to determine a sampling pattern of \mathcal{N} such that the total deviation induced by changing \underline{U} from one sampling point to an adjacent one will be roughly uniform over \mathcal{N} . This sampling constraint will be called the *uniform deviation constraint*.

To determine an appropriate sampling pattern, let us switch to a spherical coordinate system (r, ϕ, θ) , where

$$T_X = r \sin \phi \cos \theta, \quad (5.26a)$$

$$T_Y = r \sin \phi \sin \theta \quad (5.26b)$$

and

$$T_Z = r \cos \phi. \quad (5.26c)$$

Note that ϕ is the angle between the line of sight and the translation vector, and θ is the angle between the x -axis and the projection of the translation vector on the image plane. Using these notations, \mathcal{N} can be represented by the set

$$\{(\phi, \theta) : 0 \leq \phi \leq 90^\circ, 0^\circ \leq \theta < 360^\circ\}. \quad (5.27)$$

In addition, the translational component of the flow field at the pixel (x, y) can be rewritten as

$$(\alpha_T, \beta_T) = \frac{r}{Z(x, y)} (\sin \phi \cos \theta - x \cos \phi, \quad \sin \phi \sin \theta - y \cos \phi). \quad (5.28)$$

Changing the translation parameters (ϕ, θ) by $(\Delta\phi, \Delta\theta)$, where $\Delta\phi$ and $\Delta\theta$ are

small, induces the following deviation in the flow field:

$$(\Delta\alpha_T, \Delta\beta_T) \approx \left(\frac{\partial\alpha_T}{\partial\phi} \Delta\phi + \frac{\partial\alpha_T}{\partial\theta} \Delta\theta, \frac{\partial\beta_T}{\partial\phi} \Delta\phi + \frac{\partial\beta_T}{\partial\theta} \Delta\theta \right). \quad (5.29)$$

The partial derivatives of α_T and β_T with respect to ϕ and θ can be derived from (5.28):

$$\begin{pmatrix} \partial\alpha_T/\partial\phi & \partial\alpha_T/\partial\theta \\ \partial\beta_T/\partial\phi & \partial\beta_T/\partial\theta \end{pmatrix} = \frac{r}{Z(x,y)} \begin{pmatrix} \cos\phi \cos\theta + x \sin\phi & -\sin\phi \sin\theta \\ \cos\phi \sin\theta + y \sin\phi & \sin\phi \cos\theta \end{pmatrix}. \quad (5.30)$$

In this stage of the analysis we can already derive a useful observation on the desirable sampling density of ϕ and θ as a function of ϕ . If x and y are small (in focal units), then from (5.29) and (5.30) it follows that:

$$\Delta^2\alpha_T + \Delta^2\beta_T \approx \frac{r^2}{Z^2(x,y)} (\cos^2\phi \Delta^2\phi + \sin^2\phi \Delta^2\theta). \quad (5.31)$$

Thus, the sensitivity of the flow vector at (x, y) to changes in ϕ is higher as ϕ becomes smaller. On the other hand, as ϕ becomes larger, the flow vector is more sensitive to changes in θ .

A concrete sampling pattern which reflects this observation will now be determined. We will compute the total deviation (5.25) for a representative case in which the surface is a plane parallel to the image, the flow field is dense and uniformly distributed over the image, and $W(x, y) = 1$ for each pixel (x, y) . For this case the total deviation is given by

$$\sum_{x,y} (\Delta^2\alpha_T + \Delta^2\beta_T) \approx \int_{-A}^A \int_{-A}^A (\Delta^2\alpha_T + \Delta^2\beta_T) dx dy, \quad (5.32)$$

where $A = \tan(\text{fov}/2)$ (fov denotes the field of view of the camera). Note that A , which is an upper bound of x and y values, is not assumed to be small. Substituting $\Delta\alpha_T$ and $\Delta\beta_T$ with the corresponding expressions in equations (5.29) and (5.30),

and expanding the double integral, we get

$$\sum_{x,y} (\Delta^2 \alpha_T + \Delta^2 \beta_T) \approx \frac{4r^2 A^2}{Z^2} \left((\cos^2 \phi + \frac{2A^2}{3} \sin^2 \phi) \Delta^2 \phi + \sin^2 \phi \Delta^2 \theta \right). \quad (5.33)$$

Suppose now that we select sampling points along a line of longitude on the unit hemisphere \mathcal{N} ; that is, we keep θ constant. Then, in order to satisfy the uniform deviation constraint, the difference $\Delta\phi$ between adjacent sampling points should be (based on (5.33))

$$\Delta\phi \approx \frac{c}{\sqrt{\cos^2 \phi + \frac{2A^2}{3} \sin^2 \phi}}, \quad (5.34)$$

where c is a proportion factor. Similarly, if we select sampling points along a line of latitude, we should have

$$\Delta\theta \approx \frac{c}{\sin \phi}. \quad (5.35)$$

Based on these observations, let us now construct a sampling pattern of \mathcal{N} which satisfies the uniform deviation constraint. First, we determine a sequence $\phi_1 < \phi_2 < \dots < \phi_n$ such that $\phi_1 = 0^\circ$, $\phi_n = 90^\circ$ and

$$\phi_{i+1} - \phi_i \approx \frac{c}{\sqrt{\cos^2 \phi_i + \frac{2A^2}{3} \sin^2 \phi_i}}, \quad 1 \leq i \leq n-1, \quad (5.36)$$

Then, for each $2 \leq i \leq n$, we choose a sequence $\theta_{i,1} < \theta_{i,2} < \dots < \theta_{i,m_i}$ where $\theta_{i,1} = 0^\circ$, $\theta_{i,m_i} = 360^\circ$ and

$$\theta_{i,j+1} - \theta_{i,j} \approx \frac{c}{\sin \phi_i}, \quad 1 \leq j \leq m_i - 1. \quad (5.37)$$

In addition, we determine m_1 to be 1 and $\theta_{1,1}$ to be 0° . Given the desirable number of samples, the value of c can correspondingly be determined, and the function $\delta_{\mathcal{N}}$ is sampled in the points

$$\left\{ \{(\phi_i, \theta_{i,j}) : j = 1, \dots, m_i\}, i = 1, \dots, n \right\}. \quad (5.38)$$

Note that for $180^\circ \leq \theta < 360^\circ$, the points $(90^\circ, \theta)$ and $(90^\circ, \theta - 180^\circ)$ correspond to unit vectors which differ only by their sign and, therefore, $\hat{\sigma}_N(90^\circ, \theta) = \hat{\sigma}_N(90^\circ, \theta - 180^\circ)$. Hence, points of the form $(90^\circ, \theta)$, where $180^\circ \leq \theta < 360^\circ$, are actually excluded from the sampling set (5.38). An example of a sampling pattern, for $fov = 30^\circ$, is shown in Figure 5.3.

V.1.2.3 Recovering environmental structure. The solution of \underline{U} , found in the last iteration of the unit hemisphere sampling procedure, and the corresponding sign s^* and the rotation parameters $\underline{\Omega}^*$, defined in the procedure for computing $\hat{\sigma}_N$, are the determined motion parameters. Substituting these parameters in equation (5.10), the relative depth, corresponding to each flow vector, can be estimated as well.

V.1.3 Comments on inherent ambiguity and instability

The flow field equations (3.5) may, in general, have more than one solution. If, for example, the surface is planar, then usually two solutions exist [TSA84, WAX83]. Thus, multiple solutions should be searched for on the error surface. This can be done by locating locally minimal error values, which are also very close to the global minimum.

Another problem is the inherent instability which may exist in recovering 3-D motion and structure from *noisy* flow fields. Such a situation can be identified as one in which the error function $\hat{\sigma}_N$ (equation (5.17)) is very close to its minimal value in a large portion of the search space (see, for example, Figures 7.2h and 7.2i). In this case, reasonably reliable estimates of the motion parameters may be impossible to obtain. The influence of certain parameters on this instability will be analyzed in Section VI.1. Approaches which may be taken in order to deal with this problem will also be discussed.

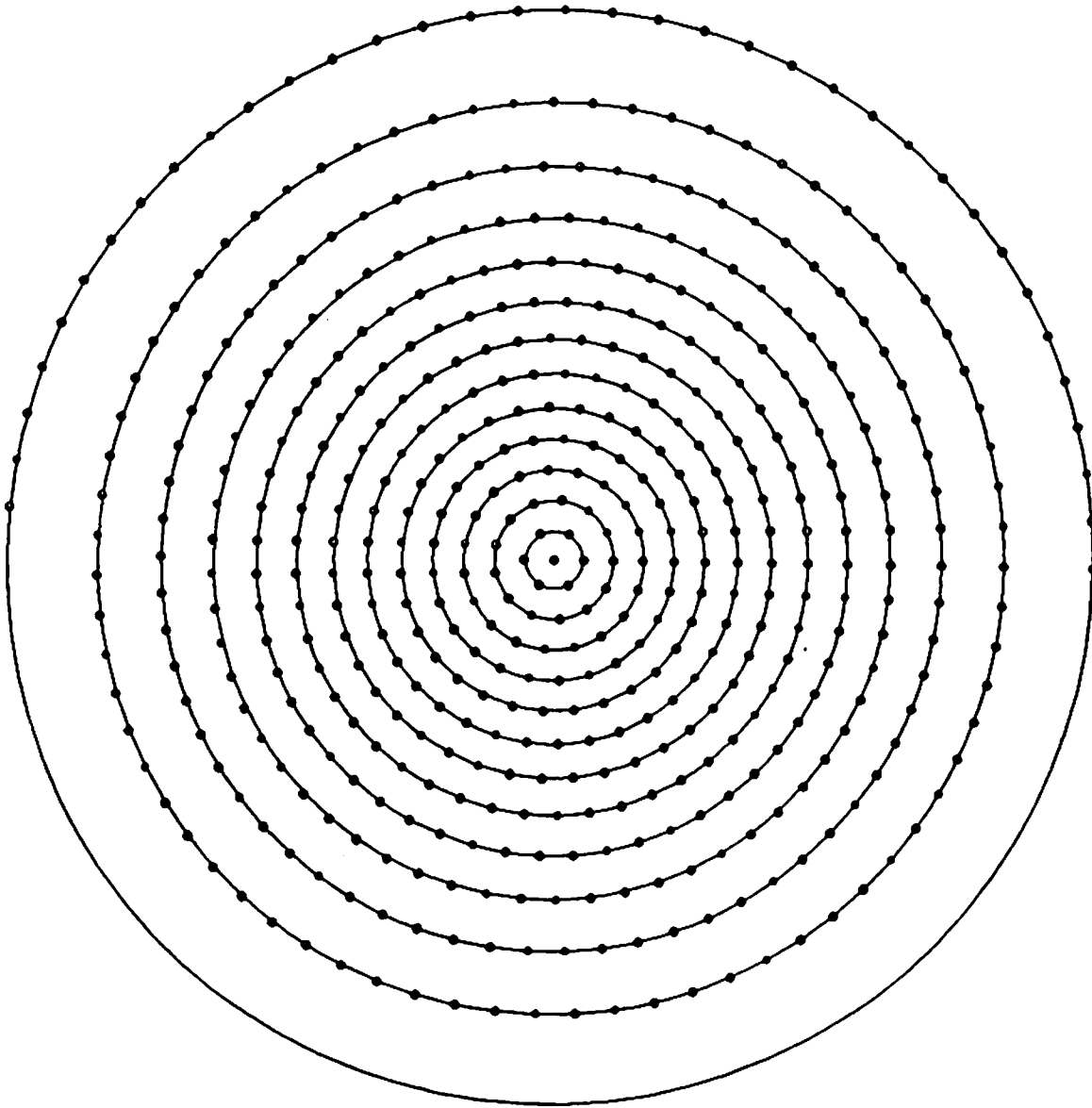


Figure 5.3: A sampling pattern for $fov = 30^\circ$. The total number of samples, represented in the figure by dots, is 513. The angles (ϕ, θ) serve here as polar coordinates, where ϕ ranges from 0° at the center up to 90° at the boundary. Each circle corresponds to a given value of ϕ . Sampling points on one half of the outer circle are excluded, as explained in the text.

V.2 Testing Object Hypotheses

Employing a rigidity assumption similar to the one in [ULL79], segments of the flow field which are consistent with the same 3-D motion parameters can be assumed to be induced by one rigidly moving object (or by the camera motion). Using the algorithm described in the previous section, such a consistency is detected by computing optimal motion parameters and related error values for sets of segments. The criteria for accepting object hypotheses, which are similar to the criteria described in Section IV.3.2.2 for merging components into segments, are primarily based on these error values.

It will be shown in Section VI.2 that the decomposition of the environment into independently moving objects may be ambiguous. For example, two such objects induce, in some cases, optical flows which can be interpreted as resulting from one rigidly moving object. In order to deal with this ambiguity, one may have to find a *set* of possible decompositions, not only one. Therefore, each group of segments is tested for consistency with 3-D motion parameters, unless it contains a sub-group which has already been shown to be associated with more than one rigid motion.

At this point, we would like to raise the question of the necessity of the flow field segmentation as a pre-processing step for the formation of object hypotheses. At first sight, by employing a multipass Hough technique, one would think that it might be possible to directly decompose the flow field into sets corresponding to independently moving objects and, simultaneously, recover the associated motion parameters. In this approach, a 5-dimensional parameter space can be used, where each dimension corresponds to one of the recoverable motion parameters. Flow vectors would vote for compatible transformations represented in the parameter space, and the maximally supported transformations are supposed to correspond to rigid objects in the environment. Thus, the segmentation process described in Chapter IV could be avoided. However, even if the computational issues involved

in developing such a scheme can be solved, at least two problems still remain: (a) flow vectors corresponding to small but distinct surfaces, which may resolve ambiguities in recovering the motion parameters (see Section VI.1), do not receive the appropriate weight in the voting process; (b) it is difficult to detect possible ambiguities in the flow field decomposition.

Returning to the scheme developed in this thesis, each segment is sampled, using the method in step (2) of the multipass Hough technique (Section IV.3.1.2), and only the selected flow vectors are used for testing object hypotheses and computing the corresponding motion parameters. This sampling procedure considerably reduces the computation time. The desired number of vectors selected from each segment does not depend on the size of the segment. Therefore, all the distinct surfaces and independently moving objects, even the small ones, are appropriately represented, thus preventing the suppression of valuable data.

Only one iteration of the unit hemisphere sampling procedure, described in Section V.1.2.2, is used for testing object hypotheses, since for this purpose we mainly need the minimal values of the corresponding error functions, rather than the motion parameters which give these minima. Good approximations of these values has experimentally been shown to be obtained from one iteration which includes about 500 samples (see, for example, Figure 5.3).

Let us now denote by \mathcal{F} a set of segments $\{S_j : j = 1, \dots, n\}$ which are not necessarily adjacent. To determine the consistency of these segments with the same motion parameters, an error measure δ_j (which is the minimal value of the corresponding error function $\hat{\sigma}_j$, defined in equation (5.17)) is separately computed for each segment S_j , $j = 1, \dots, n$. In addition, optimal motion parameters, corresponding to all the vectors sampled from the segments in \mathcal{F} , are computed. Substituting these 'global' parameters and the flow vectors sampled from S_j in equation (5.17), a new error measure, δ'_j , is obtained for each segment S_j .

As we have already discussed in Section IV.3.2.2, when an object hypothesis is tested, the allowed level of δ'_j should be a function of p_j , which is the ratio of the sum of vector weights in S_j to the total sum of weights in \mathcal{F} (considering only the selected vectors). If p_j is close to 1, then δ'_j is expected to be close to δ_j . If, however, p_j is close to 0, then δ'_j is allowed to be significantly larger than δ_j , up to a given upper bound m_j (see below). Therefore, an object hypothesis is accepted if and only if, for each segment S_j in \mathcal{F} ,

$$\delta'_j \leq p_j^2 \delta_j + (1 - p_j^2) m_j. \quad (5.39)$$

Thus, the allowed level of δ'_j ranges from δ_j to m_j as p_j varies from 1 to 0.

Notice that the allowed deviation of δ'_j from the reference error δ_j in (5.39) is larger than the allowed deviation of σ'_j from σ_j in equation (4.14). As will be shown below, this can be justified by continuity considerations of the flow field; while the flow field is continuous across the boundary between two components merged into one segment, this is often not true for segments associated with one rigid object because the segments may correspond to surfaces which are spatially non-adjacent.

Now let C_{j_1} and C_{j_2} be two adjacent components, corresponding to surface patches on the same plane, which should be merged together to create a segment. Because the components correspond to spatially adjacent surfaces, the flow field is continuous across the boundary between the two components. If C_{j_1} is much larger than C_{j_2} , then we expect the flow field in C_{j_2} , because of its relatively small size and continuity considerations, to be fairly similar to the flow field predicted by the Ψ transformation computed for C_{j_1} . Thus, the Ψ transformation computed for $C_{j_1} \cup C_{j_2}$ should not be significantly influenced by C_{j_2} . Therefore, we expect the error measure σ'_{j_1} to be very close, in this case, to σ_{j_1} .

On the other hand, suppose that two segments, S_{j_1} and S_{j_2} , which are indeed

induced by the same rigid motion, are tested as corresponding to one object. Such segments may be non-adjacent, and even if they are adjacent, they often correspond to *spatially* non-adjacent surfaces, and then the flow field is usually not continuous across the boundary between them. In addition, motion parameters which are computed for S_{j_i} and are adjusted to the noise associated with this segment may be completely different from the correct parameters, because of the instability discussed in Section VI.1. Such incorrect parameters are not necessarily compatible with the flow field in S_{j_i} , since continuity of the flow field can not be assumed. In such a case, the optimal motion parameters for the combined pair of segments will have to be adjusted to S_{j_i} in addition to the noise in S_{j_i} , and δ'_{j_i} may be significantly larger than δ_{j_i} , even if p_{j_i} is fairly close to 1.

The upper bound m_j , $1 \leq j \leq n$, is determined to be

$$m_j = \max\{m_a \delta_j, m_b\}. \quad (5.40)$$

In a manner similar to the choice of the thresholds described in Section IV.3.2.2, m_a is chosen to be 1.5 and m_b represents a reasonable upper bound of the noise level. m_b should be smaller, by a factor of about $\sqrt{2}$, from l_b in equation (4.15). To see this, notice that for each flow vector the depth value Z can be adjusted to the associated noise to obtain a predicted flow vector which is the projection of the measured vector on the line l connecting its initial point to the FOE (see Figure 5.1). Thus, assuming that the computed motion parameters are correct, the deviation between the measured and predicted flow values is given only by the component of the noise which is perpendicular to l .

To summarize, in the first section of the current chapter we presented an algorithm for searching 3-D motion and structure values which minimize the deviation between the flow field predicted by these values and the given flow field. In the second section, this algorithm is combined with the segmentation results to form and test object hypotheses and to recover the 3-D information associated with each

of them. In Chapter VII this scheme, as well as difficulties in ambiguous situations, will experimentally be demonstrated.

CHAPTER VI

INHERENT AMBIGUITIES IN RECOVERING 3-D INFORMATION FROM A NOISY FLOW FIELD

In the last chapter we presented a technique for recovering the motion parameters and structure of independently moving objects. However, as has already been mentioned, two problems may arise due to the presence of noise in the flow field. First, motion parameters of the sensor or a moving object may be extremely difficult to estimate because there may exist a large set of significantly incorrect solutions which induce flow fields similar to the correct one. The second problem, which is closely related to the first one, is in the decomposition of the environment into independently moving objects. Two such objects may induce optical flows which are compatible with the same motion parameters and, hence, there is no way to refute the hypothesis that these flows are generated by one rigid object. These ambiguities are inherent in the sense that they are algorithm-independent.

In this chapter we will employ mathematical analysis to characterize situations where these problems are likely to arise. A few examples will demonstrate the conclusions. Constraints and parameters which can be recovered even in ambiguous situations, as well as appropriate modifications of the interpretation goals, will be presented.

VI.1 Ambiguity in Determining Motion Parameters of a Rigid Object

VI.1.1 Introduction

Given a flow field induced by a rigid object, it will be shown that in certain situations the flow field induced by totally incorrect motion and structure may be similar to the correct one. In the presence of noise which is statistically larger than

the difference between these flow fields, it may be impossible to obtain reasonably accurate estimates of the motion parameters. The influence of certain factors on this ambiguity will be analyzed.

Let us start by examining the cases of pure rotation and pure translation. In a purely rotational motion the flow field is represented by equations (3.7a,c) which can be rewritten as

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} -xy \\ -1 - y^2 \end{pmatrix} \Omega_X + \begin{pmatrix} 1 + x^2 \\ xy \end{pmatrix} \Omega_Y + \begin{pmatrix} -y \\ x \end{pmatrix} \Omega_Z. \quad (6.1)$$

Thus, each rotation parameter has a distinct signature in the flow field, and in most cases it can reliably be recovered.

In a purely translational motion, the direction of translation is represented by the focus of expansion (FOE) which, in this case, is the intersection of the straight lines corresponding to the flow vectors. Usually, the FOE can be robustly recovered [LAW82,84], unless the absolute value of the translation is small relative to the distance of the surface from the observer, in which case the flow vectors are small and the determination of the corresponding intersection is sensitive to noise.

In the general case, an ambiguity in determining the motion parameters becomes a much more severe problem because rotation and translation may induce similar flows. To demonstrate this, let us examine the case of a planar surface which is parallel to the image plane, and denote by d the distance of this plane from the camera. We wish now to compare the flow field generated by a purely translational motion $(P_X, P_Y, 0)$ to the flow field generated by the purely rotational motion $(-P_Y/d, P_X/d, 0)$. The flow field in the first case is

$$\begin{pmatrix} \alpha_T \\ \beta_T \end{pmatrix} = 1/d \begin{pmatrix} P_X \\ P_Y \end{pmatrix}, \quad (6.2)$$

while in the second case the flow is given by

$$\begin{pmatrix} \alpha_R \\ \beta_R \end{pmatrix} = 1/d \begin{pmatrix} xyP_Y + (1+x^2)P_X \\ (1+y^2)P_Y + xyP_X \end{pmatrix}. \quad (6.3)$$

Hence,

$$\begin{pmatrix} \alpha_R \\ \beta_R \end{pmatrix} - \begin{pmatrix} \alpha_T \\ \beta_T \end{pmatrix} = 1/d \begin{pmatrix} xyP_Y + x^2P_X \\ y^2P_Y + xyP_X \end{pmatrix}. \quad (6.4)$$

If the field of view (FOV) is small, then the second-order terms of the image coordinates, x and y , are small, and the difference between the flow fields is small as well (see Figure 6.1). In such a case it may be very difficult, in the presence of noise, to distinguish between these fields and to determine whether the motion is purely translational, purely rotational or a combination of both. Note, however, that if the FOV is large, then the second-order components of the flow field are relatively large and, therefore, the ambiguity is more likely to be resolved.

Ambiguity in determining the motion parameters is affected not only by the FOV, but also by variations in the surface structure. This can be concluded from the work of Rieger and Lawton [RIE83], who examined the case of large discontinuities in the depth map. In this case the differences between flow vectors near occlusion boundaries are oriented towards the FOE of the translational field and, therefore, the ambiguity in distinguishing between the translational and rotational components can be resolved.

In addition, it has been experimentally shown [PRA80, LON81, FAN83b] that the accuracy of the estimated 3-D motion parameters is improved when the translational component of the motion is large relative to the distance of the object from the sensor. The results can also be improved by using a large number of flow vectors [ROA80, TSA84], and by increasing the size of the region containing these vectors [PRA80, FAN83a].

To summarize, ambiguity (or, using another term, instability) in determining

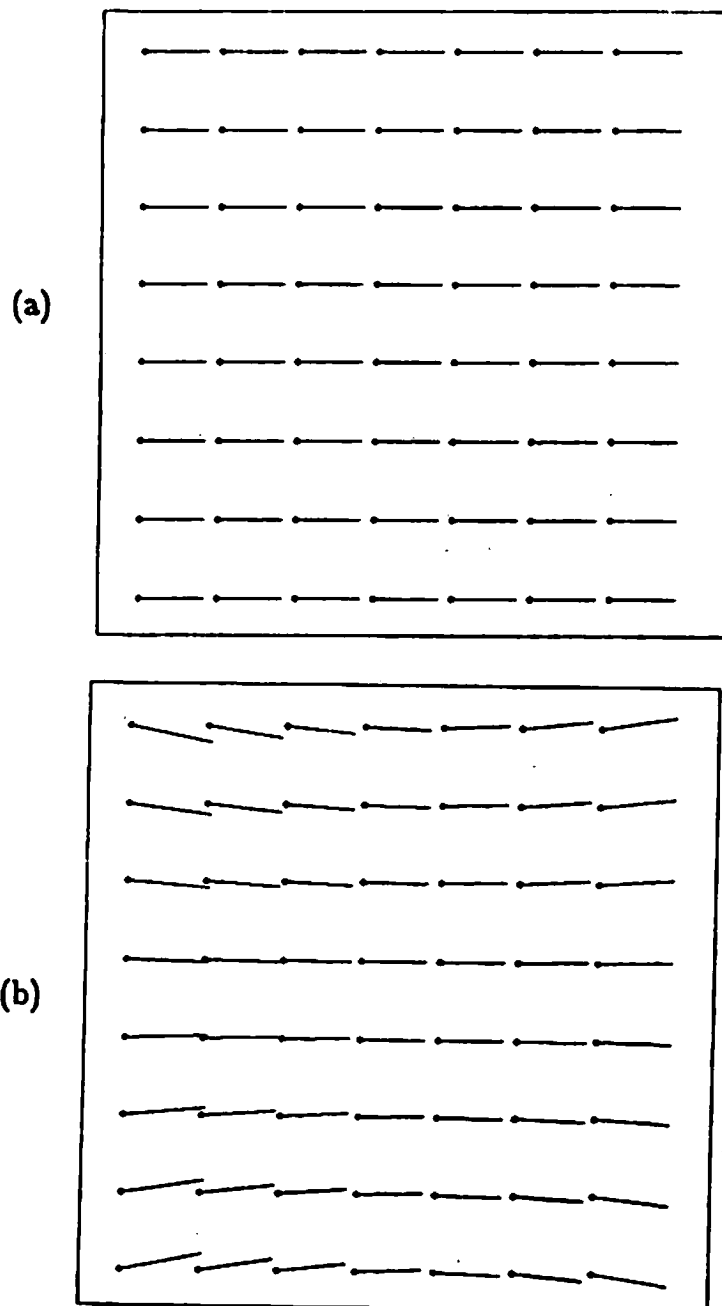


Figure 6.1: The flow field in (a) is purely translational, whereas the flow field in (b) is purely rotational. In both cases the field of view is 60° . Note the similarity of the flow fields in the central portion of the image, where the values of x and y are small. On the other hand, the difference between the flow fields is not negligible near the boundary of the image.

3-D information can be expected if the FOV, the depth variations, and the ratio of the translation to the distance of the object from the camera are all small. In addition, local techniques, in which only the information in a small region of the image is utilized, are more sensitive to noise. In the next section we will employ a mathematical analysis in order to show that these conditions, as well as other conditions, generally contribute to ambiguity in recovering 3-D motion and structure. Combined together these are sufficient conditions for such instability. Note that the error analyses existing in the literature are experimental and algorithm-dependent, whereas here we develop a mathematical and algorithm-independent analysis.

VI.1.2 Mathematical analysis

VI.1.2.1 A planar surface. In this section we restrict ourselves to flow fields induced by a rigid motion of a planar surface given by equation (4.1): $k_X X + k_Y Y + k_Z Z = 1$. Since

$$Z = \frac{1}{k_Z} - \frac{k_X}{k_Z} X - \frac{k_Y}{k_Z} Y, \quad (6.5)$$

$1/k_Z$, denoted by d , is the distance from the camera to the surface along the line of sight, and the values $-k_X/k_Z$ and $-k_Y/k_Z$, denoted, respectively, by s_X and s_Y , represent the slopes of the surface relative to the image plane.

In the following analysis, the image is assumed to be square and the FOV is defined to be the visual angle corresponding to each side of the image, which, therefore, is $2 \tan(\text{fov}/2)$ focal units. The region \mathcal{R} corresponding to the perspective projection of the planar patch on the image is contained in a square for which the proportion between its side and the image side is γ , where $0 < \gamma \leq 1$. For simplicity, the center of this square is assumed to be $(0, 0)$, but even if this is not the case, results similar to those which we will obtain in this section can be derived by expanding the flow equations around this center. γ , which we shall call the *locality factor*, will be small if the ratio of the object size to its distance from the camera is

small relative to the image size (in focal units), or if a technique based on a local analysis of the flow field is employed.

The flow field generated by the motion parameters $\{\underline{T}, \underline{\Omega}\}$ can be described by a Ψ transformation (equations (4.3)) with the coefficients given in equations (4.4). Let \underline{k} denote the vector (k_X, k_Y, k_Z) and r denote the absolute value of the translation, which is assumed to be non-zero. Using the normalization $\underline{U} = \underline{T}/r$ and $\underline{l} = r\underline{k}$, the coefficients of the Ψ transformation can be rewritten as:

$$a_1 = \Omega_Y + l_Z U_X, \quad (6.6a)$$

$$a_2 = l_X U_X - l_Z U_Z, \quad (6.6b)$$

$$a_3 = -\Omega_Z + l_Y U_X, \quad (6.6c)$$

$$a_4 = -\Omega_X + l_Z U_Y, \quad (6.6d)$$

$$a_5 = \Omega_Z + l_X U_Y, \quad (6.6e)$$

$$a_6 = l_Y U_Y - l_Z U_Z, \quad (6.6f)$$

$$a_7 = \Omega_Y - l_X U_Z \quad (6.6g)$$

and

$$a_8 = -\Omega_X - l_Y U_Z. \quad (6.6h)$$

Employing the constraint $U_X^2 + U_Y^2 + U_Z^2 = 1$, we obtain 9 non-linear equations with 9 unknowns. Usually, these equations have 2 sets of solutions [TSA84, WAX83], where, of course, only one of them is the correct one. Let us now denote by \hat{U} , $\hat{\Omega}$ and \hat{l} estimates of the motion and structure parameter values. We will show that in some situations, vectors \hat{U} significantly different from the corresponding values in each of the two exact solutions produce flow fields which are very similar to the correct one, if combined with appropriate values of $\hat{\Omega}$ and \hat{l} .

The basic idea is that if the region \mathcal{R} is rather small (in focal units) and l_X, l_Y are not large, then, based on equations (4.3) and (6.6g,h), a change in \underline{U} has only

a small effect on the second-order components of the flow field. Therefore, given an arbitrary \hat{U} , we concentrate on the lower-order components, and try to find $\hat{\Omega}$ and \hat{I} such that the correct values of the coefficients a_1, \dots, a_6 will be maintained. This will lead to hypothesized values of the motion and structure parameters. We can substitute these parameters in the expressions for a_7 and a_8 and measure the deviation of the obtained values from the correct ones. These deviations determine what we shall call the *error field*, that is, the discrepancy between the correct flow field and that predicted from the hypothesized parameters. Note that this error field is actually an upper bound to the 'minimal' error field which could be obtained for the same \hat{U} by optimizing the values of $\hat{\Omega}$ and \hat{I} across all the eight coefficients instead of a_1, \dots, a_6 .

Given a vector \hat{U} , the equations (6.6a) to (6.6f) associated with the coefficients a_1, \dots, a_6 produce six linear equations with six unknowns: $\hat{\Omega}_X, \hat{\Omega}_Y, \hat{\Omega}_Z, \hat{I}_X, \hat{I}_Y$ and \hat{I}_Z . These equations can be represented by

$$F\mathbf{u} = \mathbf{a}, \quad (6.7a)$$

where

$$F = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & \hat{U}_Y \\ 0 & 1 & 0 & 0 & 0 & \hat{U}_X \\ 0 & 0 & -1 & 0 & \hat{U}_X & 0 \\ 0 & 0 & 1 & \hat{U}_Y & 0 & 0 \\ 0 & 0 & 0 & \hat{U}_X & 0 & -\hat{U}_Z \\ 0 & 0 & 0 & 0 & \hat{U}_Y & -\hat{U}_Z \end{pmatrix}, \quad (6.7b)$$

$$\underline{u} = \begin{pmatrix} \hat{\Omega}_X \\ \hat{\Omega}_Y \\ \hat{\Omega}_Z \\ \hat{i}_X \\ \hat{i}_Y \\ \hat{i}_Z \end{pmatrix} \quad \text{and} \quad \underline{a} = \begin{pmatrix} a_4 \\ a_1 \\ a_3 \\ a_5 \\ a_2 \\ a_6 \end{pmatrix}. \quad (6.7c,d)$$

A unique solution is guaranteed if the determinant of F , denoted by D , is non-zero. As can easily be verified

$$D = \hat{U}_Z(\hat{U}_X^2 + \hat{U}_Y^2). \quad (6.8)$$

Thus, if the translation vector is not exactly perpendicular ($\hat{U}_X = \hat{U}_Y = 0$), or parallel ($\hat{U}_Z = 0$), to the image plane, then there exist $\hat{\Omega}$ and \hat{i} which keep the correct values of a_1, \dots, a_6 . Note, however, that this solution may still be not physically realizable if the depth constraint $Z > 0$ is not satisfied by one of the depth values predicted by the vector \hat{i} . We will return to this problem in the end of this section.

The solution of the equations set (6.7) is, if $D \neq 0$,

$$\underline{u} = F^{-1}\underline{a}. \quad (6.9)$$

To find F^{-1} we employ the decomposition technique in [RAL65]. We represent F by

$$F = \begin{pmatrix} F_1 & F_2 \\ F_3 & F_4 \end{pmatrix}, \quad (6.10)$$

where each of F_i ($i = 1, 2, 3, 4$) is a 3×3 matrix. Then

$$F^{-1} = \begin{pmatrix} G_1 & G_2 \\ G_3 & G_4 \end{pmatrix}, \quad (6.11)$$

where

$$G_1 = F_1^{-1} - F_1^{-1}F_2G_3, \quad (6.12a)$$

$$G_2 = -F_1^{-1}F_2G_4, \quad (6.12b)$$

$$G_3 = -G_4F_3F_1^{-1} \quad (6.12c)$$

and

$$G_4 = (F_4 - F_3F_1^{-1}F_2)^{-1}. \quad (6.12d)$$

Necessary and sufficient conditions for applying this technique is that F_1 and $F_4 - F_3F_1^{-1}F_2$ are non-singular matrices. These conditions are satisfied if F is non-singular.

Employing the decomposition technique, we can now obtain

$$F^{-1} = \frac{1}{D} \begin{pmatrix} -D & 0 & \hat{U}_X \hat{U}_Y^2 & \hat{U}_X \hat{U}_Y^2 & -\hat{U}_Y^3 & -\hat{U}_X^2 \hat{U}_Y \\ 0 & D & -\hat{U}_X^2 \hat{U}_Y & -\hat{U}_X^2 \hat{U}_Y & \hat{U}_X \hat{U}_Y^2 & \hat{U}_X^3 \\ 0 & 0 & -\hat{U}_Y^2 \hat{U}_Z & \hat{U}_X^2 \hat{U}_Z & -\hat{U}_X \hat{U}_Y \hat{U}_Z & \hat{U}_X \hat{U}_Y \hat{U}_Z \\ 0 & 0 & \hat{U}_Y \hat{U}_Z & \hat{U}_Y \hat{U}_Z & \hat{U}_X \hat{U}_Z & -\hat{U}_X \hat{U}_Z \\ 0 & 0 & \hat{U}_X \hat{U}_Z & \hat{U}_X \hat{U}_Z & -\hat{U}_Y \hat{U}_Z & \hat{U}_Y \hat{U}_Z \\ 0 & 0 & \hat{U}_X \hat{U}_Y & \hat{U}_X \hat{U}_Y & -\hat{U}_Y^2 & -\hat{U}_X^2 \end{pmatrix}. \quad (6.13)$$

Substituting the definitions of \underline{u} and \underline{a} in (6.9),

$$\underline{u} = \begin{pmatrix} \hat{\Omega}_X \\ \hat{\Omega}_Y \\ \hat{\Omega}_Z \\ \hat{l}_X \\ \hat{l}_Y \\ \hat{l}_Z \end{pmatrix} = F^{-1} \begin{pmatrix} a_4 \\ a_1 \\ a_3 \\ a_5 \\ a_2 \\ a_6 \end{pmatrix} = F^{-1} \begin{pmatrix} -\Omega_X + U_Y l_Z \\ \Omega_Y + U_X l_Z \\ -\Omega_Z + U_X l_Y \\ \Omega_Z + U_Y l_X \\ U_X l_X - U_Z l_Z \\ U_Y l_Y - U_Z l_Z \end{pmatrix}, \quad (6.14)$$

where \underline{U} , $\underline{\Omega}$ and \underline{l} are the correct values of the motion and structure parameters. Multiplying F^{-1} by \underline{a} , and using the notations $\epsilon_{XY} = \hat{U}_X U_Y - \hat{U}_Y U_X$, $\epsilon_{XZ} =$

$\hat{U}_X U_Z - \hat{U}_Z U_X$ and $\epsilon_{YZ} = \hat{U}_Y U_Z - \hat{U}_Z U_Y$, we can derive:

$$\hat{\Omega}_X = \Omega_X + \frac{\hat{U}_Y \epsilon_{XY} (\hat{U}_Y l_X - \hat{U}_X l_Y)}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)} + \frac{\epsilon_{YZ} l_Z}{\hat{U}_Z}, \quad (6.15a)$$

$$\hat{\Omega}_Y = \Omega_Y + \frac{\hat{U}_X \epsilon_{XY} (\hat{U}_X l_Y - \hat{U}_Y l_X)}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)} - \frac{\epsilon_{XZ} l_Z}{\hat{U}_Z}, \quad (6.15b)$$

$$\hat{\Omega}_Z = \Omega_Z + \frac{\epsilon_{XY} (\hat{U}_X l_X + \hat{U}_Y l_Y)}{\hat{U}_X^2 + \hat{U}_Y^2}, \quad (6.15c)$$

$$\hat{i}_X = \frac{(\hat{U}_X U_X + \hat{U}_Y U_Y) l_X - \epsilon_{XY} l_Y}{\hat{U}_X^2 + \hat{U}_Y^2}, \quad (6.15d)$$

$$\hat{i}_Y = \frac{(\hat{U}_X U_X + \hat{U}_Y U_Y) l_Y + \epsilon_{XY} l_X}{\hat{U}_X^2 + \hat{U}_Y^2} \quad (6.15e)$$

and

$$\hat{i}_Z = \frac{U_Z}{\hat{U}_Z} l_Z + \frac{\epsilon_{XY} (\hat{U}_Y l_X - \hat{U}_X l_Y)}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)}. \quad (6.15f)$$

The error field corresponding to the values \hat{U} , $\hat{\Omega}$ and \hat{i} is the deviation between the flow field predicted by these values and the correct field. Referring to equations (4.3), its value at the (x, y) pixel is:

$$\begin{pmatrix} \Delta \alpha_{\Phi} \\ \Delta \beta_{\Phi} \end{pmatrix} = \begin{pmatrix} \Delta a_7 x^2 + \Delta a_8 xy \\ \Delta a_7 xy + \Delta a_8 y^2 \end{pmatrix}, \quad (6.16)$$

where Δa_7 and Δa_8 are the errors induced in the coefficients a_7 and a_8 . Recall now that the distance, denoted by d , from the camera to the surface along the line of sight is $1/k_Z$, and the slopes, denoted by s_X and s_Y , of the surface relative to the image plane are, respectively, $-k_X/k_Z$ and $-k_Y/k_Z$. Hence,

$$r/d = l_Z, \quad s_X = -l_X/l_Z \quad \text{and} \quad s_Y = -l_Y/l_Z. \quad (6.17a,b,c)$$

Thus, using equations (6.6g,h) and (6.15), we can obtain:

$$\begin{aligned}
 \Delta a_7 &= (\hat{\Omega}_Y - \hat{U}_Z \hat{l}_X) - (\Omega_Y - U_Z l_X) \\
 &= \frac{\hat{U}_X \epsilon_{XZ} + \hat{U}_Y \epsilon_{YZ}}{\hat{U}_X^2 + \hat{U}_Y^2} l_X + \frac{-\hat{U}_X \hat{U}_Y l_X + (1 - \hat{U}_Y^2) l_Y}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)} \epsilon_{XY} - \frac{\epsilon_{XZ} l_Z}{\hat{U}_Z} \\
 &= \frac{-r}{d} \left[\frac{\hat{U}_X \epsilon_{XZ} + \hat{U}_Y \epsilon_{YZ}}{\hat{U}_X^2 + \hat{U}_Y^2} s_X + \frac{-\hat{U}_X \hat{U}_Y s_X + (1 - \hat{U}_Y^2) s_Y}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)} \epsilon_{XY} + \frac{\epsilon_{XZ}}{\hat{U}_Z} \right] \quad (6.18a)
 \end{aligned}$$

and

$$\begin{aligned}
 \Delta a_8 &= (-\hat{\Omega}_X - \hat{U}_Z \hat{l}_Y) - (-\Omega_X - U_Z l_Y) \\
 &= \frac{\hat{U}_X \epsilon_{XZ} + \hat{U}_Y \epsilon_{YZ}}{\hat{U}_X^2 + \hat{U}_Y^2} l_Y + \frac{(\hat{U}_X^2 - 1) l_X + \hat{U}_X \hat{U}_Y l_Y}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)} \epsilon_{XY} - \frac{\epsilon_{YZ} l_Z}{\hat{U}_Z} \\
 &= \frac{-r}{d} \left[\frac{\hat{U}_X \epsilon_{XZ} + \hat{U}_Y \epsilon_{YZ}}{\hat{U}_X^2 + \hat{U}_Y^2} s_Y + \frac{(\hat{U}_X^2 - 1) s_X + \hat{U}_X \hat{U}_Y s_Y}{\hat{U}_Z (\hat{U}_X^2 + \hat{U}_Y^2)} \epsilon_{XY} + \frac{\epsilon_{YZ}}{\hat{U}_Z} \right]. \quad (6.18b)
 \end{aligned}$$

Therefore, if the translation is not large relative to the distance of the surface from the camera (i.e., r/d is small), and the surface is not very slanted (i.e., s_X and s_Y are small), then Δa_7 and Δa_8 are not large for vectors \hat{U} in a relatively large neighborhood of U . If, in addition, the region \mathcal{R} is small, then x and y are small as well, and thus the deviation $(\Delta \alpha_{\Phi}, \Delta \beta_{\Phi})$ is very small. Under these conditions, any error surface corresponding to possible values of \hat{U} can be expected to be very flat around the correct solution U and, therefore, the process of recovering 3-D motion and structure will be very unstable and sensitive to noise.

To determine more precisely how these instability problems depend on factors related to the camera and the flow field, we will normalize the error field given in equation (6.16) to the noise level and thus, for each vector \hat{U} , obtain a measure of a signal-to-noise ratio (SNR). Note that the error field is used as a 'signal' measure, since high values of this field reduce ambiguity. The probability that the vector \hat{U}

will be selected as the correct solution is a decreasing function of the corresponding SNR values. Hence, if these values are small for a large set of translation axes, then instability in recovering 3-D information can be expected.

The noise in the flow field is assumed to be additive, its expectation is 0 and its standard deviation, in focal units, is

$$\sigma_f = \sigma_p \frac{2 \tan(\text{fov}/2)}{N}, \quad (6.19)$$

where the image contains $N \times N$ pixels and σ_p is the standard deviation in pixel units. To obtain an SNR measure we divide the error field by the square root of the sum of the second moments of the noise samples, which are assumed to be independent, in both axes. Thus, for each pixel (x, y) where a flow vector is defined,

$$\text{snr}(x, y) = \frac{\sqrt{(\Delta a_7 x^2 + \Delta a_8 xy)^2 + (\Delta a_7 xy + \Delta a_8 y^2)^2}}{2\sqrt{2}\sigma_p \tan(\text{fov}/2)/N}. \quad (6.20)$$

Employing the definition, in the beginning of this section, of the locality factor γ , x and y satisfy the inequalities:

$$|x| \leq \gamma \tan(\text{fov}/2), \quad |y| \leq \gamma \tan(\text{fov}/2), \quad (6.21a,b)$$

and, therefore,

$$\text{snr}(x, y) \leq \frac{N\gamma^2 \tan(\text{fov}/2)(|\Delta a_7| + |\Delta a_8|)}{2\sigma_p}. \quad (6.22)$$

Even when the SNR values are small, it may be possible to successfully recover the desired parameters, if there exist many flow vectors and the noise samples associated with them are independent. However, in many cases, especially if γ and N are small and the flow field is sparse, the number of flow vectors is small. In addition, if the flow field is dense, then usually the noise samples in neighboring pixels are highly correlated. This is the case, for example, if the noise is induced by a round-off error.

Before we can conclude this section we still have to deal with the depth constraint. This constraint is satisfied if, for any pixel (x, y) in the region \mathcal{R} , the estimated value of τ/Z is positive, that is, the following inequality, derived from (4.1), holds:

$$\hat{l}_z + \hat{l}_x x + \hat{l}_y y > 0. \quad (6.23)$$

Substituting \hat{l}_x , \hat{l}_y and \hat{l}_z with the corresponding expressions in equations (6.15) and dividing by l_z , we obtain the equivalent constraint:

$$\frac{U_z}{\hat{U}_z} + \frac{\epsilon_{XY}(\hat{U}_x s_Y - \hat{U}_y s_X)}{\hat{U}_z(\hat{U}_x^2 + \hat{U}_y^2)} - \frac{(\hat{U}_x U_x + \hat{U}_y U_y)(s_X x + s_Y y) + \epsilon_{XY}(s_X y - s_Y x)}{\hat{U}_x^2 + \hat{U}_y^2} > 0. \quad (6.24)$$

If the slopes s_X and s_Y are small and the region \mathcal{R} is small, then, usually, the second and third terms in (6.24) are small and the inequality is satisfied by vectors \hat{U} in a large neighborhood of \underline{U} . Note that anyway these conditions are among those already specified as contributing to ambiguity.

To conclude, the following conditions contribute to ambiguity in recovering 3-D motion and structure parameters:

- The FOV is small.
- The locality factor γ is small.
- The planar surface is at most moderately slanted.
- The object is far away.
- The absolute value of the translation is small.
- The resolution of the image is coarse.
- The noise level (in pixels) is high.
- The flow field is sparse.

- The noise values in adjacent flow vectors are highly correlated.

VI.1.2.2 The general case. Referring to equation (4.2), the 'reciprocal depth' map, $1/Z$, can be generally represented by

$$\frac{1}{Z(x, y)} = k_Z + k_X x + k_Y y + \xi(x, y) \quad (6.25)$$

where $\xi(x, y)$ is the difference between $1/Z$ and the approximating linear function $k_Z + k_X x + k_Y y$. Using this representation and the normalization $\lambda = r\xi$, we can rewrite the flow field equations (3.5):

$$\alpha(x, y) = \alpha_{\Psi}(x, y) + (U_X - xU_Z)\lambda(x, y) \quad (6.26a)$$

and

$$\beta(x, y) = \beta_{\Psi}(x, y) + (U_Y - yU_Z)\lambda(x, y), \quad (6.26b)$$

where $(\alpha_{\Psi}, \beta_{\Psi})$ is the Ψ transformation corresponding to the planar surface $k_X X + k_Y Y + k_Z Z = 1$.

Given \underline{U} , we can usually choose rotation parameter values $\hat{\Omega}$, and normalized plane parameter values \hat{l} , which maintain the correct zeroth and first order components of the Ψ transformation. If, in addition, for each flow vector we choose the value of λ as the correct one, then the error field corresponding to these motion and structure parameters is

$$\begin{pmatrix} \Delta\alpha_{\Psi} \\ \Delta\beta_{\Psi} \end{pmatrix} + \begin{pmatrix} \Delta U_X - x\Delta U_Z \\ \Delta U_Y - y\Delta U_Z \end{pmatrix} \lambda(x, y), \quad (6.27)$$

where $\Delta\alpha_{\Psi}$ and $\Delta\beta_{\Psi}$ are the errors associated with the planar surface (equation (6.16)) and $(\Delta U_X, \Delta U_Y, \Delta U_Z)$ is the error in the normalized translation vector. Therefore, we can expect instability in determining the 3-D motion and structure if, in addition to the conditions associated with planar surfaces, the function $1/Z$ can

be reasonably approximated by a linear function, i.e., $\lambda(x, y)$ is small. Note that this condition means that the environmental surface can be relatively approximated by a planar surface, that is, the distance between the two surfaces is small relative to the distance from the sensor to the real surface. This approximation can usually be improved as the FOV or the locality factor γ are reduced, unless there is a significant discontinuity in the depth map.

VI.1.3 Examples

In this section we demonstrate the influence of three parameters on the degree of instability in recovering 3-D information from the flow field induced by a rigid motion of a planar surface. These parameters are the locality factor γ , the ratio r/d of the translation magnitude r to the distance d from the camera to the surface along the line of sight, and the slope s_X of the planar surface. The demonstration is based on several examples, where in each example a dense flow field is simulated. The technique presented in Chapter V for computing the error function $\hat{\sigma}_X(\underline{U})$ (equation (5.17)), which is an approximation to the average value of the minimal error field corresponding to \underline{U} , is then employed. The sharpness of this function around the correct value determines the sensitivity to noise in estimating the translation axis and, therefore, also in estimating the rotation parameters and the environmental structure.

In all the examples the FOV is 60° , the number of pixels is 128×128 , the camera translation is $\underline{T} = (0, 0, 10)$ and the rotation is $(0, 0, 0)$. In the first three cases the surface, defined by the equation $Z = 100$, is parallel to the image plane. The locality factor γ , on the other hand, is different in each of these experiments: 1 in the first, $1/2$ in the second and $1/4$ in the third. The contour maps in Figures 6.2, 6.3 and 6.4 show the drastic change in the sharpness of the corresponding error functions. Recall that these functions are defined on the unit hemisphere $\mathcal{X} = \{\underline{U} : |\underline{U}| = 1, U_Z \geq 0\}$. The spherical coordinates (ϕ, θ) , employed in

equation (5.27) for representing this hemisphere, are used in the figures as polar coordinates. The contours are labeled by the corresponding error values, given in pixels, and the correct solution is marked by a black dot.

In examples 4 to 7 we choose again $\gamma = 1$, but the planar surface is varied. In examples 4 and 5 the surface is still parallel to the image plane, but its distance from the camera is 200 in example 4, and 400 in example 5. Thus, the influence of the relative translation magnitude r/d is demonstrated by examples 1, 4 and 5, in which r/d is 0.1, 0.05 and 0.025, respectively. The results in Figures 6.2, 6.5 and 6.6 clearly show that smaller values of r/d are associated with higher levels of ambiguity.

In examples 6 and 7 the distance d is kept at 100, but the surfaces, defined respectively by $Z = 100 + 0.414X$ and $Z = 100 + X$, are slanted: 22.5° in example 6 and 45° in example 7. Figures 6.2, 6.7 and 6.8 show that the error function becomes sharper as the surface becomes more slanted. The basic reason for this relation is the depth variation associated with slanted surfaces. This variation helps in resolving the ambiguity in distinguishing between the translational and rotational components of the flow field, since the translational component is affected by variations in depth, while the rotational component is independent of the depth values.

Note the second accurate solution in Figures 6.7 and 6.8, which corresponds, according to equations developed in [WAX83], to a situation where the surface is non-slanted but the translatory motion is not along the line of sight; instead, the motion deviates by 22.5° and 45° , respectively, from this line. The relative translation along the line of sight, that is, T_z/d , is still 0.1 for these alternative solutions. Since in these cases the translational motion along the X -axis is non-zero, the ratio r/d is larger than 0.1, specifically 0.1082 in experiment 6, and 0.1414 in experiment 7.

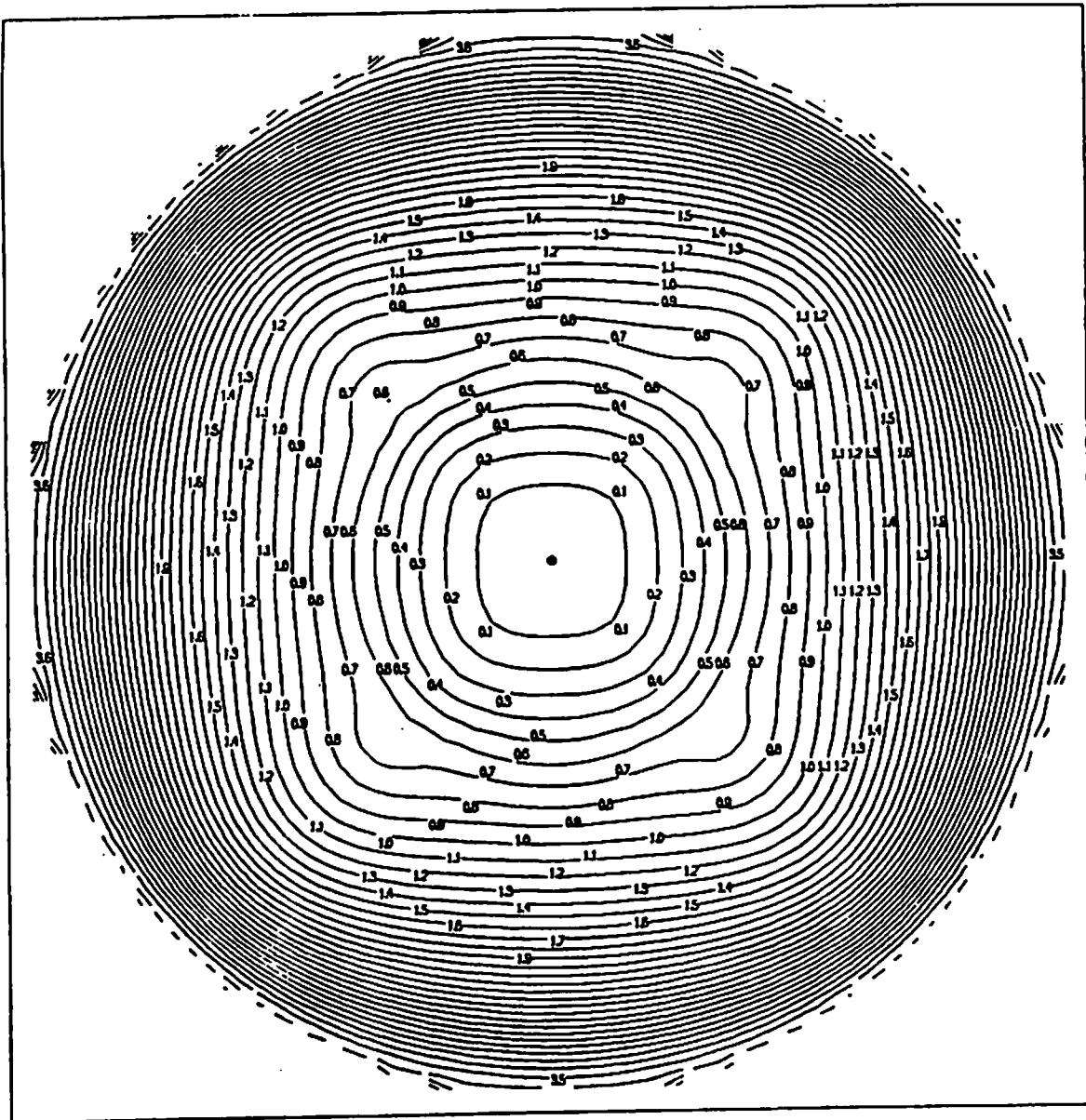


Figure 6.2: The error function in example 1. The surface is non-slanted, $r/d = 0.1$, and $\gamma = 1$.

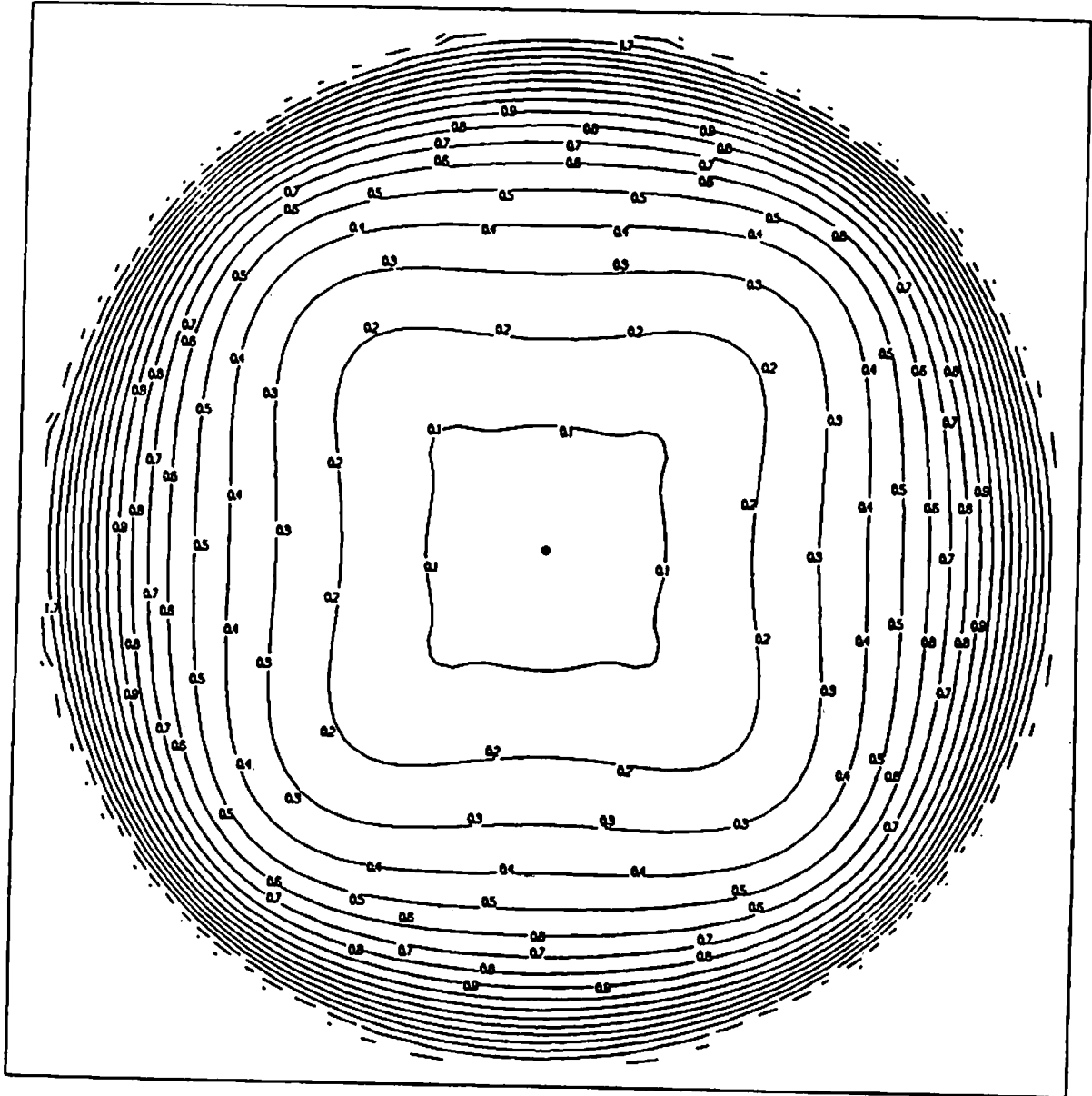


Figure 6.3: The error function in example 2. The surface is non-slanted, $r/d = 0.1$, and $\gamma = 0.5$.

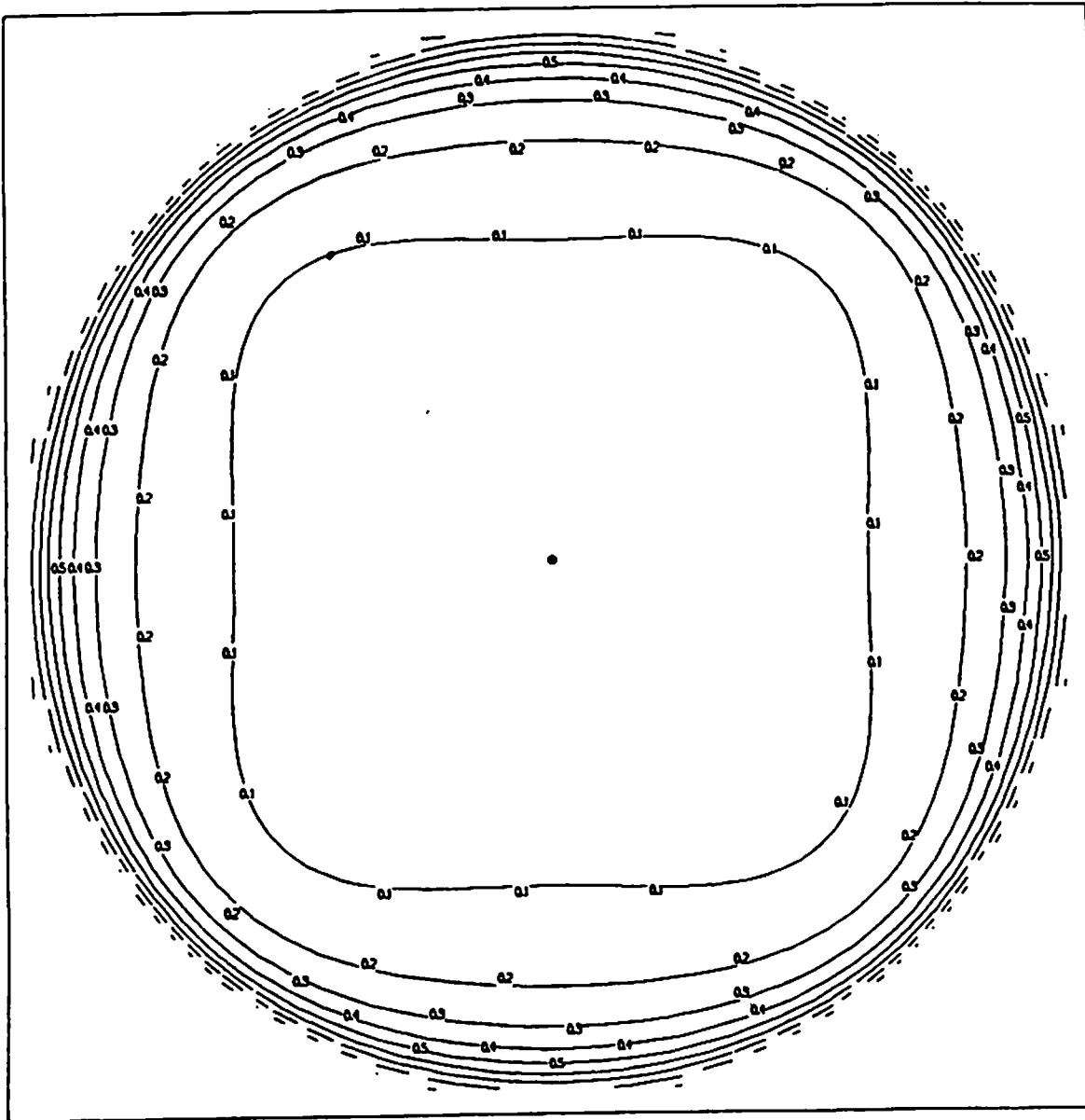


Figure 6.4: The error function in example 3. The surface is non-slanted, $r/d = 0.1$, and $\gamma = 0.25$.

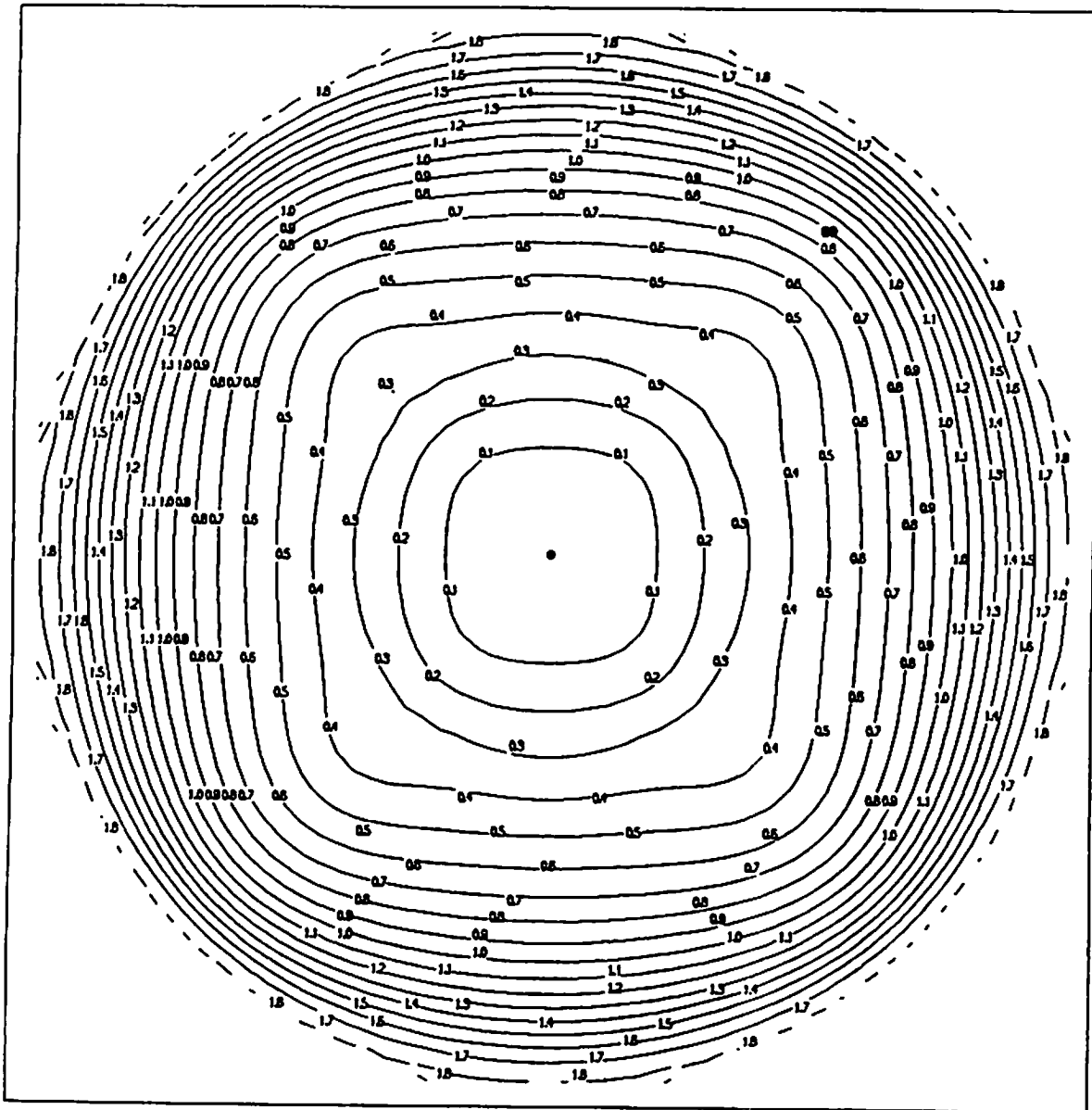


Figure 6.5: The error function in example 4. The surface is non-slanted, $r/d = 0.05$, and $\gamma = 1$.

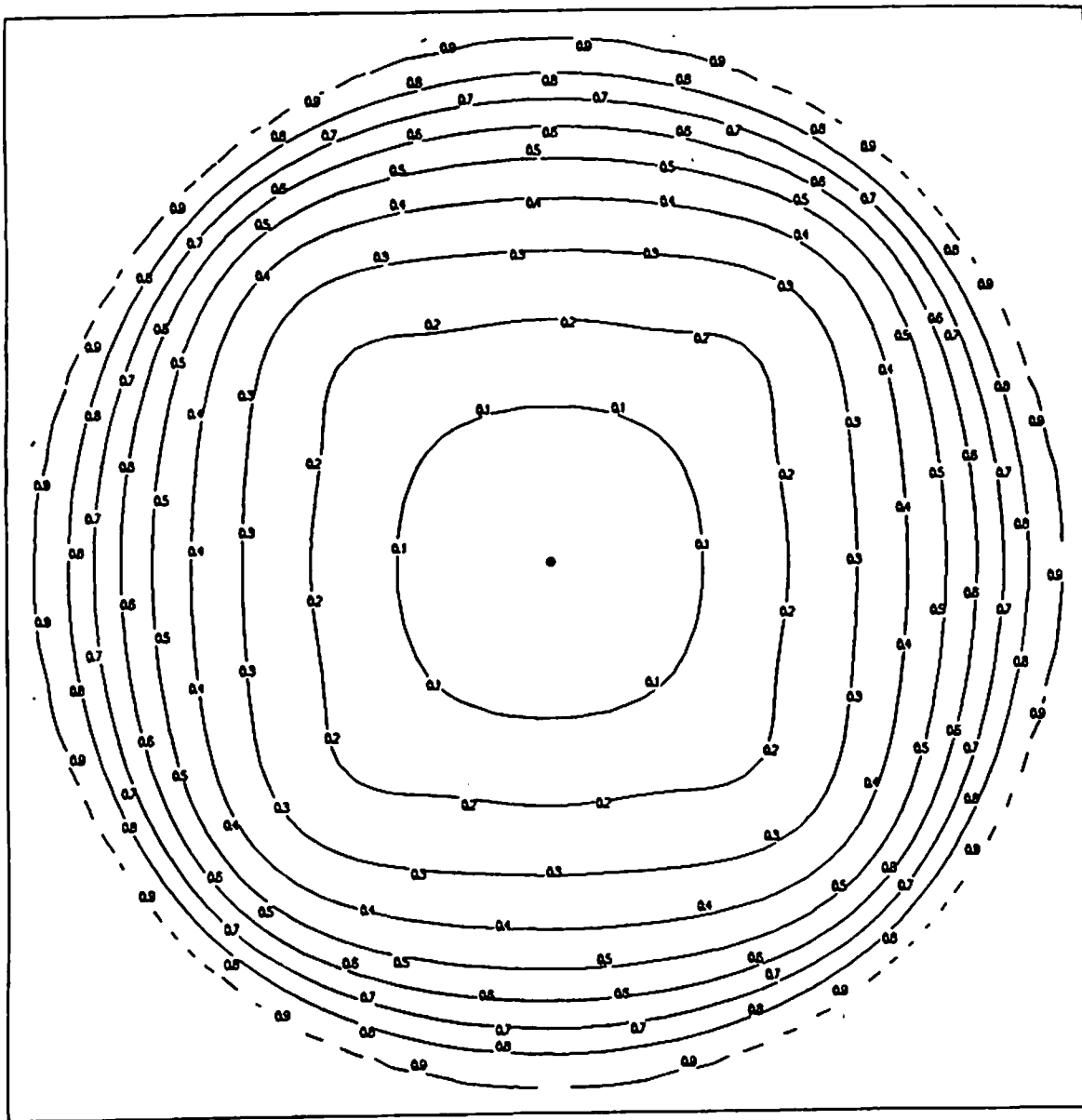


Figure 6.6: The error function in example 5. The surface is non-slanted, $r/d = 0.025$, and $\gamma = 1$.

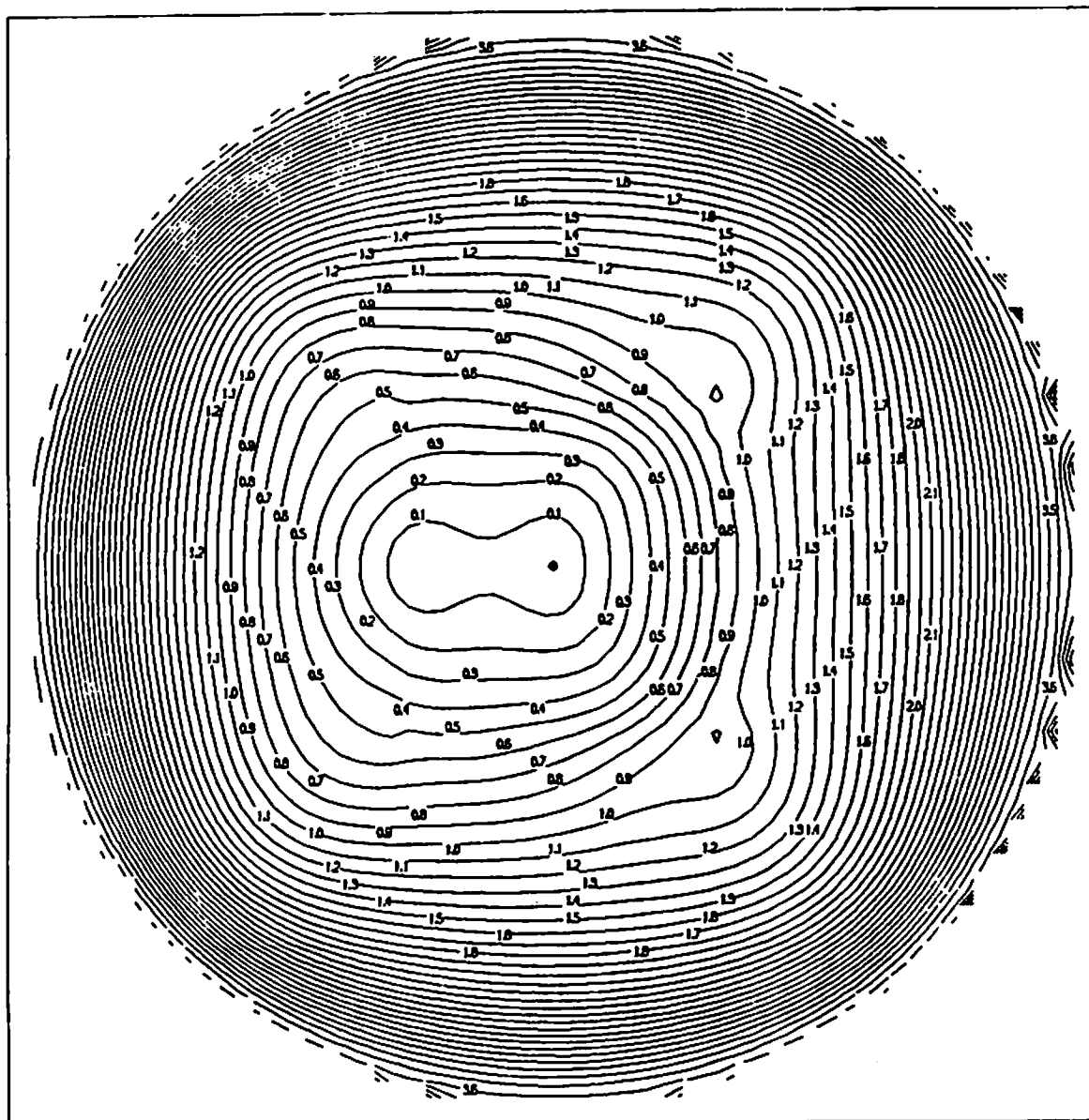


Figure 6.7: The error function in example 6. The surface is slanted (22.5°), $r/d = 0.1$, and $\gamma = 1$. Note the second solution which corresponds to a non-slanted surface and a translation not along the line of sight (the angle between the translation vector and the line of sight is 22.5°).

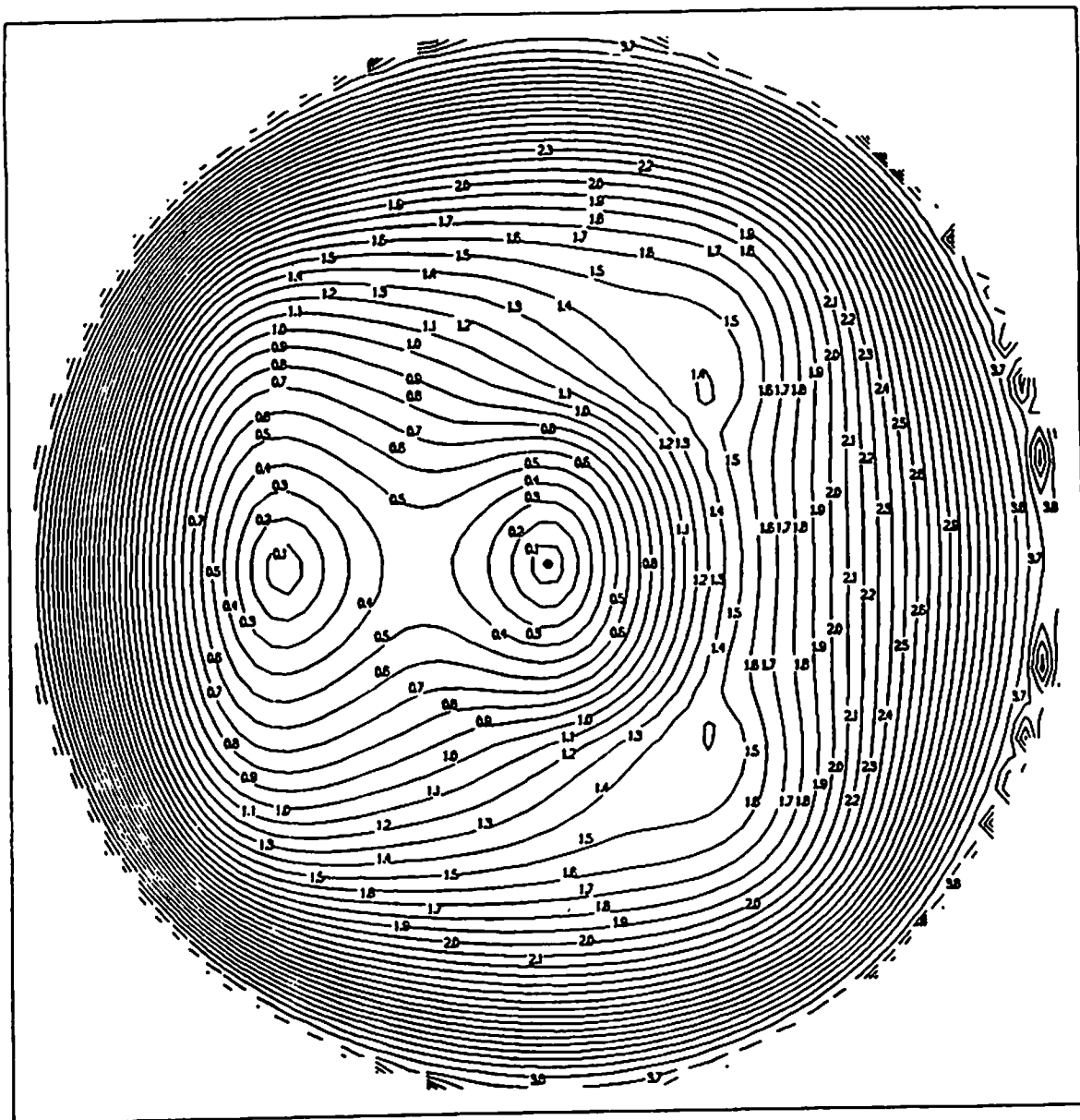


Figure 6.8: The error function in example 7. The surface is slanted (45°), $r/d = 0.1$, and $\gamma = 1$. The second solution corresponds again to a non-slanted surface and a translation not along the line of sight (this time, the angle between the translation vector and the line of sight is 45°).

The solution in experiment 1 and the alternative solutions in experiments 6 and 7 correspond to situations where the surface is parallel to the image plane. Yet, there is a large difference in the sharpness of the error functions around these solutions. This difference may partly be due to the change in r/d . The second factor which apparently influences the degree of ambiguity in these cases is the deviation between the line of sight and the translation axis. At least in the case of a non-slanted planar surface, it seems that the instability is reduced as the translation vector increasingly deviates from the line of sight.

VI.1.4 Constraints and parameters which can be determined

In ambiguous situations, when the surface can be relatively approximated by a plane, we can still recover useful information in terms of partial constraints on the motion and structure parameters. Usually, the coefficients of the 0th and 1st order components of the flow field, that is, the coefficients a_1, \dots, a_6 of the Ψ transformation (see equations (6.6a-f)), can be reliably estimated. Integration of these constraints over a time sequence of flow fields may, eventually, resolve the ambiguity and result in a unique interpretation.

If a planar patch is independently moving and the camera is stationary, then the ambiguity is, at least partially, the result of using a *camera coordinate system*. In this coordinate system a_1 and a_4 are sums of the X and Y translations (normalized by the distance d from the camera to the object along the line of sight) and rotations. It may be very difficult, however, to determine the correct decomposition to the rotational and translational components. On the other hand, it is possible to define an '*object coordinate system*' which is parallel to the camera coordinate system, but its center is shifted to the surface along the line of sight. In this coordinate system a_1 and a_4 are, respectively, the X and Y translations normalized by d . Hence, *these* normalized translations can be reliably recovered.

Let us now examine the situation where at least one of the two following conditions is satisfied: (a) the translation is along the line of sight, that is, $U_X = U_Y = 0$; (b) the surface can be relatively approximated by a planar surface parallel to the image plane, that is, $l_X = l_Y = 0$. Note that this situation is very common in real scenes. Employing equations (6.6c,e), $a_3 = -a_5$ in this case, and Ω_Z can be estimated by $(a_5 - a_3)/2$. In addition, $a_2 = a_6$ and T_Z , normalized by the distance to the object along the line of sight, can be estimated by $(a_2 + a_6)/2$. In this situation, this is the inverse of the *time-to-collision* and, therefore, we can usually obtain a reasonably accurate estimate of this important parameter, even when ambiguity in recovering 3-D information does exist.

In order to show how the situation discussed above can be detected, we will prove that when $a_3 = -a_5$ and $a_2 = a_6$, then $U_X = U_Y = 0$ and/or $l_X = l_Y = 0$. That is, the first equalities are not only necessary but also sufficient conditions for the latter ones. To prove this, notice that the equalities $a_3 = -a_5$ and $a_2 = a_6$, combined with equations (6.6b,c,e,f), lead to the equalities:

$$l_Y U_X = -l_X U_Y, \quad (6.28a)$$

$$l_X U_X = l_Y U_Y. \quad (6.28b)$$

Assuming that U_X , U_Y , l_X and l_Y are all non-zero, we can divide each side of the first equation with the corresponding side of the second equation, and thus obtain $l_Y/l_X = -l_X/l_Y$ which leads to a contradiction: $(l_Y/l_X)^2 = -1$. Therefore, at least one of the quantities U_X , U_Y , l_X , l_Y must be 0. Suppose now that $U_X = 0$; examining equations (6.28), it follows that $l_X U_Y = 0$ and $l_Y U_Y = 0$ and, therefore, $U_Y = 0$ and/or $l_X = l_Y = 0$. Similarly, each of the conditions $U_Y = 0$, $l_X = 0$, $l_Y = 0$ leads to the desired result.

Another approach which may be taken in order to deal with instability in recovering motion parameters is based on representing possible values of these parameters by a probabilistic distribution function. Such a function can be defined, for example,

on the unit hemisphere \mathcal{N} , using the computed values of $\hat{\sigma}_\mathcal{N}$.

VI.2 Ambiguity in Grouping Flow Vectors into Sets Corresponding to Rigidly Moving Objects

VI.2.1 Introductory discussion

We demonstrated in Section VI.1 that in some situations there exists a large set of motion parameters which, assuming the presence of noise, are consistent with the flow field generated by a rigidly moving object. Suppose now that two independently moving objects are given. If the two corresponding solution sets of motion parameters are large, then the possibility that these sets intersect each other is not negligible. Such an intersection corresponds to 3-D motion parameters which are consistent with both objects. In this case the optical flows can be interpreted as resulting from one rigidly moving object. Note that, to the best of our knowledge, this ambiguity has not been addressed yet in the literature.

To demonstrate the possibility of ambiguity in decomposing the flow field into sets corresponding to rigid objects, we wish to show that there exist non-trivial situations in which we can find motion parameters compatible with the flows generated by two independently moving objects. If the surfaces of the objects can be relatively approximated by planes, then, following section VI.1.2, we can examine this possibility of ambiguity by trying to compute motion and structure parameters which are consistent with the coefficients a_1, \dots, a_6 of the associated Ψ transformations. The resulting errors in the second-order components of the optical flows will be small if, for example, the field of view is small enough. Since the six motion parameters \underline{U} and $\underline{\Omega}$ should be the same for both objects but the three structure parameters can be different, we obtain, including the constraint $U_X^2 + U_Y^2 + U_Z^2 = 1$, 13 equations with 12 unknowns. It is reasonable to expect that in many situations these equations do have a solution.

VI.2.2 Analysis of a specific case

Continuing the introductory discussion, let us now examine, as we did in Section VI.1.4, the common situation where, for each of the two objects, at least one of the following conditions (not necessarily the same) must be satisfied: the translation is along the line of sight, or the surface can be relatively approximated by a planar surface parallel to the image plane. In such a situation $U_X = U_Y = 0$ or $l_X = l_Y = 0$, and $U'_X = U'_Y = 0$ or $l'_X = l'_Y = 0$, where the parameters associated with the second object are marked by the symbol "'". In addition, let us assume that $\Omega_Z = \Omega'_Z$ and that the signs of U_Z and U'_Z are the same, where $U_Z = 0$ if and only if $U'_Z = 0$. Finally, to guarantee a solution, it is assumed that if $U_Z \neq 0$ and $l_Z U_Z = l'_Z U'_Z$, then $\Omega_Y + l_Z U_X = \Omega'_Y + l'_Z U'_X$ and $-\Omega_X + l_Z U_Y = -\Omega'_X + l'_Z U'_Y$.

Employing equations (6.6a) to (6.6f), we can obtain the following equations, related to the first object, where the unknowns are denoted by the symbol "hat" :

$$\hat{\Omega}_Y + \hat{l}_Z \hat{U}_X = a_1 (= \Omega_Y + l_Z U_X), \quad (6.29a)$$

$$\hat{l}_X \hat{U}_X - \hat{l}_Z \hat{U}_Z = a_2 (= -l_Z U_Z), \quad (6.29b)$$

$$-\hat{\Omega}_Z + \hat{l}_Y \hat{U}_X = a_3 (= -\Omega_Z), \quad (6.29c)$$

$$-\hat{\Omega}_X + \hat{l}_Z \hat{U}_Y = a_4 (= -\Omega_X + l_Z U_Y), \quad (6.29d)$$

$$\hat{\Omega}_Z + \hat{l}_X \hat{U}_Y = a_5 (= \Omega_Z) \quad (6.29e)$$

and

$$\hat{l}_Y \hat{U}_Y - \hat{l}_Z \hat{U}_Z = a_6 (= -l_Z U_Z). \quad (6.29f)$$

A similar set of equations can be obtained for the second object.

Let us start the solution process by choosing $\hat{\Omega}_Z = \Omega_Z$, $\hat{l}_X = \hat{l}_Y = 0$ and $\hat{l}'_X = \hat{l}'_Y = 0$, thus satisfying equations (6.29c) and (6.29e), as well as the corresponding equations associated with the second object. We proceed by examining the case $U_Z \neq 0$, in which we constrain \hat{U}_Z to be non-zero and to have the same sign as the

sign of U_Z . Thus, from (6.29b) and (6.29f) we can obtain

$$\hat{l}_Z = l_Z \frac{U_Z}{\hat{U}_Z}. \quad (6.30)$$

Substituting this expression in (6.29a) and (6.29d) yields

$$\hat{\Omega}_Y + l_Z U_Z \hat{m}_x = a_1 \quad (6.31a)$$

and

$$-\hat{\Omega}_X + l_Z U_Z \hat{m}_y = a_4, \quad (6.31b)$$

where $(\hat{m}_x, \hat{m}_y) = (\hat{U}_X/\hat{U}_Z, \hat{U}_Y/\hat{U}_Z)$ is the corresponding FOE. Similarly, we can obtain the following equations, corresponding to the second object:

$$\hat{\Omega}_Y + l'_Z U'_Z \hat{m}'_x = a'_1 \quad (6.32a)$$

and

$$-\hat{\Omega}_X + l'_Z U'_Z \hat{m}'_y = a'_4. \quad (6.32b)$$

Combining (6.31) with (6.32) yields

$$a_1 - l_Z U_Z \hat{m}_x = a'_1 - l'_Z U'_Z \hat{m}'_x \quad (6.33a)$$

and

$$a_4 - l_Z U_Z \hat{m}_y = a'_4 - l'_Z U'_Z \hat{m}'_y. \quad (6.33b)$$

If $l_Z U_Z = l'_Z U'_Z$, then, according to our assumptions, $a_1 = a'_1$ and $a_4 = a'_4$, and, therefore, we can choose arbitrary values of \hat{m}_x and \hat{m}_y . Otherwise,

$$\hat{m}_x = \frac{a'_1 - a_1}{l'_Z U'_Z - l_Z U_Z} \quad (6.34a)$$

and

$$\hat{m}_y = \frac{a'_4 - a_4}{l'_Z U'_Z - l_Z U_Z}. \quad (6.34b)$$

The values of $\hat{\Omega}_X$ and $\hat{\Omega}_Y$ can now be computed from equations (6.31) or (6.32).

Let us now examine the complementary case where $U_Z = 0$. In this case, to satisfy equations (6.29b) and (6.29f), we choose $\hat{U}_Z = 0$. Combining equations (6.29a) and (6.29d) with the corresponding equations associated with the second object yields

$$a_1 - \hat{l}_Z \hat{U}_X = a'_1 - \hat{l}'_Z \hat{U}_X \quad (6.35a)$$

and

$$a_4 - \hat{l}_Z \hat{U}_Y = a'_4 - \hat{l}'_Z \hat{U}_Y. \quad (6.35b)$$

Therefore,

$$(\hat{l}'_Z - \hat{l}_Z) \hat{U}_X = a'_1 - a_1 \quad (6.36a)$$

and

$$(\hat{l}'_Z - \hat{l}_Z) \hat{U}_Y = a'_4 - a_4. \quad (6.36b)$$

Thus, since $\hat{U}_X^2 + \hat{U}_Y^2 = 1$,

$$\hat{l}'_Z - \hat{l}_Z = \pm \sqrt{(a'_1 - a_1)^2 + (a'_4 - a_4)^2}. \quad (6.37)$$

If $a_1 = a'_1$ and $a_4 = a'_4$, then $\hat{l}_Z = \hat{l}'_Z$, and any \hat{U}_X and \hat{U}_Y which satisfy $\hat{U}_X^2 + \hat{U}_Y^2 = 1$ are legitimate solutions. Otherwise, that is, if $a_1 \neq a'_1$ or $a_4 \neq a'_4$, then

$$\hat{U}_X = \frac{a'_1 - a_1}{\pm \sqrt{(a'_1 - a_1)^2 + (a'_4 - a_4)^2}} \quad (6.38a)$$

and

$$\hat{U}_Y = \frac{a'_4 - a_4}{\pm \sqrt{(a'_1 - a_1)^2 + (a'_4 - a_4)^2}}. \quad (6.38b)$$

To finish the solution process, we should choose \hat{l}_Z and \hat{l}'_Z which satisfy the constraint (6.37) and, then, using equations (6.29a) and (6.29d), determine the values of $\hat{\Omega}_X$ and $\hat{\Omega}_Y$. It is optimal to select the values of \hat{l}_Z and \hat{l}'_Z such that the resulting errors in the coefficients a_7 and a_8 of the Ψ transformations will be minimal.

VI.2.3 An example

In order to demonstrate how different motions can be interpreted as one rigid motion, let us examine the case where two planar patches, parallel to the image plane, are independently translating. Both translations are assumed to be parallel to the image plane, but one object is translating in parallel to the X-axis generating flow values of $(-0.04, 0)$ (in focal units), and the second object is translating in parallel to the Y-axis generating flow values of $(0, 0.03)$ (see Figure 6.9). Note that $a_1 = -0.04$, $a'_4 = 0.03$ and the other coefficients of the Ψ transformations associated with the objects are 0.

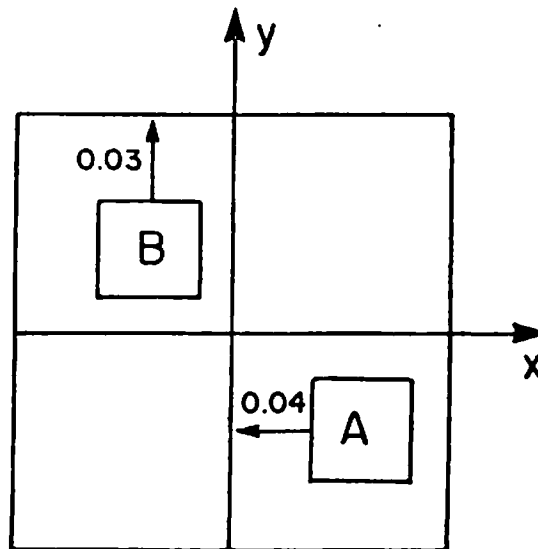


Figure 6.9: The optical flows induced by the translation of two objects.

We wish to recover motion parameters \hat{U} and $\hat{\Omega}$ which are compatible with both sets of flow vectors and with structure parameters \hat{I} and \hat{I}' corresponding, respectively, to the first and second object. Employing the results in Section VI.2.2 for the case $U_z = 0$ (equations (6.38)), we have $\hat{U} = (\pm 0.8, \pm 0.6, 0)$, $\hat{\Omega}_z = 0$,

$\hat{l}_X = \hat{l}_Y = 0$, $\hat{l}'_X = \hat{l}'_Y = 0$ and $\hat{l}'_Z - \hat{l}_Z = \pm 0.05$. In addition, using equations (6.29a) and (6.29d), $\hat{\Omega}_X = \pm 0.6\hat{l}_Z$ and $\hat{\Omega}_Y = -0.04 \mp 0.8\hat{l}_Z$.

Since $\hat{\Omega}_X$ and $\hat{\Omega}_Y$ are exactly the errors in the coefficients a_7 and a_8 of the Ψ transformations, we wish to minimize

$$\chi(\hat{l}_Z) \stackrel{\text{def}}{=} \hat{\Omega}_X^2 + \hat{\Omega}_Y^2 = 0.36\hat{l}_Z^2 + (0.04 \pm 0.8\hat{l}_Z)^2. \quad (6.39)$$

Let us now distinguish between the cases $\hat{l}'_Z > \hat{l}_Z$ and $\hat{l}'_Z < \hat{l}_Z$. In the first case $\hat{U} = (0.8, 0.6, 0)$ and the first derivative of $\chi(\hat{l}_Z)$ is $2\hat{l}_Z + 0.064$. Since \hat{l}_Z is constrained to be positive, the minimum of $\chi(\hat{l}_Z)$, obtained for $\hat{l}_Z = 0$, is, in this case, 0.0016. Note that $\hat{l}_Z = 0$ means that the object is at infinity, which is, of course, unrealistic; however, taking a sufficiently large distance of the object from the camera, \hat{l}_Z can be arbitrarily close to 0.

In the second case $\hat{U} = (-0.8, -0.6, 0)$, the derivative of $\chi(\hat{l}_Z)$ is $2\hat{l}_Z - 0.064$ and \hat{l}_Z should be at least 0.05. Hence, the minimum of $\chi(\hat{l}_Z)$, achieved for $\hat{l}_Z = 0.05$, is, in this case, 0.0009. The optimal solution is, therefore, $\hat{U} = (-0.8, -0.6, 0)$, $\hat{l} = (0, 0, 0.05)$, $\hat{l}' = (0, 0, 0)$ and $\hat{\Omega} = (-0.03, 0, 0)$. Assuming small second order terms of the rotational component, this solution can be graphically represented by Figure 6.10.

Since $\hat{\Omega}_Y = 0$, there is no error in a_7 ; on the other hand, there is an error in a_8 which is $-\hat{\Omega}_X = 0.03$. The corresponding discrepancy between the correct flow field and that predicted from the above parameters is small if the FOV is small, or both the size of the objects and their distance from the line of sight are small relative to their distance from the camera.

VI.2.4 Concluding remarks on the rigidity assumption

We have just shown that the rigidity assumption is not appropriate when the flow field is noisy, that is, the consistency of a set of flow vectors with the same

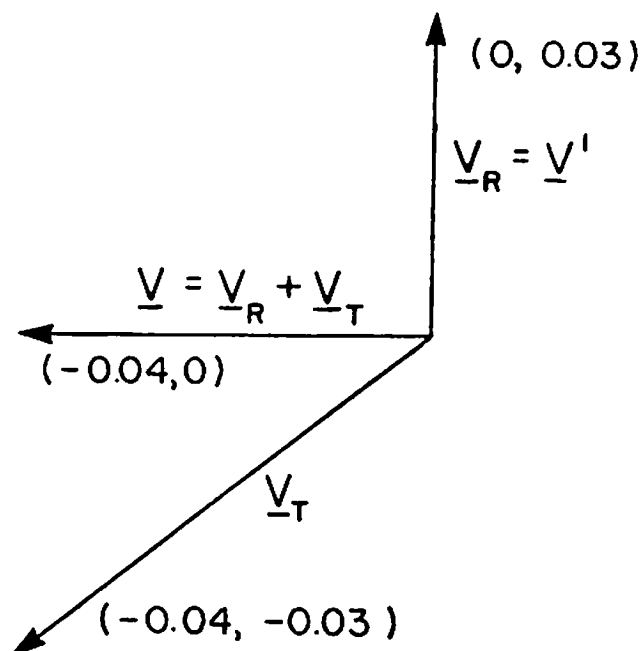


Figure 6.10: Graphical representation of the optimal solution. The flow vectors corresponding to the first and second object are denoted by \underline{v} and \underline{v}' , respectively. The rotational component of the optimal solution is \underline{v}_R , while the translational component corresponding to the first object is \underline{v}_T and the translational component corresponding to the second object is $(0, 0)$.

3-D motion parameters does not reasonably guarantee that they are really induced by one rigidly moving object. Observing, in addition, that almost any set which contains less than 5 flow vectors is consistent with some Ψ transformation, we propose a modified assumption: *a set of at least 5 adjacent flow vectors, which are compatible, up to the estimated noise level, with a rigid motion of a planar patch, will be assumed to be induced by one rigidly moving object.* This assumption has

been successfully applied (see Chapters IV and VII) to segment noisy flow fields.

In some situations, however, the consistency of two sets of flow vectors with the same motion parameters is still a strong evidence for the hypothesis that these sets are generated by one rigidly moving object. This is the case, for example, when accurate motion parameters can be separately recovered for each set. In such a situation, similarity of the results is not likely to be accidental.

Nevertheless, in general we still must accept the possibility of ambiguity in grouping flow vectors into sets corresponding to rigidly moving objects. Hence, as has already been proposed in Section V.2, the interpretation of the flow field should result in a *set* of possible decompositions, rather than only one decomposition. Each hypothesized object can be assigned a probability value, based on the number of segments composing the object's flow and on the degree of ambiguity in separately recovering the motion parameters associated with each of them.

CHAPTER VII

EXPERIMENTS

In this chapter we present five experiments which demonstrate our proposed scheme for the interpretation of optical flow fields. The first two experiments are based on simulated data, and the last three are based on images taken from a video camera in the UMASS Robotics Lab. In all the experiments, values that appear in translation vectors and surface equations are given in focal units, whereas rotation parameters are given in degrees and flow vectors are given in pixel units. Actually, the flow values in the experiments based on simulated data are rounded to integers, thus inducing noise uniformly distributed between $-1/2$ and $+1/2$ pixels. The methods employed for computing the real data in experiments 3, 4 and 5 also produce flow values given in integer units, hence the noise level in these experiments should be at least as high as in experiments 1 and 2 (actually it is higher). The image, in all the experiments, contains 128×128 pixels. The field of view of the camera is 45° in the experiments with simulated data and 30° in the experiments with real data.

VII.1 Experiment 1

The first experiment simulates a translatory motion of the camera, represented by the vectors $\underline{T}_C = (0., 0.02, 1.)$ and $\underline{\Omega}_C = (0^\circ, 0^\circ, 0^\circ)$. The environment consists of two distinct surfaces: a plane described by the equation $Z = 50Y + 100$ and an ellipsoid represented by $(X - 2)^2 + [(Y - 2)/4]^2 + (Z - 5)^2 = 1$. These surfaces are displayed in Figure 7.1a. A flow vector with a weight of 1 is computed for each pixel, unless the corresponding ray of light does not intersect any of the surfaces, in which case the related weight is assumed to be 0. A sample of the flow field is shown in Figure 7.1b.

The results of the three stages of the segmentation, shown in Figures 7.1c, 7.1d and 7.1e, demonstrate the role and importance of each of these stages. The over-fragmentation in the first stage is due to the large second-order terms of the flow field. The non-smooth and repetitive shapes of the components were caused by the round-off error. In order to reduce the computational cost of the first and second stages, the grouping of vectors belonging to small connected sets was postponed to the third stage. A few vectors were still not merged into any of the segments in this last segmentation stage, because of incompatibility with the corresponding Ψ transformations.

The two segments, found in this process, were determined to be consistent with the same rigid motion. The error function $\hat{\sigma}_N$ (equation 5.17), defined on the unit hemisphere $\{\underline{U} : |\underline{U}| = 1, U_Z \geq 0\}$, was computed using 64 vectors from each segment. The spherical coordinates (ϕ, θ) , utilized in equation (5.21) for representing the unit hemisphere, are used as polar coordinates for displaying the function $\hat{\sigma}_N$ in Figure 7.1f. Recall that $\hat{\sigma}_N(\underline{U})$ is approximately the average deviation between the measured flow field and the one predicted by the optimal motion and depth values when the translation axis is constrained to be either \underline{U} or $-\underline{U}$. Employing the non-uniform sampling procedure (which is not represented in the display of Figure 7.1f) for minimizing $\hat{\sigma}_N$, the motion parameters were determined, after two iterations, to be $\underline{U} = (0.0017, -0.0204, -0.9998)$ and $\underline{\Omega} = (-0.03^\circ, -0.02^\circ, -0.02^\circ)$. Note that, assuming a stationary environment, the camera motion is given by $\hat{\underline{U}}_C = -\underline{U}$ and $\hat{\underline{\Omega}}_C = -\underline{\Omega}$. These results are in a good agreement with the correct values; the error in determining the translational axis is only 0.1° .

Substituting the computed motion parameters in equation (5.10), the 'reciprocal depth' map, that is, the function r/Z shown in Figure 7.1g, was obtained. Errors in the depth values are mainly a direct result of the noise induced by rounding the flow values to integers. They are especially large near the FOE which is close to the center of the image. The average of the relative errors in the estimated values of

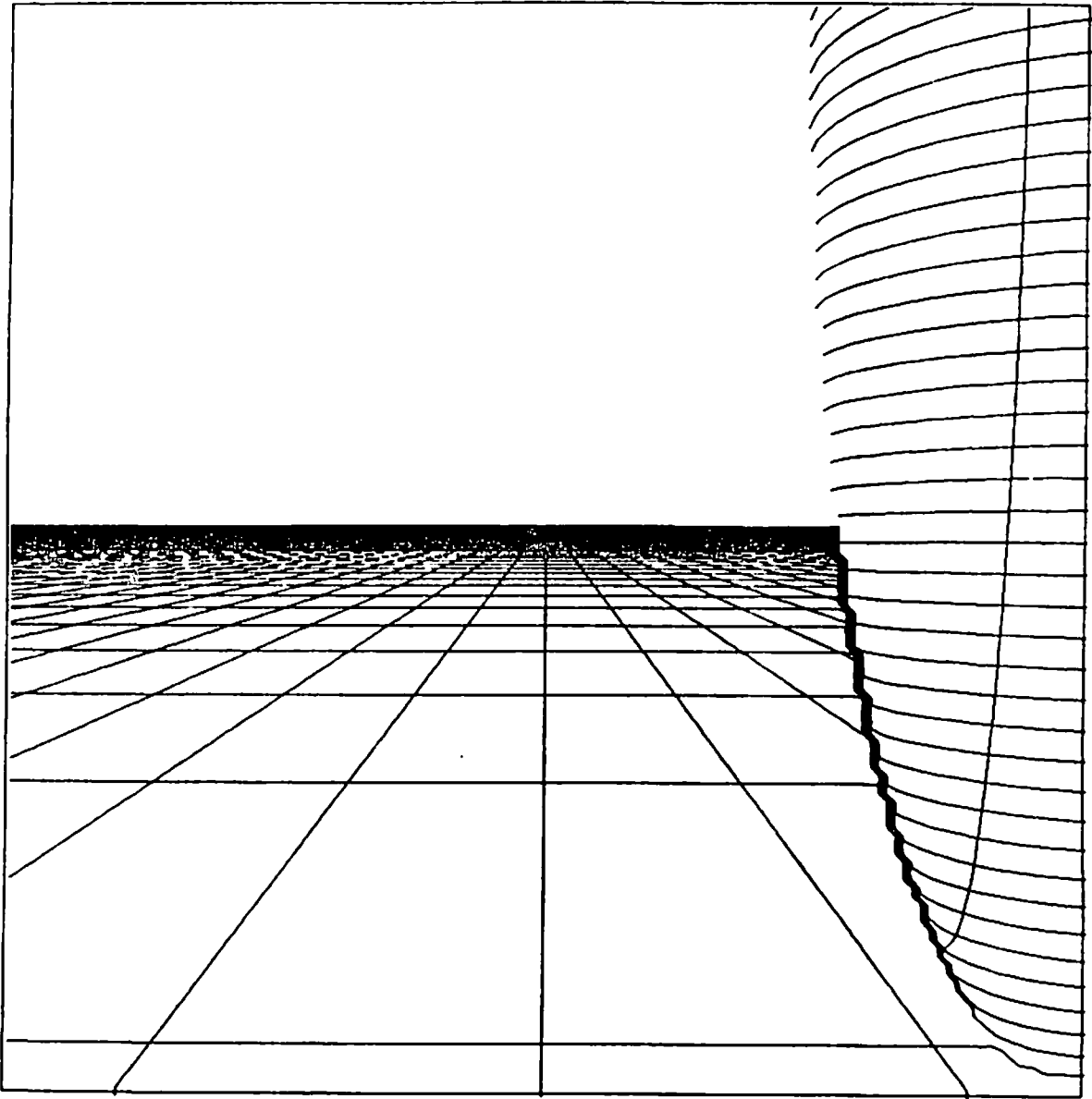


Figure 7.1: Experiment 1. (a) The simulated environment. Lines corresponding to constant values of X , as well as lines corresponding to constant values of Y , are drawn. The environment is stationary, but the camera is moving: $\mathcal{L}_C = (0., 0.02, 1.)$ and $\underline{\Omega}_C = (0.^{\circ}, 0.^{\circ}, 0.^{\circ})$.

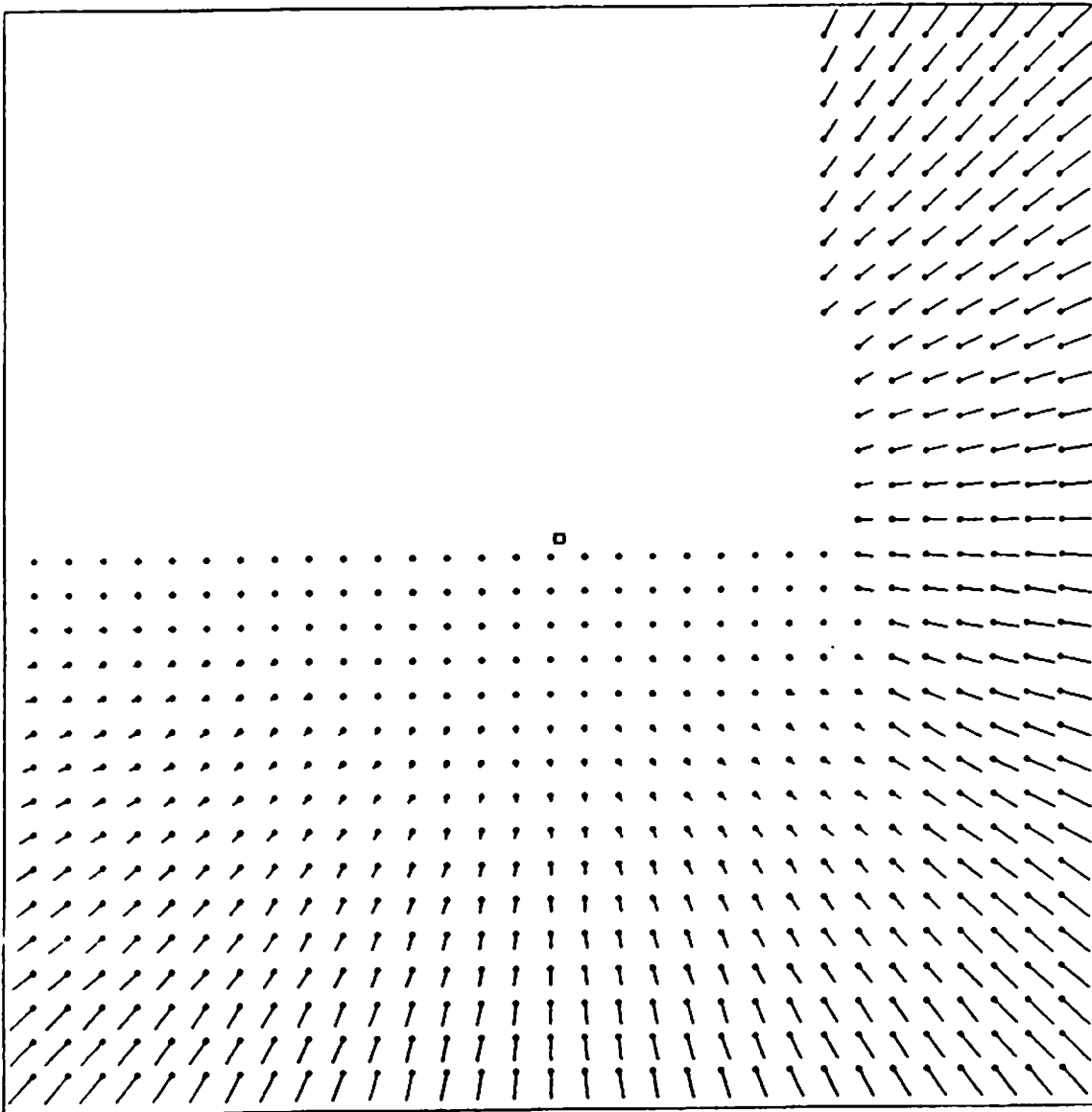


Figure 7.1, continued: (b) A sample of the flow field. The initial point of each vector is marked by a dot. The length of the vectors is scaled by 0.25. The FOE is marked by a square.

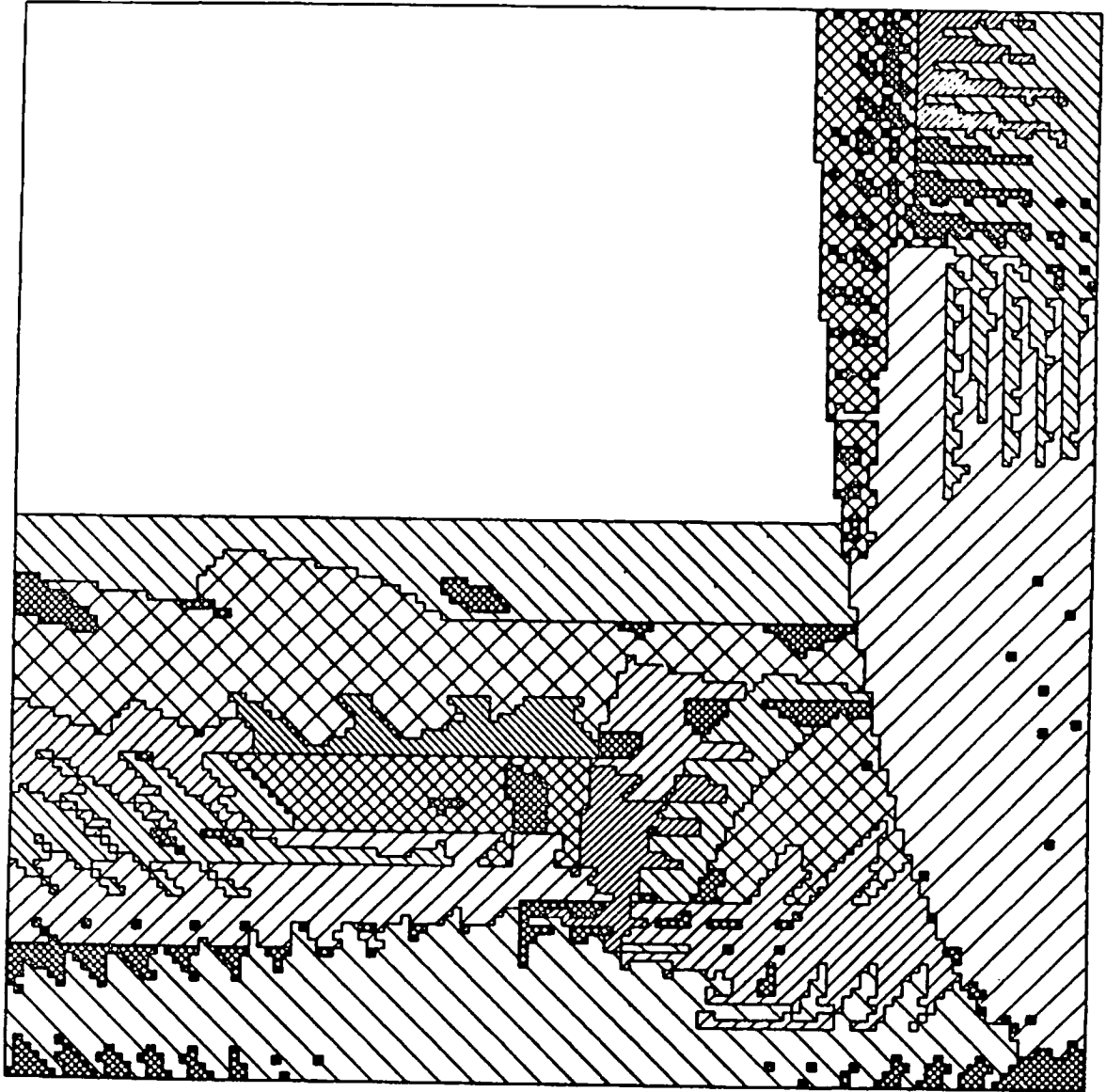


Figure 7.1, continued: (c) Components, represented by line patterns, which were determined in the first step of the segmentation. The small areas with the densest pattern correspond to vectors which are not contained in any of the components.

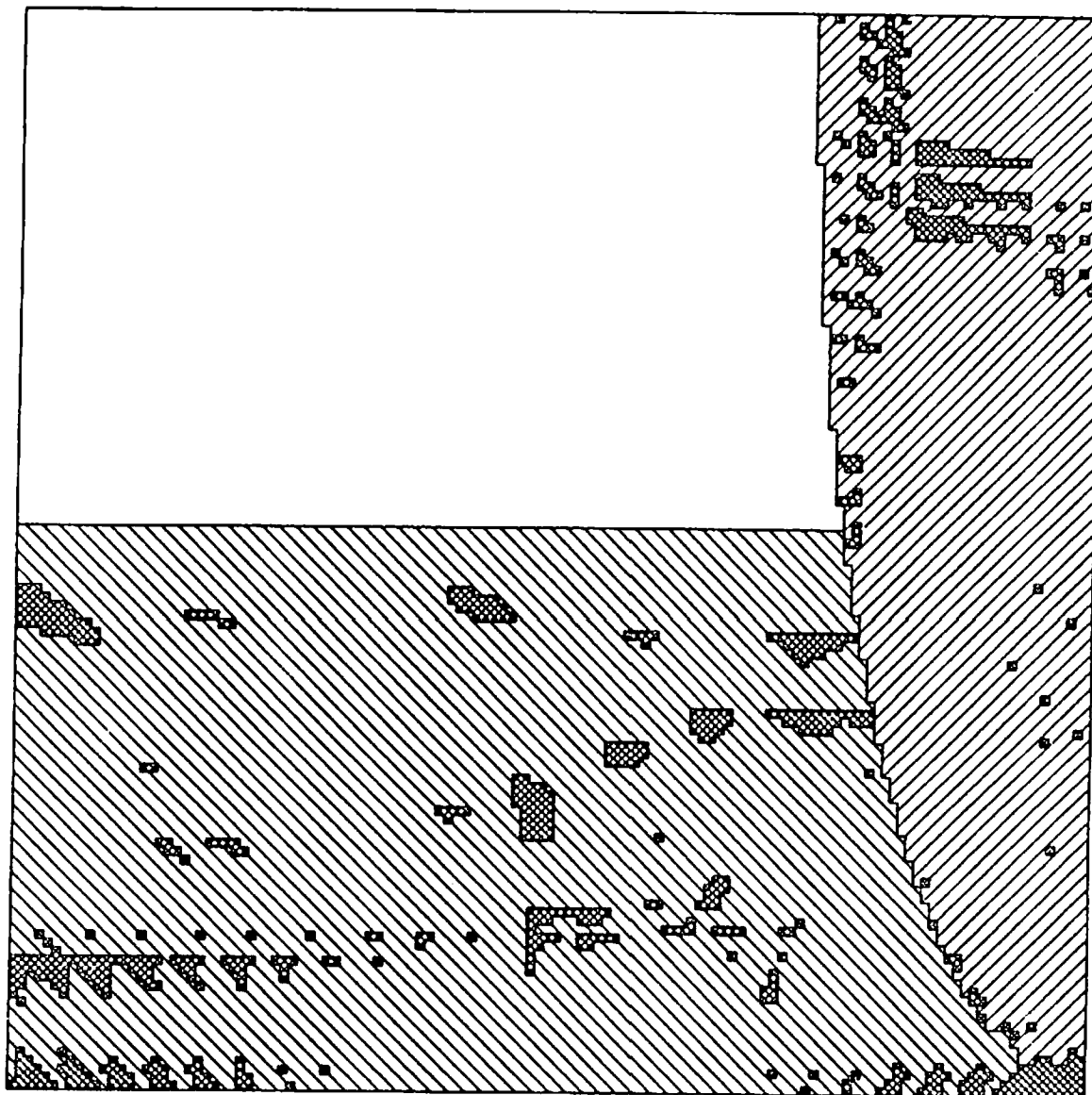


Figure 7.1, continued: (d) Segments obtained by merging components consistent with the same Ψ transformation.

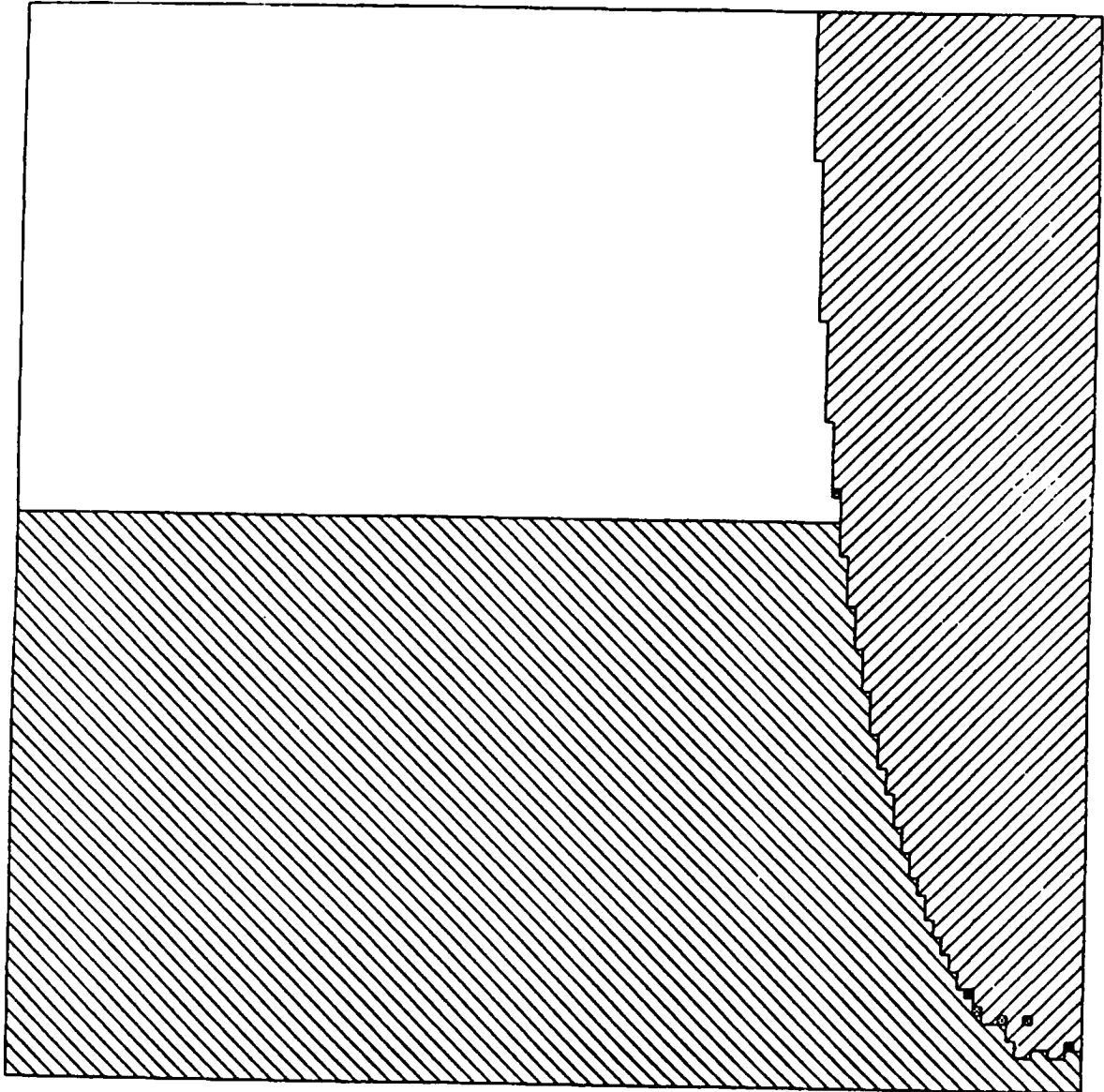


Figure 7.1, continued: (e) Final segmentation. Note that almost all the flow vectors which, after the first stage, had not been included in any of the components, were merged into the segments.

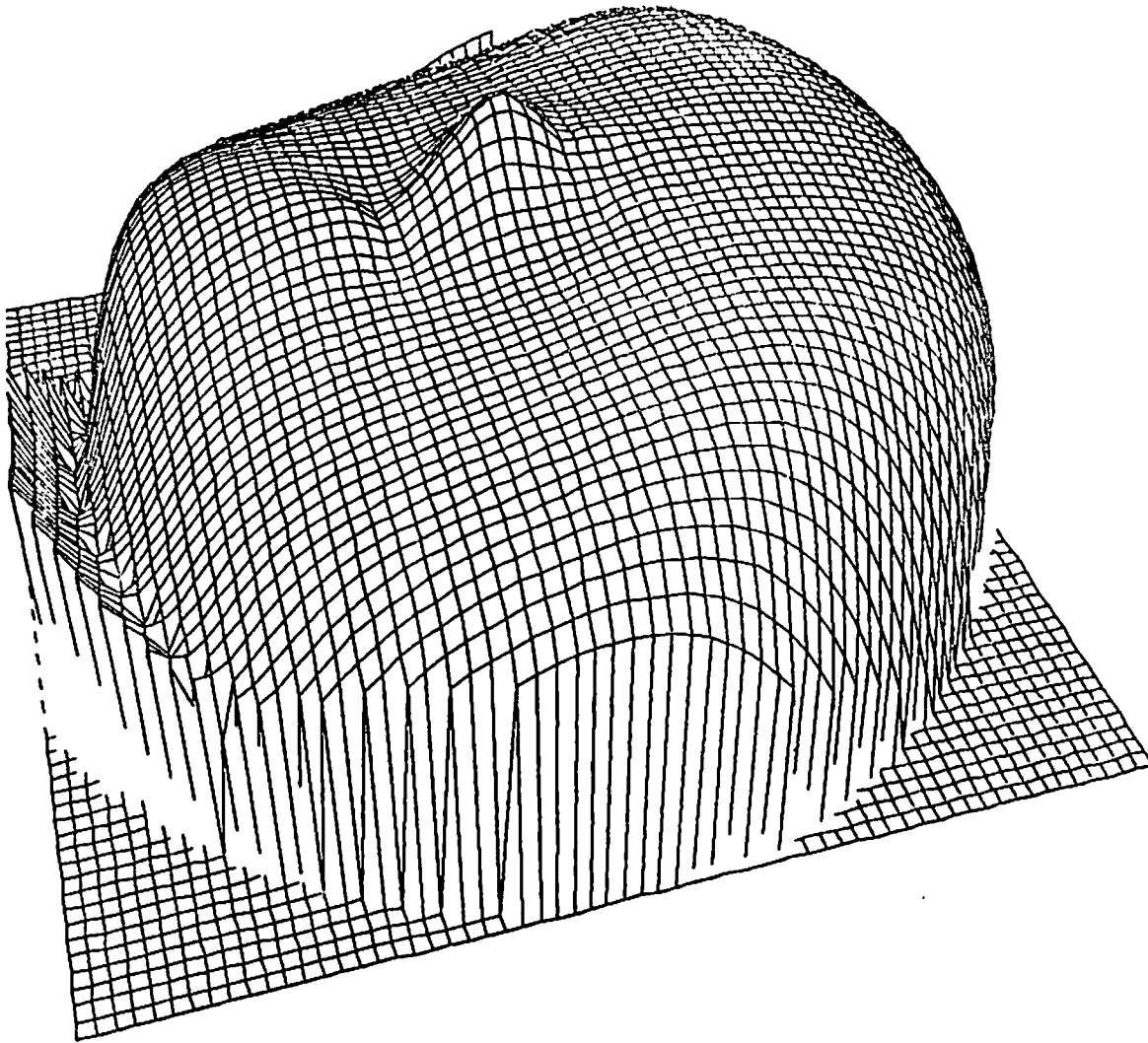


Figure 7.1, continued: (f) The error function $\hat{\sigma}_\chi$, shown inverted, defined on the hemisphere $\{\underline{U} : |\underline{U}| = 1, U_z \geq 0\}$. The spherical coordinates (ϕ, θ) , employed in equation (5.21) for representing this hemisphere, are used here as polar coordinates. The range of the function $\hat{\sigma}_\chi$ is $[0.275, 8.605]$ (in pixels); that is, 0.275 pixels is the average error value, per flow vector, associated with the peak of the displayed surface (this peak corresponds to the estimated translation axis), and 8.605 is the error value associated with the reference plane at the bottom of the surface. The solution found by minimizing this function is $\hat{\underline{U}}_C = (-0.0017, 0.0204, 0.9998)$ and $\hat{\underline{\Omega}}_C = (0.03^\circ, 0.02^\circ, 0.02^\circ)$.

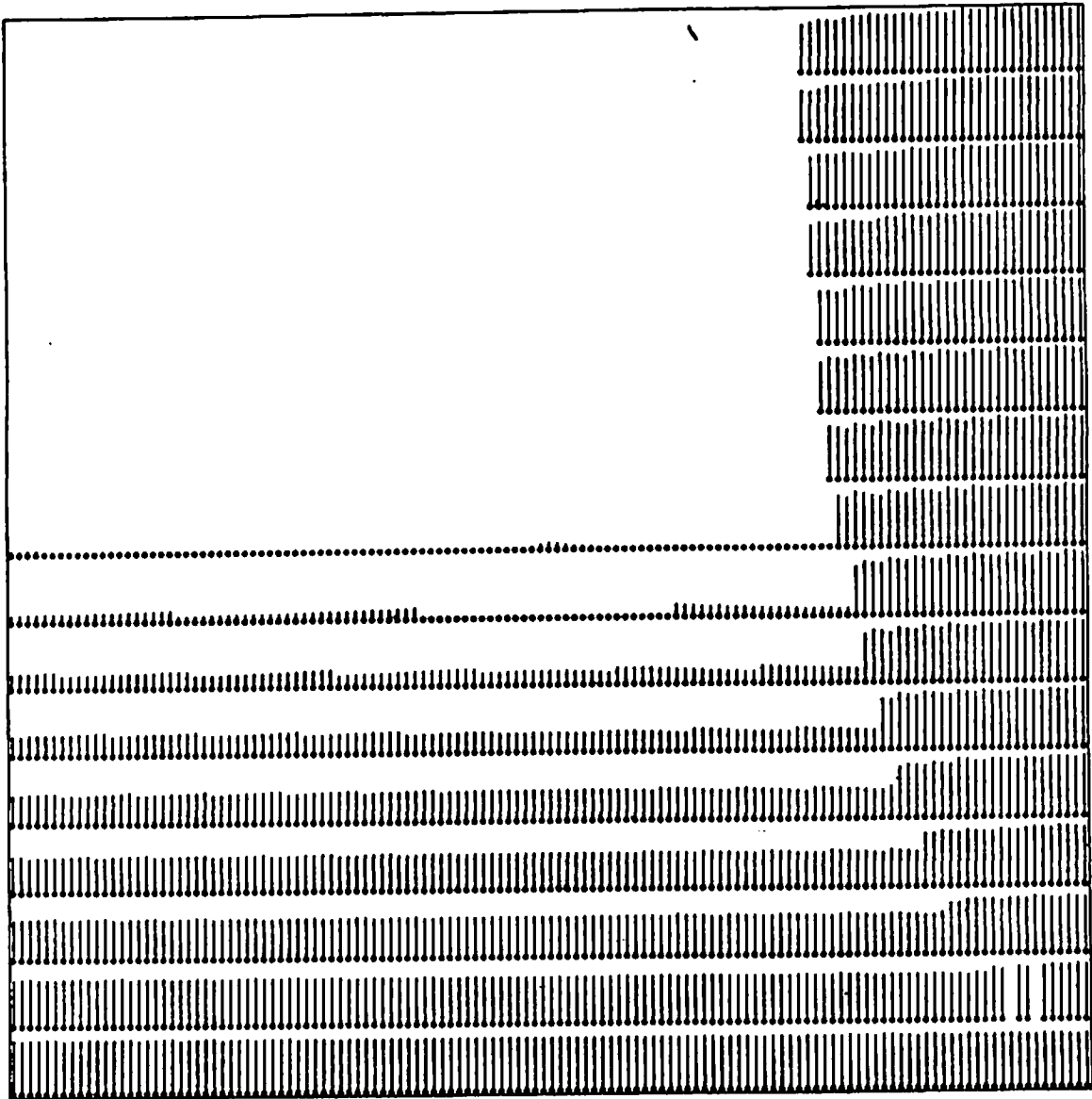


Figure 7.1, continued: (g) The function r/Z , where r is the length of the translation vector and Z is the environmental depth. The length of each bar represents the relative value of r/Z at the image pixel corresponding to the attached dot.

r/Z is 111.1% for vectors which are not more than 16 pixels away from the FOE. The corresponding average for all the other vectors is only 8.4%. The total average of the relative errors, calculated for the entire depth map, is 12.3%. Notice that these errors can considerably be reduced by using smoothing techniques or fitting parameterized surfaces to the depth map.

VII.2 Experiment 2

In the second experiment, the camera motion is composed of both translation and rotation, described by $T_C = (0.5, 0.5, 1.)$ and $\Omega_C = (1.15^\circ, -1.15^\circ, 2.86^\circ)$. The environment contains an independently moving sphere, defined by $(X - 9)^2 + (Y - 9)^2 + (Z - 30)^2 = 4$. An object coordinate system is defined, which is parallel to the camera coordinate system, but its origin is in the sphere center $(9, 9, 30)$. The motion of the object, in this coordinate system, is represented by $T_O = (0.5, -0.5, 0.)$ and $\Omega_O = (0^\circ, 0^\circ, -11.46^\circ)$. Thus, the object is translating in parallel to the image plane and rotating around an axis perpendicular to this plane. The stationary environment is composed of two surfaces: a plane described by $Z = X + 0.5Y + 50$ and an ellipsoid described by $[(X + 3)/2]^2 + [(Y + 1)/5]^2 + [(Z - 20)/2]^2 = 1$. These surfaces and the moving object are displayed in Figure 7.2a. A 32×32 sample of the flow field corresponding to this scene is shown in Figure 7.2b.

The results of the three segmentation stages are shown in Figures 7.2c, 7.2d and 7.2e. The fragmentation in the first stage is not excessive, because the second-order terms of the Ψ transformations associated with the segments are small relative to those in the first experiment.

The two segments associated with the stationary environment were determined to be consistent with the same rigid motion, while no rigid motion compatible with the third segment was also found to be consistent with one of the other seg-

ments. Thus, the decomposition of the flow field into sets corresponding to independently moving objects could be uniquely (and correctly) determined. The error function $\hat{\delta}_M$ corresponding to the stationary environment is displayed in Figure 7.2f. The associated motion parameters of the camera were determined to be $\hat{U}_C = (0.3899, 0.4037, 0.8277)$ (the corresponding actual values were $U_C = (0.4082, 0.4082, 0.8164)$ — an error of 1.2°) and $\hat{\Omega}_C = (1.17^\circ, -1.12^\circ, 2.83^\circ)$.

The depth map of the stationary environment is represented by the function r/Z in Figure 7.2g. Again, the relative errors in r/Z are mainly a direct consequence of the round-off errors in the corresponding flow values. Their average value, in this experiment, is 14.7%, while the corresponding average for vectors which are within 32 pixels of the upper right corner of the image (which is close to the FOE) is 79.1%.

The error function corresponding to the independently moving object is shown in Figures 7.2h and 7.2i. This function is very close to its minimal value in a large portion of the search space, thus demonstrating the ambiguity in recovering motion parameters from a noisy flow field, which is discussed in Section VI.1. The main reason for the ambiguity in the current case is the small size of the sphere's projection on the image, since the extraction of the 3-D motion can be based in such a case only on local information. In addition, the depth variation is small relative to the distance of the object from the camera. The correct parameter values of the motion of the object relative to the camera are, in camera coordinates, $U_{O/C} = (-0.7809, 0.3471, -0.4338)$ and $\Omega_{O/C} = (-1.15^\circ, 1.15^\circ, -14.32^\circ)$, while the estimated values were determined, after one iteration, to be completely different (see Figure 7.2i): $\hat{U}_{O/C} = (0.7925, -0.4661, -0.3932)$ and $\hat{\Omega}_{O/C} = (-5.96^\circ, -9.37^\circ, -9.13^\circ)$. The error measure associated with the correct parameter values is indeed larger than the error measure associated with the estimated values, although only slightly. Note the somewhat symmetric shape of the error surface in Figure 7.2i. This may be related to the duality of planar surface solutions, because the sphere

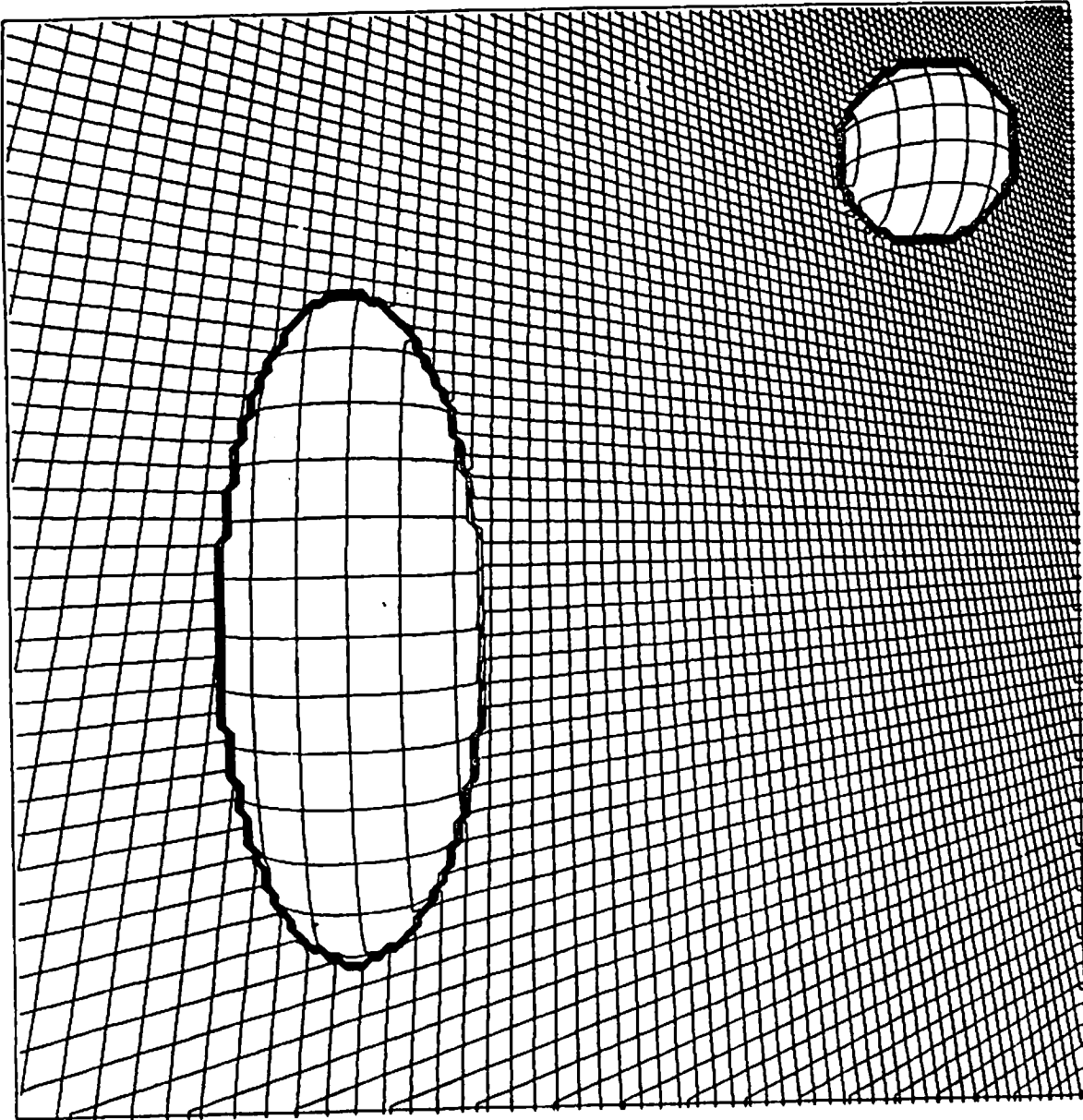


Figure 7.2: Experiment 2. (a) The simulated environment, graphically represented by lines corresponding to constant values of X and Y . The motion of the camera is represented by $\mathcal{T}_C = (0.5, 0.5, 1.)$ and $\Omega_C = (1.15^\circ, -1.15^\circ, 2.86^\circ)$. The motion of the independently moving object, that is, the sphere projected on the upper right corner of the image, is, in its own coordinate system, $\mathcal{T}_O = (0.5, -0.5, 0.)$ and $\Omega_O = (0.^\circ, 0.^\circ, -11.46^\circ)$.

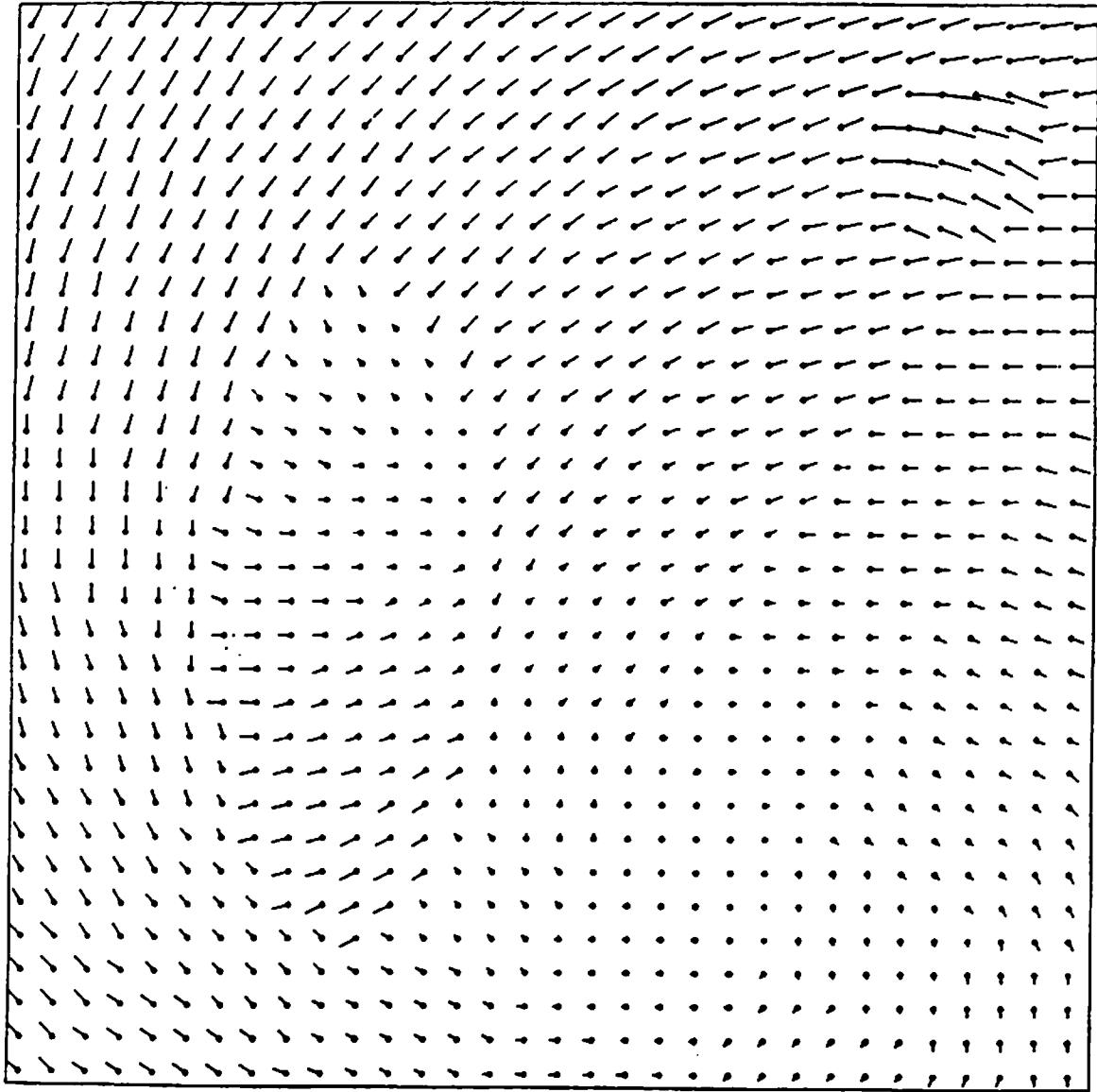


Figure 7.2, continued: (b) A 32×32 sample of the flow field. The vectors are scaled by 0.5. The FOE associated with the camera motion is outside the image, near the upper right corner.

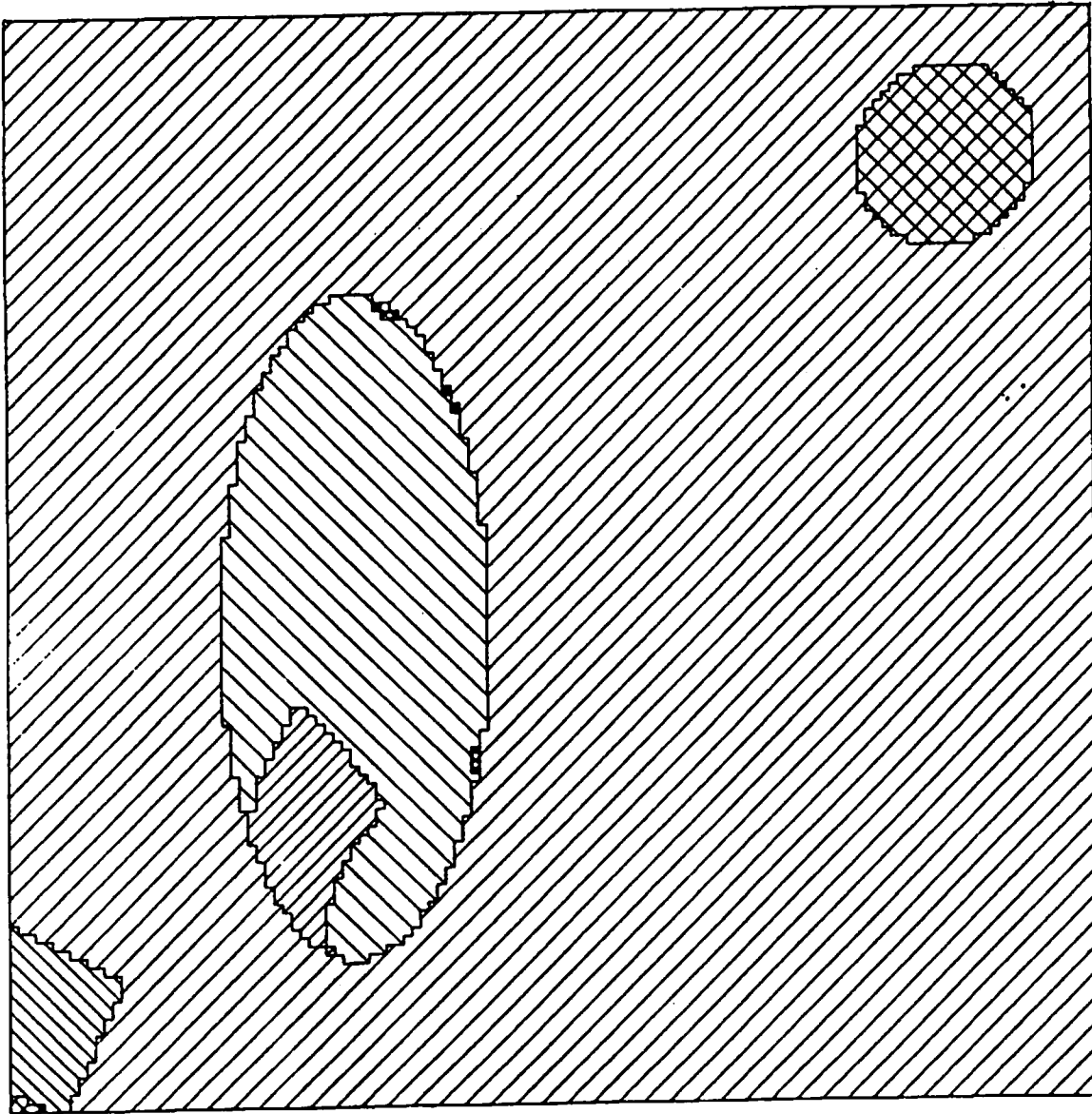


Figure 7.2, continued: (c) Components obtained in the first stage of segmentation.

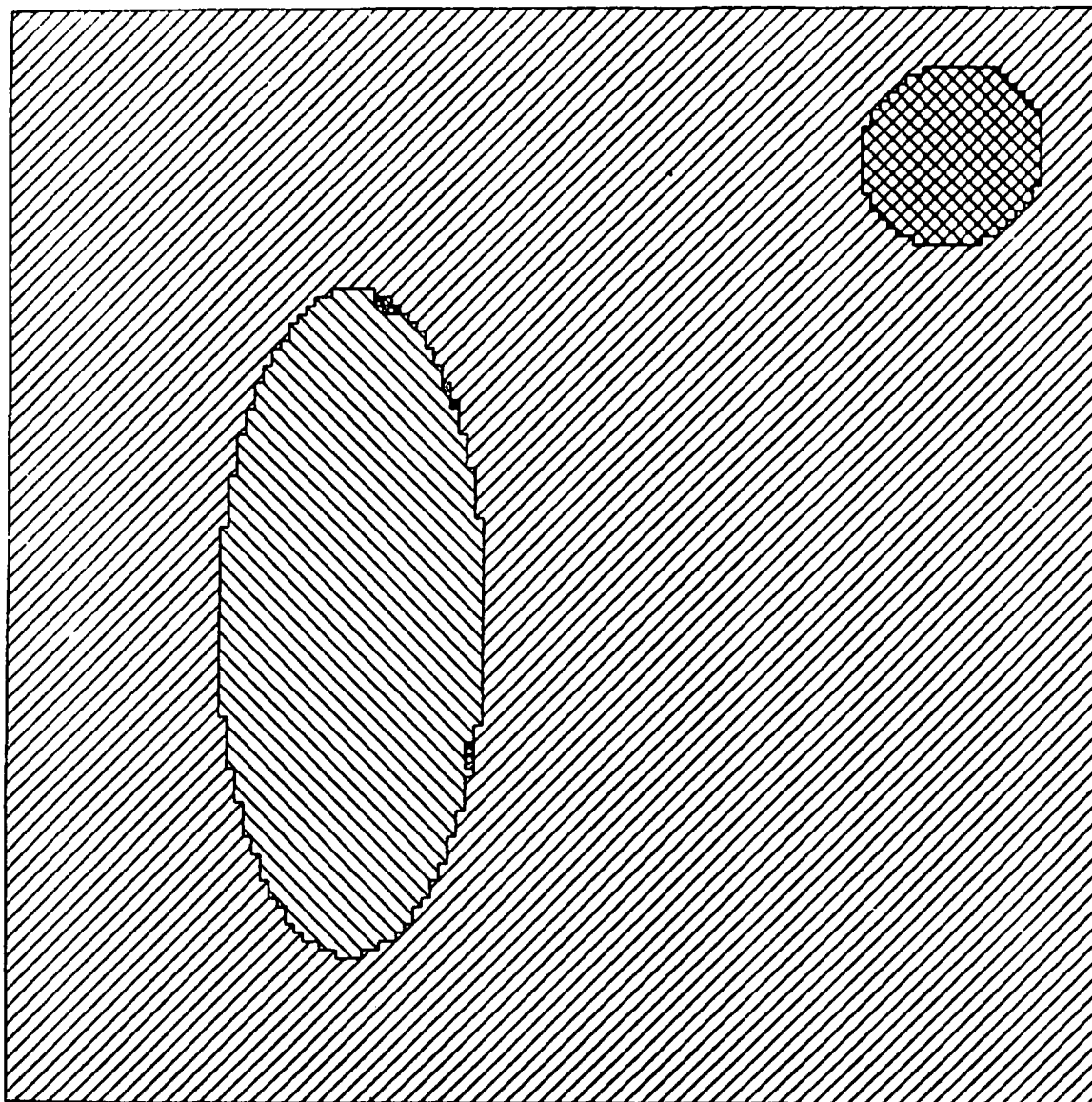


Figure 7.2, continued: (d) Segments obtained by merging components.

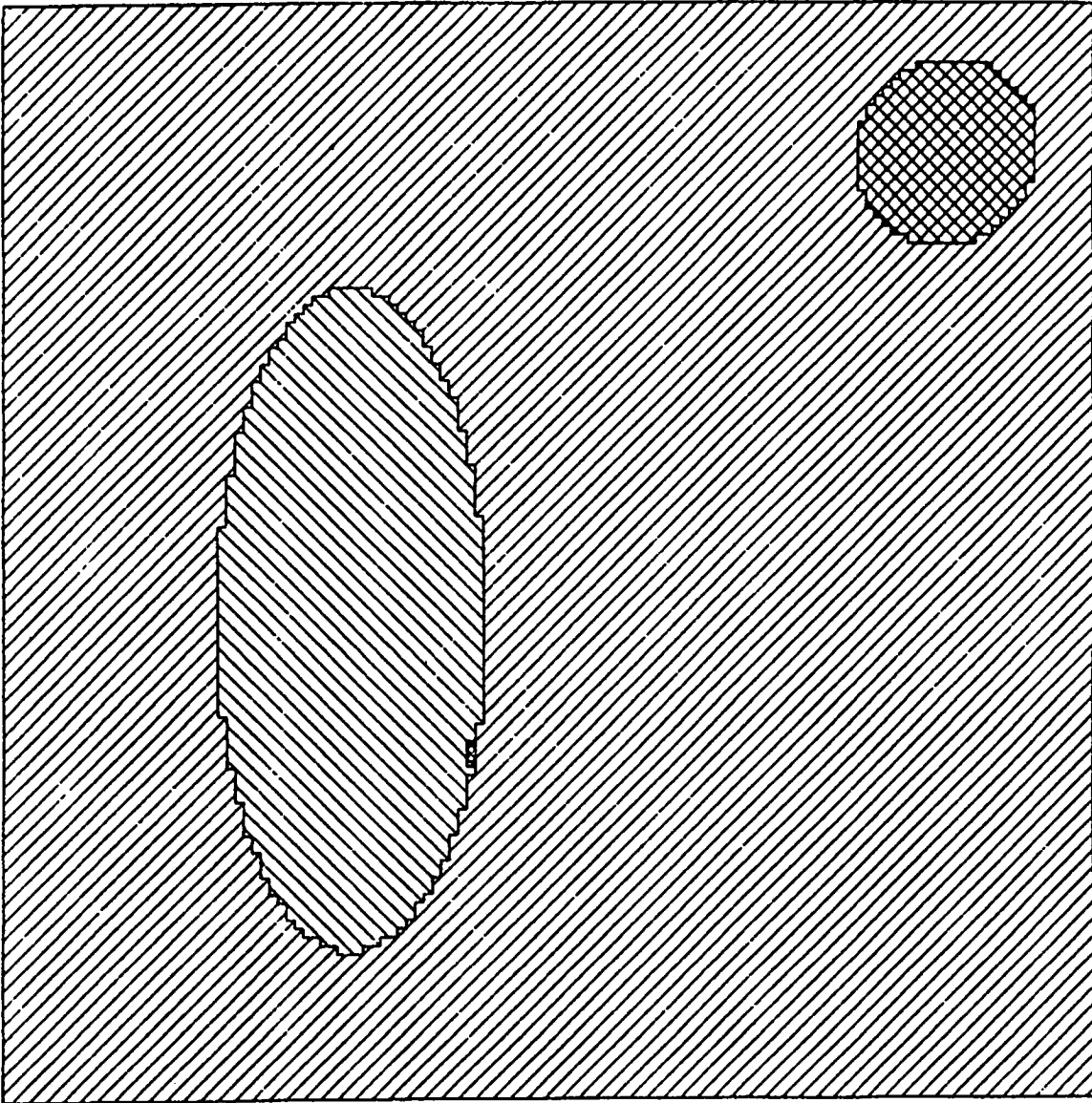


Figure 7.2, continued: (e) Final segmentation.

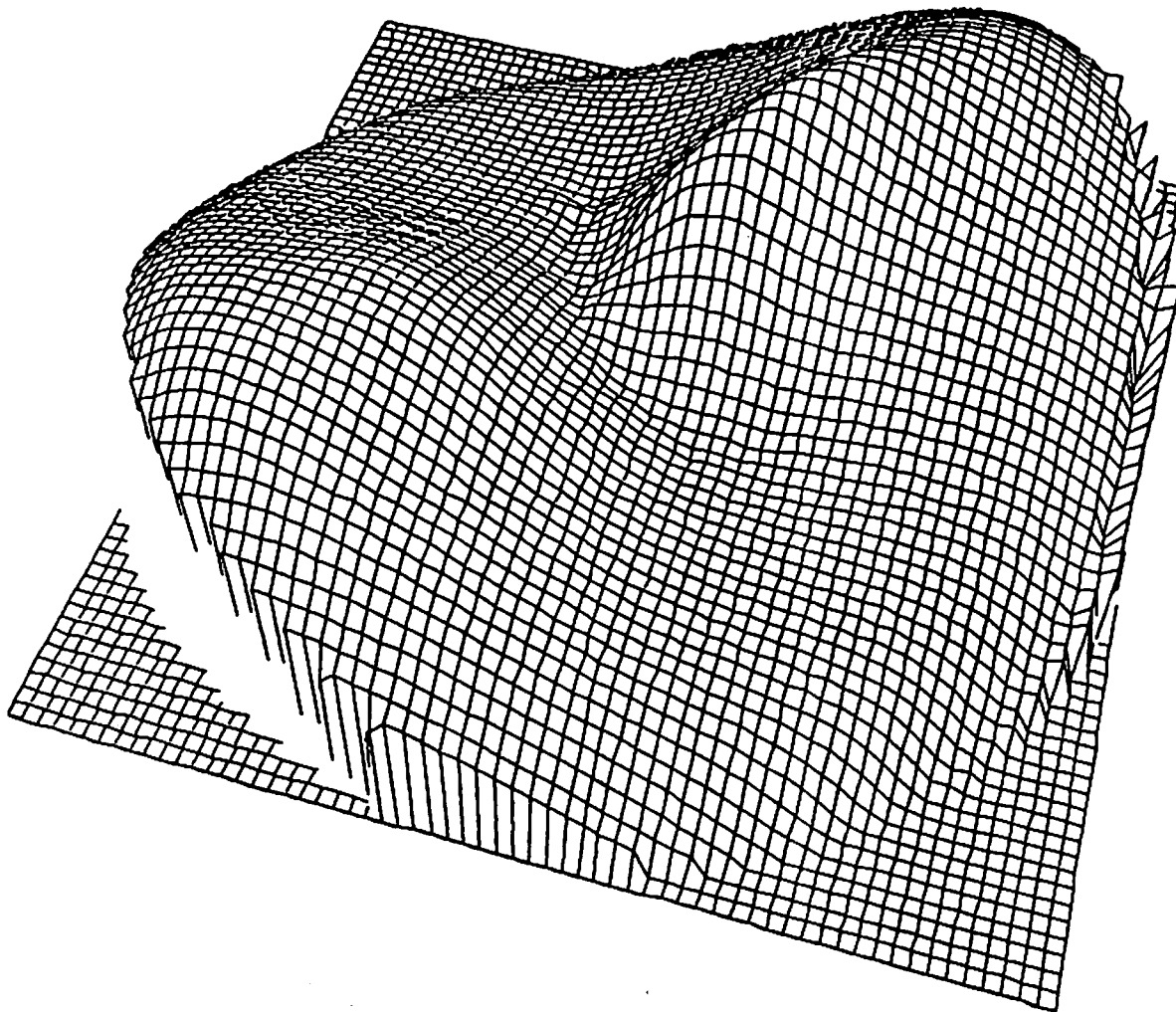


Figure 7.2, continued: (f) The error function $\delta_{\mathcal{N}}$, shown inverted, corresponding to the stationary environment. The range of the function is $[0.281, 2.710]$. The solution obtained by minimizing this function is $\hat{\mathcal{U}}_C = (0.3899, 0.4307, 0.8277)$ and $\hat{\mathcal{Q}}_C = (1.17^\circ, -1.12^\circ, 2.83^\circ)$.

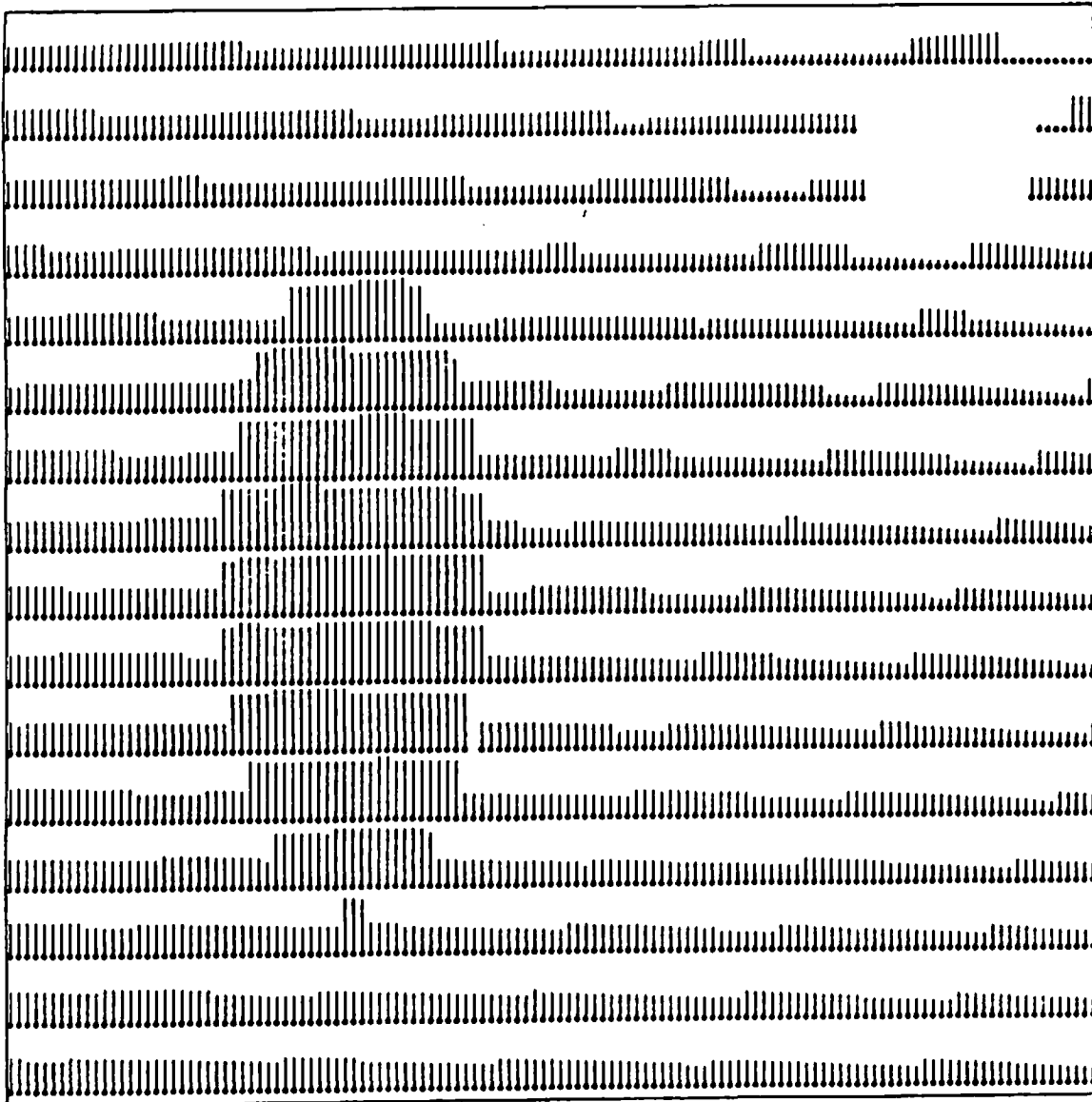


Figure 7.2, continued: (g) The depth function r/Z corresponding to the stationary environment. The round-off error has a strong effect, especially in the upper right corner, near the focus of expansion.

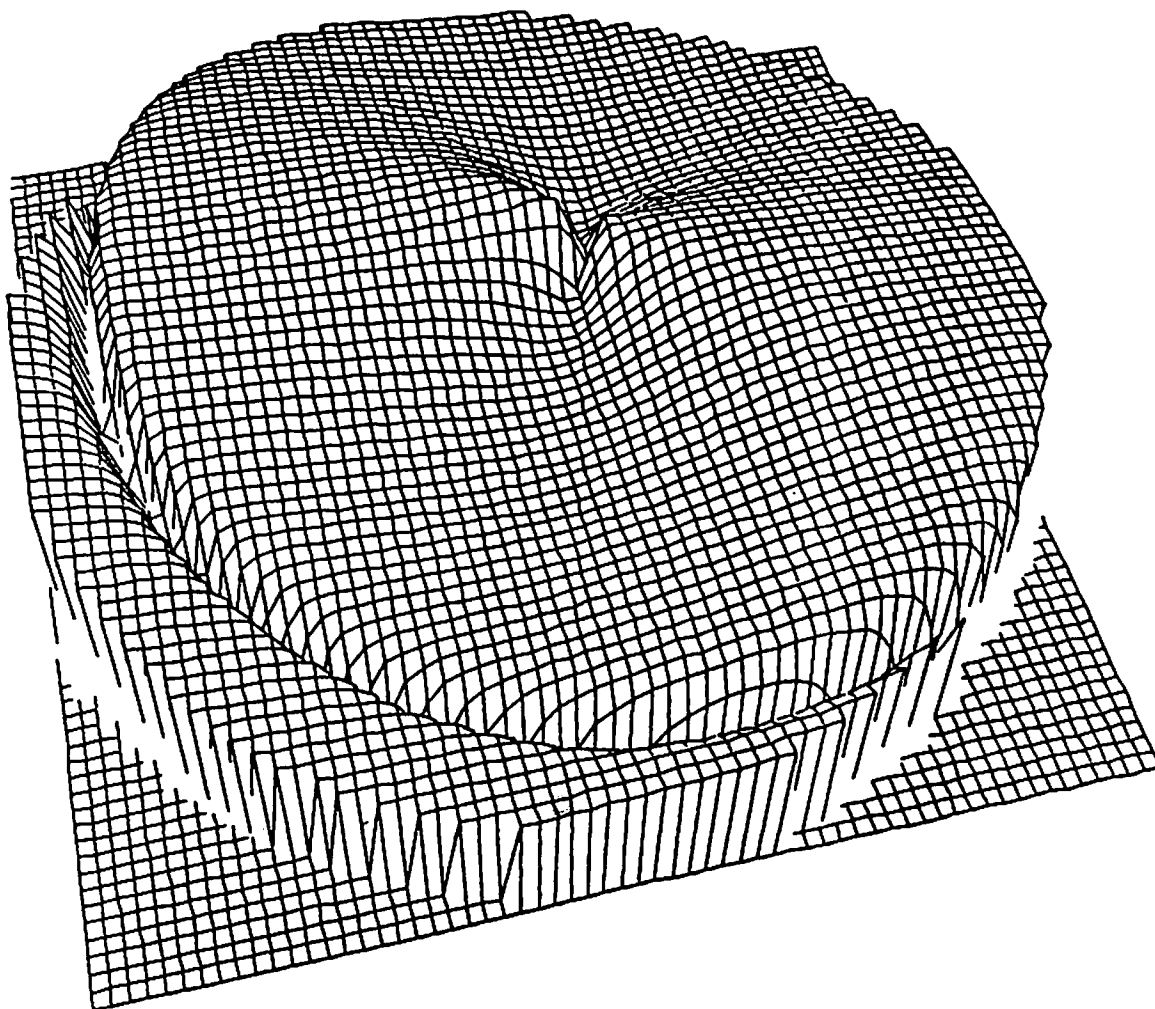


Figure 7.2, continued: (h) The error function δ_N , shown inverted, corresponding to the moving object. The range of this function is $[0.257, 0.390]$. The solution found by minimizing this function is $\hat{U}_{0j} = (0.7925, -0.4661, -0.3932)$ and $\hat{\Omega}_{0j} = (-5.96^\circ, -9.37^\circ, -9.13^\circ)$.

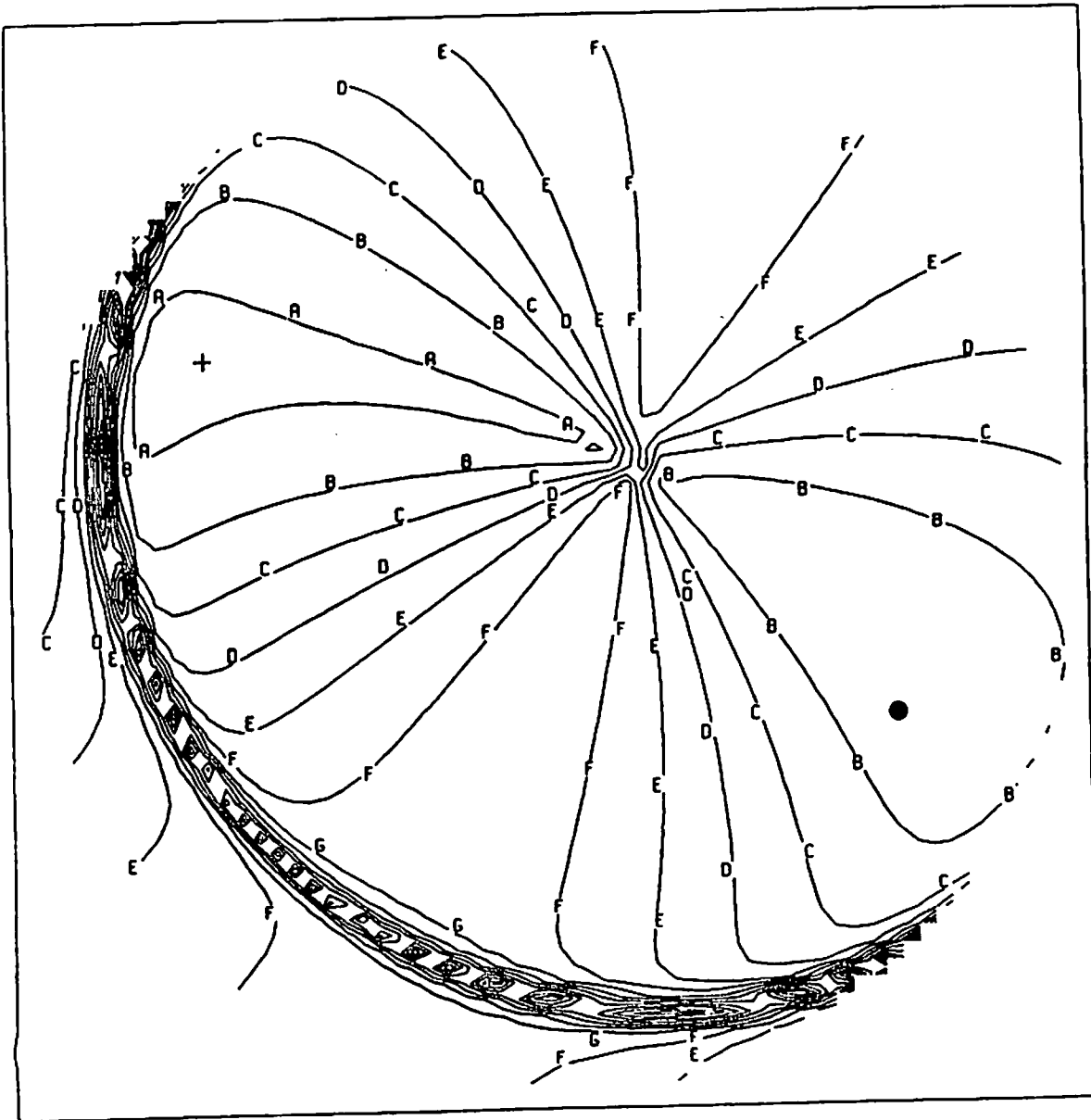


Figure 7.2, continued: (i) The error function corresponding to the sphere is shown again, this time by using a contour map. The contours are labeled by the symbols 'A', 'B', 'C', etc. which represent, respectively, error values (in pixels) of 0.26, 0.27, 0.28, etc. The correct FOE is marked by a dot, and the computed FOE is marked by a cross.

can be relatively approximated by a plane (see Section IV.1).

VII.3 Experiment 3

The third experiment demonstrates the ability of our scheme to interpret sparse flow fields. This experiment is based on real data, originally utilized in [RIE83], which is shown in Figures 7.3a and 7.3b. The scene contains two textured cylinders (oriented vertically and positioned on the left and right sides) in front of a textured plane parallel to the image plane. The camera was translated approximately in the direction of the Z -axis ($\underline{U}_C \approx (0.0261, 0.0249, 0.9993)$), and then rotated to the left about its Y -axis. Figure 7.3c shows the flow vectors determined by correlating windows which were centered at interesting points extracted from the image [RIE83]. The weight assigned to each vector is 1, since no reliability measure was computed.

The results of the segmentation process are shown in Figure 7.3d. Note that these results were already produced by the first segmentation stage, since the three surfaces can be relatively approximated by planar surfaces (see Section IV.1) which are parallel to the image plane and, therefore, the flow field corresponding to each of them is compatible with an affine transformation.

The three segments are consistent with the same camera motion. Figure 7.3e displays the corresponding error function $\hat{\sigma}_\lambda$. Assuming stationary environment, the recovered motion parameters of the camera are $\hat{\underline{U}}_C = (-0.0079, 0.0181, 0.9998)$ and $\hat{\underline{\Omega}}_C = (-0.1^\circ, -1.16^\circ, 0.01^\circ)$. The error in determining the translational axis is 2° . This relatively large error may partly be due to the use of the approximating velocity equations (3.5) instead of the accurate displacement equations (3.11), while the values of T_Z/Z are relatively large for the left and right surfaces.

The three distinct surfaces in the environment and their relative distances from

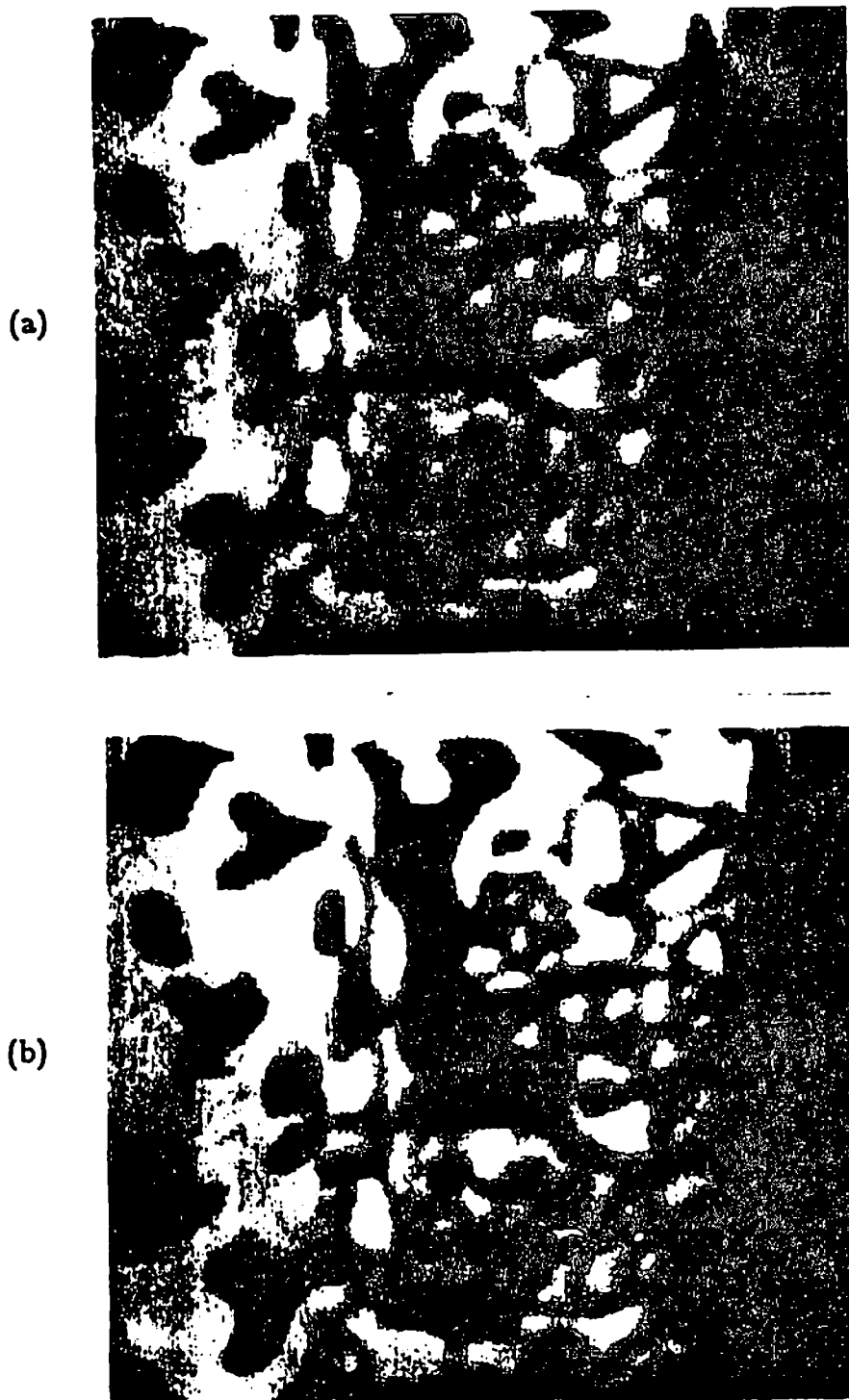


Figure 7.3: Experiment 3. (a) The first intensity image. (b) The second intensity image.

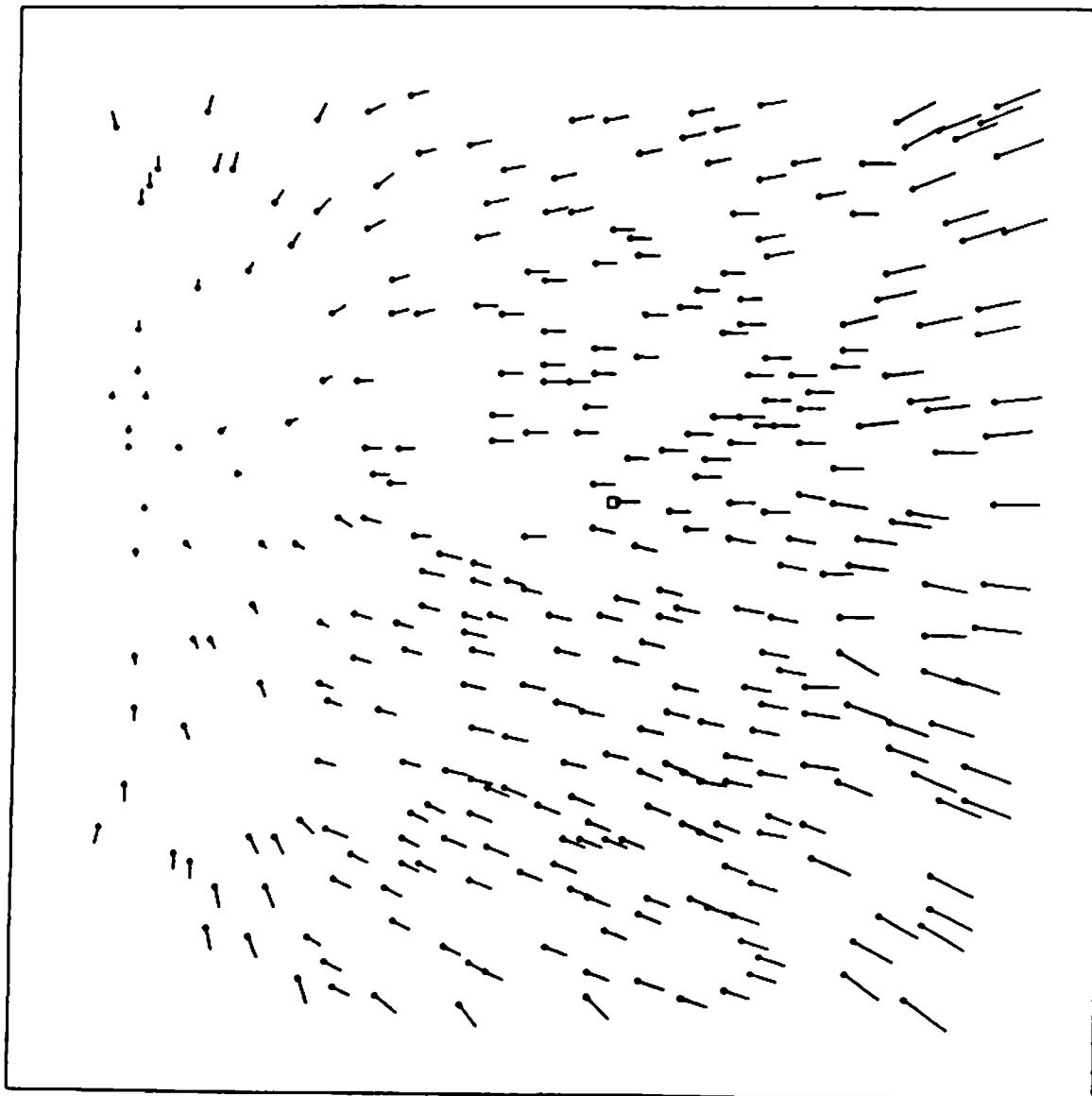


Figure 7.3, continued: (c) The flow field produced in [RIE83]. The vectors are scaled by 0.5. The FOE is marked by a square near the image center.

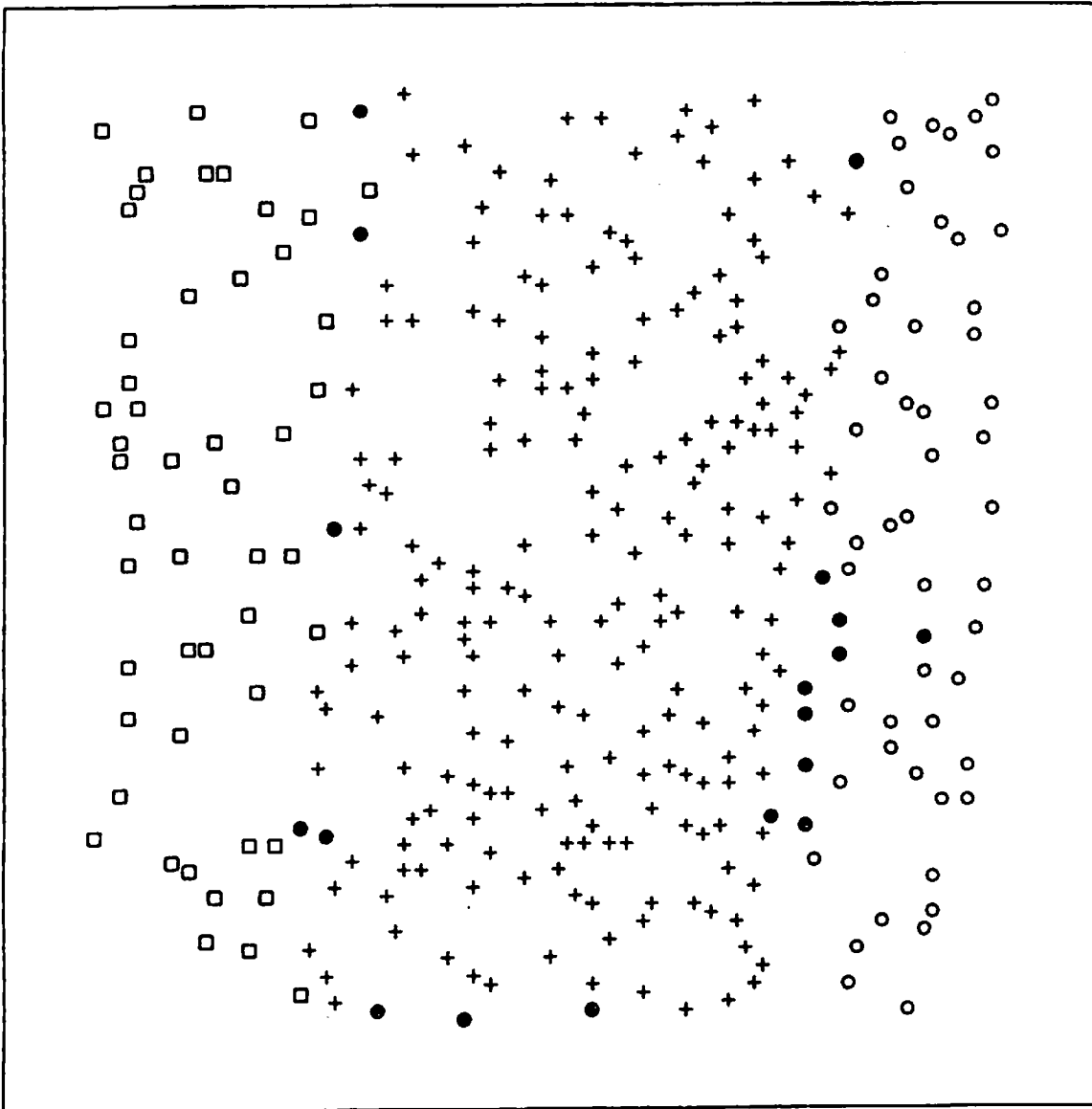


Figure 7.3, continued: (d) Final segmentation. Each segment is represented by a distinct shape; the black dots correspond to flow vectors which are not contained in any of the segments.

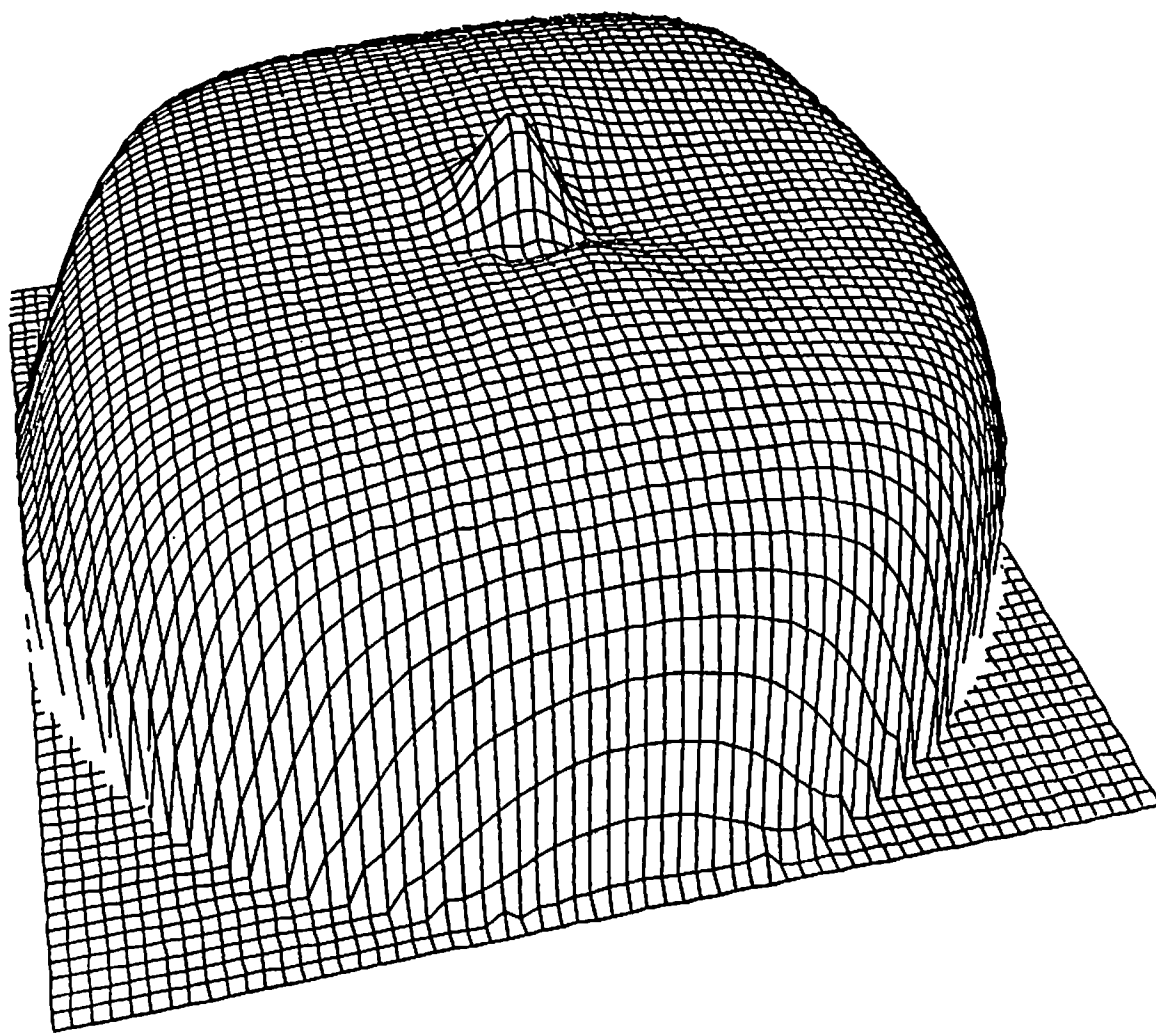


Figure 7.3, continued: (e) The error function $\delta_{\mathcal{N}}$ shown inverted. The range of the function is $[0.356, 3.571]$. The solution obtained by minimizing this function is $\hat{\underline{U}}_C = (-0.0079, 0.0181, 0.9998)$ and $\hat{\underline{\Omega}}_C = (-0.1^\circ, -1.16^\circ, 0.01^\circ)$.

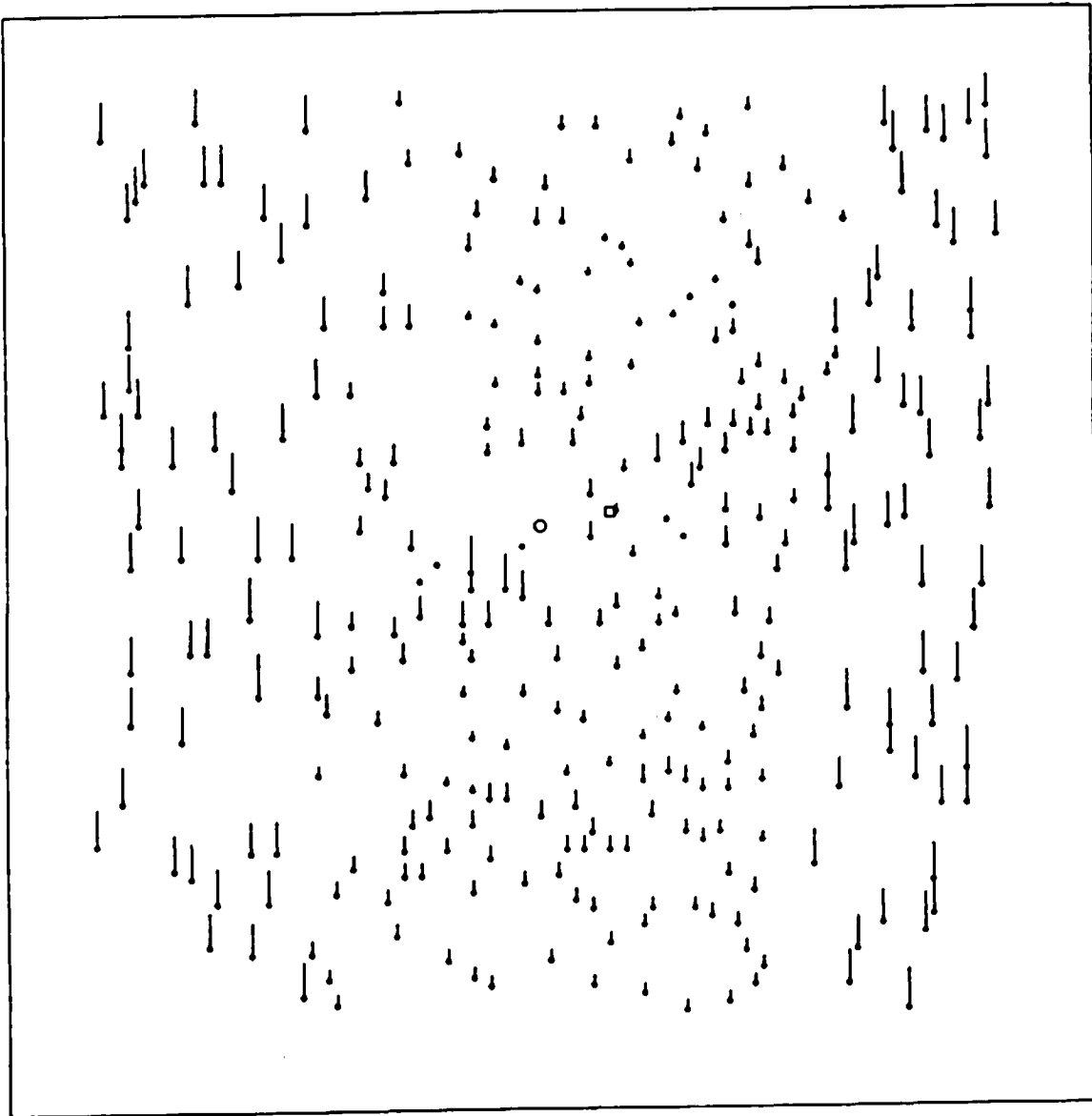


Figure 7.3, continued: (f) The estimated depth function r/Z . The correct FOE is marked by a square, and the computed FOE is marked by a circle.

the camera can be perceived in the depth map in Figure 7.3f. It is evident that the surfaces on the left and right sides are much closer to the camera than the central surface; the average values of r/Z for the left, right and central surfaces are, respectively, 0.1049, 0.1036 and 0.0356. It is also clear that the depth variations for the left and right surfaces are small compared to their distance from the camera; the standard deviations of r/Z values for these surfaces are, respectively, 0.0079 and 0.0083. Notice, however, the relatively large variation in the depth values computed for the central surface (standard deviation of 0.0152), whereas the actual values are approximately constant in this area. These errors can be related to the location of the FOE in the region corresponding to this surface, and to the error in estimating the FOE location; as in experiments 1 and 2, they are particularly large near the computed FOE (see Figure 7.3f).

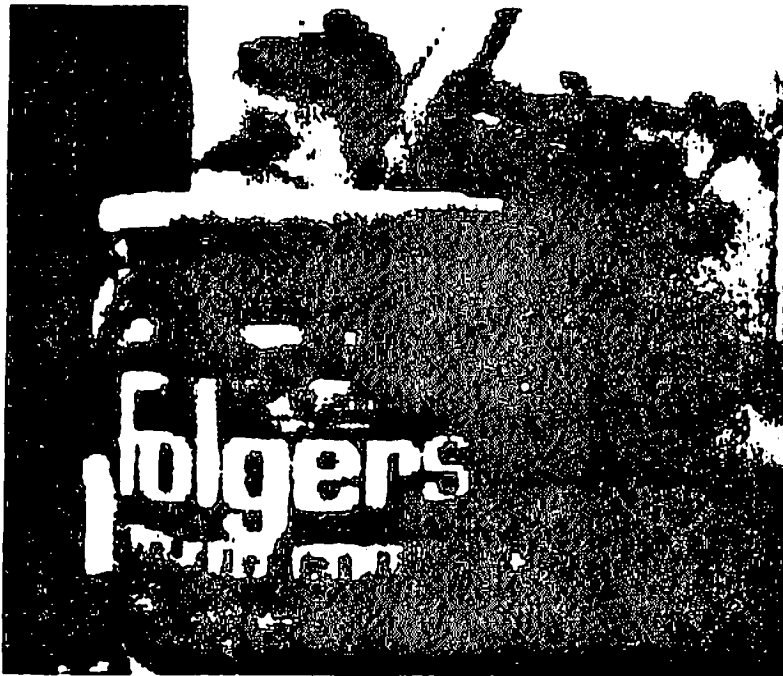
VII.4 Experiment 4

Figures 7.4a and 7.4b are images taken from a camera translated in the direction of its X -axis. The scene mainly contains a coffee can in the front and a plant in the background. The flow field (Figure 7.4c) and a related confidence measure (Figure 7.4d) were computed employing a technique developed by Anandan [ANA84]. The confidence values were used for weighting the contributions of the flow vectors.

The four segments in Figure 7.4e were already obtained in the first segmentation stage, because, as in experiment 3, the surfaces are approximately parallel to the image plane and the flow field in each of them is compatible with an affine transformation. The segments roughly correspond to distinct surfaces in the image. Their boundaries are inaccurate, however, because of errors in the given flow field, especially at occlusion boundaries and in homogenous areas.

All the segments were found to be consistent with the same motion parameters. The corresponding error function is shown in Figure 7.4f. The optimal motion pa-

(a)



(b)



Figure 7.4: Experiment 4. (a) The first intensity image. (b) The second intensity image.

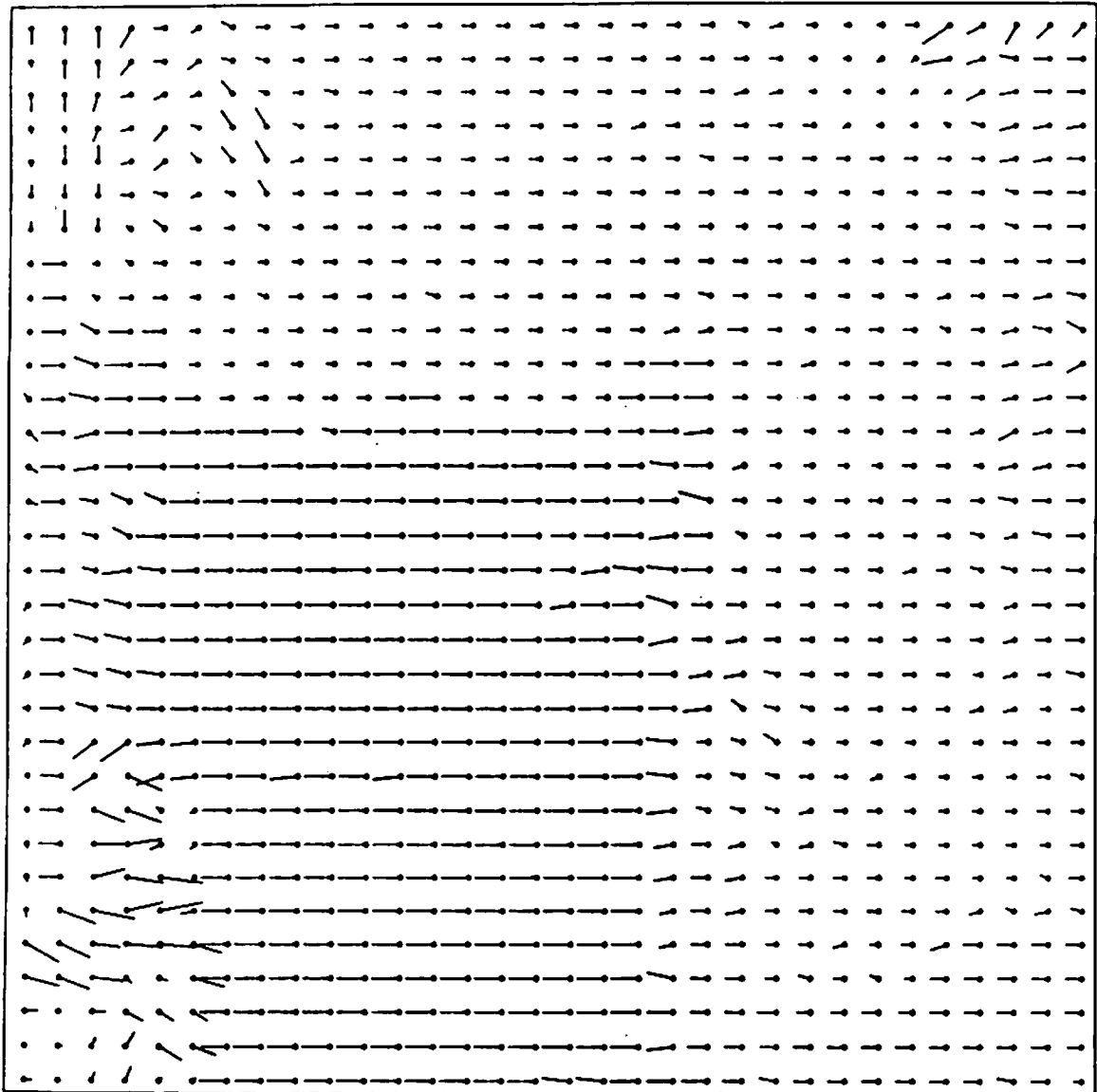


Figure 7.4, continued: (c) A 32×32 sample of the computed flow field. The vectors are scaled by 0.33.



Figure 7.4, continued: (d) The weight plane. High values are represented by bright gray levels.

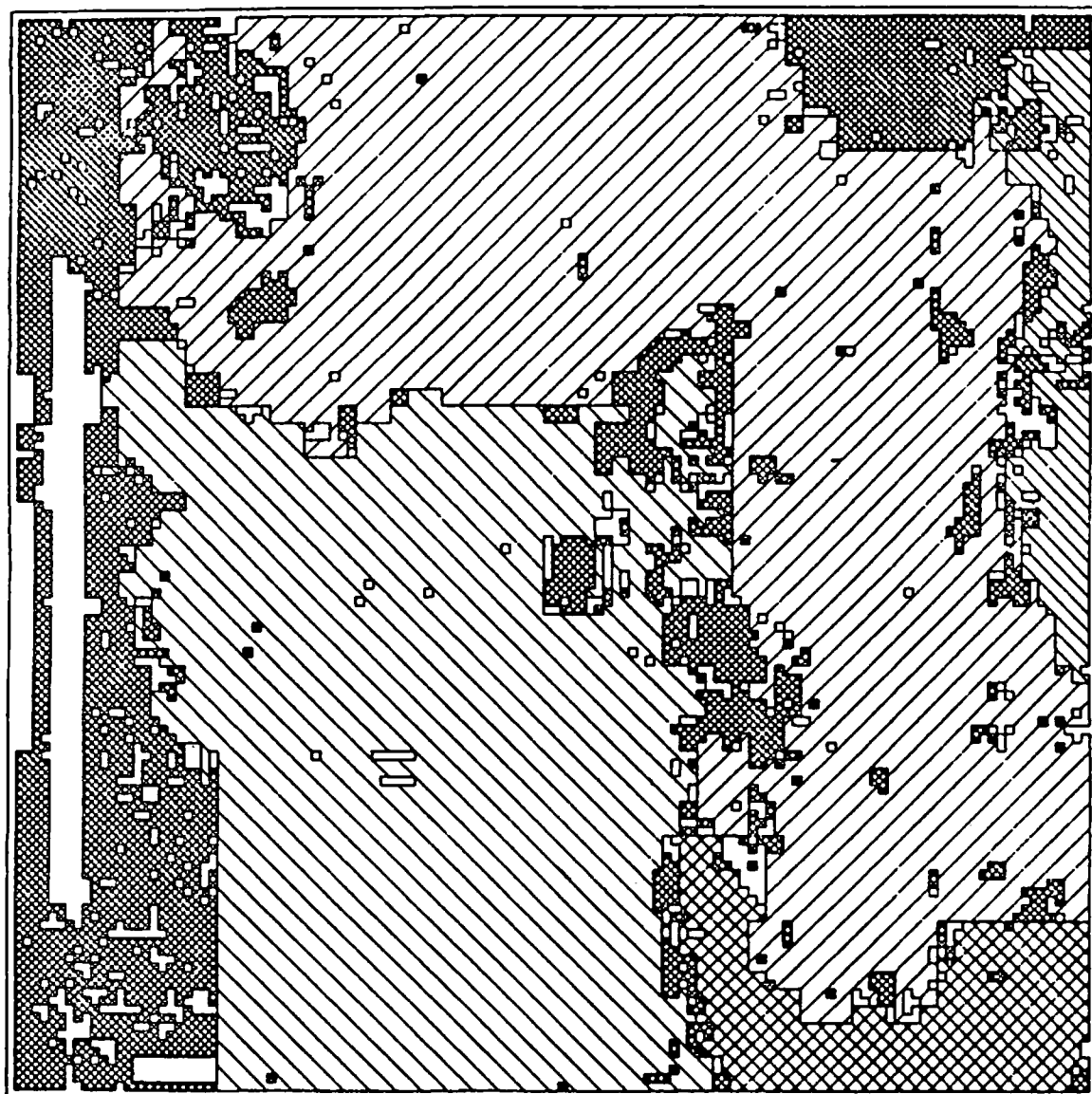


Figure 7.4, continued: (e) Final segmentation. The white areas correspond to flow vectors assigned weight 0. The areas with the densest pattern correspond to unsegmented vectors.

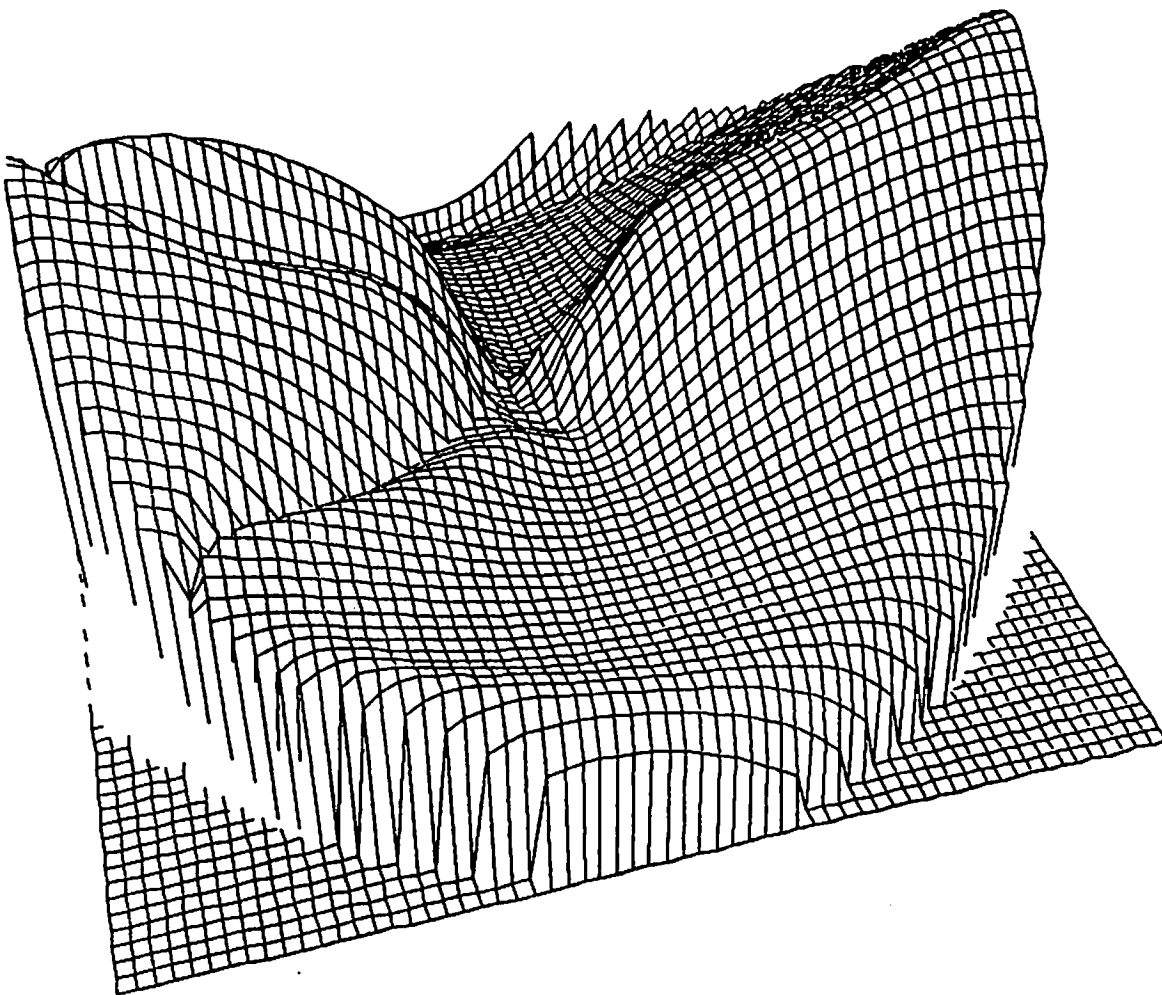


Figure 7.4, continued: (f) The error function $\hat{\sigma}_\lambda$ shown inverted. Note the two peaks which actually correspond to the same translation, because $\hat{\sigma}_\lambda$ is invariant to sign change in the translation vector. The range of the function is $[0.120, 2.190]$. The motion parameters obtained by minimizing this function are $\hat{U}_C = (1., 0., 0.)$ and $\hat{\Omega}_C = (0^\circ, 0.11^\circ, 0^\circ)$.

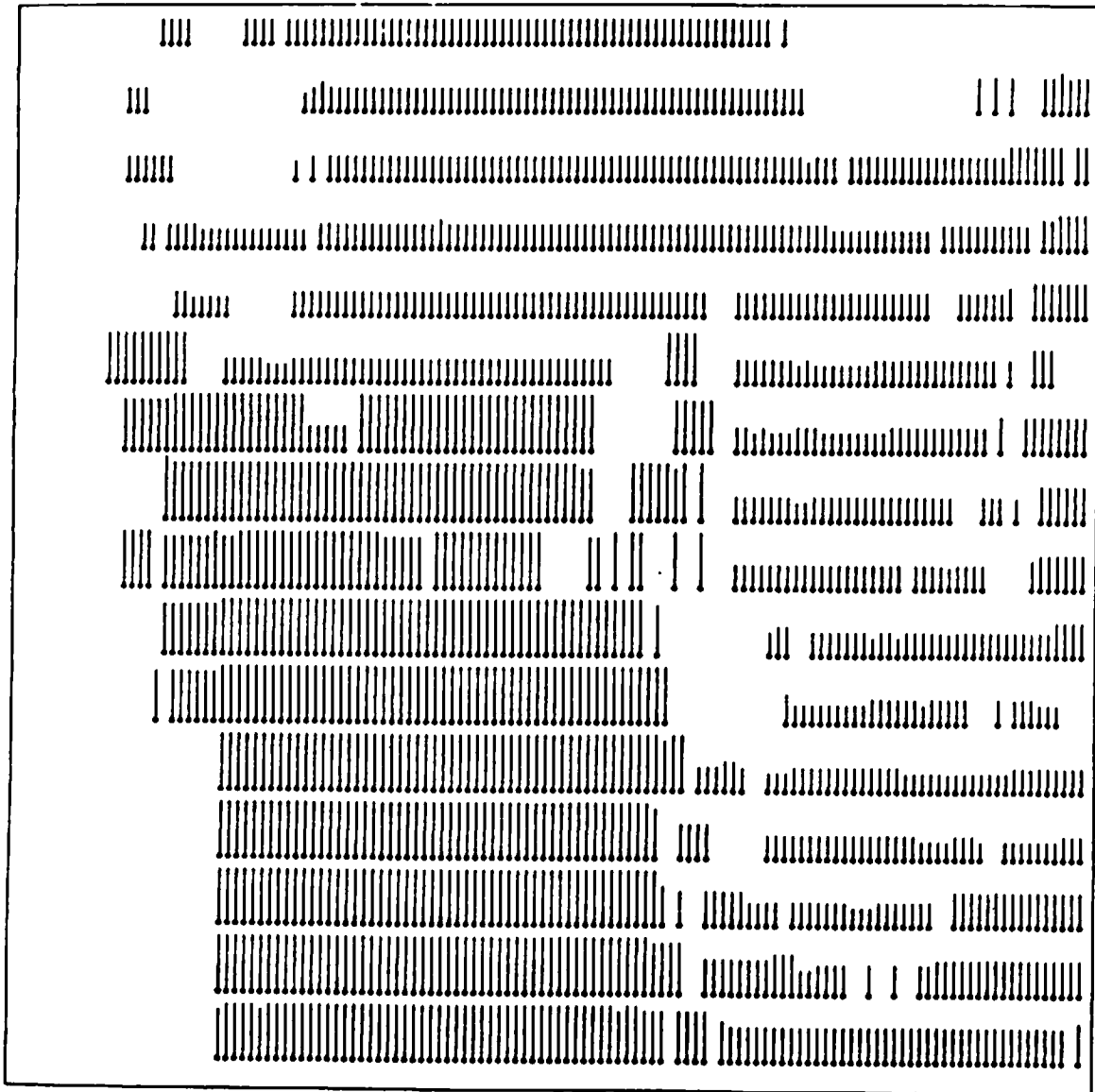


Figure 7.4, continued: (g) The estimated depth function r/Z . The depth values are computed only for the segmented flow vectors.

rameters of the camera, obtained by minimizing this function, are $\hat{\underline{U}}_C = (1., 0., 0.)$ and $\hat{\underline{\Omega}}_C = (0.^{\circ}, 0.11^{\circ}, 0.^{\circ})$. These results are very close to the correct ones; there is no error at all in determining the translation axis. Note the two peaks in the error surface which actually correspond to the same solution, that is, a translation along the X -axis. These two peaks occur because the error function $\hat{\sigma}_\mu$ is invariant to sign change in the translation vector and, therefore, the solution can be represented by either $\hat{\underline{U}}_C = (1., 0., 0.)$ or $-\hat{\underline{U}}_C = (-1., 0., 0.)$. In other words, the FOE can be either on the right, and then the translation is *from* the FOE, or on the left, and then the translation is *towards* the FOE.

Figure 7.4g shows the corresponding 'reciprocal depth' map, namely, r/Z . The fact that the surfaces are roughly parallel to the image plane, as well as the relative distances of the surfaces from the camera, can easily be deduced from this depth map.

VII.5 Experiment 5

Figures 7.5a and 7.5b are images taken from a camera translated in the direction of its X -axis and then rotated 1.5° about its Y -axis. In addition, to introduce an independently moving object, a toy 'dinosaur' was rotated approximately 4.2° about an axis perpendicular to the image plane. This moving object was placed on a tea can covered by a sheet of white paper. The flow field (Figure 7.5c) and a related weight plane (Figure 7.5d) were computed employing again the technique in [ANA84].

The three segmentation stages are shown in Figures 7.5e, 7.5f and 7.5g. As in experiment 4, the segments correspond to the main surfaces in the environment, but their boundaries are inaccurate because of errors in the computed flow field, and because of the continuity of flow fields across boundaries between regions corresponding to surfaces in similar depths (relative to their distance from the sensor).

One should remember that the segmentation is based purely on the flow field and no information from the intensity images is utilized other than that employed in the computation of the flow vectors. When evaluating the segmentation results, the reader should examine Figure 7.5h which shows together the flow field used as input to the segmentation process, the grouping of the flow vectors into segments, and the 'correct' boundaries. In general, the problems mentioned above in computing flow fields imply that only limited goals should be set for segmentation based on motion analysis or, more importantly, it should be combined with analysis of intensity data.

A correct and unique grouping of the segments into objects was determined. An error function, based on the three segments corresponding to the stationary environment, is displayed in Figure 7.5i. Minimizing this function, the translation axis and the rotation parameters of the camera were determined to be $\hat{U}_C = (0.9996, -0.0258, -0.0059)$ and $\hat{\Omega}_C = (-0.05^\circ, -1.68^\circ, 0.61^\circ)$, in a reasonable agreement with the actual values; the error in determining the translation axis is 1.5° .

In Figure 7.5j, which shows the 'reciprocal depth' map r/Z , three distinct surfaces can be detected. Two of these surfaces are roughly parallel to the image plane, and the third one is slanted; these orientations are consistent with the actual environment.

The error function associated with the independently moving object is displayed in Figures 7.5k and 7.5l. As in experiment 2, the error function is very flat and, therefore, the translation axis can not reliably be determined, thus demonstrating again the instability which may exist in recovering the motion parameters. The main reason for the ambiguity in this case is the small depth variation in the object relative to its distance from the camera. The correct parameter values of the motion of the object relative to the camera are $\underline{U}_{O/O} \approx (-0.9962, -0.0872, 0.)$ and $\underline{\Omega}_{O/O} \approx (0.^\circ, 1.5^\circ, 4.2^\circ)$, while the computed values

are $\hat{U}_{O/C} = (-0.9411, -0.3365, 0.0332)$ and $\hat{\Omega}_{O/\sigma} = (-2.2^\circ, 6.3^\circ, 4.7^\circ)$. The correct and computed values of the FOE are shown in Figure 7.51; they differ from each other by about 15° . The large errors in Ω_X and Ω_Y are associated with large errors in the relative depth values r/Z ; the estimated values of r/Z are approximately 0.13, while the correct values are about 0.04. On the other hand, the computed values of $T_Z/Z (= U_Z r/Z)$ and Ω_Z are fairly accurate as can be expected from the analysis in Section VI.1.4 for the case where the surface can be relatively approximated by a planar surface parallel to the image plane. Finally, note the additional peak of the error surface near the center of the display in Figure 7.51. This peak may be related to the duality of planar surface solutions.

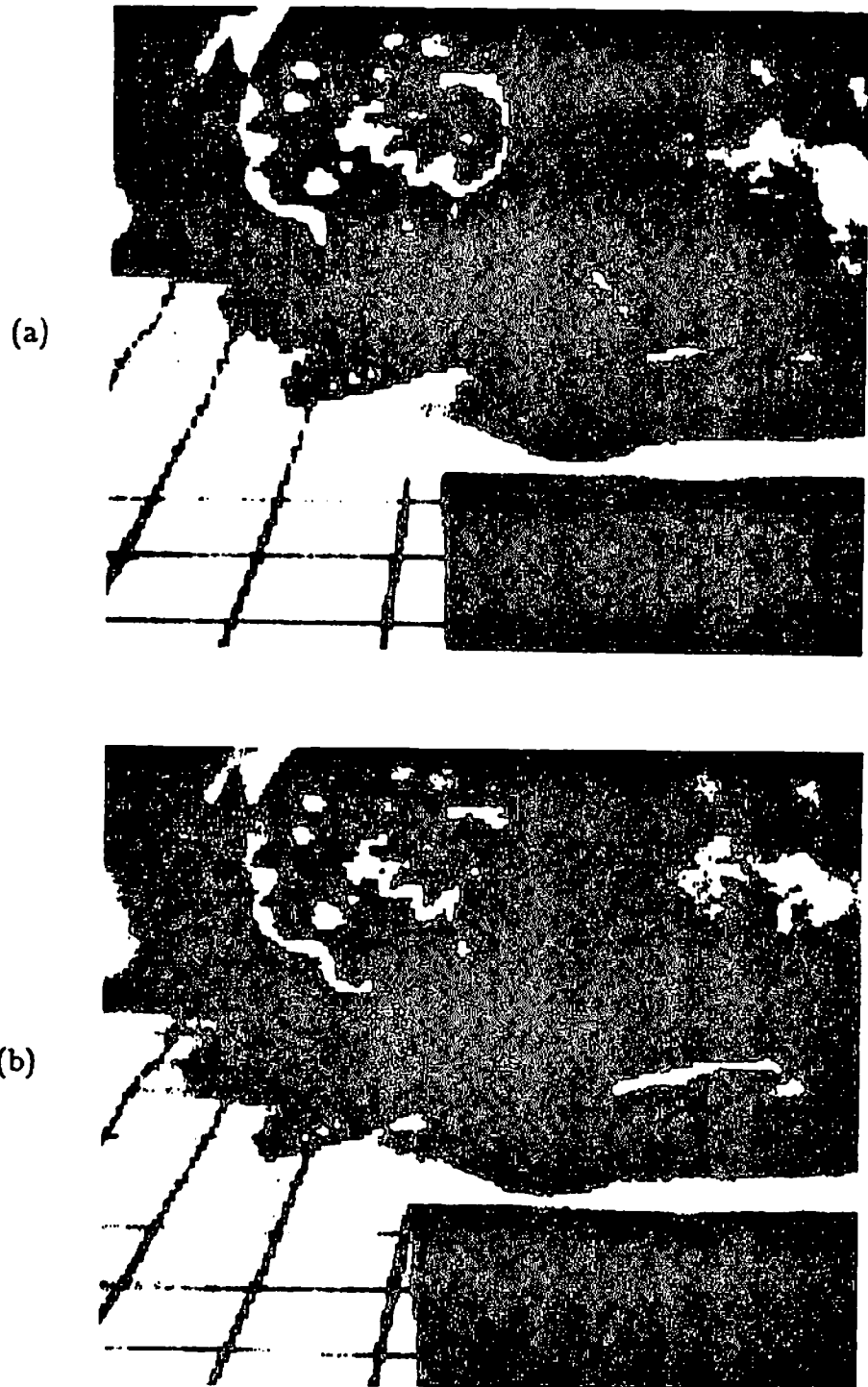


Figure 7.5: Experiment 5. (a) The first intensity image. (b) The second intensity image.

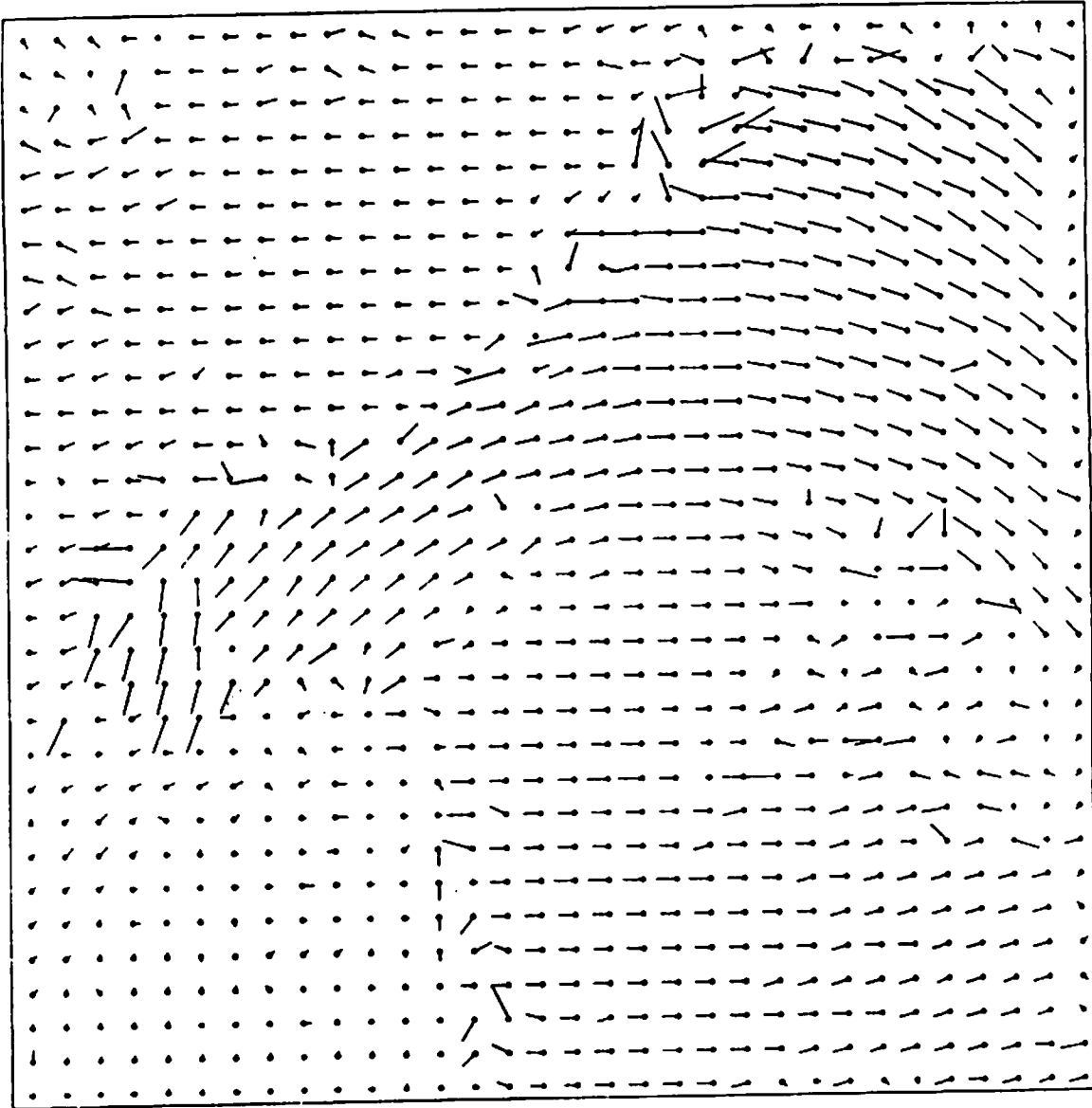


Figure 7.5, continued: (c) A 32×32 sample of the computed flow field. The vector lengths are scaled by 0.5.



Figure 7.5, continued: (d) The weight plane. High values are represented by bright gray levels.



Figure 7.5, continued: (e) Components found in the first segmentation stage. The white areas correspond to flow vectors assigned weight 0. The areas with the densest pattern correspond to ungrouped vectors.

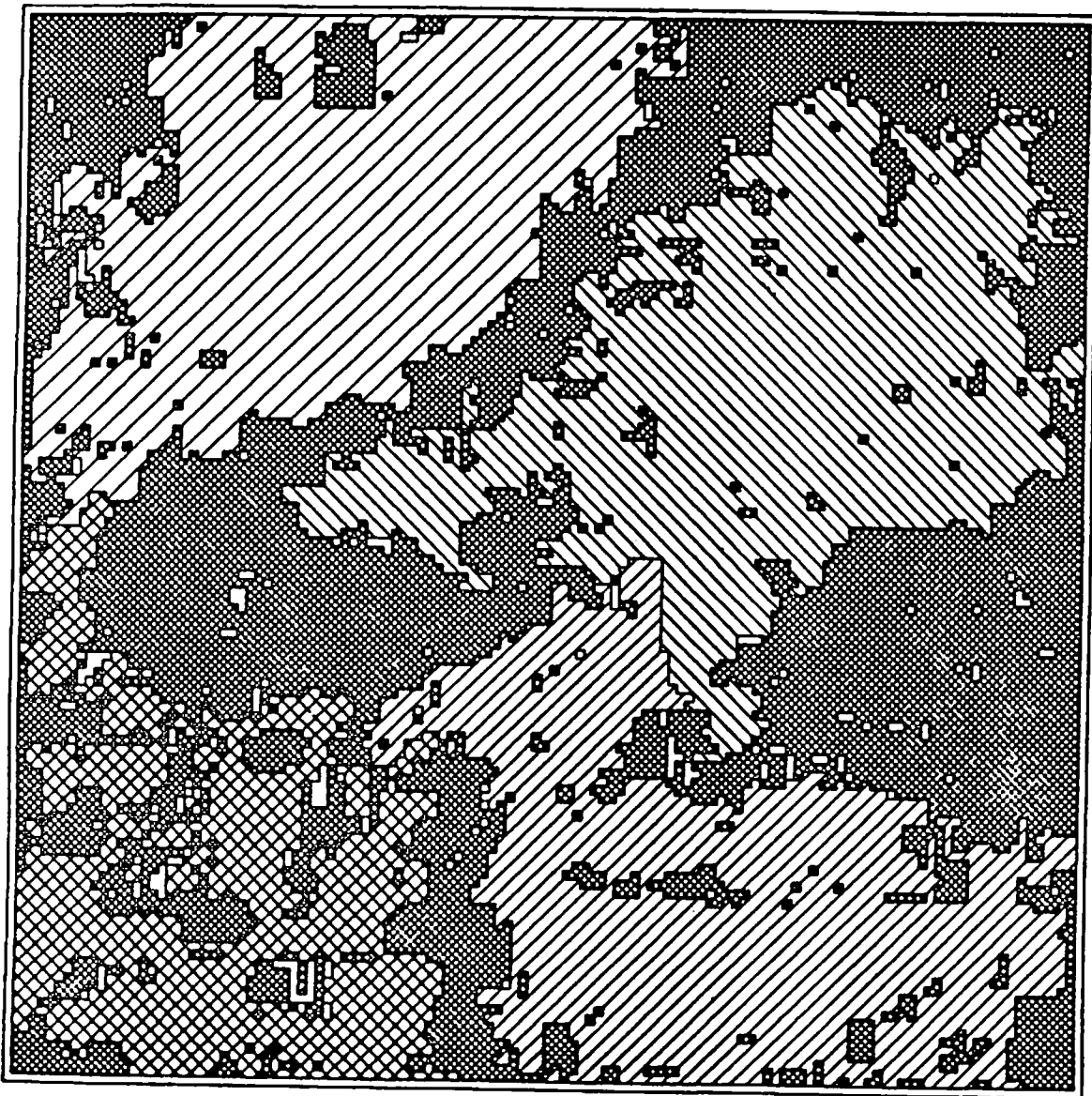


Figure 7.5, continued: (f) Segments obtained by merging components in the second segmentation stage.

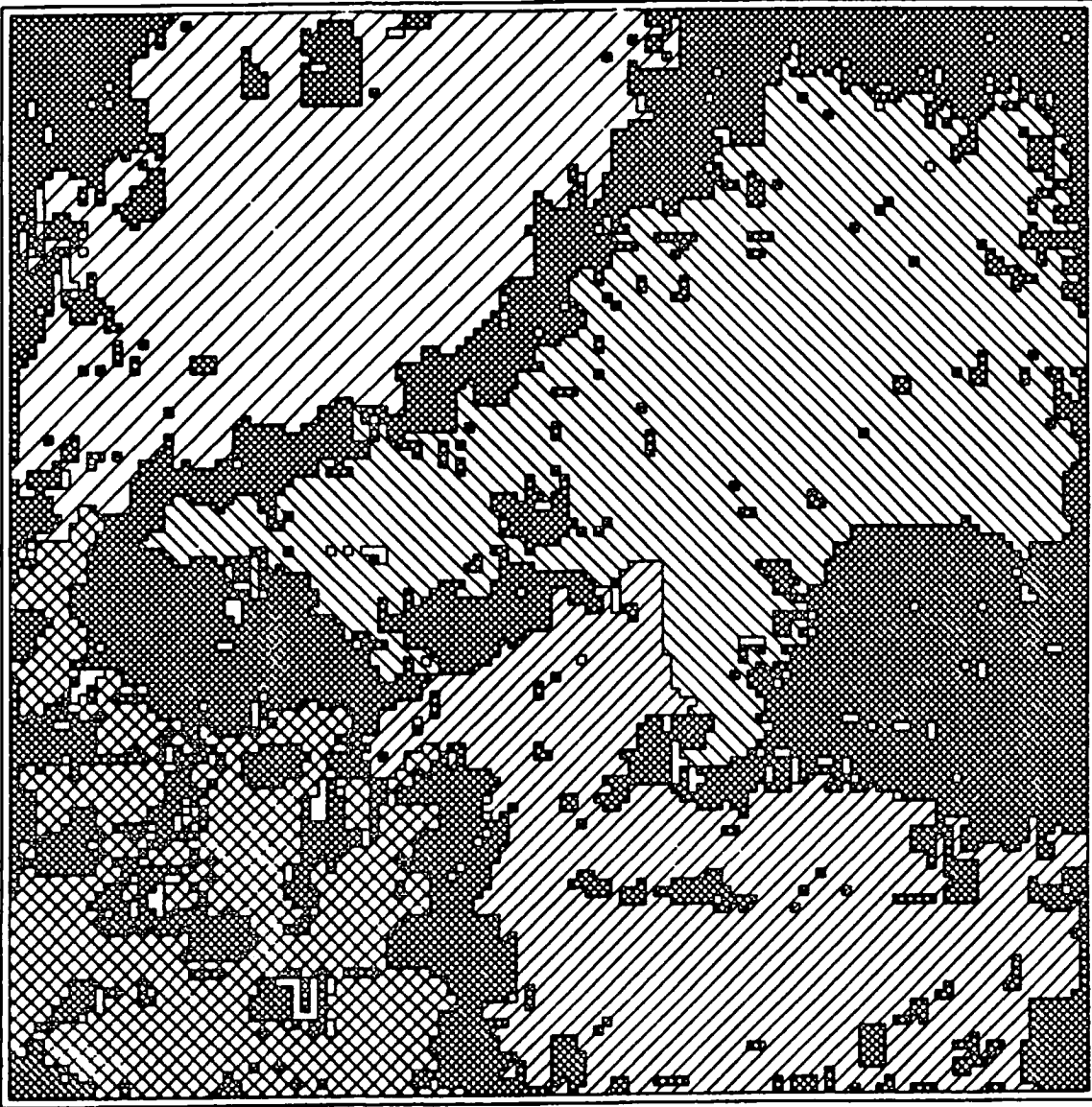


Figure 7.5, continued: (g) Final segmentation.

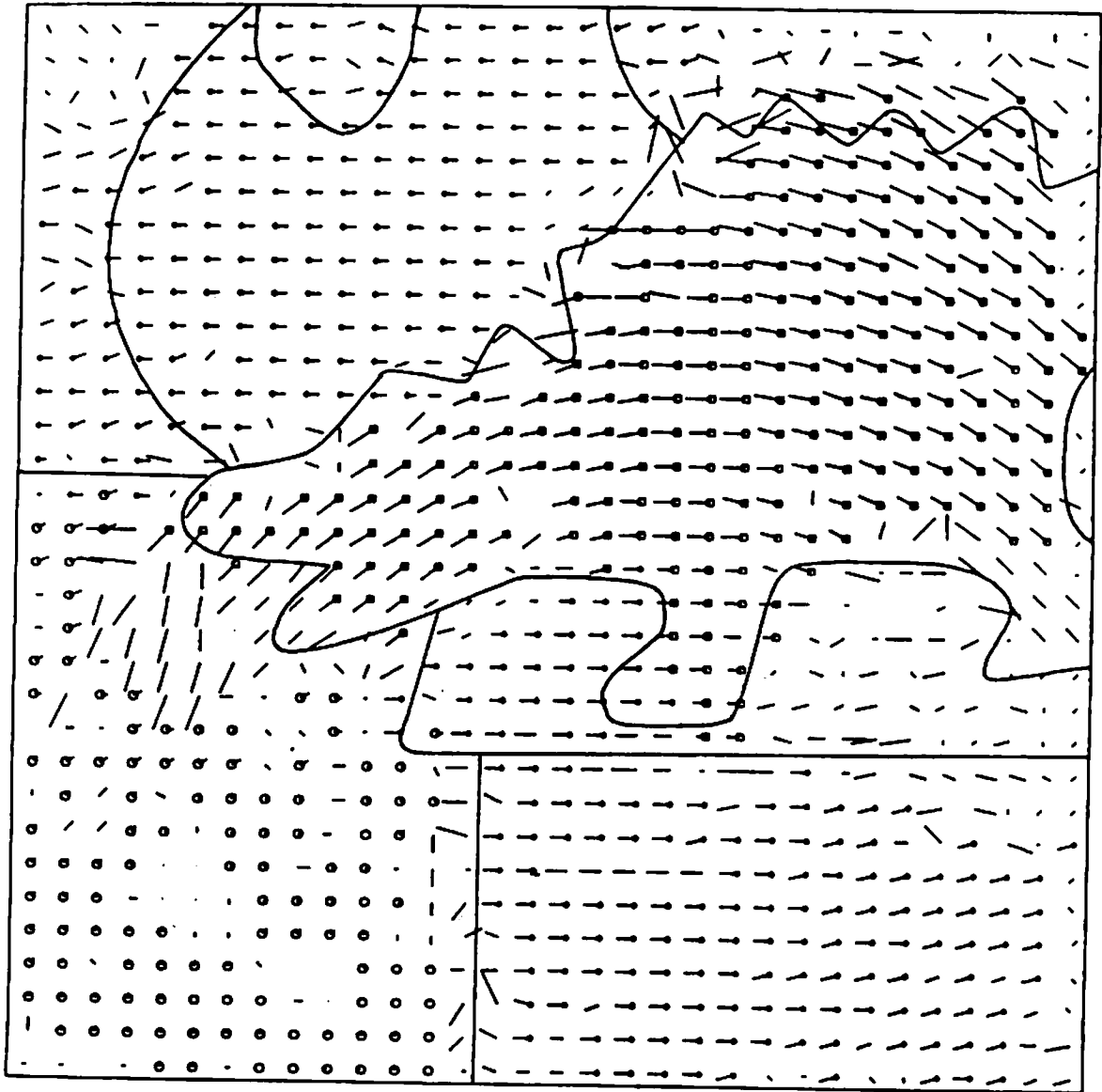


Figure 7.5, continued: (h) The grouping of the flow vectors into segments is shown by using various shapes of the vector tails. Vectors without a tail are ungrouped. In addition, the 'correct' boundaries are drawn.

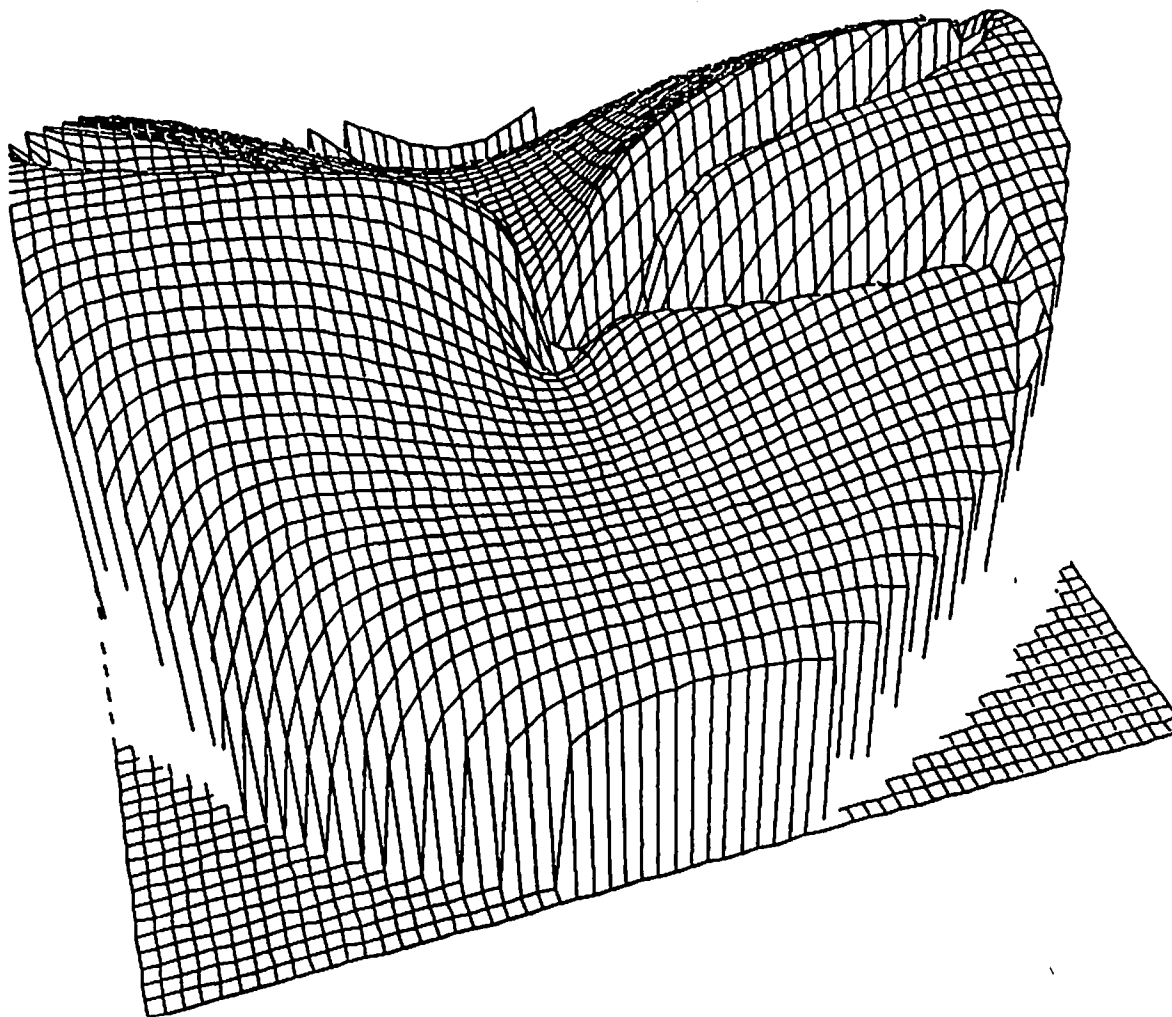


Figure 7.5, continued: (i) The error function δ_M , shown inverted, corresponding to the stationary environment. The range of the error function is $[0.297, 2.216]$. The solution obtained by minimizing this function is $\hat{U}_C = (0.9996, -0.0258, -0.0059)$ and $\hat{\Omega}_C = (-0.05^\circ, -1.68^\circ, 0.61^\circ)$.

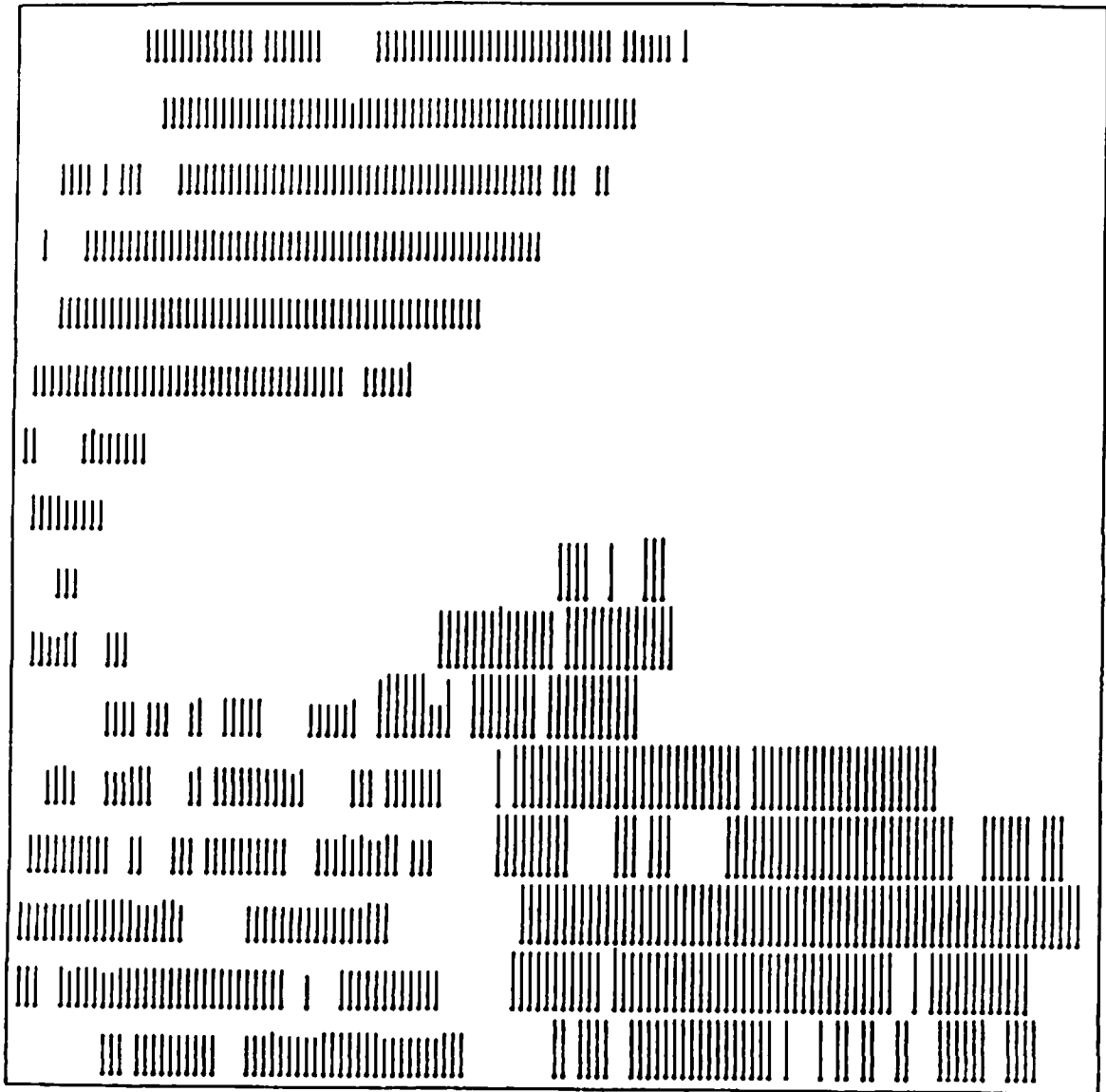


Figure 7.5, continued: (j) The estimated depth function r/Z .

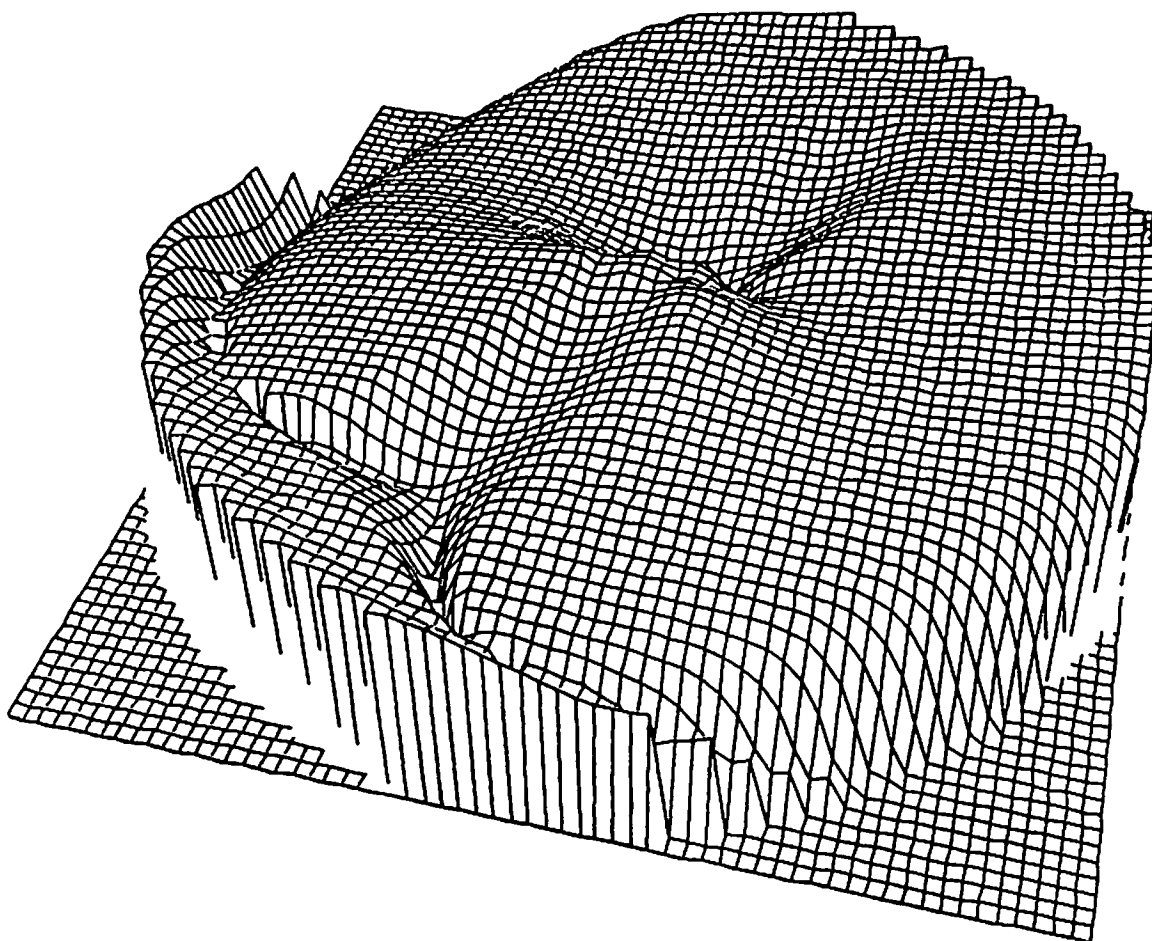


Figure 7.5, continued: (k) The error function, shown inverted, corresponding to the independently moving object. The range of this function is $[0.241, 0.707]$. The solution was determined to be $\hat{\underline{U}}_{O/C} = (-0.9411, -0.3365, 0.0332)$ and $\hat{\underline{\Omega}}_{O/C} = (-2.2^\circ, 6.3^\circ, 4.7^\circ)$.

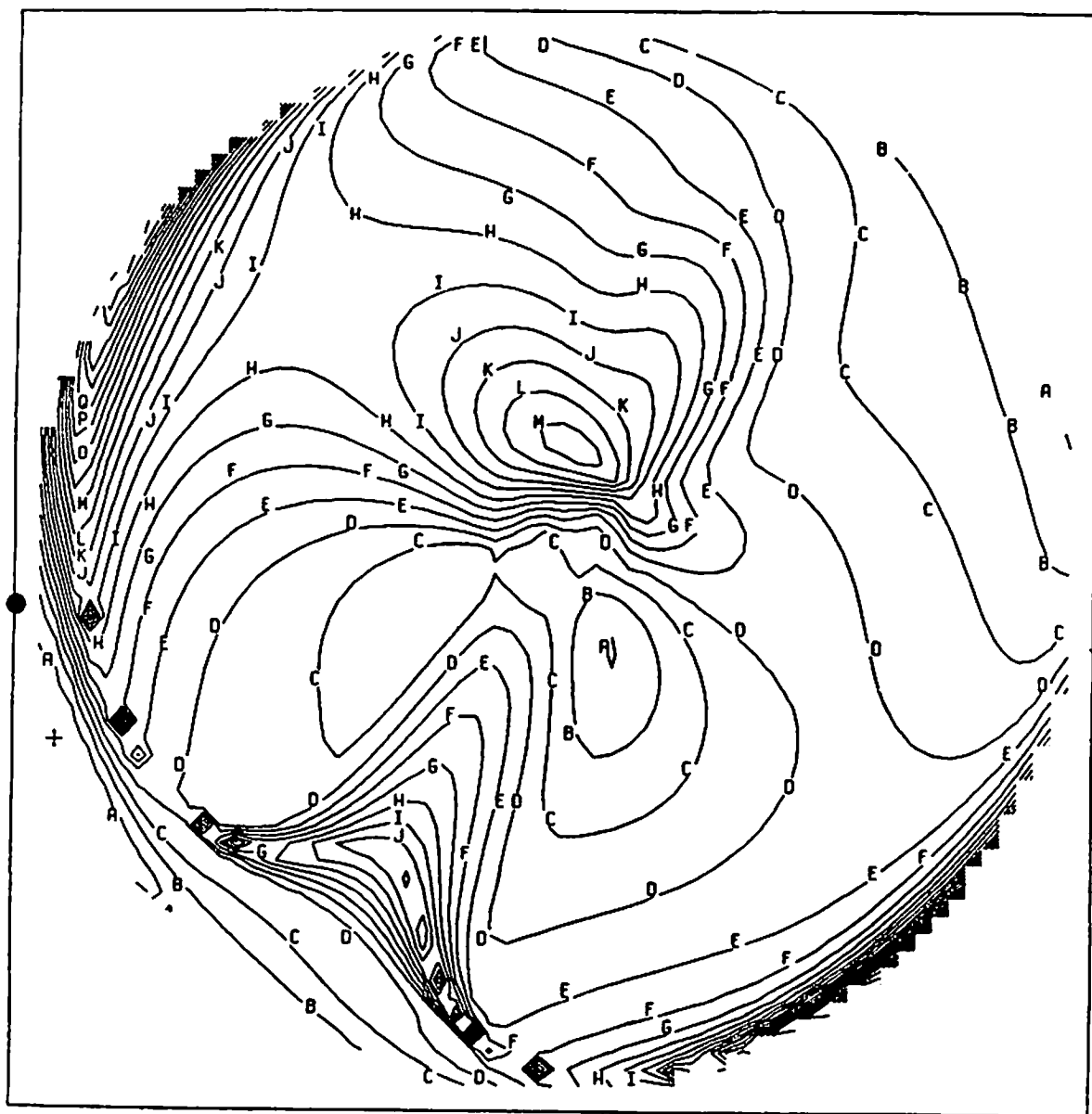


Figure 7.5, continued: (1) A contour map representing the error function which is associated with the moving object. The symbols 'A', 'B', 'C', etc. represent, respectively, error values (in pixels) of 0.25, 0.27, 0.29, etc. The correct FOE is marked by a dot, and the computed FOE is marked by a cross.

CHAPTER VIII

SUMMARY

We have presented a new approach for the interpretation of optical flow induced by general motion of the camera as well as independent motions of several rigid objects in the environment. The interpretation goals included decomposition of the flow field into sets corresponding to independently moving objects, recovery of the rotation parameters and the translation axis of the camera and each moving object, and estimation of the relative depth of environmental surfaces. The feasibility of these goals has been examined for noisy, partially incorrect, and possibly sparse flow fields. An algorithm based on our approach was demonstrated to successfully work with such data derived from both artificial and real images. On the other hand, a mathematical error analysis discovered algorithm-independent ambiguities which may exist in interpreting noisy flow fields. These ambiguities were also demonstrated by experimental results.

Taking into account the sensitivity to noise indicated by experiments in the literature (see Chapter II) and by the ambiguity analysis in Chapter VI, we found it necessary to improve the performance of the algorithm as much as possible by employing a global approach in which all the available information is effectively and efficiently utilized. This goal was achieved by segmenting the flow field and then combining segments to form object hypotheses. Thus, we could deal with independently moving objects while employing all the information associated with each object. Using this method, we also prevented the suppression of valuable data in distinct, but possibly small, surfaces, and we excluded incorrect flow vectors which were inconsistent with their neighbors.

The technique presented in Chapter V for recovering the 3-D motion and structure of a rigid object by a direct minimization of the deviation between the actual

and predicted flow fields also contributed to the robustness to noise. An error measure, based on the least-squares criterion, was defined on the 5-dimensional parameter space corresponding to the recoverable motion parameters. The depth constraints (5.7) were fully incorporated into the procedure for computing the error measure, as opposed to other techniques in the literature. A method for reducing the dimensionality of the search space from five to two made the minimization task relatively straightforward. This method employs an error measure corresponding to possible locations of the FOE. For each hypothesized FOE, corresponding rotation parameters are computed by solving 3 linear equations and, then, the related error measure is estimated by computing lower and upper bounds of this value. A minimum value of the resulting error function is determined, using a multi-resolution sampling technique.

It should be noted that the scheme proposed in Chapters IV and V is based on a hierarchical structure, in which four levels of organization in the flow field are employed. In the interpretation process units from each level are combined into larger units in the next level based on their consistency with appropriate parameter values. Thus, flow vectors, consistent with an affine transformation, are combined into one component; then, components that are compatible with the same Ψ transformation (motion of a planar surface) are merged into a segment; and, finally, segments which satisfy the same 3-D motion parameters possibly correspond to one rigid object. The techniques for computing the parameter values in each level has been based, whenever possible, on solving linear equations derived from the least-squares criterion. Otherwise, sampling techniques combined with multi-resolution search schemes, have been employed. Combining all these techniques together, an effective and efficient algorithm has been developed.

In some situations, however, there exist inherent ambiguities in the interpretation of noisy flow fields. In Chapter VI we characterized and demonstrated such situations. The first ambiguity is in recovering the motion parameters from a noisy

flow field generated by a rigid motion. We found that if the field of view corresponding to the region containing the interpreted flow field is small, and the depth variation and translation magnitude are small relative to the distance of the object from the camera, then the determination of the 3-D motion and structure can be expected to be very sensitive to noise and, in the presence of a realistic level of noise, practically impossible. We experimentally found that there is also a relation between the location of the FOE and the degree of ambiguity. This relation should be mathematically investigated in future research.

The second ambiguity is in the decomposition of the flow field into sets corresponding to independently moving objects. We found that the rigidity assumption is not appropriate for noisy flow fields, that is, the consistency of a set of flow vectors with the same motion parameters, up to the estimated noise level, does not reasonably guarantee that they are really induced by one rigid motion. As an alternative to this assumption, we have assumed that a connected set of flow vectors, which is consistent with a rigid motion of a *planar* surface, is induced by a single rigid motion. This is the central assumption in the scheme developed in this thesis. It is weaker than the first version of the rigidity assumption in the sense that it can only be applied in more restricted situations and, therefore, it is more likely to be correct.

The results of the ambiguity analysis can be used when the effectiveness of motion algorithms is evaluated for real-world tasks. They can help to decide which algorithm to choose, and in what situations this algorithm can be expected to be effective.

Constraints and parameters which can be extracted, even in ambiguous situations, were also introduced in Chapter VI. Integration of such partial information over a time sequence of flow fields may, eventually, resolve the ambiguity and result in a unique interpretation. In addition, combining this information with other

knowledge sources (e.g., a fiber optic rotation sensor [LAW84]) can be considered.

Recovering motion and structure of independently moving objects may be particularly difficult, as was demonstrated by the flat error surfaces obtained for such objects in the second and fifth experiments in Chapter VII. In general, ambiguity in recovering 3-D motion and structure of independently moving objects can be expected, since the effective field of view and the ratio of the depth variation to the distance between the object and the camera are usually small. Furthermore, additional information from other knowledge sources may be hard to acquire. Therefore, the possibility of partially resolving the ambiguity in such a case, by using an *object* coordinate system, is especially interesting and should be investigated in future research.

Another limitation of optical flow interpretation is related to the segmentation process. The segments found in the experiments usually correspond to distinct surfaces in the environment. Their boundaries, however, are sometimes inaccurate because of errors in the given flow fields, especially at occlusion boundaries and homogenous areas, and because of the continuity of flow fields across boundaries between regions corresponding to surfaces in similar depths (relative to their distance from the camera). It should be emphasized that the segmentation results in Chapter VII are entirely based on the flow fields and no other information derived from the intensity images is utilized. These results imply that only limited goals should be set for segmentation based on motion analysis. In order to obtain better results, it might be combined with segmentation based on intensity data or other knowledge sources, e.g., stereo or direct depth measurements.

An additional problem which should be further investigated is the accuracy of the approximation used in Chapter V for evaluating the error measure associated with each hypothesized location of the FOE. This approximation may fail if the deviation between the actual and predicted flow fields can significantly be reduced by

allowing negative depth values. We believe that such an occurrence depends on an unlikely coincidence and, based on our experimental results, we conjecture that the approximation is satisfactory in almost any practical situation. More work should be done, however, in order to characterize situations where this approximation may fail, because then an almost full search of the original 5-dimensional parameter space may be required.

The results of this thesis are valid for velocity fields generated by objects moving rigidly, without any significant restrictions on the motion parameters. However, for our analysis to be valid in the case of displacement fields, small amounts of rotation and translation along the line of sight (relative to the depth values) must be assumed (see Chapter III). Thus, in the experiments the displacement fields were restricted to relatively small values. The possibility of extending our scheme to displacement fields generated by unrestricted motion should be examined.

Embedding our scheme in a computational framework for dealing with a sequence of frames seems to be necessary not only for resolving ambiguities but, in general, for real-world tasks such as controlling a robot arm or navigating an autonomous vehicle. In addition, these tasks may require developing a technique for constructing a parameterized model of the environment from a depth map. Other goals of our future research are: real-time implementation, more extensive mathematical and experimental error analyses, and integration with other knowledge sources for image understanding systems. We will also have to deal with the possibility that changes in the image are not a function of motion and structure alone, but also of lighting conditions and surface properties [WEB82]. Another issue to be addressed is non-rigid motion which is very common in biological systems.

REFERENCES

- [ADI83a] G. Adiv, *Recovering 2-D Motion Parameters in Scenes Containing Multiple Moving Objects*, in Proc. DARPA Image Understanding Workshop, Arlington, VA, 1983, pp. 285-292 (also TR 83-11, Computer and Information Science Dept., Univ. of Mass.).
- [ADI83b] G. Adiv, *Recovering Motion Parameters in Scenes Containing Multiple Moving Objects*, in Proc. IEEE Conf. Computer Vision and Pattern Recognition, Washington, D.C., 1983, pp. 399-400.
- [ANA84] P. Anandan, *Computing Dense Displacement Fields with Confidence Measures in Scenes Containing Occlusion*, in Proc. DARPA Image Understanding Workshop, New Orleans, Louisiana, 1984, pp. 236-246.
- [BAL81a] D.H. Ballard, *Parameter Networks: Towards a Theory of Low-Level Vision*, in Proc. Int. Joint Conf. Artificial Intell., Vancouver, Canada, 1981, pp. 1068-1078.
- [BAL81b] D.H. Ballard and O.A. Kimball, *Rigid Body Motion from Depth and Optical Flow*, TR70, Computer Science Dept., Univ. of Rochester, 1981.
- [BRU81] A.R. Bruss and B.K.P. Horn, *Passive Navigation*, MIT A.I. Memo 662, 1981.
- [FAN83a] J.-Q. Fang and T.S. Huang, *Solving Three Dimensional Small-Rotation Motion Equations*, in Proc. IEEE Conf. Computer Vision and Pattern Recognition, Washington, D.C., 1983, pp. 253-258.
- [FAN83b] J.-Q. Fang and T.S. Huang, *Estimating 3-D Movement of a Rigid Object: Experimental Results*, in Proc. Int. Joint Conf. Artificial Intell., Karlsruhe, Germany, 1983, pp. 1035-1037.

- [FEN79] C.L. Fennema and W.B. Thompson, *Velocity Determination in Scenes Containing Several Moving Objects*, Computer Graphics and Image Processing, vol. 9, pp. 301-315, Apr. 1979.
- [HAN78] A. Hanson and E. Riseman (Eds.), "Computer Vision Systems", Academic Press Inc., New York, NY, 1978, pp. 303-334.
- [JER83] C. Jerian and R. Jain, *Determining Motion Parameters for Scenes with Translation and Rotation*, in Proc. Workshop Motion: Representation and Perception, Toronto, Canada, 1983, pp. 71-77.
- [LAW82] D.T. Lawton, *Motion Analysis via Local Translational Processing*, in Proc. Workshop on Computer Vision, Rindge, NH, 1982, pp. 59-72.
- [LAW84] D.T. Lawton, *Processing Dynamic Image Sequences from a Moving Sensor*, Ph.D. Dissertation (TR 84-05), Computer and Information Science Dept., Univ. of Mass., 1984.
- [LON80] H.C. Longuet-Higgins and K. Prazdny, *The interpretation of a Moving Retinal Image*, Proc. Roy. Soc. Lond., B, vol. 208, pp. 385-397, July 1980.
- [LON81] H.C. Longuet-Higgins, *A Computer Algorithm for Reconstructing a Scene from Two Projections*, Nature, vol. 293, pp. 133-135, Sep. 1981.
- [NAG81a] H.H. Nagel, *On the Derivation of 3D Rigid Point Configurations from Image Sequences*, in Proc. IEEE Conf. Pattern Recognition and Image Processing, Dalas, Texas, 1981, pp. 103-108.
- [NAG81b] H.H. Nagel and B. Neumann, *On 3D Reconstruction from Two Perspective Views*, in Proc. Int. Joint Conf. Artificial Intell., Vancouver, Canada, 1981, pp. 661-663.
- [NEU80] B. Neumann, *Motion Analysis of Image Sequences for Object Grouping*

- and Reconstruction*, in Proc. Int. Conf. Pattern Recognition, Miami, Florida, 1980, pp. 1262-1265.
- [ORO81] J. O'Rourke, *Motion Detection Using Hough Techniques*, in Proc. IEEE Conf. Pattern Recognition and Image Processing, Dalas, Texas, 1981, pp. 82-87.
- [PRA80] K. Prazdny, *Egomotion and Relative Depth Map from Optical Flow*, Biol. Cybernetics, vol. 36, pp. 87-102, 1980.
- [PRA81] K. Prazdny, *Determining the Instantaneous Direction of Motion from Optical Flow Generated by a Curvilinearly Moving Observer*, in Proc. IEEE Conf. Pattern Recognition and Image Processing, Dalas, Texas, 1981, pp. 109-114.
- [RAL65] A. Ralston, "A First Course in Numerical Analysis", McGraw-Hill, New York, NY, 1965.
- [RIE83] J.H. Rieger and D.T. Lawton, *Determining the Instantaneous Axis of Translation from Optic Flow Generated by Arbitrary Sensor Motion*, in Proc. Workshop Motion: Representation and Perception, Toronto, Canada, 1983, pp. 33-41.
- [ROA80] J.W. Roach and J.K. Aggarwal, *Determining the Movement of Objects from a Sequence of Images*, IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-2, pp. 554-562, Nov. 1980.
- [SLO81] K.R. Sloan, *Dynamically Quantized Pyramids*, in Proc. Int. Joint Conf. Artificial Intell., Vancouver, Canada, 1981, pp. 734-736.
- [THO59] E.H. Thompson, *A Rational Algebraic Formulation of the Problem of Relative Orientation*, Photogrammetric Record, vol. 3, pp. 152-159, Oct. 1959.
- [THO84] W.B. Thompson, K.M. Mutch and V.A. Berzins, *Analyzing Object Mo-*

- tion Based on Optical Flow*, in Proc. Int. Conf. Pattern Recognition, Montreal, Canada, 1984, pp. 791-794.
- [TSA84] R.Y. Tsai and T.S. Huang, *Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces*, IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-6, pp. 13-27, Jan. 1984.
- [ULL79] S. Ullman, "The Interpretation of Visual Motion", MIT Press, Cambridge, Mass., 1979.
- [ULL81] S. Ullman, *Analysis of Visual Motion by Biological and Computer Systems*, Computer, vol. 14, pp. 57-69, Aug. 1981.
- [WAX83] A.M. Waxman and S. Ullman, *Surface Structure and 3-D Motion from Image Flow: A Kinematic Approach*, CAR-TR-24, Center for Automation Research, Univ. of Maryland, 1983.
- [WAX84a] A.M. Waxman, *An Image Flow Paradigm*, in Proc. Workshop on Computer Vision, Annapolis, Maryland, 1984, pp. 49-57.
- [WAX84b] A.M. Waxman and K. Wohn, *Contour Evolution, Neighborhood Deformation and Global Image Flow: Planar Surfaces in Motion*, CAR-TR-58, Center for Automation Research, Univ. of Maryland, 1984.
- [WEB82] J.A. Webb and J.K. Aggarwal, *Shape and Correspondence*, in Proc. Workshop on Computer Vision, Rindge, NH, 1982, pp. 95-101.
- [WIL81] T.D. Williams, *Computer Interpretation of a Dynamic Image from a Moving Vehicle*, Ph.D. Dissertation (TR 81-22), Computer and Information Science Dept., Univ. of Mass., 1981.