COMPUTING INTRINSIC IMAGES

John (Yiannis) Aloimonos
Department of Computer Science
The University of Rochester
Rochester, N.Y. 14627

TR 19?
August 1986

Low level modern computer vision is not domain dependent, but concentrates on problems that correspond to identifiable modules in the human visual system. Several theories have been proposed in the literature for the computation of shape from shading, shape from texture, retinal motion from spatiotemporal derivatives of the image intensity function and the like.

The problems with the existing approach are basically the following:

(1) The employed assumptions are very strong (they are not present in a large subset of real images), and so most of the algorithms fail when applied to real images.

(2) Usually the constraints from the geometry and the physics of the problem are not enough to guarantee uniqueness of the computed parameters. In this case, strong additional assumptions about the world are used, in order to restrict the space of all solutions to a unique value.

(3) Even if no assumptions at all are used and the physical constraints are enough to guarantee uniqueness of the computed parameters, then in most cases the resulting algorithms are not robust, in the sense that if there is a slight error in the input (i.e. a small amount of noise in the image), this results in a catastrophic error in the output (computed parameters).

It turns out that if several available cues are combined, then the above mentioned problems disappear in most cases; the resulting algorithms compute robustly and uniquely the intrinsic parameters (shape, depth, motion etc.).

In this thesis the problem of machine vision is explored from its basics. A low level mathematical theory is presented for the unique and robust computation of intrinsic parameters. The computational aspect of the theory envisages a cooperative highly parallel implementation, bringing in information from five different sources (shading, texture, motion, contour and stereo), to resolve ambiguities and ensure uniqueness and stability of the intrinsic parameters. The problems of shape from texture, shape from shading and motion, visual motion analysis and shape and motion from contour are analyzed in detail.

This research was supported in part by the National Science Foundation under Grant DCR-8320136, by the Defense Advanced Research Projects Agency U.S. Army Engineering Topographic

TR 198 • [

TITLE *(and Subtitle)* Computing Intrinsic Images	**5. TYPE OF REPORT & PERIOD COVEREI** Technical Report **6. PERFORMING ORG. REPORT NUMBER**
AUTHORfs; John (Yiannis) Aloimonos	**8. CONTRACT OR GRANT NUMBERfs;** DACA76-85-C-0001
PERFORMING ORGANIZATION NAME AND ADDRESS Computer Science Department The University of Rochester Rochester, New York 14627	**10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS**
CONTROLLING OFFICE NAME AND ADDRESS DARPA/1400 Wilson Blvd. Arlington, VA 22209	**12. REPORT DATE** August 1986 **13. NUMBER OF PAGES** 245
. MONITORING AGENCY NAME & ADDRESSCi/ different from Controlling Office) Office of Naval Research Information Systems Arlington, VA 22217	**15. SECURITY CLASS,** *(of this report)* Unclassified **15a. DECLASSIFICATION/DOWN GRADING SCHEDULE**

DISTRIBUTION STATEMENT *(of this Report)*

Distribution of this document is unlimited

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

SUPPLEMENTARY NOTES

None

KEY WORDS *(Continue on reverse aide it neceaaary and identify by block number)*

ntrinsic images, robust computations, shape, texture, motion, co-operating ision computations, multiple cameras

ABSTRACT *(Continue on reverae aide it neceaaary and identify by block number)*

Low level modern computer vision is not domain dependent, but concentrates on problems that correspond to identifiable modules in the human visual system. Several theories have been proposed in the literature for the computation of shape from shading, shape from texture, retinal motion from spatiotemporal derivatives of the image intensity function and the like. The problems with the existing approach are basically the following:
(1) The employed assumptions are very strong (They are not present in a

) . Fi?^, 1473 EDITION OF 1 NOV 65 IS OBSOLETE

large subset of real images), and so most of the algorithms fail when a
to real images.

(2) Usually the constraints from the geometry and the physics of the problem are not enough to guarantee uniqueness of the computed parametei In this case, strong additional assumptions about the world are used, ir order to restrict the space of all solutions to be a unique value.

(3) Even if no assumptions at all are used and the physical constraint* are enough to guarantee uniqueness of the computed parameters, then in r cases the resulting algorithms are not robust, in the sense that if thei is a slight error in the input (i.e. a small amount of noise in the ima< this results in a catastrophic error in the output (computed parameters),

It turns out that if several available cues are combined, then the abow mentioned problems disappear in most cases; the resulting algorithms compute robustly and uniquely the intrinsic parameters (shape, depth, me etc.).

In this thesis the problem of machine vision is explored from its basics A low level mathematical theory is presented for the unique and robust computation of intrinsic parameters. The computational aspect of the theory envisages a cooperative highly parallel implementation, bringing information from five different sources (shading, texture, motion, contc and stereo), to resolve ambiguities and ensure uniqueness and stability of the intrinsic parameters. The problems of shape from texture, shape from shading and motion, visual motion analysis and shape and motion from contour are analyzed in detail.

Computing Intrinsic Images

by

John Aloimonos

Submitted in Partial Fulfillment
of the
Requirements for the Degree

Doctor of Philosophy

Supervised by Christopher M. Brown

Department of Computer Science

University of Rochester

Rochester, New York

1986

Curriculum Vitae

John Aloimonos was born in Sparta, Greece, in 1957. After spending nine years in Sparta, he moved to Athens, where he completed his high school education. In 1976, he entered the Department of Mathematics of the National University of Athens after a national matriculation test (rank 2nd), and he graduated with first class honors in 1981. While at the National University of Athens, he directed a private school which prepared students for the matriculation test necessary to enter the Greek Universities, he conducted research in numerical analysis at the National Technical University of Athens, he founded and published for a year a journal (Mathematical Forms, Terzakis Publ. Co., Papanastassiou 56, Athens), with mathematical articles addressing university students in mathematics and sciences, and he wrote three books on Galois theory, differential equations and functional analysis.

In January 1982 he joined the Department of Computer Science of the University of Rochester, where he worked toward his Ph.D. degree. During his studies in Rochester he did research on theoretical computer science (with Professor Peter Gacs), and on artificial intelligence (computer vision) under the supervision of Professor Christopher Brown.

John Aloimonos has published several articles on computer vision and his research interests include knowledge representation, information theory, chaotic phenomena, parallel computing and computational complexity, in addition to robot vision.

Acknowledgments

Professor Christopher Brown, my thesis supervisor, has been a constant source of technical advice during my studies. He introduced me to the basic problems of my thesis, and with his constant moral support and encouragement helped me understand the field of computer vision. Being Chairman of the Computer Science Department, his calendar is extremely busy yet he always had time for intellectually stimulating discussions and advice which have enriched my way of thinking. Mere words cannot express my gratitude to this teacher, colleague and friend.

I would also like to thank Professor Jerome Feldman for very stimulating discussions, for technical advice and constructive criticism. His comments on the first drafts on my thesis were invaluable and his advice during my studies helped me keep a holistic view on vision problems.

Professor Dana Ballard helped me to focus on vision problems under a vertical integration paradigm. The discussions with Dana contributed a great deal to my view of vision research. He has been a source of inspiration over the years, and I thank him deeply for his advice and for being a good friend.

I wouldn't have been able to complete my studies at the University of Rochester without the help of my friend and colleague Amit Bandyopadhyay. I would like to give him special thanks.

Special thanks to Vally Koubi, without whom I wouldn't have started and without whom I wouldn't have finished.

Discussions and correspondence with B.K.P. Horn, Shariat Neghadharipour and Demetri Terzopoulos of the MIT AI Lab, Robert Hummel and David Lowe of the Courant Institute of Mathematical Sciences, Ramesh Jain of the University of Michigan, Azriel Rosenfeld and Ken-ichi Kanatani of the University of Maryland, Takeo Kanade and

Steve Shafer of Carnegie Mellon University, William Thompson of the University of Minnesota, Thomas Huang of the University of Illinois at Urbana-Champaign, Roger Tsai of IBM Research and John Tsotsos and David Fleet of the University of Toronto have greatly infuenced my work. I wish to thank them all.

Finally, I wish to thank all my friends and the members of the Rochester Vision group, Mike Swain, David Sher, Paul Chou, Anup Basu, Gary Cottrell, Isidore Rigoutsos, Barun Chandra, Rich Pelavin, Richard Newmann-Wulf, Stu Friedberg, Cesar Quiroz, Art Altman, Rabi Dutta, Hide Ohkami, Rose Peet, Peggy Meeker, Peggy Frantz and Jill Orioli, for their tremendous help.

Abstract

Low-level modern computer vision is not domain dependent, but concentrates on problems that correspond to identifiable modules in the human visual system. Several theories have been proposed in the literature for the computation of shape from shading, shape from texture, retinal motion from spatiotemporal derivatives of the image intensity function, and the like.

The problems with the existing approach are basically the following:
(1) The employed assumptions are very strong (they are not present in a large subset of real images), and so most of the algorithms fail when applied to real images.
(2) Usually the constraints from the geometry and the physics of the problem are not enough to guarantee uniqueness of the computed parameters. In this case, strong additional assumptions about the world are used, in order to restrict the space of all solutions to a unique value.
(3) Even if no assumptions at all are used and the physical constraints are enough to guarantee uniqueness of the computed parameters, then in most cases the resulting algorithms are not robust, in the sense that if there is a slight error in the input (*i.e.* small amount of noise in the image), this results in a catastrophic error in the output (computed parameters).

It turns out that if several available cues are combined, then the above-mentioned problems disappear; the resulting algorithms compute uniquely and robustly the intrinsic parameters (shape, depth, motion, etc.).

In this thesis the problem of machine vision is explored from its basics. A low level mathematical theory is presented for the unique and robust computation of intrinsic parameters. The computational aspect of the theory envisages a cooperative highly parallel implementation, bringing in information from five different sources (shading, texture, motion, contour and stereo), to resolve ambiguities and ensure uniqueness and stability of the intrinsic parameters. The problems of shape from texture, shape from shading and motion, visual motion analysis and shape and motion from contour are analyzed in detail.

Table of Contents

List of Figures

1

Introduction

A large part of Low and Intermediate Level Vision, i.e. problems such as *shape from texture, shape from shading, structure from motion, three-dimensional motion analysis and shape from contour,* is studied. All these problems have been studied in the past, using very few information sources. It was proved that in order to achieve uniqueness of the underlying computations, some restrictive assumptions should be employed. This resulted in algorithms that worked partially in synthetic laboratory images and not at all in natural images. Furthermore, the stability of the proposed algorithms was never studied; this resulted in algorithms very sensitive to noise, i.e. a very small amount of noise in the input could result in a very high percentage error of the computed parameters. In this work, we study the above problems, by using as much information is available from different cues (stereo, motion, contour, shading and texture). This results in algorithms that provably compute uniquely what they are supposed to compute and they are very stable (robust), in the sense that small noise in the input, will create a small percentage error in the computed parameters.

The basic ideas in this thesis are centered around the fact that minimal assumptions, uniqueness and stability, should (and can) be the first and basic requirements for a visual computation. We support this argument by analyzing several problems.

The first chapter introduces the reader to the Held of computer vision and establishes some of the required nomenclature. The second chapter criticizes a large part of previous work, gives the motivation for the research needed and describes the results that we have obtained. The chapter after this analyzes in detail the problem of shape from

texture, and the fourth chapter examines the problem of shape from shading. The fifth chapter analyzes visual motion and the sixth studies the perception of shape and motion from contour. The final chapter summarizes the results and describes future research in the field. Finally, our technical results are listed at the beginning of the third, fourth, fifth and sixth chapter.

Background to this work

In this chapter we discuss what a machine vision system is perceived to be by today's research as well as the relationship of machine vision to other scientific fields. We introduce concepts that will be used throughout the thesis.

LI Computer Vision

There is no doubt that vision is our most powerful sense. It gives us information about our environment and the ability to interact with the environment in a very intelligent way. Because of this, there has been a major effort in the last twenty years to give machines a visual sense. (Computer Vision is the field of computer science, and subfield of artificial intelligence, which attempts to understand vision and provide machines with a visual sense). But vision, our most powerful sense, is also our most complicated sense. Research in the field of neuroscience has shown that more than half of our brain is engaged in visual processing. Our knowledge about biological vision systems is still very poor, and we can say that what we do know about biological vision is that it is very complex. No wonder, then, that all the attempts up to now to provide machines with a rich and general sense of vision have failed. But, some progress has been made in industrial applications, where the visual environment can be controlled and so be very restricted, resulting in a clear-cut task with which the machine vision system is faced.

Building a universal machine vision system, or understanding the animal visual system, is far from reality. It is undoubtedly of the nature of research in a difficult field that some early ideas have to be abandoned and **new** concepts introduced as time passes. Some believed, for example, that understanding the image formation process was not necessary. Other researchers became very excited about specific computing methods of rather narrow utility. No doubt some of the ideas presented in this thesis will also be revised or abandoned in due course. The field is evolving too rapidly for it to be otherwise.

The next section deals with the central problem in computer vision, and constitutes the basis for the rest of the chapter.

1.2 The Central Goal of Machine Vision

It is very difficult to define the central problem of computer vision or vision in general, as in many other scientific fields. What goes on inside our heads when we see? Most people take seeing so much for granted that few will ever have considered the question seriously. Here we attempt to give the following loose definition of the central problem of computer vision:

"The central problem of computer vision is: from one or a sequence of images of a moving or stationary object or a scene, taken by a monocular (one eye) or polynocular (many eyes) of a moving or stationary observer, to understand the object or the scene and its three-dimensional properties."

The reader will immediately observe that all the terms in the above definition are well defined, with the exception of the term "understand." What is really the meaning of "understand" with respect to this problem? The problem of finding meaning is the central one in artificial intelligence and it is by no means answered. For this reason, because various researchers understand meaning in different ways, there have basically been two schools of thought in computer vision. Although no clear distinction between them can be made, we can safely differentiate them into two schools: *Reconstruction* and *Recognition.* The reconstruction school worries about the reconstruction of the physical parameters of the visual world, such as the depth or orientation of surfaces, the boundaries of objects, the direction of light sources and the like. The recognition school worries about the recognition or description of objects that we see and involves processes whose end product is some piece of behavior like a decision or a motion. Both schools have strong ties with psychology and neuroscience and it is strongly believed at this point that both schools will merge into a new one that will, it is hoped, find an answer to the difficult questions of the vision problem.

Although the author of this thesis does not put himself in any of the schools, most of the work presented here could be classified in the reconstruction school, for computing in

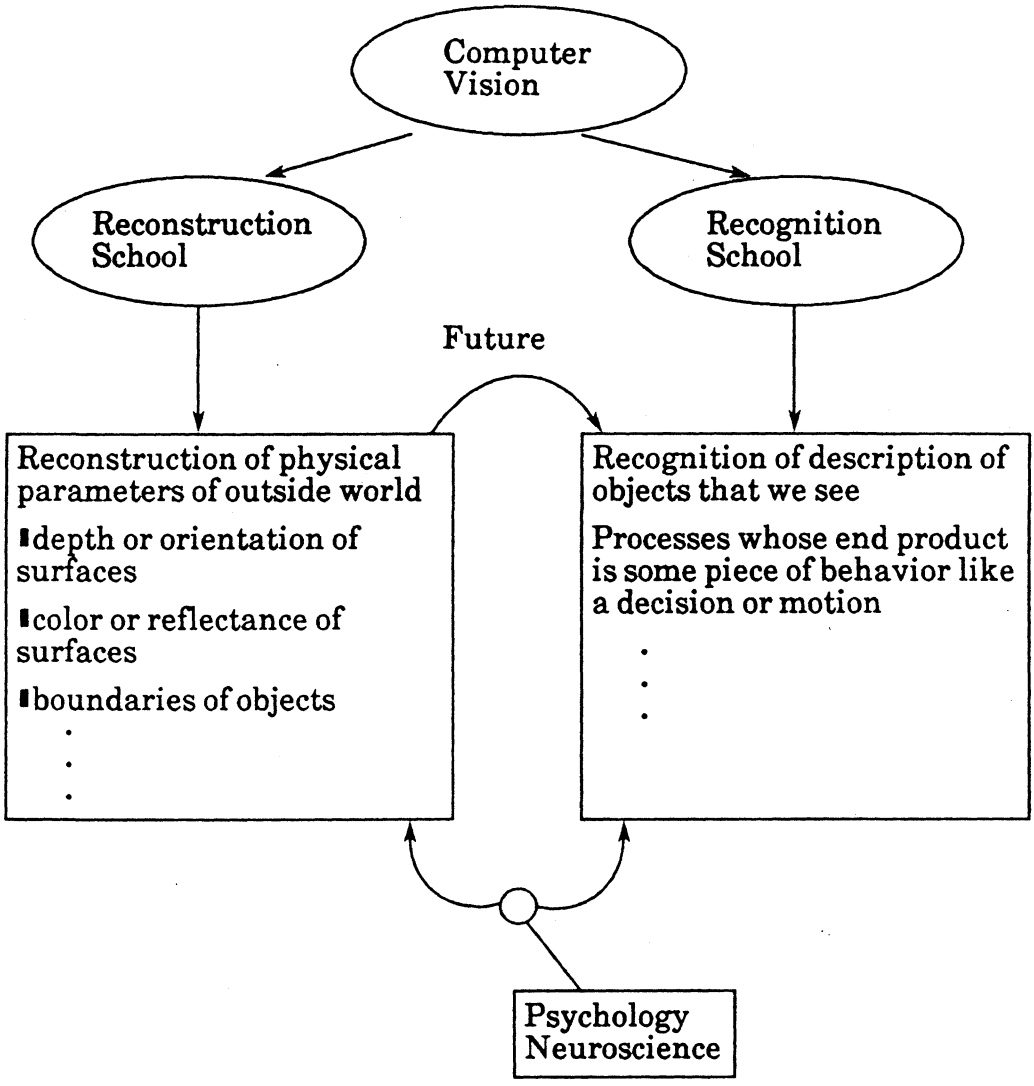


Figure 1.1: The two schools in computer vision

a mathematical way three-dimensional properties from 2-dimensional image properties. The next section reconsiders the central problem of vision from another point of view.

1.3 The Machine Vision Goal Revisited

Up to this point, we have been rather general, since we have been talking about computer vision as having as its goal the development of a universal visual system. Being more specific, a machine vision system analyzes images and produces descriptions of what

is imaged. These descriptions must capture the aspects of the objects being imaged that are useful in carrying out some task. So, we consider the machine vision system as part of a larger entity that interacts with the environment. The vision sytem can be considered an element of a feedback loop that is concerned with sensing, while other elements are dedicated to decision making and the implementation of these decisions. The input to the machine vision system is an image, or several images, while its output is a description that should satisfy at least the following two criteria:

a) *It must bear a relevant relationship to what is being imaged;*

b) *It must contain all the information needed for the specific task.*

Obviously, the first criterion ensures that the description depends in some way on the visual input. The second, ensures that the information provided is useful. Something has to be said about the concept of description that we used above. An object does not have a unique description. We can think of descriptions at many levels of detail and from many points of view. It is impossible to describe an object completely. Fortunately, we can avoid this potential philosophical snare by considering the task for which the description is intended. That is, we do not want just any description of what is imaged, but one that allows us to take appropriate action.

An example may help to clarify these ideas. Consider the task of picking up parts from a conveyor belt. The parts may be randomly oriented and positioned on the belt. There may be several different types of parts, with each to be loaded into a different fixture. The vision system is provided with images of the objects as they are transported past a camera mounted above the belt. The descriptions that the system has to produce in this case are simple. It need only give the position, orientation and type of each object. This description may be just a few numbers. In other situations an elaborate symbolic description may be needed. Figure 1.2 depicts a vision system.

1.4 Success up to Now

We have already noted that a universal vision system is very far from reality. But even systems that are not universal but are supposed to carry out a nontrivial task, are difficult to design. We think that this introduction would be incomplete if we did not

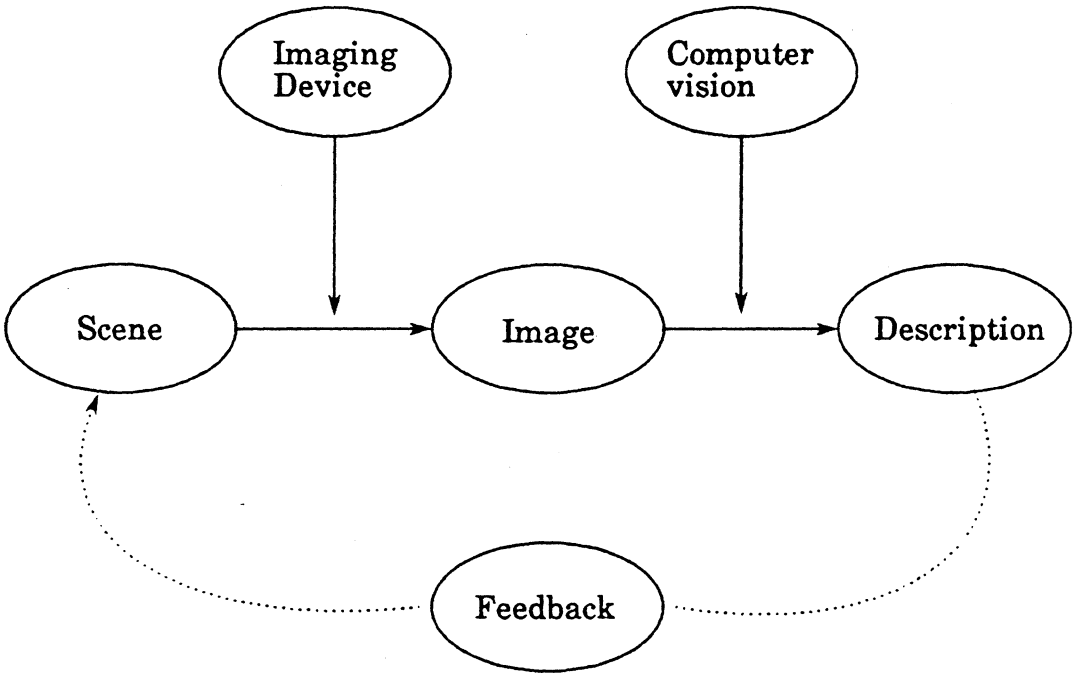


Figure 1.2: A computer vision task

mention the status of the state of the art research on computer vision. We will give two examples, one from the reconstruction school and the other from the recognition school.

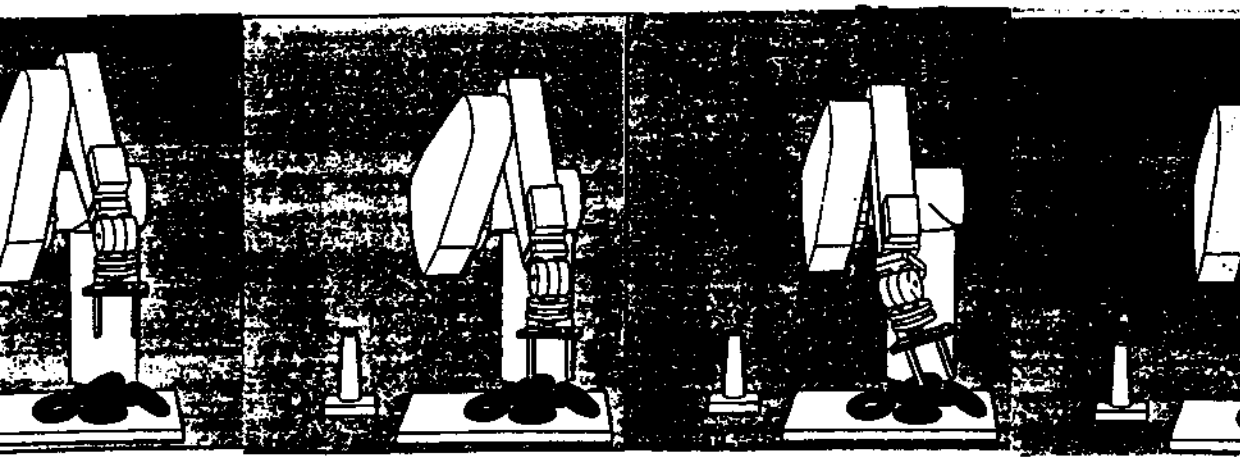
1.4.1 The Terregator

The terregator (terrestrial navigator) is an example of the state of the art research in goal-oriented vision. It is a car equipped with six wheels and television cameras which propels itself and needs no driver. The success of this robot up to now is a clear witness to the level of basic research in vision at this point. From what we know, the terregator is a primitive robot. More importantly, scientists in the field disagree about whether or not we will be able very soon to have machines that will navigate autonomously in unconstrained environments. The autonomous land vehicle (ALV), a similar vehicle developed at Martin-Merrieta with the help of several American Universities, is still in a primitive stage; it can navigate autonomously with low speed, in a constrained

environment, but even shadows or dust or unpredicted features in the environment can affect considerably the operation of the vehicle.

1.4.2 The Robot that Picks Up Donuts

The robot with visual capabilities which can pick up a donut from a pile of them and put it in a specific place was developed at MIT under the leadership of Katsushi Ikeuchi and Berthold Horn, and can be considered an example of state of the art research in the reconstruction school of computer vision. The robot is quite successful under some restrictive assumptions. Figure 1.3 shows the robot's action. There is no doubt that the most difficult part of this operation is in stages 1 and 2. During these stages, the robot takes a picture of the pile of donuts, then from this picture segments the donuts, differentiates one of them, and then it picks it up. The method to do that is quite complicated and the interested reader is referred to [Scientific American, Aug. 1984]! Figure 1.4 shows three pictures of the pile of donuts, as seen by the "robot's brain," under three different lighting conditions. Figure 1.5 shows a part of the images with the surface normals computed. Figure 1.6 shows the same part of the image, where the donuts have been segmented with the help of the surface normals. Finally, Figure 1.7 shows one donut segmented. This is the basis of an algorithm that will enable the robot's arm to pick up the donut under consideration. The robot is quite successful, but if certain conditions are not satisfied, it fails. For example, if the lighting conditions are not accurate, the surface of the donuts is not special, the illuminating source is not near to point source, there are shadows on the donuts, etc., the robot will err in its task.



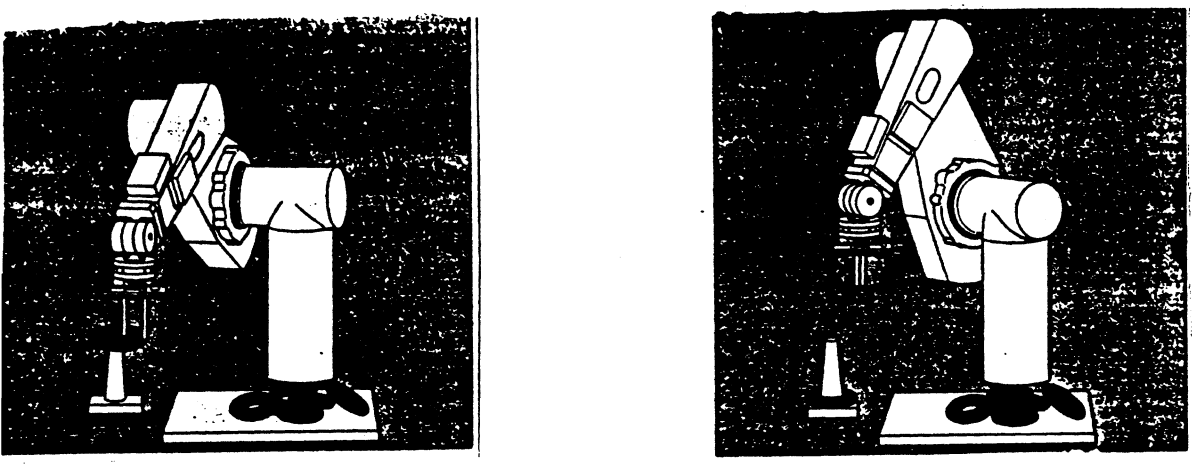


Figure 1.3: The robot's task

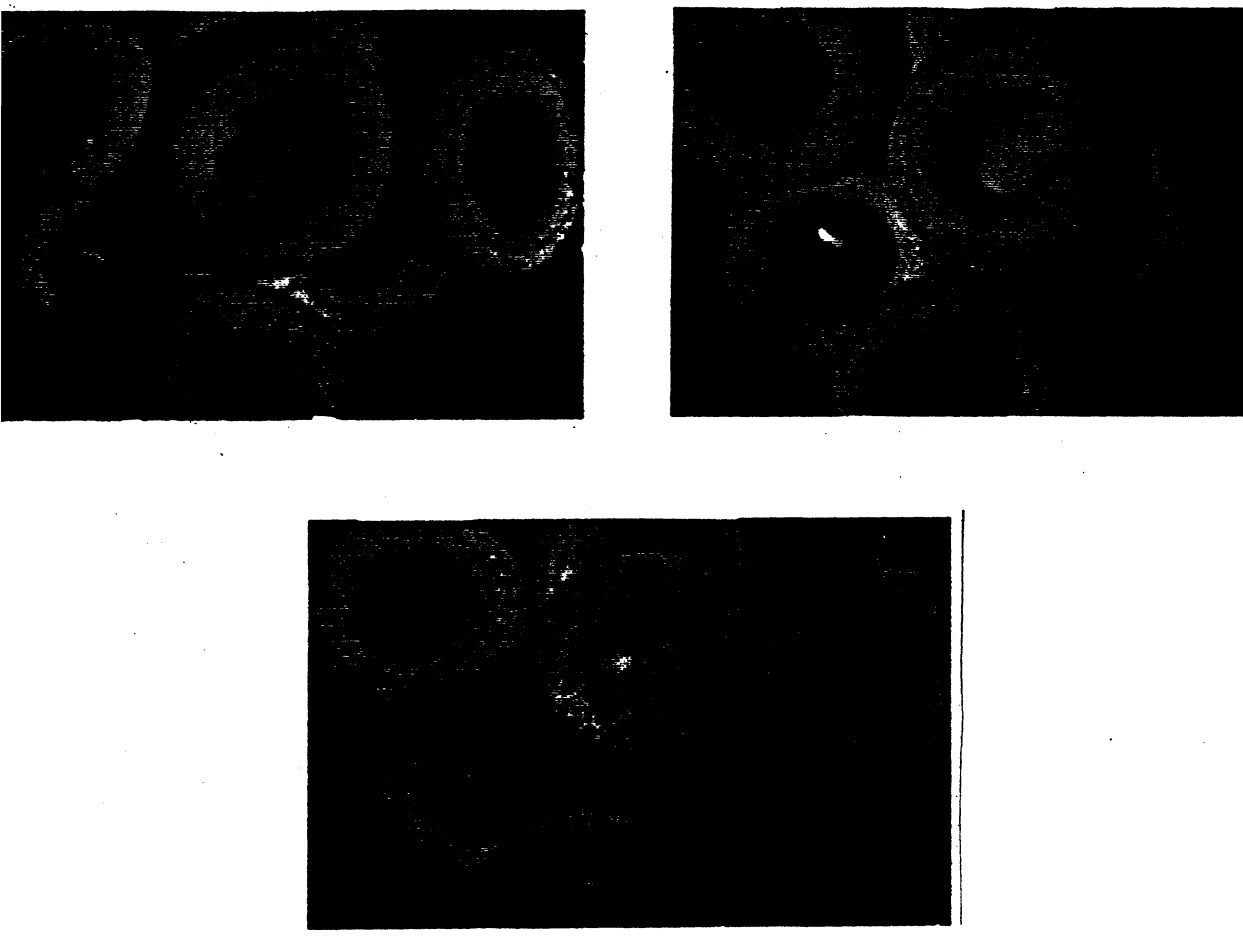


Figure 1.4: Three pictures of the pile of donuts

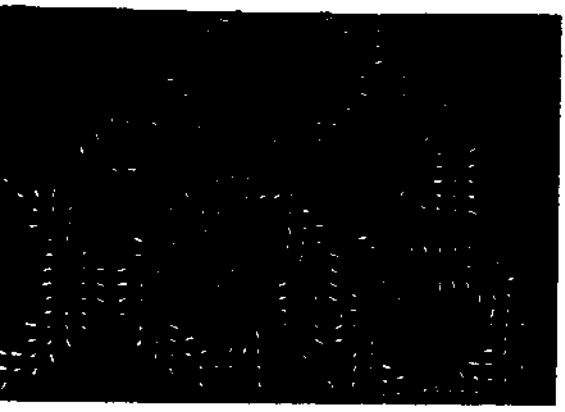


Figure 1.5

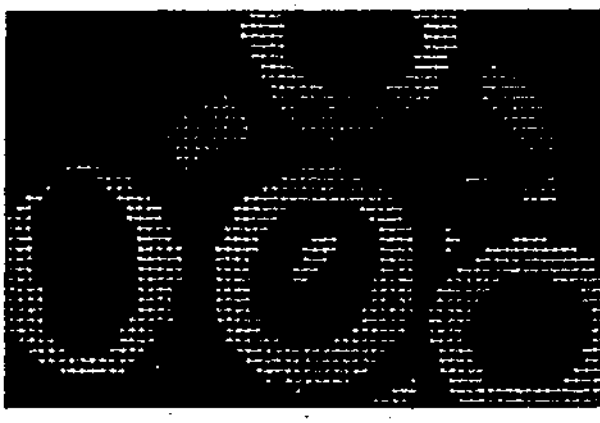


Figure 1.6

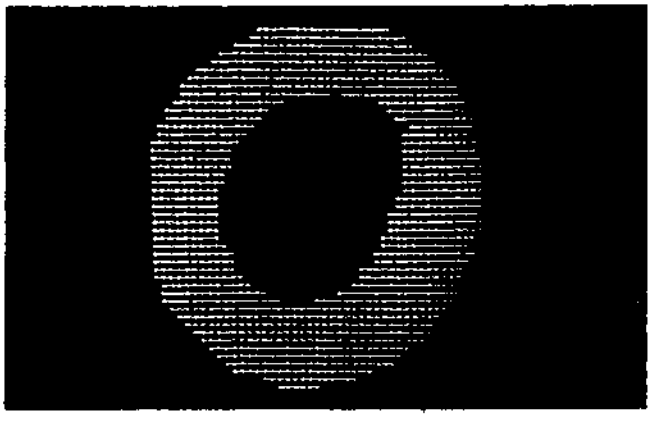


Figure 1.7

1.5 A Quick Passage Through Computer Vision History

In this section we will describe the changing character of computer vision, emphasizing the basic reasons that lead to this change. It is not a complete survey of vision research; more specific references will follow in later technical sections.

1.5,1 As it was in the beginning

Even as late as 1975, computer vision looked very different from its appearance today. A lot of research had been devoted to the blocks microworld of scenes of polyhedra. Huffman [Huffman, 1971] and Clowes had noted the advantage of making the image formation process explicit. They realized that image lines and junctions were the images of 3-D scene edges and vertices, and they made an extensive catalog of those interpretations of lines and junctions that were possible, given assumptions of planarity and the restriction that at most three surfaces were allowed to meet at a vertex. These interpretations amounted to local constraints on the volume occupied by a vertex. The local constraints propagated along picture lines since planar polyhedral edges cannot change their nature between two vertices.

Huffman showed further [Huffman, 1971] that the local vertex constraints were not enough to capture the important restriction that picture regions were the images of planar surfaces. Mackworth's algorithm using gradient space [Mackworth, 1973] was intended to repair this deficit. Despite this, most line drawings had a remarkable number of possible interpretations. Waltz's work [Waltz, 1975] introduced the inherently global constraint afforded by shadows cast from a single distant sourse, and showed that the multiple ambiguities possible without lighting were often resolved to a unique interpretation with lighting. More importantly, the process by which the unique interpretation was discovered naturally lent itself to parallel processing of a particular sort. Each vertex had an associated processor, and they all operated in strict synchrony. At each time, a processsor changed its state according to the state of those directly connected to it. Rosenfeld, Hummel and Zucker noted the connection between this scheme and relaxation processes in numerical analysis [Rosenfeld et al, 1976].

A second strand in the development of computer vision concerns what was referred to as "low level" processing. It was more art than science, and consisted largely of methods for the extraction of the "important" intensity changes in an image. The approach mostly

consisted of convolving images with local operators to estimate the position, contrast and orientation of the important intensity changes. Operators were tuned to particular applications and failed badly outside their domain in the presence of noise. Little serious analysis of actual intensity changes including the signal to noise characteristics of real images had been carried out. Other work in low level vision largely consisted of the design and construction of region finders. Region finding aimed at isolating those regions of an image that were the images of perceptual surface patches. It was thought that such regions might be isolated by defining some descriptor with respect to which they were uniform, and distinguishable from surrounding regions. It was soon clear [Barrow et al, 1971, Brice and Fennema, 1970] that even if such descriptors existed, they were not defined simply in terms of color or grey level intensity values.

By the early 1970's, the consensus was that low level vision was inherently incapable of producing rich, useful descriptions. It was observed, by analogy to the apparent need for semantics in parsing English sentences, that downward flowing knowledge of the scene could provide additional constraints. This in turn could inform local decision making. A number of program structures were proposed to effect this interaction between top down and bottom up processing of information [Barrow et al 1976, Brady 1979, Freuder 1974, Minsky and Papert 1972, Shirai 1973, Winston 1972]. Similar ideas were advanced about natural language understanding and speech perception. This influenced the design of, for example, Hearsay2 [Lesser and Erman, 1977]. To experiment with these ideas, entire systems were constructed which mobilized knowledge at all levels of the visual system as well as information specific to some domain of application. In order to complete the construction of all these systems, it was inevitable that corners were cut and many over-simplified assumptions were made. By and large, the performance of these systems did not give grounds for unbridled celebration. The authors of the KRL proposal [Bobrow and Winograd, 1977], for example, listed several common failings.

1.5.2 Is now and ... should be

Perhaps the most fundamental difference between computer vision now and a decade ago stems from the current concentration on topics corresponding to identifiable modules in the human visual system. The focus of research today is more narrowly

defined in terms of a domain, and the depth of analysis is correspondingly greater. This change has produced a number of far-reaching effects in the way vision is researched.

One obvious effect was a sharp decline in the construction of entire vision systems, in the 1975-1985 period. Most AI workers have gratefully abandoned the idea that visual perception can profitably be studied in the context of *a priori* commitment to a particular program or machine architecture. There is, for example, no more reason to believe that relaxation style processing will of itself tell us more about vision than did the excursions into heterarchy. There is no obvious reason to be encouraged by Reddy's [Reddy, 1978] claim that the Hearsay 2 model can be adapted *mutatis mutundis* to vision. However, this opinion is subject to criticism. There is probably reason to believe that if one thinks in the context of parallel architectures (i.e., connectionist networks) [Feldman, 1986], there is a chance of formulating vision problems in a context that is closer to animal visual processing capabilities, and so a greater chance of solving the problem. But that is subject to more research that will show if thinking in terms of particular architectures is of any help.

Unfortunately, dogmas have been developed during the last decade and leading researchers in the area have antidiametric opinions on the issue of whether or not a particular architecture is of help when formulating and solving vision problems. Although we think that more research is required for the answer to this question, in this thesis we do not worry about specific machine or system architectures; we rather worry about abstract visual computations and the development of algorithms that will carry out a specific computation, in the spirit of methodology as it was introduced by David Marr [Marr, 1981]. There is a standard way of designing large and complex information processing systems. We have to start addressing the question of what the system must do and have a clear understanding of the constraints on the available resources.

The first step is to divide the whole system into functional components that break the overall task into autonomous parts. Then, we should choose the representation of information within the subsystems and the languages of communication among them. After this, the details of the subsystems are tested individually, in pairs, and all together. Essentially the same methods are used for analyzing unknown large information-processing systems. It is at least possible that a similar paradigm would be of some use in studying complex biological systems, including the primate visual system, or for that

matter, the development of machines with visual sense. So, if we want to study the animal visual system or construct seeing machines, we must first understand what the system should do. In the previous sections we tried to define as clearly as possible what a visual system should do. Next, we should break the system into functional components that are somewhat autonomous. Exactly this is attempted by much of today's research, i.e., to concentrate on topics that correspond to identifiable modules in the human visual system.

We have at this point a clear idea that cues such as *shading, motion, texture, contours* and *stereopsis* are very important for the perception of the 3-D world. For this reason, almost every computer vision research paper published in the last few years has to do with the perception of shape from shading, shape from texture, shape from contour, shape from motion, depth from stereo, illuminant direction from shading, three-dimensional motion from retinal motion, and the like.

Not all modules operate directly on the image. Indeed, it seems that few do. Instead they operate on representations of the information computed, or made explicit by other processes. In the case of stereopsis, Marr and Poggio [Marr and Poggio, 1979] argue against correlating the intensity information in the left and right views. Instead, they suggest that so-called zero-crossings are matched [Marr and Hildreth, 1980]. In any case, a great deal of attention has centered on the isolation and study of individual modules, and in each case on the development of the representations on which they operate, and on those that they produce. The first of these representations, and the one whose structure is the least subject to dispute is the image itself. Not surprisingly, then, most attention has centered on those modules that operate upon the image. As we shall see, the further we progress up the process hierarchy, the less secure the story becomes as the exact structure of the representations becomes more subject to dispute. Again, this is not surprising. The image aside, any representation is one module's co-domain and another's domain. All of them shape an eventual structure. In the next two sections we will spend some time on modules that operate on the image and other representations.

1*5.2*1 Modules operating directly on the image

A great deal of effort has been devoted to understanding how the important intensity changes in an image can be extracted. Marr [Marr,D., 1976] coined the term primal sketch to describe such a representation, and he described an algorithm by which

it might be computed. His work with Poggio led to a revision of the process of construction of the primal sketch. Instead they advocated the use of zero-crossings of the second derivative of the filtered image. This idea was developed in turn by Marr and Hildreth [Marr and Hildreth, 1980] who propose that an image is first filtered by four Gaussians having different band pass characteristics. One of the novel features, (as far as Computer Vision work is concerned) of the Marr-Hildreth account is the size of the operators involved, the smallest being roughly 35 pixels square. This is in stark contrast to conventional operators, which, in most Computer Vision work today, are still typically on the order of 5 x 5. Such a large operator can be in much closer agreement with a Gaussian (or any filter for that matter) than any small operator, and its effects are therefore more predictable. Unfortunately, it is no longer obvious how to compute the assertions that Marr had previously advocated for inclusion in the primal sketch. The whole issue of constructing the primal sketch from zero-crossings is far from being resolved.

Intensity changes aside, Horn and his colleagues [Horn, B.K.P., 1977,1979, 1980,1982, Ikeuchi and Horn, 1981, Woodham, 1981, Strat, 1981,] have studied the perception of surface shape from shading. In brief outline, Horn formulated a second order differential equation that he calls the image irradiance equation, which relates the orientation of the local surface normal of a visible surface, the surface reflectance characteristics, and the lighting to the intensity value recorded at the corrresponding point in the image. The output of shape from shading is a representation that makes explicit the orientation of visible surfaces, and may make other information such as depth and surface discontinuities explicit also. Horn suggests the name needle map. Other representations have been proposed that make substantially the same information explicit. Marr [Marr,D., 1978] uses the name $2\frac{1}{2}$ sketch, and Barrow and Tenenbaum [Barrow and Tanembaum, 1976] discuss intrinsic images. Again, the exact nature of the representation is currently far from clear. In part, this is because very little research has been devoted to modules that operate upon it.

Finally, methods for computing optic flow (image motion) from spatio-temporal derivatives of image intensity have been published lately.

1.5.2.2 Modules operating on zero-crossings, points and the primal sketch

We have already stated that there remain a vast number of unresolved issues concerning the nature of the primal sketch and its computation from zero-crossings or whatever kind of filtered image. Nevertheless, the broad outlines are clear enough for work to proceed to investigate modules that are assumed to operate upon these representations. Indeed, it is necessary that it does, as it will also contribute to our understanding of the information that needs to be made explicit in the primal sketch, and thus its eventual form. Motion is an important source of information for determining structure, and much work has been done in this area. Considerable attention has been paid to stereopsis and to the detection of surface orientation from texture. In addition, much research has been devoted to the analysis of line drawings (of planar and curved surfaces), and contours.

1.6 Where Do We Stand (Current Research Status)

It is clear by now that modern computer vision worries about concentrating on topics that correspond to identifiable models in the human visual system. And although we don't know what exactly these modules are, we understand that there should exist modules that compute 3-D parameters from specific cues, such as shading, motion, stereo, contours and texture. When we say 3-D parameters, we mean intrinsic images, such as shape, depth, reflectance, three-dimensional motion, illuminant direction and the like. So, one could say that today's research is:

Compute Y from X.

where Y is an intrinsic property (shape, depth, retinal and three-dimensional motion, etc.) and X is a cue in the image or a property of the observer (shading, texture, stereopsis, etc.).

The following figure broadly summarizes the status of contemporary reconstructionist computer vision. On the right, we see the various cues, and on the left the intrinsic parameters. Research tries to recover from any of the cues in the right some of the intrinsic properties in the left. An arrow from box 1 to box 2 indicates that the property in box 2 is recovered form the cue in box 1. The names along the arrows represent some of the researchers who have worked on this specific recovery. More complete references can be found in the rest of the thesis. At this point we have to make clear that

the intrinsic parameters about which we are writing a lot, can basically be classified in two categories. The *retinotopic* and *non-retinotopic* ones . Non-retinotopic ones can be divided into features (physical parameters) and objects and relations [Ballard, 1985]. The retinotopic ones (shape, depth and the like) are the ones of most interest in this thesis. These parameters are spatially indexed at every image point. We can actually say, that the retinotopic parameters are the basic subject of the Reconstruction School, and the non-retinotopic ones (features) of the Recognition School. In this thesis we will mostly be talking about Low-Level Vision, and so the analysis of three-dimensional shape models and transformations, as part of High-Level Vision modules, won't be treated. Finally, it has to be said that the current status figure of the next page, is by no means complete. Other sources of information such as color and nonplanar contours are of great importance, but we will not discuss them here.

1.7 A Word of Caution and What is to Come

In the preceding sections, we have emphasized that contemporary computer vision is worrying about the recovery of three-dimensional properties (world) from two-dimensional image properties. By no means do we imply that this is the only issue of today's research. There is a lot of excellent research on low and high level vision, object recognition and navigation. We feel that the bulk of research (from 2-D properties to 3-D properties) is the most important because a clear understanding of these issues will contribute a great deal to our knowledge of extrapersonal space perception, to our understanding of the cortex and to our ability to construct machines with visual sense. Of course, several leading researchers may think otherwise and unfortunately the field is too young to be able to justify our claim. But simple and even naive thinking may convince us that if we ever hope to understand how the visual system works, we must first understand that our only input is two-dimensional images, and so in order to reason about the three-dimensional world, we must discover constraints between the images and the three-dimensional world that is imaged. On the other hand, prior knowledge about the world can be of great help. We are not opposed to using *a priori* knowledge about the world in order to help the process of understanding the 3-D space from its images. But before we do that, we should first analyze the various vision problems with as few assumptions as possible, and if no solution is possible, then we should resort to additional assumptions.

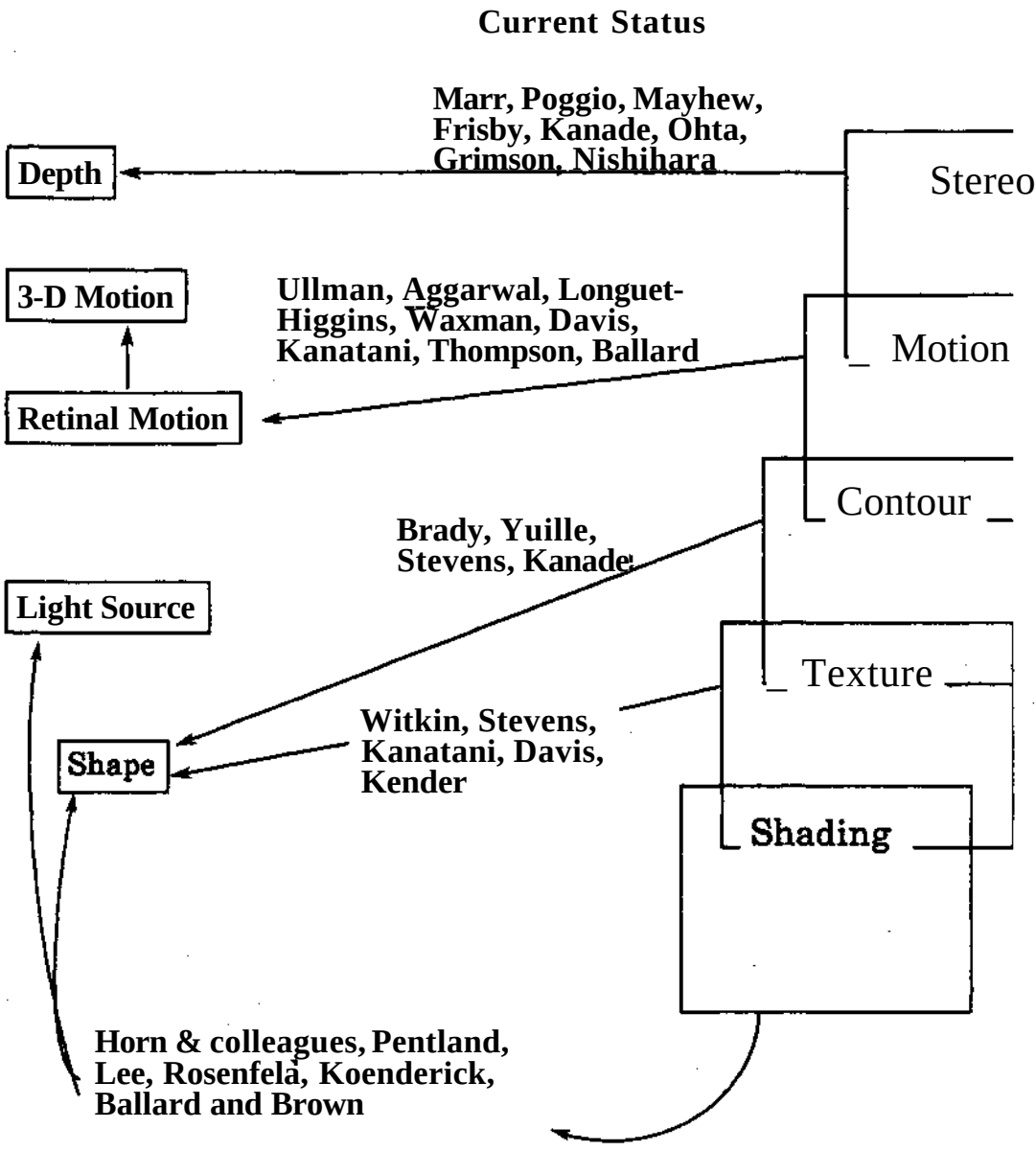


Figure 1.8: Current research status

The next chapter introduces the technical background necessary for the understanding of the rest of the thesis, presents a positive critique of current research from a technical point of view, and finishes with a proposal on how 3-D vision problems should be approached.

2

Unique and robust intrinsic images: The problem, the answer and the technical prerequisites.

In this chapter we describe what the problems of the current research status are and we propose a new aproach. In the rest of the chapter we discuss how images are formed and how they are sensed by a computer, and we give the technical prerequisites for the foundation of the technical work described in later chapters.

2.1 The current research picture revisited

Recalling the current research picture from Section 1.6, we see that the intrinsic parameters that will be described extensively in the rest of this Chapter, are computed from some particular image cue. Indeed, *shading, texture, contours, motion* and *stereo* are very important cues for obtaining three-dimensional information, and later chapters will present evidence for that. If we look carefully at the research picture from Section 1.6, we will realize that an intrinsic parameter is computed using only a particular cue. So we have algorithms for shape from shading, shape from motion, depth from stereo, and the like. There are, however, three basic problems with this approach.

The first problem has to do with employing the right assumptions. Some of these algorithms are based on assumptions which despite their generality are not present in the real world and so the algorithms fail when applied to a variety of natural images. An example of this is all the algorithms for the computation of shape from texture [Witkin, 1981, Stevens, 1980, Davis et al, 19831. In these algorithms the basic assumption is the directional isotropy. In other words, it is assumed that contours and line segments in natural images have orientations which are uniformly distributed over all directions. Obviously, if we look around us for natural or man-made surfaces, we won't find that this assumption is true.

The second problem has to do with uniqueness properties of the resultin algorithms. Some of the problems in Figure 1.8, as formulated, cannot have unique solution. So, in order to bring down the space of all solutions to a uniqu point, assumptions are made about the world which usually are unrealistic and th algorithms fail when applied to real images. An example of this is all the shap from shading algorithms [Horn 1977, Ikeuchi and Horn 1981, Brooks, 1984] th* use assumptions about the global smoothness of the surfaces in view.

The third problem with the current research status is the one which has to do wit the robustness or stability of the resulting algorithms. Even if theoretical analysi shows that given the constraints at hand a particular problem has a uniqu solution, in practice it turns out that the solution is very unstable. In other words, very small error in the input results in a catastrophic error in the output. A example of this is all the algorithms that compute 3-D motion from retinal motioi using only one camera [Waxman et al, 1984,1985, 1986, Tsai and Huang, 198' Longuet Higgins and Prazdny, 1984, Prazdny 1984, Bruss and Horn, 1984; fc additional references, see Chapter 5]. The basic problems with the current researc status can be summarized in the following table.

Problems of Current Research Status

Problem	Example
Use of restrictive assumptions about the world	*Shape from texture algorithms*
To make a problem solvable (uniquely), unreasonable assumptions are made	*Shape from shading, optic flou from image sequences*
Even if an algorithm is proved to have a unique solution, usually the resulting algorithm is unstable	*3-D motion from optic flow, image reconstruction frrom zero-crossings and gradients*

2.2 The regularization paradigm and our criticism

One of the best definitions of early vision is that it is the inverse of optics, i.e., a set of computational problems that both machines and biological organisms have to solve. While in classical optics the problem is to determine the images of physical objects, vision is confronted with the inverse problem of determining properties of the 3-dimensional world from the light distribution in an image, or a dynamic sequence of images. In 1923 Hadamard defined a mathematical problem to be well-posed when its solution:

a) exists,

b) is unique,

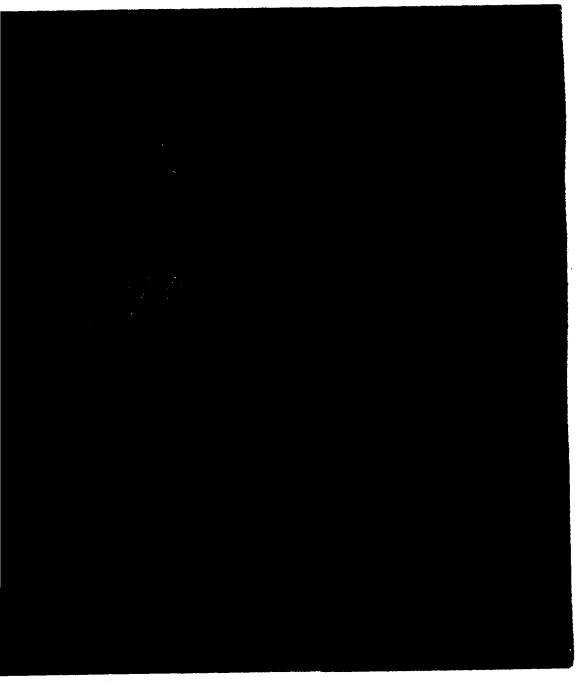
c) depends continuously on the initial data (is robust against noise).

Most of the problems in classical physics are well posed, and Hadamard argued that physical problems had to be well-posed. However, it seems that inverse problems are usually ill-posed. Consider, for example, the equation: $y = Ax$, where A is a known operator. This equation can represent optics, where y is the image, A is the imaging process, and x is the world. So, in this case, the problem is to determine y from x. The inverse problem, i.e. find x from;y, is usually ill-posed when x,y belong to a Hilbert space.

The regularization paradigm claims that most early vision problems are ill-posed (shape from shading, texture, contour, optic flow from image brightness and the like). Rigorous regularization theories for solving ill-posed problems have been developed during the past years [Tichonov, and Arsenin, 1977, Tichonov, 19631. The basic idea of regularization techniques is to restrict the space of acceptable solutions by choosing the function that minimizes an appropriate functional. The regularization of the ill-posed problem of finding x from y such that y = Ax requires the choice of norms 11H I and of a stabilizing functional HPxII. Of course this choice is dictated by mathematical considerations and most importantly, by a physical analysis of the generic constraints of the problem. Then, several methods can be applied as for example, find x that minimizes 11Ax - yll2 +AliPxll2, where \ is the so-called regularization parameter, or among x that satisfies HPxIl^k, where k is a constant, find x that satisfies 11Ax - yli = minimum, etc. The reader interested in regularization techniques is referred to [Tichonov, and Arsenin,

regularization paradigm may be unrealistic for addressing low-level vision problems in natural images.

What led to the regularization paradigm is the fact that several published algorithms for the computation of intrinsic images were basically of the same flavor. In other words it is a post-facto legitimization of a class of methods in early vision.The constraints were not sufficient, additional assumptions were made, and a functional from all these was constructed, with its ultimate goal a minimization that would lead to a solution. Basically, all the additional assumptions had to do with smoothness, because smoothness, when expressed in mathematical terms, gives very strong constraints. But our visual world is anything but smooth. We can safely say that a very small subset of the surfaces that we see are twice continuously differentiable. But even if we forget this for a moment and accept that the smoothness assumption is a good one (in the sense that it is present in our visual world), even then the performance of the regularization algorithms cannot serve as a strong rationale of the feasibility of the approach. Putting aside natural images and concentrating only on synthetic ones, the performance is not excellent. The following figures show the image of an object and the extracted shape using a regularization-based algorithm [Ikeuchi and Horn, 1981] for shape form shading. The poor performance in this particular example is also due to the fact that the constraint from the shading is very weak. Several such examples for other early vision problems can be found.



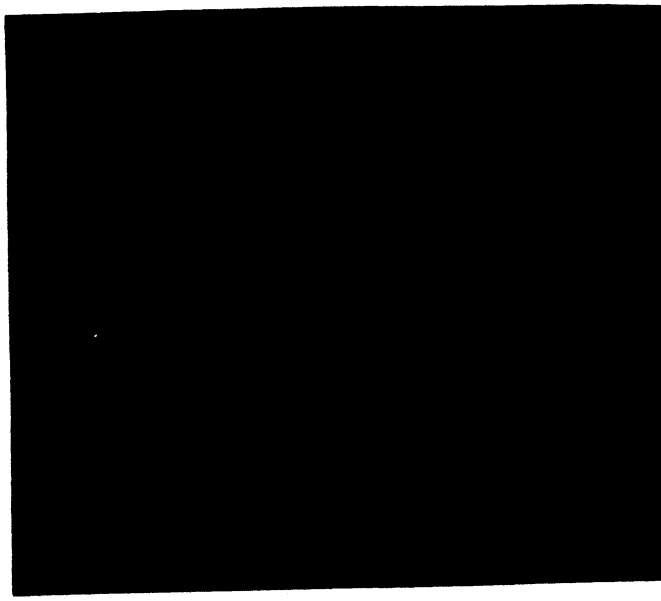


Figure 2.0.1: Intensity image **Figure 2.0.2:** Reconstructed shape

A very positive aspect of the regularization-based approach is that it presents a unified approach for the early vision problems. But this is not at all convincing, since the assumptions used are very restrictive. Of course, *if other functionals (recall in Section 2.2 the functional Px) are used instead of the ones that incorporate smoothness, then this might be proven promising*. Another negative aspect of the regularization based approach is that it examines several problems separately, i.e., it investigates shape from shading, shape from motion, shape and depth from stereo, for example, separately without taking into account that existing, well-working biological vision systems live in a dynamic world and have two eyes.

> *Our claim is that vision is full of redundancy, because organisms can get information from many different sources. Vision seems to be, at least for biological organisms, a very well-posed problem. If our knowledge about vision is very limited today, we should not make the problems ill-posed. If a problem turns out to be mathematically ill-posed, then we should not try to solve it by imposing unrealistic restrictions. Instead, we should investigate what kind of information is missing from the situation at hand, and search for a source which will provide this missing information. In other words, it is the vision researchers that pose the vision problems in such a way that they become ill-posed. The vision problems are well-posed, as it can be very well demonstrated empirically. It is evident that in order to be able to answer vision questions in the right way we must first ask the questions in the right way.*

Our criticism of the regularization-based approach ends at this point, except for stating that if we cannot solve a vision problem as formulated, this means that we have not formulated the problem in the right fashion. Restrictive assumptions about a problem will never enrich our understanding of computational vision. Finally, our position is enforced by recent psychological results by Todd *et al.* [Todd et al, 1986] that state that for the case of shape from shading no algorithm from the regularization-based paradigm seems to have any connection with the computational human mechanisms for the detection of shape from shading.

It has to be noted however that the regularization techniques are very powerful (in a mathematical sense) for attempting a unique solution, when the required assumptions are present in the image under consideration. No wonder then, that several reguralization based algorithms [Terzopulos, 1984, 1985, Negadharipur and Horn, 1986, Maroquin, 1986] perform very well for their domains, that satisfy smoothness assumptions. What we are against for, is the use of regularization as a general theory for low-level vision, for the very simple reason that our visual world is anything but smooth.But if the problem under consideration obeys smothness assumptions, then regularization based approaches are very powerful and give good results.

2.3 Mathematical algorithms and biological vision systems

Even though this thesis is on machine vision, we make no basic distinction between machines and biological systems. In other words, our results could be very well applied for the explanation of biological visual abilities, even though this is not the goal of this thesis. In the rest of the thesis, our results will be formulated in terms of mathematical propositions and algorithms. Two difficulties are immediately raised regarding the applicability of such results to biological visual systems. The first is that unlike an electronic computer a biological system cannot be expected to solve the equations used in deriving the mathematical results. The second is that a biological system does not have access to the perfectly accurate data used in the mathematical abstraction.

A comprehensive examination of the first objection would be beyond the goals of this thesis. The main answer lies, however, in the distinction between different levels of analysis: *competence* vs. *performance* [Chomsky, 1965] or *computational* vs *algorithmic* [Marr and Poggio, 1977]. The computational studies aim primarily at establishing principles that apply to any visual system facing the problem of interpreting something (3-D property) from something else (2-D image property). Certain equations may be used in the derivation of such principles, but it does not follow that a system utilizing these principles would have to solve these equations in the process of the interpretation.

The problem of accuracy in the measurement and computation is an important one. To be of practical value, the interpretation scheme should be robust. Small errors in the

input measurements should not lead to a complete breakdown of the interpretation scheme. This means that computational studies should not only explore what is possible under idealized conditions, but also examine the effects of small perturbations and errors. Unfortunately, current research does not worry about the last issue, and only very recently began to consider the uniqueness of the computations [Ullman, 1983, Tsai and Huang, 1984, Bruss and Horn 1984]. Until then, everything was based on an ad hoc fashion.

2.4 Results: More information from cooperative sources yields unique and reliable solutions.

Looking back at the current research status diagram of Section 1.6 (Figure 1.8), we see that from a particular cue a particular intrinsic property is computed. That is, no cues are combined, in most of the published work, to recover an intrinsic image. As we have already seen, this has as a result the fact that several computations do not have uniqueness properties (and so additional assumptions are needed about the world) and several computations that have uniqueness properties under ideal conditions break down in the presence of small amounts of noise. In order to take care of these problems, more information is needed. In particular, if we combine information from the different image cues, then several computations that did not have uniqueness properties might now have them, simply because the unknown parameters are subject to more constraints that guarantee uniqueness and several computations which even though they had uniqueness properties were very unstable are now robust, simply because the additional constraints do not let the solution escape from its actual position. The proposed framework for the computation of intrinsic images is given in the following figure 2.1. The reader should compare this with Figure 1.8 of Section 1.6 to realize that new information is combined from different cues to recover the intrinsic parameters.

It is worth noting that very recently a few researchers have realized the need for combination of information from different image cues for better estimation of intrinsic parameters. In particular, there is the work of Waxman *et al.* [Waxman et al, 1986] for combining stereo and motion, the work of Grimson [Grimson, 1984] for combining shading and stereo, the work of Richards, and Huang for stereo of motion [Richards, 1985, Huang and Blonstein, 1985], and the work of Milenkovich and Kanade [Milenkovich et al, 1985]. So the need for such an approach has already been realized by some and the hope is

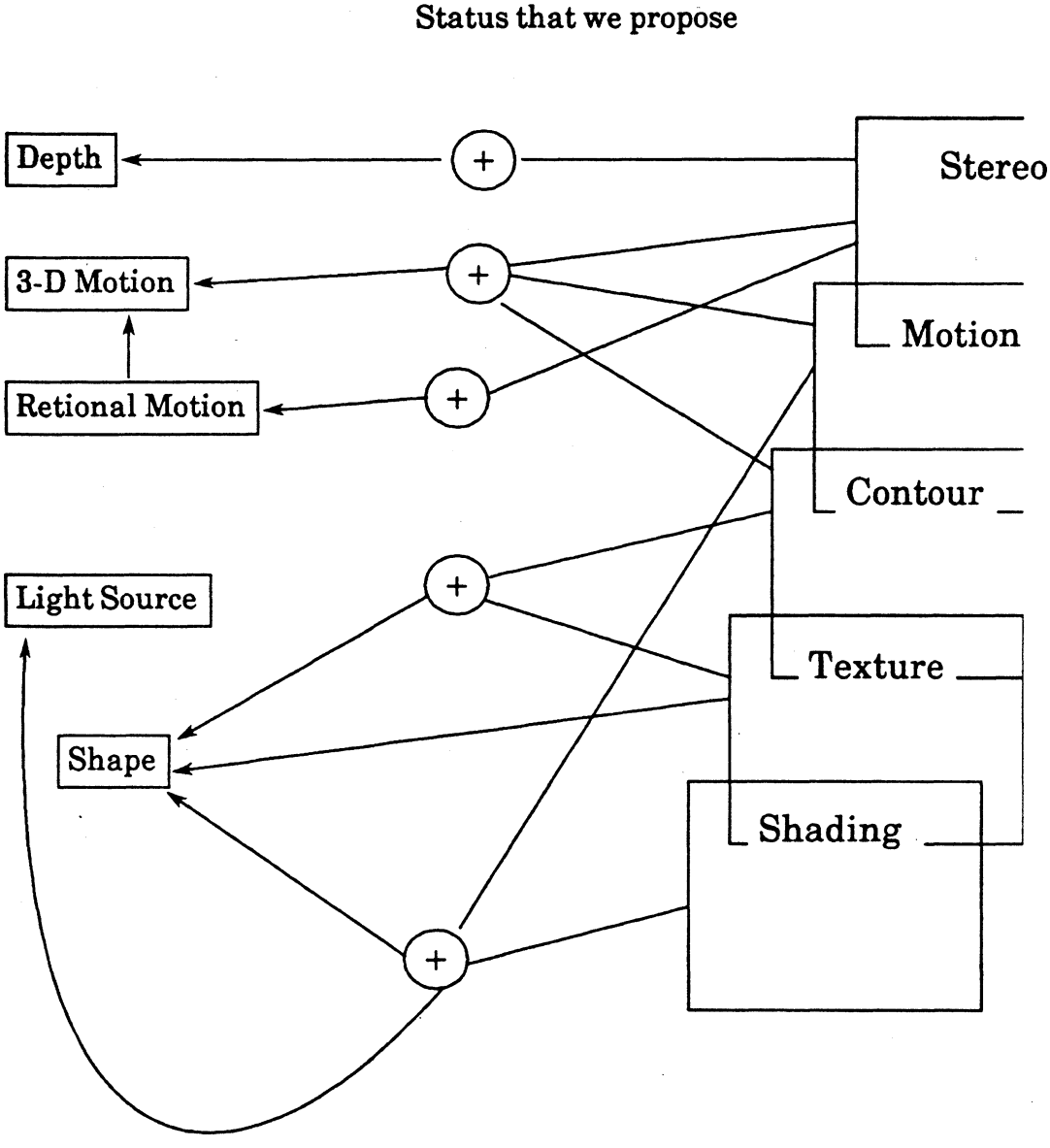


Figure 2.1: Proposed status

that this thesis will contribute to a better understanding of this approach and that it will generate more related research. Due to the different nature of the intrinsic parameters and the image cues, a unified approach, i.e. a general theory for computing intrinsic parameters from combination of image cues (with the intrinsics and the image cues as parameters) seems at this point very difficult, if not impossible. Our approach will be based on a case-by-case analysis. That is, we will consider each individual problem separately, analyze it, see that a solution without additional assumptions or stability is impossible, and then combine different cues to obtain unique or robust results.

The basic structure of the thesis is depicted in the following diagram (Figure 2.2). In the elipses (top) are the different image cues (we take the liberty to call stereo or motion a cue.) It is obvious that by cue we mean a source of information, either coming from the image(s) or from the particular set up or condition of the visual system (stereo-motion). In the squares are the results we obtain (in terms of propositions) when we combine information from two different cues. Two or more different cues are combined with arcs which lead to small circles containing a plus. Then, a different arc from the plus leads to a square containing the result from this combination. The numbers at a plus or an arc indicate that the theory for this particular computation can be found in the corresponding chapter.

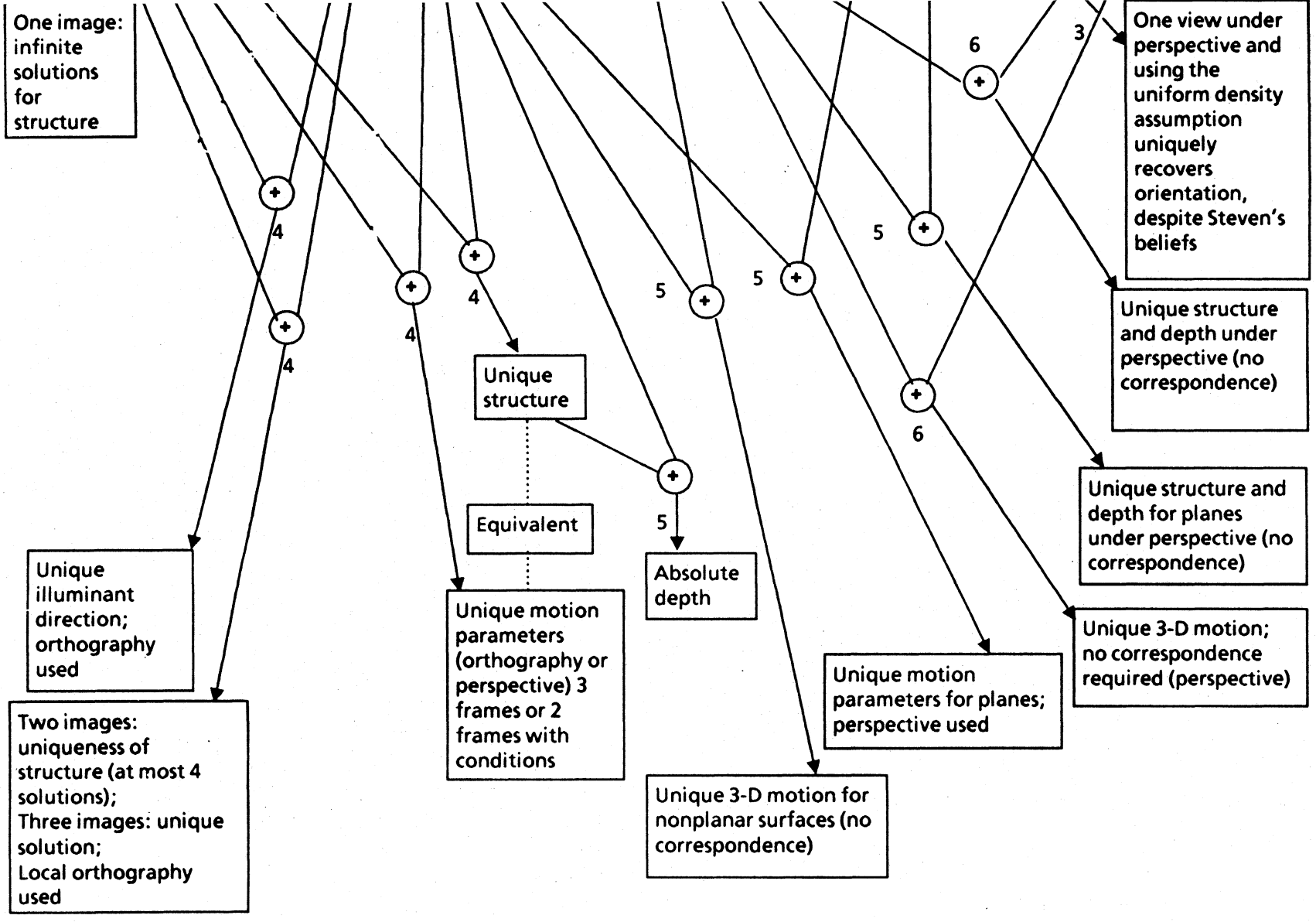
2.5 Technical Prerequisites: Image formation and intrinsic images

It is very important to understand how the images are formed, because this is a prerequisite for being able to extract information from images. There are basically two questions about image formation:

a) What determines where the image of some point will appear?

b) What determines how bright the image of some surface will be?

Agreeing that it is very important to know how an image is formed in order to analyze it, we have to study two things: First, we need to find the geometric correspondence between points in the scene and points in the image, and second, we must find out what determines the brightness at a particular point in the image. The next section addresses the first issue.


One image: infinite solutions for structure
4
4
4
4
4
5
5
5
6
6
3
Unique structure
Equivalent
Unique illuminant direction; orthography used
Two images: uniqueness of structure (at most 4 solutions); Three images: unique solution; Local orthography used
Unique motion parameters (orthography or perspective) 3 frames or 2 frames with conditions
Absolute depth
Unique 3-D motion for nonplanar surfaces (no correspondence)
Unique motion parameters for planes; perspective used
Unique 3-D motion; no correspondence required (perspective)
Unique structure and depth for planes under perspective (no correspondence)
Unique structure and depth under perspective (no correspondence)
One view under perspective and using the uniform density assumption uniquely recovers orientation, despite Steven's beliefs

2.6 Geometric Correspondence Between Points in the Scene and the Image

2.6.1 Perspective projection

Consider an ideal pinhole at a fixed distance in front of an image plane (see Figure 2.3). Let us assume that an enclosure is provided so that only light coming through the pinhole can reach the image plane. Given that light travels along straight lines, each point in the image corresponds to a particular direction defined by a ray from that point through the pinhole. This is what we know as perspective projection.

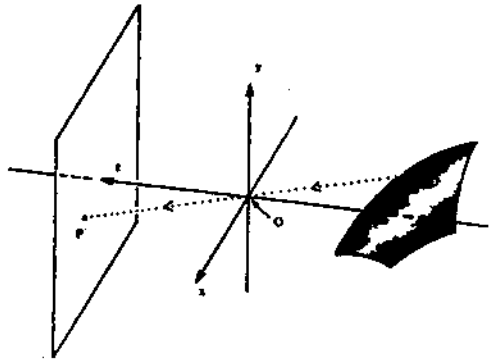


Figure 2.3: Perspective projection

In the sequel, in order to simplify the resulting equations, we consider the nodal point of the eye (pinhole) behind the image plane. This is only for simplifying the analysis; all the results can be transformed automatically to the actual case. The system we will be using is depicted in Figure 2.4.

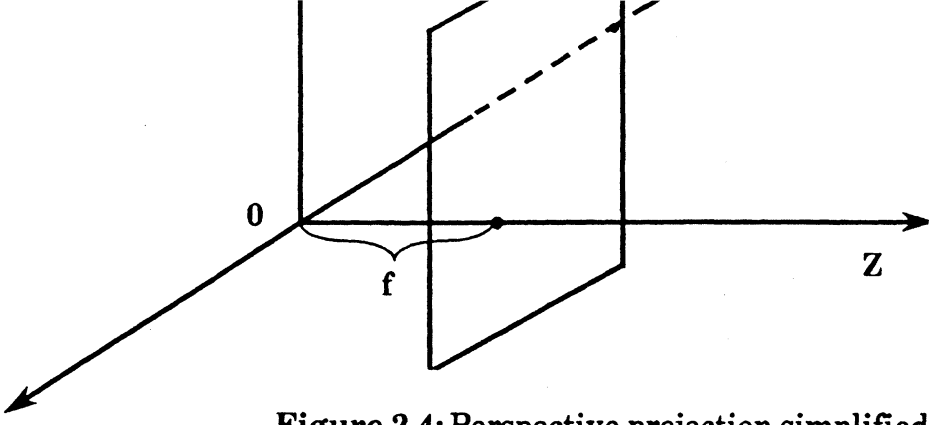


Figure 2.4: Perspective projection simplified

We define the optical axis in this case to be the perpendicular from the pinhole to the image plane. We introduce a cartesian coordinate system with the origin at the nodal point and the z-axis aligned with the optical axis and pointing toward the image (Figure 2.4). We would like to compute where the image A' of the point A on some object in front of the camera will appear. We assume that nothing lies on the ray from point A to the nodal point O. Let $V = (X,Y,Z)$, the vector connecting O to A and $V' = (x,y,f)$, the vector connecting O to A', with f the focal length, i.e., the distance of the image plane from the nodal point O, and (x,y) are the coordinates of the point A' on the image plane in the naturally induced coordinate system with origin the point of the intersection of the image plane with the optical axis, and axes x and y parallel to the axis of the camera coordinate system OX and OY. It is trivial to see that

$$x = \frac{fX}{Z}, \; y = \frac{fY}{Z} \qquad (2.1)$$

Equations (2.1) relate the image coordinates to the world coordinates of a point. Very often, to further simplify the equations we [illegible] without loss of generality.

2.6.2 Orthographic projection

The orthographic projection model seems unrealistic to the eye of the beginner and so we will motivate its use. If, in the perspective projection model, we have a plane that lies parallel to the image plane at $Z = Z_0$, then we define as magnification, mg, the ratio of the distance between two points measured in the image to the distance between the corresponding points on the plane. So, if we have a small interval on the plane (dX, dY, O) and the corresponding small interval (dx, dy, O) in the image, then:

$$mg = \frac{(dx)^2 + (dy)^2}{(dX)^2 + (dY)^2} = \frac{f}{Z_0} < 1$$

So a small object at an average distance Z_0 will produce an image that is magnified by mg. It is obvious that the magnification is approximately constant when the depth range of the scene is small relative to the average distance of the surfaces from the camera. In this case we can simply write for the projection (perspective) equations, that:

$$x = mX, \quad and \quad y = mY \qquad (2.2)$$

with $m = f/Z_0$ and Z_0 the average value of the depth Z. For our convenience, we can set $m = 1$. Then equations (2.2) are further simplified to the form:

$$x = X, \ and \ y = Y \qquad (2.3)$$

These equations (2.3) model the orthographic projection model, where the rays are parallel to the optical axis (see Figure 2.5). So, the difference between orthography and perspective is small when the distance to the scene is much larger than the variation in distance among objects in the scene. A rough rule of thumb is that perspective effects are significant when a wide angle lens is used, while images taken by telephoto lenses tend to approximate orthographic projection, but, of course, this is not exact [Horn, 1986].

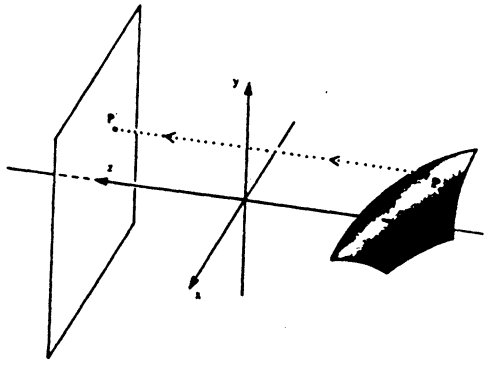


Figure 2.5: Orthographic projection

2.6.3 Paraperspective projection

The orthographic projection is a very rough approximation of the projection of light on the fovea, but it seems unrealistic for machine vision applications at this point. The perspective projection, a true model, sometimes produces very complicated equations for most of the problems and makes the subsequent analysis very hard. The paraperspective projection is a very good approximation of the perspective, and stands between orthography and perspective. A very similar form of the paraperspective projection was first introduced by Ohta *et al.* [Ohta et al, 1983]. Let a coordinate system $OXYZ$ be fixed with respect to the camera, with the $-Z$ axis pointing along the optical axis and O the nodal point of the eye. Again we consider the image plane perpendicular to the X axis at the point (0,0,-1) (i.e. focal length $f = 1$, without loss of generality). Let a small planar surface patch SP on a surface S, with the planar patch obeying the equation $-Z = pX + qY = C$ (see Figure 2.6).

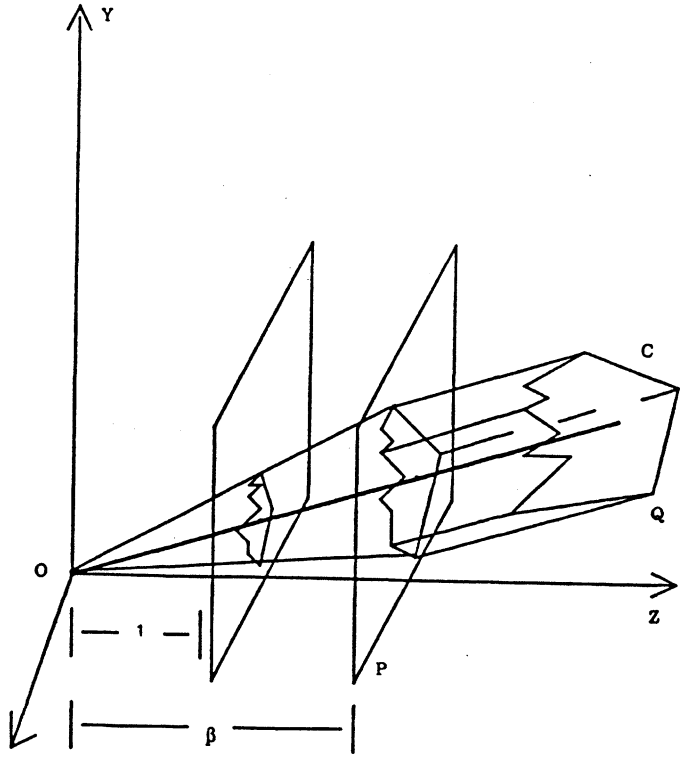


Figure 2.6: Paraperspective projection

Under perspective, any point $(X,Y,Z) \varepsilon SP$ is projected onto the point $(X/Z, Y/Z)$ on the image plane. Let us now see how the small patch SP is projected under the paraperspective projection model.

Consider the plane $-Z = d$, where $-d$ is the Z-coordinate of the center of mass of the region SP. The paraperspective projection is realized by the following two steps:

a) First, the small region SP is projected onto the plane $-Z = d$, which plane is parallel to the image plane and includes the center of mass of the region SP. The projection is performed by using the rays that are parallel to the central projecting ray OG, where G is the center of mass of the region SP.

b) The image on the plane $-Z = d$ is now projected perspectively onto the image plane. Since the plane $-Z = d$ is parallel to the image plane, the transformation is a reduction by a scaling factor $1/d$ (see Figure 2.7 which illustrates a cross sectional view of the projection process sliced by a plane which includes the central projecting ray and is perpendicular to the XZ plane). Finally it is clear that the introduced model decomposes the image distortions in two parts: Step (a) captures the foreshortening distortion and part of the position effect, and step (b) captures both the distance and the position effects.

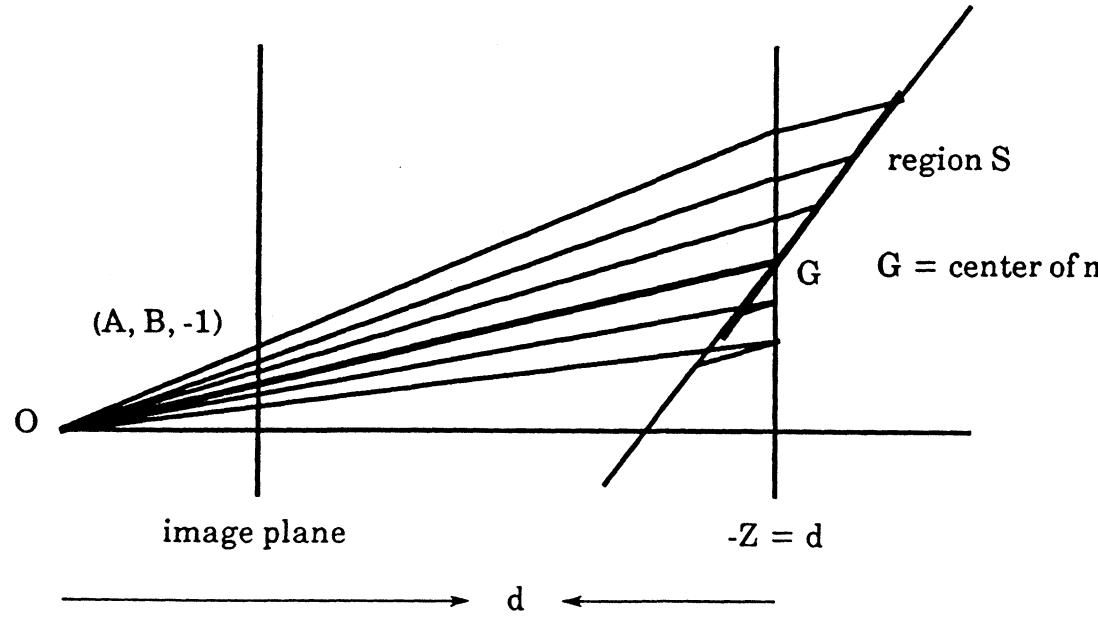


Figure 2.7: Cross sectional view of paraperspective

The paraperspective projection process turns out to have nice mathematical properties, since it is an affine transformation. Chapter 3 describes in detail the properties of this projection and its comparison with perspective and orthographic projections.

After having discussed the geometric correspondence between points in the image and points in the scene, we need now to determine the brightness at each image point. But to do that we need some technical prerequisites, which will be found in the next section on intrinsic images.

2.7 Intrinsic Images

In the previous chapter we stressed the fact that a very large percentage of modern computer vision is exploiting the recovery of three-dimensional properties (i.e. intrinsic images) from two-dimensional image properties. This section will define mathematically what we mean by intrinsic images, i.e. shape, motion, depth, etc.

Consider again a coordinate system $OXYZ$, fixed with respect to a camera, whose nodal point is the origin 0 and the image plane perpendicular to the Z-axis (which is also the optical axis), with focal length f. Consider also the naturally induced image plane xy coordinate system, with origin at the point where the optical axis intersects the image plane and x,y axes parallel to OX and OY respectively. Image coordinates will be denoted by small letters and world coordinates by capital letters. Suppose that the system is imaging a surface S with equation $Z = Z(X,Y)$.

2.7.1 What we mean by shape

We will examine shape under both orthography and perspective projection. Surface orientation is usually represented as the surface normal vector. In intrinsic images, shape means the local surface orientation, not some global property of the surface. If the surface is expressed as $Z(X,Y)$ it can be reconstructed from the local shape orientation.

The meaning of shape under perspective

Consider a point $(X,Y,Z) \varepsilon S$ whose image under perspective projection is the point $(x = fX/Z, y = fY/Z)$. If we say that we know the shape of the object in view at the point (x,y), we mean that we know the surface normal vector n of surface S at the point (X,Y,Z), in particular

$$\bar{n} = \left(\frac{\partial Z}{\partial X}, \frac{\partial Z}{\partial Y}, -1\right) \Big/ \left[\left(\frac{\partial Z}{\partial X}\right)^2 + \left(\frac{\partial Z}{\partial Y}\right)^2 + 1\right]^{1/2}.$$

Suppose now that for every point (x,y) in the image we know the surface normal of the surface patch whose image is the point (x,y). Then, this new image (a surface normal for each point (x,y) of the image) is called intrinsic shape image. But from only one image we can never hope to compute the exact (X,Y,Z) point, and from it $(\partial Z/\partial X, \partial Z/\partial Y, -1)$. What we can compute, though, is the quantity $(\partial Z/\partial x, \partial Z/\partial y)$, i.e., the gradient of the surface expressed in retinal coordinates. But then, what is the relationship between the gradient in retinal and world coordinates, or in other words, what do we know when we know the quantities $(\partial Z/\partial x, \partial Z/\partial y)$?

Consider a point (x,y) on the image and a small displacement in the image (dx,dy) from the point, which corresponds to a displacement (dX, dY, dZ) in the world, on the surface $Z = Z(X,Y)$. Then, from the perspective projection equations, we have:

$$dX = \frac{dx \cdot Z + x\,dZ}{f} \quad and \quad dY = \frac{dy \cdot Z + y\,dZ}{f}$$

Now, given that $Z(X+dX, Y+dY) = Z(x + dx, y+dy)$, and expanding both sides of this equation in a Taylor series and ignoring the higher order terms, we get that:

$$\frac{\partial Z}{\partial X}\frac{Z}{f - x\frac{\partial Z}{\partial X} - y\frac{\partial Z}{\partial Y}}dx + \frac{\partial Z}{\partial Y}\frac{Z}{f - x\frac{\partial Z}{\partial X} - y\frac{\partial Z}{\partial X}}dy = dx\frac{\partial Z}{\partial x} + dy\frac{\partial Z}{\partial y}$$

from which

$$\frac{\partial Z}{\partial x} = \frac{Z\frac{\partial Z}{\partial X}}{f - x\frac{\partial Z}{\partial X} - y\frac{\partial Z}{\partial Y}} \quad and \quad \frac{\partial Z}{\partial y} = \frac{Z\frac{\partial Z}{\partial Y}}{f - x\frac{\partial Z}{\partial X} - y\frac{\partial Z}{\partial Y}}$$

From equations (2.5) it is easy to see that if $\delta Z/\delta X, \delta Z/\delta Y$ are known, then the quantity

$$\frac{Z(x + dx, y + dy)}{Z(x,y)}$$

is computable. But this means that if the surface normals are known indexed by retinal coordinates, then the depth function *(Z(x,y))* can be computed up to a constant factor. In other words, if shape is known, then for any two points (x_i,y_i) and (x_j,y_j) on the image, we know the ratio

$$\frac{Z(x_i, y_i)}{Z(x_j, y_j)} .$$

So, an object whose shape we know under perspective projection can be small and near the camera or large and far away.

The meaning of shape under orthography

Under orthographic projection, the image coordinates of a point are equal to the corresponding 3-D coordinates, i.e. (x,y) = (X,Y). So

$$\left(\frac{\partial Z}{\partial X}, \frac{\partial Z}{\partial Y} \right) = \left(\frac{\partial Z}{\partial x}, \frac{\partial Z}{\partial y} \right) .$$

Obviously, if we know shape in this case, since

$$Z(x + dx, y + dy) - Z(x,y) = \frac{\partial Z}{\partial x}\ dx + \frac{\partial Z}{\partial y}\ dy + (h.o.t.),$$

we know that the depth function can be computed up to constant additive term. So, if we know shape under orthography, we know exactly the object in view, but we do not know its depth.

Other representations for shape

We have stated that the surface normal

$$\frac{\left(p, q, -1\right)}{(p^2 + q^2 + 1)^{\frac{1}{2}}}$$

with $p = \delta Z/\delta X$, $q = \delta Z/\delta Y$ at a point of a surface $Z = Z(X,Y)$ represents the shape. This

$$(p,q) = \left(\frac{\partial Z}{\partial X}, \frac{\partial Z}{\partial Y} \right)$$

is not the only representation. Obviously shape is nothing but a direction in three-dimensional space, and so there are many representations for it. The ones that we will use* quite often in this thesis are, with the exception of the gradient that we have already analyzed, the following:

a) Coordinates (a,b,c) on the Gaussian sphere .

b) Latitude and longitude angles, say, (9, p).

c) Slant and tilt. Slant is the tangent of the latitude angle and tilt is the longitude angle. The notation for (slant, tilt) is (o, x). The slant and tilt are polar versions of of the (p,q) coordinates.

The relationship among these different representations is given by the following equations:

$$o = \tan 8 = V(p2 + q2)$$

$$p/q = \tan \phi = \tan t$$

Finally, if (a,b,c) are the coordinates on the Gaussian sphere, then:

$$(a,6,c) = \wedge \ , \ | \ , \ \wedge \ ) \quad \textit{with} \quad * = (p^2 + q^2 + 1) *$$

2.7.2 What we mean by retinal motion

If the object in view is moving with a general motion, or if the camera is moving, or if both move, then the image is moving too. Let the retinal velocity at an image point be *(u,v).* The resulting vector field (the velocity of every image point) is called retinal motion field or optic flow field. This flow field is an intrinsic retinal motion image.

2.7.3 What we mean by depth

Consider again a surface *S* with equation Z = *Z(X,Y)* in front of the camera. Every point *(x,y)* in the image is the projection of a point *(X,Y,Z)* e S. If for every point *(xy)* on the image we know the *Z* coordinate (depth) of the corresponding 3-D point $(X,Y,Z)_f$ then we know exactly where the surface is with respect to the camera coordinate system. The

resulting image (for every point in the image there corresponds a number (depth) of the corresponding 3-D point), is called intrinsic depth image.

2.7.4 Intrinsic parameters that are not images

There exist intrinsic parameters which do not correspond to every point in the image. These are global constants and every point in the image is in some relation to them. Examples of these parameters are the 3-D motion and lighting direction parameters.

3-D motion parameters

If an object moves in front of a camera with a general motion, then this motion can be considered as the sum of a translation (U,V,W) and a rotation (A,B,C). These six parameters will be called motion parameters

Lighting direction parameters

Consider again a surface in front of a camera, illuminated by a light source in the direction (l_x,l_y,l_z), with respect to the camera coordinate system. The direction (l_x,l_y,l_z) is called the lighting or illuminant direction.

2.8 A synopsis

Up to this point we have defined mathematically so-called intrinsic parameters. These are shape, depth, retinal motion, 3-D motion, and light source direction. This of course does not mean that these are the only intrinsic parameters. There can be many more but the ones that we described here are the ones which we (and contemporary research) think that are the most important for the perception of the outside world. Again, we do not want to get involved in philosophical arguments about why these intrinsic parameters are important to compute for visual perception. The shape of objects is important for the recognition of objects that we see, the depth of objects is important for our interaction with the environment (picking up things), retinal motion is important for understanding discontinuities and segmenting the environment as well as for the computation of the 3-D motion which is important for navigation and for understanding the motion of objects in our environment as well as for avoiding moving objects.

There may very well be other important intrinsic parameters that we haven't discovered yet. There may also be no more intrinsic parameters of interest. Further research will uncover the truth on this matter.

2.9 Brightness at every image point

In this section we analyze how the brightness at every image point is determined. The amount of light reflected by a surface element depends on its microstructure, on its optical properties and on the distribution and state of polarization of the incident illumination. For several surfaces, the fraction of incident illumination reflected in a particular direction depends only on the surface orientation. The characteristics of the reflectance of such a surface can be represented as a function *f(i,g,e)* of the angles *i* = incident, *g* = phase and *e* = emergent, as they are defined in Figure 2.8.

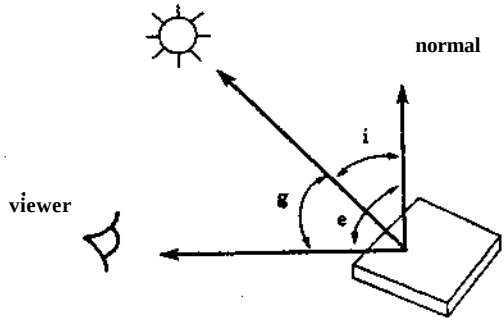


Figure 2.8: Reflectance model

The reflectance function *f(i,g,e)* determines the ratio of surface radiance to irradiance measured per unit surface area, per unit solid angle, in the direction of the viewer. If we want to be precise, we should specify the quantities and units used to define the required ratio. Here it is sufficient to point out the role that surface orientation plays in the determination of the angles *i* and *g*.

Consider the example of perfect specular (mirror-like) reflection. In this case, the incident angle equals the emergent angle and the incident, emergent and normal vectors lie on the same plane *(g = i + e)*. So, the reflectance function is

$$f(i,e,g) = \begin{cases} 1 & \text{if } i = e \text{ and } i + e = g \\ 0, & \text{otherwise} \end{cases}$$

The interaction of light with surfaces of varying roughness and composition of material leads to a more complicated distribution of reflected light. Surface reflectance characteristics can be determined empirically, derived from models of surface mircostructure or derived from phenomenological models of surface reflectance. The most widely used model of surface reflectance is given by the function *f(i,e,g)* = p cosi, where p is a constant depending on the specific surface. This reflectance function corresponds to a phenomenological model of a perfectly diffuse (Lambertian) surface which appears equally bright from all viewing directions; the cosine of the incident angle accounts for the foreshortening of the surface as seen from the source.

The surface normal vector relates surface geometry to image irradiance because it determines the angles *i* and *e* appearing in the surface reflectance function *fli,e,g).* In orthographic projection,the viewing direction and so the phase angle *g* is constant for all surface elements. So, for a fixed light source and viewer geometry and fixed material, the ratio of scene radiance to scene irradiance depends only on the surface normal vector. Furthermore, suppose that each surface element receives the same irradiance. Then, the scene radiance and hence image intensity depends only on the surface normal vector. A reflectance map *R(p,q)* determines image intensity as a function of *p* and *q* (where *(p,q,-l)/^(p2 + q^2 +1)* is the surface normal vector). Using a reflectance map, an image irradiance equation can be written as *I(x,y) = R(p,q),* where *I(x,y)* is the intensity at the image point *(x,y)* and *R(p,q)* is the corresponding reflectance map.

A reflectance map provides a uniform representation for specifying the surface reflectance of a surface material for a particular light source, object surface and viewer geometry. A comprehensive survey of reflectance maps derived for a variety of surface and light source conditions has been given by Horn [Horn, 1977]. Furthermore, a unified approach to the specification of surface reflectance maps has been given in [Horn and Sjoberg, 1981]

Expressions for cosi, cose and cosg can be easily derived from the surface normal vector *(p,q, -1)* and the light source vector *(p_{8t} q_{8f} -1)* and the vector *(0,0,* -2) which points

in the direction of the viewer. For a Lambertian reflectance function we get

$$R(p,q) = \frac{\rho\ (1 + p\,p_s + q\,q_s)}{\sqrt{(1 + p^2 + q^2)}\ \sqrt{\ (1 + p_s^2 + q_s^2)}}$$

So, for a Lambertian surface, the intensity $I(x,y)$ at a point (x,y) of the image is given by:

$$I(x,y) = \frac{\rho\,(1 + p\,p_s + q\,q_s)}{\sqrt{(1 + p^2 + q^2)}\ \sqrt{(1 + p_s^2 + q_s^2)}}$$

with ρ the albedo constant and $(p,q,-1)$ and $(p_s, q_s, -1)$ the surface normal at the point whose image is the point (x,y) and the light source direction respectively, under orthographic projection. Under perspective projection, the model is not known yet exactly.

2.10 What is to come

Once again, this thesis does not try to present a unified theory for the computation of the intrinsic images. Much more research is required for that, and the last chapter sheds some light on this issue. Instead, it tries to prove mathematically that if several cues are combined and if the right (natural) assumptions are employed, then we can obtain visual computations which uniquely and robustly compute intrinsic images.

Chapter 3 is devoted to the problem of shape from texture where it is demonstrated that a modified Gibsonian assumption leads to an algorithm that works for a variety of natural images. In this Chapter we demonstrate that the right assumptions are bound to give good results. Chapter 4 examines the problem of shape from shading, which leads to the conclusion that it cannot be solved. After this, if shading is combined with motion (or stereo), then this leads to algorithms that uniquely compute shape from these cues. Chapter 5 is devoted to visual motion analysis. The feasibility of the problem of structure from motion is examined and several new theorems of theoretical importance are proved. Finally, it is shown that if motion is combined with stereo, then robust solutions for the structure from motion problem are obtained and that motion analysis can be done without point correspondences. Finally, Chapter 6 is devoted to the analysis of the perception of shape from contour, and the advantages of combining stereo, contour and texture. Chapter 7 presents the conclusions from this work, sets forth foundations for future work and discusses the beginning of a unified early vision theory which works in a highly

parallel fashion, where the different processes cooperate to integrate information from different sources and compute uniquely and robustly the parameters of our extrapersonal space.

3

Shape from Texture

Results

Here we study the problem of determination of shape from texture. In particular:

1) We show how to recover the shape of a surface covered with small elements (texels) of the same area. The shape of the texels is of no importance to our theory. Furthermore we indicate that there is a very strong connection between shading and texture.

2) For natural textures, we show that the uniform density assumption is enough to recover the orientation of a single textured plane in view, under perspective projection. Furthermore, when the texels cannot be found, the edges of the image are enough to determine shape, under a more general assumption, that the sum of the lengths of the contours on the world plane is about the same everywhere. The problem is examined under both perspective and paraperspective projection. The results in the case of paraperspective projection are better than in the case of perspective. Finally, several experimental results in synthetic and natural images are presented.

The basic assumption here is that we are imaging a single textured plane. For the methods developed here to be applied to an image where several planes are present, a segmentation is required first. In the conclusion of this chapter, we describe how this theory could be used for such a segmentation.

A central goal for visual perception is the recovery of the three-dimensional structure of the surfaces depicted in an image. Crucial information about three-dimensional structure is provided by the spatial distribution of surface markings, particularly for

static monocular views: projection distorts texture geometry in a manner that depends systematically on surface shape and orientation. To isolate and measure this projective distortion in an image is to recover the three-dimensional structure of the textured surface.

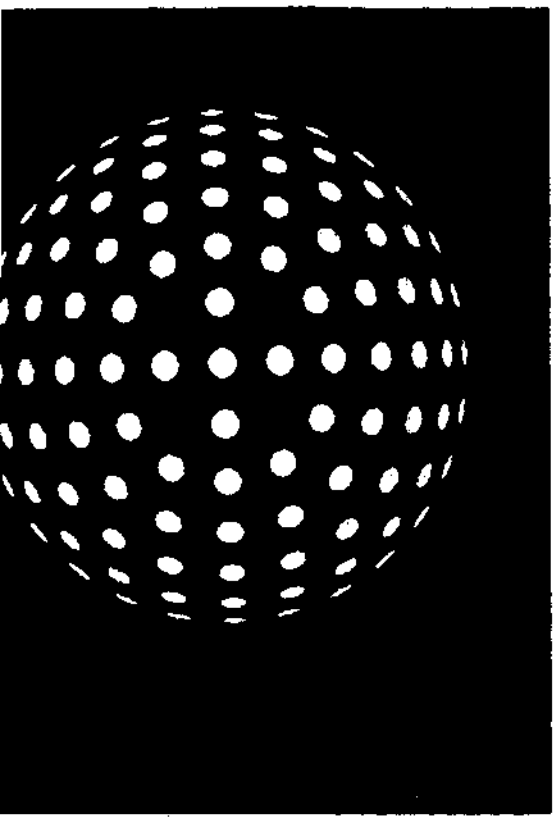
In order to study the problem of detecting surface orientation from texture, we need to distinguish between two kinds of texture: artificial texture (or pattern texture) and natural texture. When we say that an object is covered with artificial texture, we mean that the surface of that object is covered with repeated patterns of the same area. When we say that a surface is covered wtih natural texture, we mean that the surface is irregularly marked. Both kinds of texture are important for recovering 3-D structure, and for this reason we will study both of them, beginning with artificial texture. Figure 3.1 depicts the monocular images of surfaces covered with artificial and natural texture respectively. It is remarkable how humans can infer the three-dimensional structure of the imaged surfaces clearly with the help of texture.

3.1 Detecting surface orientation from artificial texture, or shape from patterns

The problem we address here is to recover the three-dimensional shape of a surface covered with repeated texture elements of the same area, which we will call texels, from a monocular view. This problem, known in the literature as shape from patterns, has already been addressed by various researchers who obtained partial solutions, under certain assumptions. Previous work in this area has been developed with the use of three different kinds of projections: orthographic, perspective and spherical.

In the above figure we show examples of artificial and natural texture.

Kender [Kender, 1980] and Kanader and Walker [Walker and Kanade, 1984] studied the problem under orthographic projection. Kender assumes the patterns to be polygonal or symmetrical and recovers orientation using skewed symmetry constraints (knowing the angle between two axes in space and the angle they make in the image, constraints between 3-D surface orientation and measurable image parameters can be developed). For this he needs prior knowledge of symmetry or specific knowledge about the pattern, as well as some heuristics about the orientation of some of the patterns. Walker and Kanade use a combination of Kender's method and Shafer's theory of generalized cylinders



(a) (b)

Figure 3.1: (a) artificial texture, (b) natural texture

[Shafer, 1982] to recover surface orientation from patterns, under orthographic projection. But their method has limited applicability as reported.

Kender [Render, 1982] and Ohta *et al.[* Ohta *et al,* 1983] study the problem under perspective projection. Render's method is based on the vanishing point of parallel scene lines and as such is very limited to special kinds of patterns. On the other hand, Ohta's method is very ingenious even though it is strictly applicable only to planar surfaces. This method permits different kinds of texels on a plane and it provides a somewhat heuristic method for their separation, which does not always work. After the image texels have been separated into clusters of the same kind, the area ratios of two texels of the same kind provide rich information for the orientation of the imaged planar suface.

Finally, Ikeuchi [Ikeuchi, 1984] studies the problem under spherical projection and provides good results for images that fit his assumptions. In his work the texture elements on the world surface have to be known *a priori* and to be symmetrical; basically he

$$\frac{S_I}{S_w} \quad \frac{1}{p^2} \quad \left[\begin{array}{cc} \dfrac{-1+pA}{V i + p^2} & \dfrac{pB}{V(i+p^2)} \\ \dfrac{g(p+A)}{V(l+p^2)\,(1+p^2+q^2)} & \dfrac{qB-p\ -1}{V(l+p^2)\,(1+p^2q^2)} \end{array} \right]$$

or

$$\ddot{s}_I = \frac{^{S}u,}{\beta^2} \cdot \frac{1-Ap-Bq}{\sqrt{1+p^2+q^2}}$$

Equation (3.1) relates the area of a world texel Sw, its gradient (p,q), the area Si of its image and its mass center (A,B). If we call the quantity Si "textural intensity," and the quantity Sw/P^2 "textural albedo," then equation (3.1) is very similar to the image irradiance equation

$$i \stackrel{x}{=} \frac{l-Ap-Bq}{\sqrt{(1+p^2+q^2)}}$$

where/is the intensity (p,q) the gradient of the surface point whose image has intensity /, X is the albedo at that point and (A,B,1) the direction of the light souece [Horn 1977; Ikeuchi, 1981].

Thus equation (3.1) can be used to recover surface orientation, using methods that have been discovered for the solution of the shape from shading problem [Ikeuchi, 1981].

3,1.4 A gradient map

Equation (3.1) of the previous section can be written as

$$/ = R\ (p,q) \qquad <^3\text{-}^2)$$

where / is the textural intensity, i.e. the area of an image texel with mass center (A,B), and

$$\sqrt{(}$$

with X the textural albedo,i.e. the quantity Sw/p^2, and (p,q) the gradient of the plane on which the world texel lies. The function R(pq,) we call textural reflectance. If we fix the albedo X, and the position (A,B) of the texel on the image, then equation (3.2) can be

3.1.2 Paraperspective projection: An approximation of the perspective projection by a 2-D affine transformation

Let a coordinate system OXYZ be fixed with respect to the camera, with the -Z axis pointing along the optical axis, and O the nodal point of the eye (center of the lens), as in 2.6.3. The image plane is assumed to be perpendicular to the Z axis at the point (0,0,-1), i.e. focal length = 1. If P is the depth of the center of mass of the world pattern that lies on a plane with gradient (p,q), then to represent the original pattern of the surface texel, we use an (a,b,c) coordinate system, with its origin at the mass center of the texel. To represent the pattern of the image texel, we use and (a', *b\ c')* coordinate system, with its origin the point (A,B, -1), i.e. the mass center of the image texel, and the axes a^1, b\ c* parallel to the axes X, Y, Z respectively. Then the transformation from (a,b) to (a',b^f) with the two step projection process of the previous section is given by the affine transformation

$$\begin{bmatrix} a' & 6' \end{bmatrix} = \begin{bmatrix} a & b \end{bmatrix} \frac{1}{S} \begin{bmatrix} \frac{-1+pA}{V(i+p^2)} & \frac{pB}{V(i+p^2)} \\ \frac{q(p+A)}{\sqrt{(1+p^2)(l+p^2+q^2)}} & \frac{qB-p^2-1}{V(l+p^2)(l+p^2+q^2)} \end{bmatrix}$$

It is clear that this transformation is the relation between two 2-D patterns, one in the 3-D space and the other its image on the image plane. We now use this affine transformation to develop the desired constraint.

3.1.3 The constraint

The determinant of the matrix of an affine transformation is equal to the ratio of the areas of the two patterns before and after the transformation. Specifically, if Sw is the area of a world texel that lies on a plane with gradient (p,q) and Si is the area of its image that has mass center (A,B), then we have:

develops constraints similar to Kender's, but in a simpler form because of the properties of the spherical projection. In this work, we determine the shape of a surface covered with repeated texels , from a monocular view, under the following assumptions:

(1) The surface in view is smooth and is covered with repeated texture elements. All the texture elements on the surface are of the same area. These texture elements we call texels. The shape of the texels is of no importance for our theory.

(2) Each texture element is assumed to lie on a plane (i.e., we assume that the surface in view is locally planar). This means that the size of the texels on the surface has to be small compared with a change of surface orientation there.

(3) The scene texture is imaged under paraperspective projection (section 2.6.3).

The fact that the surface in view is smooth, enables us to use existing techniques already applied to recover shape from shading [Ikeuchi and Horn, 1981], that make use of smoothness constraints. Although the technique that we will use falls in the regularization paradigm, it is of significant value for this case. We insist on the fact that regularization cannot be applied to unrestricted natural images, but in this case since the inherent assumption for the case of artificial texture is smoothness, and the domain that we will address in our experiments consists of smooth objects, the method that we will develop is valid and useful.

Under the above assumptions, we develop a new gradient map which will enable us to define a "textural reflectance function." Our theory is very similar to earlier work on shape from shading [Horn, 1977; Ikeuchi, 1981], with the image intensity at a point replaced with the area of the image texel at that point.

We value the following analysis and the suggested algorithms, not only because they provide a good way for detecting shape from patterns, but also because they provide insight to a possible unified approach for the perception of shape from texture and shading, since our mathematical findings with respect to this problem suggest that under the appropriate formulation, the problems of shape from shading and shape from texture can be solved in the same basic way.

represented conveniently as a series of contours of constant textural intensity. Figure 3.3 illustrates such a simple textural reflectance map.

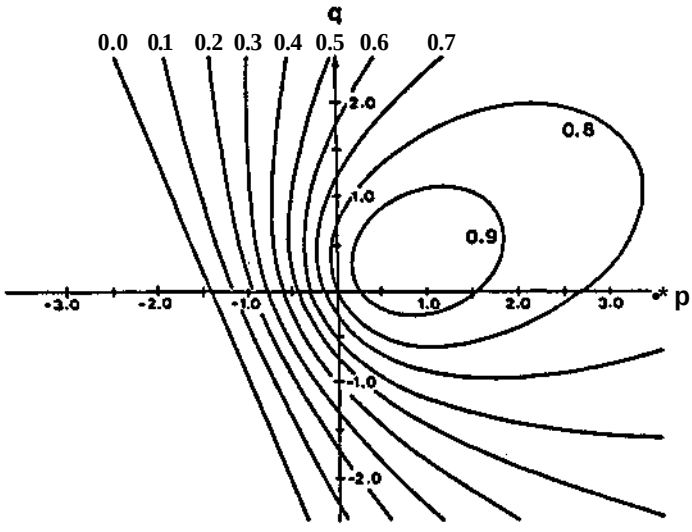


Figure 3.3: The gradient map

In the above figure, we present the textural reflectance map for a point (A,B) = (-7,-.3) with textural albedo X = 1. The reflectance map is plotted as a series of contours spaced one unit apart.

3.1.5 Recovering the textural albedo

We use equation (3.2) of the previous section to recover the local surface orientation. No matter what method we use we must know the textural albedo $X = S_W/\beta^2$.

We cannot know β from a static monocular view; neither can we know S_W in general. But it turns out that we can compute approximately the ratio S_W/β^2, i.e. the textural albedo λ.

Consider three neighboring image texels T_1, T_2 and T_3 with areas I_1, I_2 and I_3 and we suppose that the world texels whose iamges are the texels T_1, T_2 and T_3 lie on the same plane with gradient (p,q). Then the following equations arise:

$$I_1 = \lambda (s_1, n) \tag{3.3}$$

$$I_2 = \lambda (s_2, n) \tag{3.4}$$

$$I_3 = \lambda (s_3, n) \tag{3.5}$$

where $n = (p,q,1)/\sqrt{(1+p^2+q^2)}$ and $s_i = (A_i, B_i, 1)$ for $i = 1,2,3$ and (A_i,B_i) the mass center of texel T_r. Eliminating the textural albedo λ from the equations (3.3), (3.4), and (3.5) we get:

$$n = k\left[I_1 (s_2 X s_3) + I_2 (s_3 X s_1) + I_3 (s_1 X s_2)\right]$$

$$\lambda = \frac{1}{k\left[s_1, s_2, s_3\right]}$$

for some constant k that makes n a unit vector, where $[s_1,s_2, s_3] = s_1(s_2 \text{ X } s_3)$ and provided that $[s_1, s_2, s_3] \neq 0$, i.e. the vectors s_1, s_2, and s_3 are not coplanar (linearly dependent).

The result of equation (3.7) is approximate due to the hypothesis that three neighboring texels lie on the same plane. But, if we perform this process in all the triples of neighboring points, and we take the average value for the albedo, then the result is highly improved. At the same time, we can get an approximate value for the surface normals at all the texels in the image (equation (3.6)). Then we can use these initial approximations to start the iterative algorithm that will be introduced in the next section.

3.1.6 Another way to recover the albedo

Following Ohta *et al.* [1980], and assuming local planarity, i.e. three neighboring texels belong to the same plane which we call Q, we have that:

$$\frac{f_1}{f_2} = \left(\frac{s_1}{s_2}\right)^{\frac{1}{3}}$$

where fcj, & are the distances from two texels to the vanishing line of the plane Q along the line joining the two texels and $j, *S2* are the areas of the two texels in the image. Since I//-/2I is just the distance between the two texels in the image and it is known, a point on the vanishing line may be determined. With a third texel, two points may be determined, which give the equation of the vanishing line [Render, 1980]. Since the equation of the vanishing line of the plane Q is px + qy = l, the orientation of the plane Q can be determined, and from that an approximation of the textural albedo is found.

3.1.7 Additional constraints and propagation of the constraints

In this section we introduce the smoothness constraint [Ikeuchi, 1981] and we present an iterative algorithm of the same flavour as the one introduced by Ikeuchi.

3.1.8 An iterative propagation algorithm

We have already proved that every distortion value (image texel area) for a specific image position corresponds to a contour in the gradient space (See section 3.4). So, the problem has infinite solutions and this is the reason that we introduce the smoothness assumption. A smoothness constraint can be used to reduce the locus of possible orientations to a unique orientation, through an iterative algorithm.

Trying to develop a global error function that should be minimized in order to give the desired value, we measure the departure from smoothness and the error in the textural reflectance equation (equation (3.2)). The error in smoothness we measure (after [Ikeuchi, 1981]) as follows:

where py and <fy denote the orientation at the surface point whose image is the point (ij). The error in the textural reflectance equation, can be given by:

$$e_{i,j} = (I_{i,j} - R(p_{i,j}, q_{i,j}))2$$

where /;j is the distortion value (texel area) at the point *(ij)* and *R* the textural reflectance.

An acceptable solution should minimize the sum of the error terms in all the grid nodes. If *E* is such a global error function, then

$$E = \sum_i \sum_j (s_{i,j} + \omega e_{i,j})$$

and the factor co gives a weight to the errors in the textural gradient map relative to the "distance" from smoothness. To minimize E_t we differentiate with respect to *pij* and *qij* and setting the resulting derivatives to zero and rearranging the equations, we obtain:

$$p_{ij} = pa_{ij} + \circ [I_{ij} - R(pa_{ij}, qa_{ij})] \frac{dR}{dp} T^d$$

$$q_{ij} = qa_{ij} + a [I_{ij} - R(pa_{ij}, qa_{ij})] \frac{dR}{dZ}$$

where *pqij* and *qaij* are the average values of p and q around the point (ij) respectively. The above equations suggest an adjustment of p and q in the direction of the gradient of the textural reflectance function, by an amoung that is proportional to the error in the textural reflectance equation (equation (3.2)). So it is natural to use the following iterative rule for the estimation of the p and q everywhere in the image:

$$p_{ij}^{n+1} = pa_{ij}^{n} + co[I_{ij} - R(p^n, q^n)] \frac{\partial R}{\partial p}$$

$$q_{ij}^{n+1} = qa_{ij}^{n} + \omega [I_{ij} - R(p_{ij}^n, q_{ij}^n)] \frac{\partial R}{\partial q}$$

In the above equations the partial derivatives of the textural reflectance are evaluated on the values of p and q of the n-th iteration. Finally, to avoid numerical instabilities we modify the above formulas to the following form [Ikeuchi & Horn, 1981]:

$$p^{n+1} = pa^n + a[I - R(pa^n, qa^n)] \frac{\partial R}{\partial p}$$

$$q_{IJ}^{n+1} = qa_{IJ}^{n} + \cos\left[R(pa_{IJ}^{n}, qa_{IJ}^{n}) \right] \frac{\partial R}{\partial p}$$

R(Pij,qij) is a function on a four-dimensional space, unlike the R(p,q) of orthographic shape from shading. The shading *R(p,q)* can be determined empirically, but the textural reflectance *R(pij, qij)* is an analytic, geometrical entity arising from imaging geometry, and thus only the global constant (texture) albedo varies from texture to texture and scene to scene.

3*1.9 Experiments

The algorithm was tested on artificial images of a plane, cylinder, sphere, ellipsoid and a donut shaped object. There are four distinct steps into which the program may be broken down:

1) Location of texels,

2) Minimum triangulation of the texel centers,

3) Calculation of initial orientations and textural albedo,

4) Iterative process.

In 1), the connection regions in the image are detected. Their centers of gravity are taken to be the locations of the texels. Their size is recorded and the texels which are in the boundary are marked [Ballard & Brown, 1982]. In 2), the points denoting the centers of the texels are triangulated so that the sum of the length of the lines is minimum [Aho, Hopcroft & Ullman]. In 3), the estimate of *k* was calculated from the local orientation with the lowest value of p and q. Due to curvature of the surface, convex objects tend to give an overestimate of A while concave objects tend to give an underestimate. These errors are minimized when the surface of the object is most nearly perpendicular to the image plane. The algorithm is quite insensitive to initial orientations given to texels whose orientations were allowed to vary through the iterative process. Boundary texels were not allowed to change. The error in calculating their values was the predominant factor in influencing the total error. The iterative process took under 10 iterations. The process always converged for our synthetic images. The final error values were

	fractional error
plane	negligible
sphere	.005
cylinder	.015

The errors in the table denote the average percent error at each texel. The error at each texel was taken to be $1/4\pi \cdot \sigma$, where σ = solid angle subtended by rotating the calculated orientation about the actual orientation. Figure 3.4 gives a pictorial description of the error at each texel.

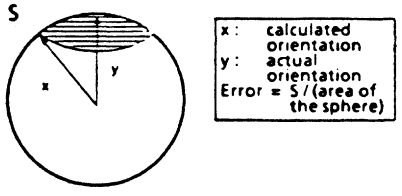


Figure 3.4: Schematic description of the error

Finally, azimuthal equidistant coordinates (AEC) [Ikeuchi&Horn, 1981] were used through the iterative process instead of the gradient space p and q, since AEC change linearly with change in orientation. The AEC can be easily understood in the following way. Consider the Gaussian sphere and the gradient space plane tangential to it at the north pole, in the origin of the gradient space. In order to find on the sphere the AEC of a point in gradient space, we roll the sphere to the direction of the gradient space point, until the sphere touches the point. The corresponding point on the sphere gives the AEC of the gradient space point. Figure 3.5 shows the image of a sphere which is covered with a repeated pattern. Figure 3.6 shows the reconstructed sphere using the algorithms of Sections 3.1.7, and Figure 3.7 shows the reconstructed sphere after the relaxation. Figures 3.8, 9, and 10 and 11 and 12 show the analogous pictures for a cylinder and plane

respectively. Figures 3.12.1, 3.12.2 (triangulation), 3.12.3 and 3.12.4 show similar experiments for an ellipsoid and figures 3.12.5, 3.12.6 (triangulation), 3.12.7 and 3.12.8 show similar experiments for a donut shaped object.

In the previous sections we studied the problem of determining surface shape from artificial texture, *i.e.* from the apparent distortion of patterns. In the sequel, we will study the problem of determining surface orientation from natural texture.

Figure 3.5: Input (sphere)

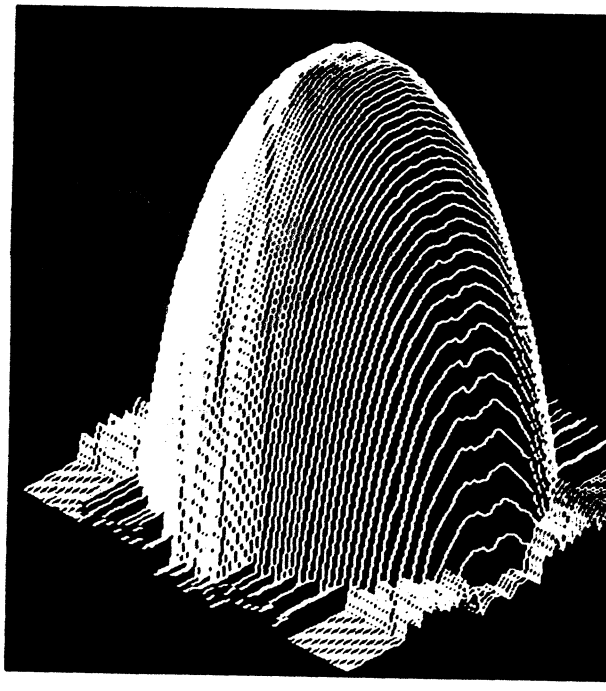


Figure 3.6: First phase

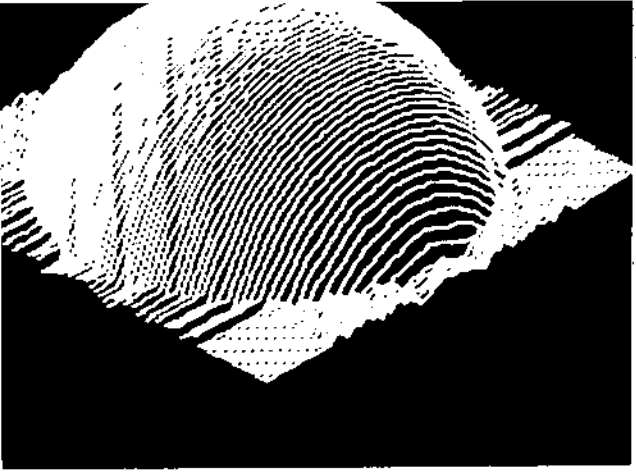


Figure 3.7: Result (reconstruction)

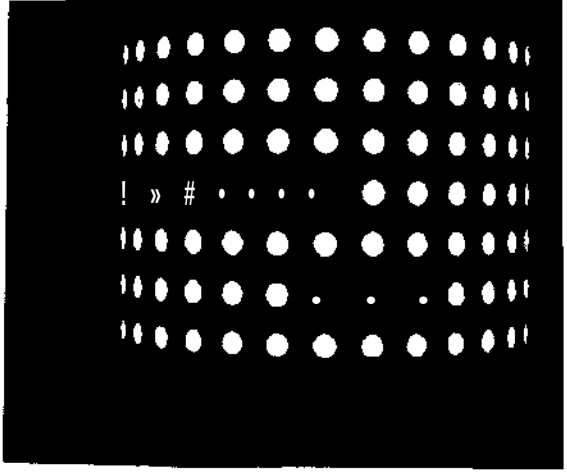


Figure 3.8: Input

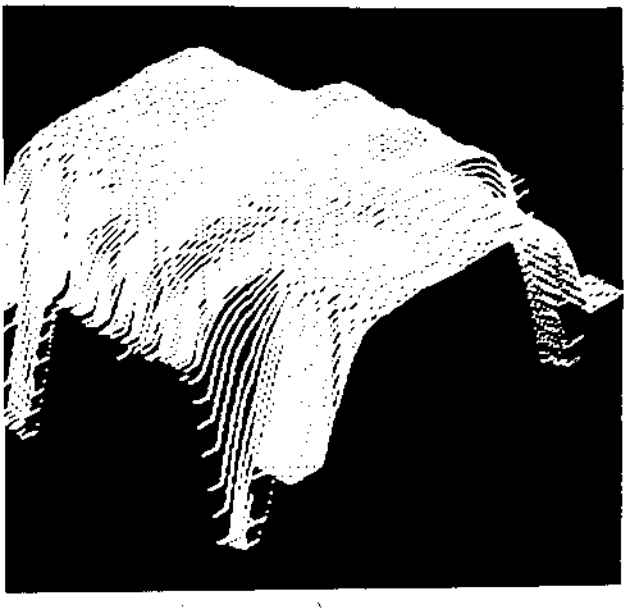


Figure 3.9: First phase

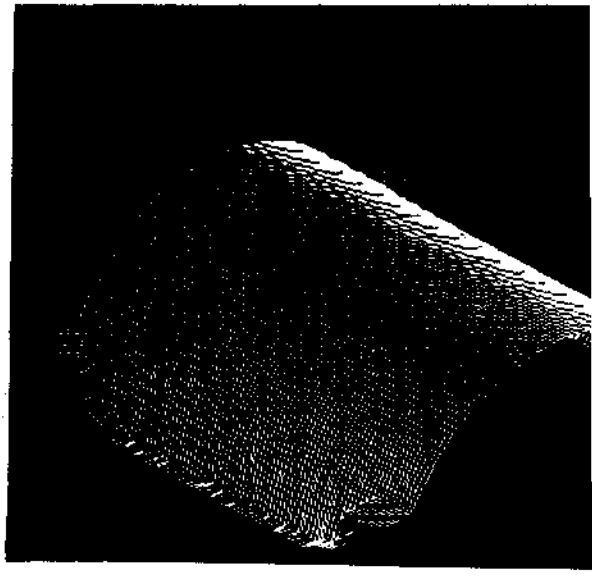


Figure 3.10: Result

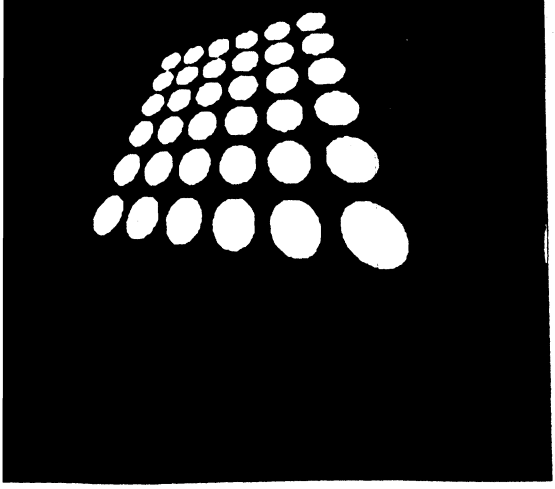


Figure 3.11: Input (plane)

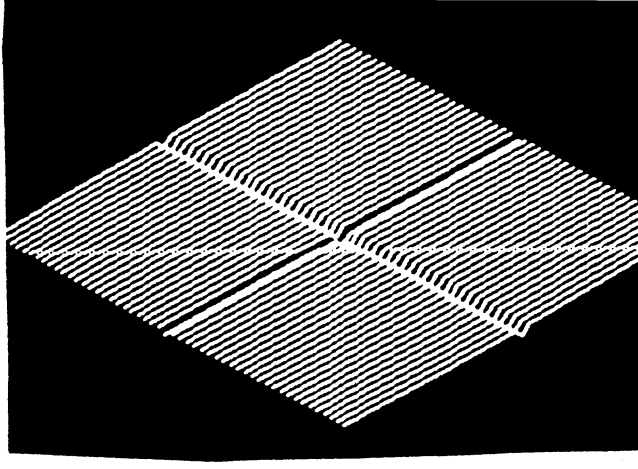


Figure 3.12: Result

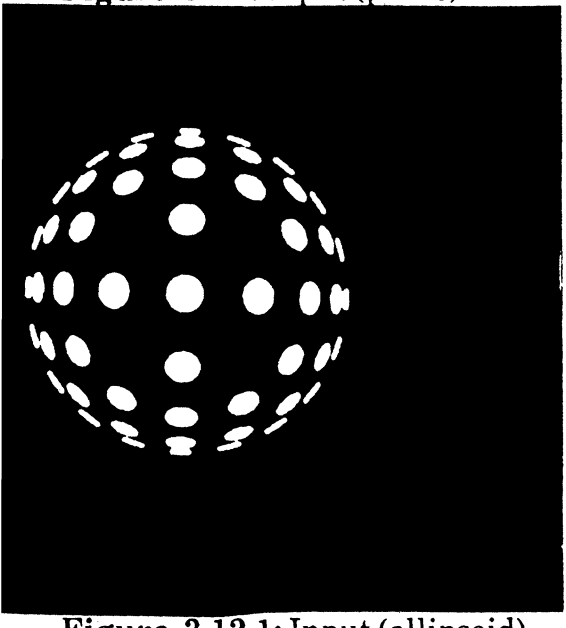


Figure 3.12.1: Input (ellipsoid)

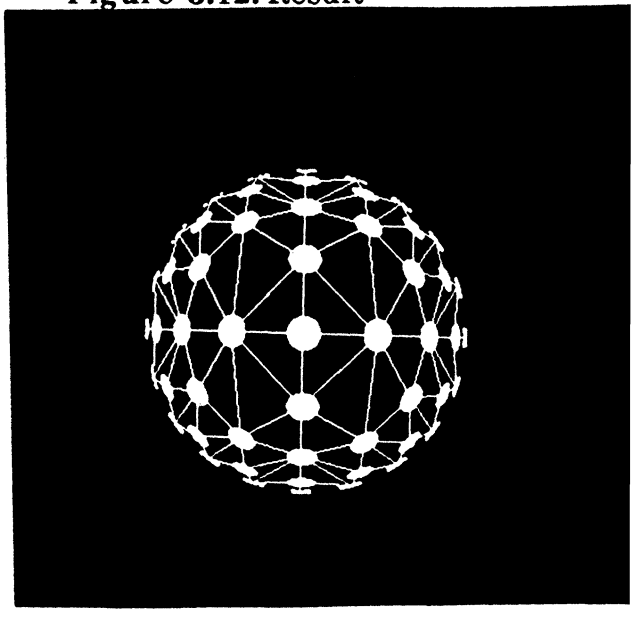


Figure 3.12.2: Triangulation

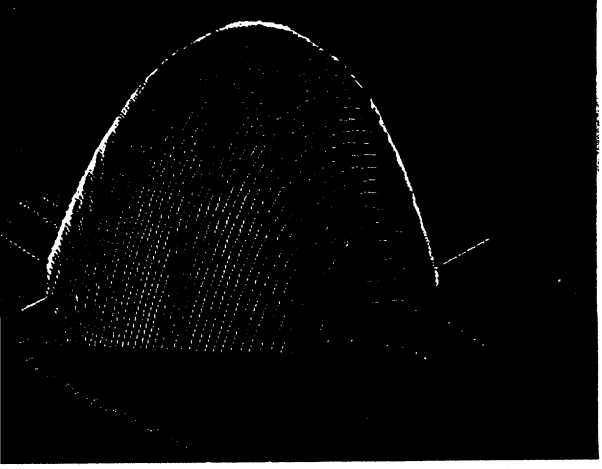


Figure 3.12.3: First phase

Figure 3.12.4: Result

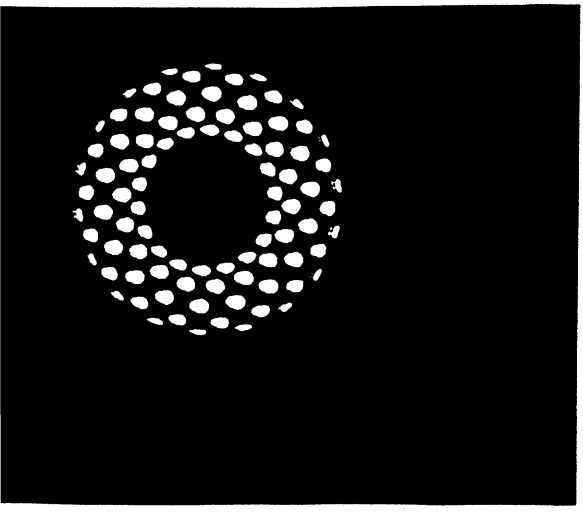


Figure 3.12.5: Input (donut)

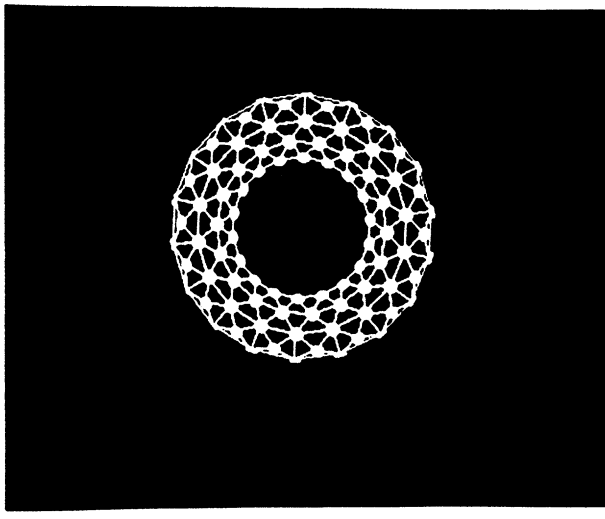


Figure 3.12.6: Triangulation

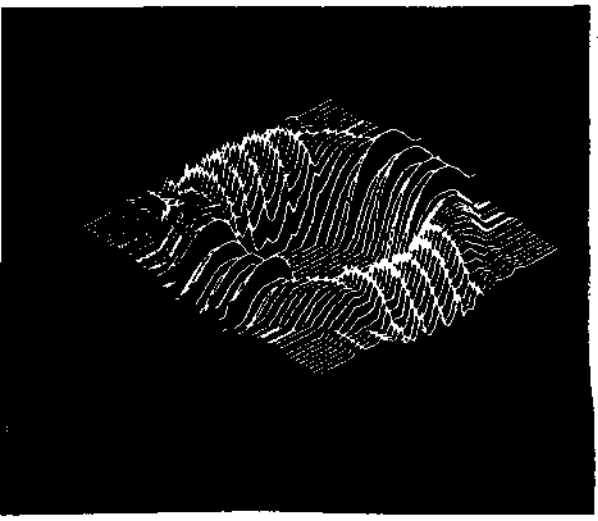


Figure 3-12.7:First phase

Figure 3-12,8: Result

3.2 Detection of surface orientation from natural texture-

It is very clear that natural texture provides an important source of information about the local orientation of visible surfaces. To recover three-dimensional structure, the distorting effects of the projection must be distinguished from properties of the texture on which the distortion acts. This requires that *assumptions must be made* about the texture. In this work we will study extensively the problem of shape from texture for the case of planes. Extension of our theory for curved surfaces will also be discussed. Several researchers have done work in this area, with the most important results presented by Gibson [Gibson, 1950], Witkin [Witkin, 1981], Stevens [Stevens, 1981], Bacjsy [Bacjsy et al, 1976], Rosinski [Rosinski, 1980) and Kanatani [Kanatani, 1984]. These researchers studied the problem under different assumptions about the texture and the imaging geometry. The next section analyzes the distortions imposed by the imaging geometry in an attempt to decide under what kind of projection we should study the problem of shape from texture, and the section following reviews and criticizes previous work.

3.2,1 Distortions imposed by the imaging geometry

An image is the projection of a three-dimensional world onto a plane. This process (projection), introduces various distortions to the objects in view. In general, the distortions can be considered as coming from the following effects: *the distance effect* (the objects in view appear larger when they are closer to the image plane), *the position effect*

(the distortion of a pattern depends also on the angle between the line of sight and the image plane, which depends on the image position of the pattern), and the *foreshortening effect* (the distortion of a pattern depends on the angle between the surface normal and the line of sight). It is clear that the orthographic projection model captures only the foreshortening effect and ignores the other two. Therefore, methods for shape from texture which use orthographic projection are valid only in a limited domain, where the other two effects can be ignored. On the other hand, the perspective projection model, which can be used as a camera model, captures all three effects, but the resulting algorithms are complicated and they involve the solution of nonlinear equations. Furthermore, the numerical errors introduced by the numerical approximation of several quantities (under perspective projection) reduce by a small amount the accuracy of any method. In this work we analyze the texture problem under both perspective projection and an approximation of the perspective projection that captures all the above three effects. It is relatively simple and gives accurate results. This approximation is called paraperspective projection and has already been described in the second chapter.

3.2.2 Previous work

Some serious work has been done in this area, and many of the published papers have reasonable results for the images that fit their assumptions. The first to approach the shape from texture problem was Gibson [1950]. Trying to develop a theory on how humans perceive surface orientation from texture, he suggested that texture consists of small elements, called texels. Of course, these small elements constitute the texture in a very irregular, non-canonical way. Gibson, realizing that he should make assumptions about the texture, proposed the following: The individual elements that constitute the texture (texels) are uniformly distributed on the world plane, in the sense that in a unit area on the world plane there is approximately the same number of texels; in other words, texture is uniformly distributed on the world plane. But when we look at it, *i.e.* take an image, then the texture density is not uniform, *i.e.* it has a gradient. So, Gibson proposed that humans perceive the orientation of naturally textured surfaces from this sameness (uniform density on the world plane) and difference (gradient of the texture in the image). Gibson, not having the necessary analytical tools, treated the case of perspective projection of a receding plane (ground plane). He assumed the plane to be covered with

elements of uniform density, and from that, the gradient of texture density in the image specifies surface orientation.

Continuing with the approach initiated by Gibson, Bajcsy and Lieberman [1976] tried a heuristic use of the two-dimensional Fourier power spectrum windows to detect texture gradient. Their work, despite its elegance, was of a very limited applicability. Their method works only for receding surfaces, and with the distance of the camera from the ground known. Furthermore, all the texture elements are assumed to have the same size, for their theory to be right. Because of the fact that the texture elements are not of the same size in the real world, their results are not accurate, as reported. After this, the Gibsonian approach was abandoned, basically due to the work of Witkin and Stevens.

Witkin [1981] presented a statistical approach without assuming spatial homogeneity. He assumed "directional isotropy", i.e. the assumption that the peripheral contours of the figures in the true texture have line segments that are uniformly distributed over all orientations. Based on an orthographic projection model, he derived the maximum likelihood estimators of the slant and tilt angles. Although the isotropic assumption is a general one, there are many natural scenes that do not agree with this assumption. In our formulation, Witkin's assumption can be used, but our experiments showed that it yields very low accuracy. The reason for that is that the directional isotropy assumption is very restrictive and seems to be present only in a small subset of natural images. The arguments of Witkin as to why he did not continue with the Gibsonian uniform density assumption are two: First, it had not been demonstrated up to that point that the uniform density assumption could be used as the basis of an algorithm to detect surface orientation in a general situation. We prove in the forthcoming sections that this is not the case. Witkin's second argument was that even if we had an algorithm that could recover surface orientation based on uniform density, this algorithm would need to know the texels, and it is not at all obvious how we can find the texels in an image. This is perhaps the strongest argument against the Gibsonian assumption of uniform density, but in later sections we will show how to overcome this problem and alter the uniform density assumption to a better, more general one that does not require our finding of the texels.

Stevens (Stevens,1980) studied the problem under perspective projection and found that texture density depends on both *scaling* (distance-position) and *foreshortening*

(surface shape). From this, he concluded that texture density is not a good measure for computing surface orientation, since it varies with both scaling and foreshortening. Stevens did not realize that despite the fact that scaling and foreshortening both affect the texture density, their effects could be separated and that the separated foreshortening effect could compute uniquely the surface orientation. Our approach performs the separation of the foreshortening and distance effects and does not make any assumptions about the shape and size of the texels. It assumes only that the texels are distributed in the world plane at uniform density. Practical difficulties (finding texels) obliged us to generalize the uniform density assumption to another form which seems to capture a very large subset of natural and man-made environments; the resulting algorithms do not require a strong segmentation (finding texels), but only a weak segmentation (finding edges).

Render [1980]) and Kanade [1979] explored the domain under orthographic projection. Render formalized the relationship between local surface orientation and two perpendicular axes of the same length. Kanade proposed using skewed symmetry to recover local surface orientation. The angle between a skewed symmetry direction and the opposite direction can be a constraint on surface orientation. Render [1980] and Ohta *et al.* [Ohta, Maenobu, Sakai, 1981] address the shape from texture problem under perspective projection. Render determines surface orientation from many parallel lines observed on a plane. Ohta *et al.* proposed using the area ratio of texture elements to recover surface orientation. Their method depends on the accuracy of measuring the areas of individual texels of the same shape. Measurement errors are amplified when the texels are very small. Furthermore, their method needs to find the individual texels, something very hard and seemingly impossible in natural images. Ikeuchi [1984] addresses the problem under spherical projection for general surfaces, but his crucial assumption is that the world texels must be regular (symmetrical) and known a priori, as it has already been noted in section 3.1. Ranatani [1984] uses the second Fourier harmonics of the number of intersections between texture and parallel scanning lines to find the surface orientation based on orthographic projection, by assuming that the texture is directionally isotropic, ie. his method is very similar to the one used by Witkin.

3.2.3 The model

The paraperspective projection model is very general, and it can slightly change everytime we change the auxiliary plane. This model has already been described in sections 2.6.3 and 3.1.2. Here we describe the inverse transformation, i.e. the transformation from the image plane to the world plane.

3.2.3.1 The inverse transformation under paraperspective projection

The transformation that was introduced in the two previous sections, was from the world to the image. In this section, we study the inverse transformation, *i.e.* the one from the image to the world, under the introduced paraperspective projection. There are two reasons for doing this. First, we will derive the same algorithm using two different methods and second, we will use the results of this section later, when we will address the problem in the case when we cannot identify the individual texels, but parts of their boundaries (edges).

Consider the function f that maps points in an area S_I of the image to their corresponding ones in the world plane under the inverse of the already introduced paraperspective projection. The function f does the following:

1) If the point $s=(A,B,-1)$ is the center of gravity of the image area S_I, then $f(s)$ is the intersection of the vector $(A,B,-1)$ and the world plane.

2) For any other point $p=(x,y,-1)$ in the area S_I, $f(p)=Q+t\,(A, B, -1)$, where W is the vector defined by the origin and the intersection of the direction $(x, y, -1)$ with the plane $z = -d$.

It is clear that the transformation f is the inverse of the imaging transformation. From (1) and (2), f can be written explicitly as:

$f(x,y,-1)=(dx+tA,dy+tB,-d-t)$ with

$$d=\frac{c}{1-Ap-Bq} \quad and \quad t=\frac{d(pX+qy-1)+c}{1-pA-qB}$$

In the rest of the chapter, whenever we use the symbol f, we will mean the inverse transformation introduced in this section. Finally, we should say that f is defined for a region S in the image, since it depends on the center of gravity of the area S. So, if the image is divided in n areas sj , s^, ..,s_n and the inverse transformation for each area is fj , $f2$,.., f_n then the inverse transformation for the whole image can be realized as the set $\{fi, f_2, .., f_n\}$.

3.2.3.2 The inverse transformation under perspective projection

Here we study the inverse transformation under perspective projection. Let us fix a coordinate system $OXYZ$ with the Z axis as the optical axis and the image plane perpendicular to the Z-axis (focal length = 1). If (x,y) is the coordinate system on the image plane (x axis parallel to X, y axis parallel to Y) with origin at the intersection of the Z-axis with the image plane, then a point (X,YJZ) in the world is projected on the image point (x,x),with:

$$x = \frac{X}{Z} \quad , \quad y = \frac{Y}{I}$$

Furthermore, let a plane $Z=pX+qY+c$ in the world, whose image is considered. The inverse imaging function, f_f is again the function that maps the image plane onto the world plane. So, if (x,y) is an image point, the 3-D world point on the plane $Z=pX+qY+c$ that has (x,y) as its image, is given by:

$$f(x,y) = \left(\frac{ex}{px-qy} \quad \frac{cy}{l-px-qy} \quad \frac{c}{l-px-qy} \right)$$

We see that in the previous case (paraperspective) the inverse transformation was defined for a small area. Here the inverse transformation is defined for the whole image plane by the same form. In the rest of this section we will develop the first fundamental form of [Lipschutz, 19691, because it will be needed later. The first fundamental form of f is the quadratic form: $Edx?+2Fdxdy+Gdy*$

with $E=f_x f_x, F=f_x f_y f G=fyfy$ where represents the dot product operation. After simple calculations we get:

$$E = \frac{c^2}{(1-px-qy)^4}\left[(1-qy)^2 + p^2y^2 + p^2\right]$$

$$F = \frac{c^2}{(l-px-qy)^4}\left[(1-qy)\,qx + (1-p^*)py + pq\right]$$

$$G = \frac{c^2}{(l-px-qy)^*}\left[4V + (1-px)^2 + q^2\right]$$

The above coefficients *E, F, G* are called first fundamental coefficients and are functions of *xy* (and so they vary from point to point). In the sequel we will examine the relation between image and world areas, as well as the relation between image and world lengths, for both perspective and paraperspective projection.

3.2.4 Relation between image and world areas

In order to study the relationship of the texture on the world plane and of the texture on the image plane, we must examine the relationship between areas in the world and in the image. The next two sections do that for both cases of paraperspective and perspective projection.

3.2.4.1 The case of paraperspective projection

It is known that the absolute value of the determinant of the matrix of a 2-D affine transformation is equal to the ratio of the areas before and after the transformation. In other words, if S_w is the area of a region on the world plane $-Z = pX + qY + c$, and S_7 is the area of its image under the introduced projection process, then:

$$\frac{S_I}{} = abs\left[\det\left(\frac{1}{d}\begin{bmatrix} \frac{-1+pA}{\sqrt{(1+p^2}} & \frac{pB}{\sqrt{(1+p^2)}} \\ \frac{q(p+A)}{V(l+p^2)(1+p^2+q^2)} & \frac{qB-p^2-l}{V(l+p^2)(1+p^2+q^2)} \end{bmatrix}\right)\right)$$

or

$$\frac{S_I}{S_w} = abs\left(\frac{1}{d^2}\frac{1-Ap-Bp}{V(l+p^2+q^2)}\right)$$

But it is clear from the intersection of the central projection ray with the world plane that

$$d = \frac{c}{1 - pA - qB}$$

The above equations give:

$$\frac{S_I}{S_W} = abs\left(\frac{1}{d^3} \quad \frac{c}{\sqrt{(1+p^2+q^2)}} \right)$$

Since the parameters p, q, c are constant, the above equation tells us that the ratio of the areas before and after the transformation is inversely proportional to the cubic of the distance of the mass center of the world region from the origin. Also, it says that an area S_I in the image is due to the projection of an area (S_I *abs* (d^3) $\sqrt{(1+p^2 + q^2)}$ /c) in the world; in other words, if we consider an area S_I in the image, then in order to find the area in the world plane whose projection is S_I, we must multiply the area S_I with the factor

$$R_I = abs\left(\frac{d^3\sqrt{(1+p^2+q^2)}}{c} \right) = abs\left(\frac{c^2\sqrt{(1+p^2+q^2)}}{(1-Ap-Bq)^3} \right)$$

where *(A,B)* is the center of gravity of the image area S_I.

The ratio S_I / S_W can also be computed in the following elegant way, using the inverse transformation *f* that was developed in section 3.2.3.4. The function *f* maps the region S_I to a region S_W on the world plane (see Fig.3.14a).

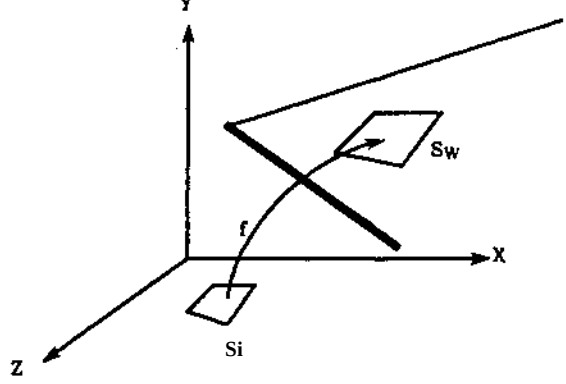


Figure 3.14a:The inverse transformation

8/78* and 8/78y represent the speed along the *x* curve and *y* curve respectively. A rectangle *Si* in the image having area Ax Ay is mapped to a parallelogram in the world plane which is determined by tide vectors 8/78x Ax and8/78yAy. Therefore the area ofthis parallelogram is the magnitude

$$\left\| \frac{\delta f}{\delta x} \Delta x \times \frac{8f}{\quad} \Delta y \right\| = \left\| \frac{8f}{\quad} \times \frac{\delta f}{\delta y} \right\| \Delta x \, \Delta y$$

The area of S_w can be computed by the double integral:

$$\int \int_{S_I} \left\| \frac{\delta f}{dx} \times \frac{\delta f}{\delta y} \right\| dx \, dy$$

But

$$\frac{\delta f}{dx} = \frac{d}{l\text{-}Ap\text{-}Bq} \; (l\text{-}qB, pB, -p)$$

and

$$\frac{\delta f}{\delta y} = \frac{d}{l\text{-}Ap\text{-}Bq} \; (qA, l\text{-}pA, -q)$$

So,

$$\frac{\delta f}{\delta x} \times \frac{\delta f}{\delta y} = \frac{d^2}{1 - Ap - Bq} \left(p, q, 1 \right)$$

and

$$\left\| \frac{8f}{8x} \text{X} \frac{}{8y} \right\| = a^{,2} \frac{\sqrt{(1 + p^2 + q^2)}}{abs(l\text{-}Ap\text{-}Bq)} = c^2 \frac{Vq + p^2 + q^2)}{abs(1 - Ap - Bq)^3}$$

and so, the equations that relate image to world area can be derived again.

3.2.4.2 The case of the perspective projection

Here we address the same problem as in the previous section but for the case of the perspective projection. We know that if we have an area S_2 in the image plane, then the image of this area through the inverse function /can be computed directly with the aid of the *first fundamental coefficients* [Lipschutz, 1969]. In other words, if we have an area S_7 on the image plane, then the area S_w in the world plane $Z=pX+qY+c$ whose projection is S_r is given by:

$$S_w = \int_{S_I} \int V(EG-F^2)dxdy$$

with E, F,G the first fundamental coefficients. If we substitute E, F, G with their values (section 3.2.3.5) we get:

$$S_W = \int_{iS_!} \int \frac{f^2}{qy^3} \sqrt{(1+p^2+q^2)}\, dx\, dy$$

It is obvious that the relation between $S_;$ and S_w becomes the same under paraperspective and perspective, when the area $S_;$ becomes very small. Finally, the above equation cannot be further simplified, since we do not have a specific area S_r

3.2.5 Relation between image and world lengths

Because of the fact that we will need the relation between edges (line segments) on the world plane and on the image plane, we need to develop them here. The next two sections examine this problem for both the cases of paraperspective and perspective projection.

3.2.5.1 The case of paraperspective projection

In this section we exploit the relation between the length of a small line segment in the world and its image under the introduced model of the paraperspective-projection. We repeat here that the inverse transformation /that was introduced in section 3.2.3.4 maps image points to world points. The speed of *f* in the direction o> = (co$9, *sinB)* is the directional derivative *of fin* the direction co. In particular,

$$f_Q'(x,y) = Df(x,y)o>^1$$

or

$$f_\theta'(x,y) = \begin{bmatrix} D_1f_1 & D_2f_1 \\ D_1f_2 & D_2f_2 \\ D_1f_3 & D_2f_3 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$

or

$$f_\theta'(x,y) = \frac{c}{(1-Ap-Bq)2} \begin{bmatrix} 1-qB & qA \\ pB & 1-pA \\ -p & -q \end{bmatrix} \cdot \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}$$

So, a line segment with length l in the direction θ in the image, is due to the projection of a line segment L in the world plane with length

$$L = \|f_\theta'\| . l .$$

But

$$\|f_q'\| = \frac{c.\sqrt{((1-qB)^2+(pB3)^2+p^2)\cos^2\theta+((1-pA)^2+(qA)^2+q^2)\sin^2\theta+2((1-qB)qA+(1-pA)pB+}}{(1-Ap-Bq)^2}$$

In other words, if we have a line segment l in the image area S_I in the direction ($\cos\theta$, $\sin\theta$), then in order to find the length of the line segment in the world plane that has image l, we have to multiply l with the factor:

$$L_I = \frac{c.\sqrt{((1-qB)^2+(pB)^2+p^2)\cos^2\theta+((1-pa)^2+(qa)^2+q^2)\sin^2\theta+2((1-qB)qA+(1-pA)pB+pq)}}{(1-Ap-Bq)^2}$$

At this point we should say that the same result could be obtained using simple analytic geometry, but the analysis was done in this way for reasons of elegance.

3.2.5.2 The case of perspective

In this section we address the same problem as in the previous section, but for the case of the perspective projection. Again, the desired relation is given directly from the *first fundamental coefficients*. Indeed, if we have a line segment L on the image plane, then the

length DL of the line segment on the world plane whose image is the line segment L, is given by the integral of the first fundamental form, i.e.

$$DL = \int_L \sqrt{(E dx^2 + 2F\, dx\, dy + G dy^2)} \quad on\ the\ image\ plane$$

where E, F, G are the first fundamental coefficients. Again we can substitute the values of E, F, G but we cannot get rid of the integral if we don't assume a specific line segment L. We will now utilize the findings of sections 3.2.4 and 3.2.5 to devise efficient and robust algorithms for the computation of the orientation of the textured plane in view.

3.2.6 Exploiting the uniform density assumption

In this section, we use the uniform density assumption to develop constraints that will enable us to recover the gradient (p,q) of the plane in view from its image. We first address the problem for the case where the texels can be located and counted (*strong segmentation-weak result*) and then for the case where the edges (texel boundaries) can be located (*weak segmentation-strong result*).

3.2.6.1. Determining shape provided that the texels can be found

In this section we study how we can recover the shape of the textured plane in view, provided that the texels can be located. Up to this point there is no known algorithm that can successfully detect texels from a natural image. There is, of course, current research effort in this direction with promising results. The following two sections are based on the assumption that the texels can be detected, even though we don't know of any algorithm that does so. The value of the forthcoming sections is theoretical, and is basically an answer to the objection raised by those who follow Witkin's approach.

3.2.6.1.1 The case of perspective projection

The uniform density assumption states that if S and S' are any two regions in the world plane, and they contain k and k' texels respectively, then

$$\frac{k}{area(S)} \sim \frac{k'}{area(S')}$$

Consider any two regions s_1 and s_2 in the image of the textured plane with areas S_1 and S_2 respectively. These regions are the projections of some regions in the world plane with areas S_{W1} and S_{W2}, where

$$S_{W1} = \int_{s} \int \frac{c^2}{\quad} \sqrt{(1 + p^2 + q^2)}\, dx\, dy$$

and

$$S_m = \int_{s_2} \int \frac{}{(1 - px - qy)^3} \sqrt{(1+p^2+q^2)}\, dx\, dy \ .$$

So, let k_1 and k_2 be the number of texels in the image regions s_1 and s_2 respectively. Then, the regions in the world plane whose projections are the image regions s_1 and s_2 contain k_1 and k_2 texels respectively. Thus, the uniform density assumption, is written as:

$$\frac{k_1}{\int \int \frac{c^2}{(1-px-qy)^3} \sqrt{(1+p^2+q^2)}\, dx\, dy} = \frac{k_2}{\int \int \frac{c^2}{(1-px-qy)^3} \sqrt{(1+p^2+q^2)}\, dx\, dy}$$

The above equation is the basis for the recovery of the gradient, provided that the texels can be located and counted. This equation, clearly is an equation in the unknowns p,q, but it is nonlinear even for the simplest choiceof the areas s_1 and s_2 (squares). Because of the nonlinearity of this equation, we do not attempt a closed form solution, something that probably is not impossible under the employment of over simplifying assumptions. Instead , we use the following simple method. We divide the image into n equal areas (squares) (see Fig. 3.15), $s_2, s_2, ..., s_n$, and suppose that each of th$se areas contains $k_1, k_2, ..., k_n$ texels respectively. What we require is that the density of the texels in the world plane is about the same; in other words, we want to find the parameters p,q so that the quantities:

$$\frac{k_i}{\int_{s_i}\int \frac{c^2}{(1-px-qy)^3}\sqrt{(1+p^2+q^2)}dx\,dy}\ ,\ i=1,\ldots,n$$

are about the same, or the quantities

$$d_i=\frac{k_i}{\int_{s_i}\int \frac{1}{(1-px-qy)^3}\sqrt{(1+p^2+q^2)}\,dx\,dy}\ ,i=1,\ldots,n$$

are about the same. Of course "about the same" has a statistical meaning; in particular, if the density (texel density on the world plane) was the same everywhere, then the quantities d_i, $i=1,..,n$ should be equal. But it is unrealistic to expect that the density will be the same everywhere on a textured surface. What is to be expected is that the density will be "about" the same everywhere. In other words, we want to find the gradient *(p,q)*, that minimizes the variance of the sample $\{d_1,..,d_n\}$. This can be done easily by trying all the different values for the orientation and choosing the one that minimizes the variance of the sample $d_1,..,d_n$. Of course we change formulation for the gradient, and instead of the gradient space (p,q), we use the (equivalent) Gaussian sphere formalism (azimuth, elevation) in a discretized fashion (180 different values for the elevation, 180 different values for the elevation = 180*180 different combinations). We do this in a hierarchical manner, *i.e.* after all the different orientations have been tried (180*180 values) and the sample with the smallest variance has been selected, we have an answer for the orientation correct up to .9 degrees for both azimuth and elevation. If this answer is, for example : *azimuth*=*a* degrees, *elevation*=*e* degrees, we continue the same process but in the interval (a-1,a+1)X(e-1,e+1) until we obtain the desired accuracy. Finally, the integral in the computation of the densities d_i, i=1 ,..,n, can be easily computed. In particular, if we consider an image

area s that is defined by the square *((m,r),(n,r),(n,s),(m,s))* (see fig. 3.15), then:

$$\int\int_{\backslash . \backslash} \frac{y/(l+p^2+q^2)}{\{l-px-qyf} \, dxdy = \frac{1}{2} \frac{\sqrt{(1+p^2+q^2)}\ (n-m)\ (s-r)\ (2-pn-pm-qs-qr)}{(1-pn-qs)(1-pm-qs)(1-pn-qr)(q-pm-qr)}$$

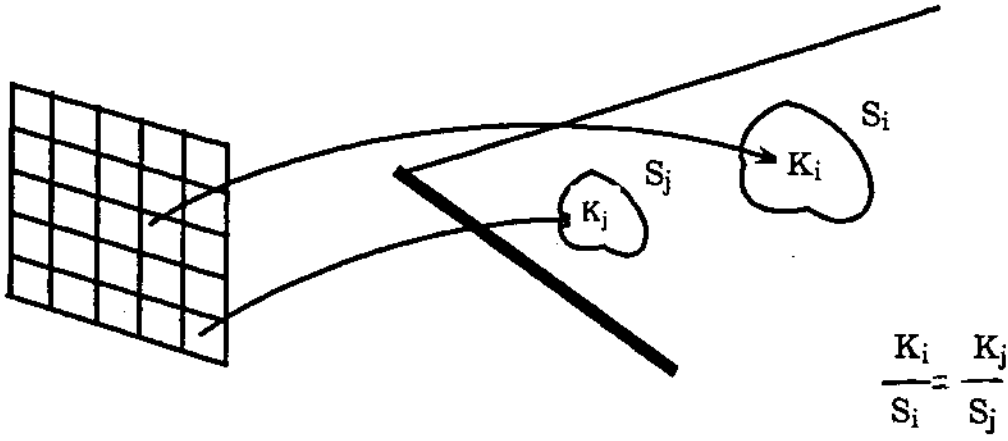


Figure 3.15: Backprojection

So, the denominators in the quantities d_i can be precomputed for all orientations and stored in a big look-up table. This table is three-dimensional. The first dimension represents the position in the image (area square in the image), and the other two orientation. Each entry of the table contains the value of the above integral (denominator of d_i) for the particular area and the particular orientation. So, the algorithm for the computation of surface orientation given a textured image (where the texels have been found and counted) is very fast, since it computes the different samples d_i, i=1,..,n (180*180 of them) by table look-up and by counting the texels (ki ,i= l,..,n) in every area.

The next section examines the same problem, but under the paraperspective projection. It turns out that in this case a closed form solution can be found (with a very simple algorithm).

3.2.6.1.2 The case of paraperspective projection

Here the same problem is treated under the paraperspective projection assumption. The uniform density assumption states that if *S* and S'are any two regions in the world plane, and they contain *K* and *K*'texels respectively, then

$$\frac{K}{areaiS)} = \frac{K^*}{area(S')}.$$

Consider any two regions s_t and s_2 in the image of the textured plane with areas S_t and S_2 respectively. These regions are the projections of some regions in the world plane with areas S_1R_1 and S_2R_2 respectively, where

$$R_1 = abs\left(\frac{c^2\sqrt{(1+p^2+q^2)}}{(1-A_1p-B_1q)^3}\right)$$

and

$$R_2 = abs\left(\frac{c^2\sqrt{(1+p^2+q^2)}}{(1-A_2p-B_2q)^3}\right)$$

with (Aj , J?2) and (A_2 , B_2) the centers of the gravity of the image regions s_{j9} s_2 respectively.

Let K_l and K_2 be the number of texels in the image regions s_t and s_2 respectively. By our assumption, the regions in the world plane whose projections are $\$_t$ and s_2 contain K_t and K_2 texels respectively. That is.

$$\frac{K_1}{S_1R_1} = \frac{K_2}{S_2R_2}$$

or

$$abs\left(\frac{K}{S_1c^2V(1+P^2+g^2)^{\wedge}}\right) = abs\left(\frac{A_2p-B_2q)^3}{S_2c^2\sqrt{(1+p^2+q^2)}}\right)$$

Since $1 - px - qy = 0$ is the vanishing line of the projected plane, (A_1, B_1) and (A_2, B_2) lie on the same side of this line. That is, $(1 - A_1 p - B_1 q)$ and $(1 - A_2 p - B_2 q)$ have the same sign. Therefore, we can drop the absolute functions in the above equation. We get

$$\frac{1 - A_1 p - B_1 q}{1 - A_2 p - A_2 q} = \left(\frac{K_2}{K_1} \frac{S_1}{S_2} \right)^{\frac{1}{3}}$$

or

$$\left[\left(\frac{K_2}{K_1} \frac{S_1}{S_2} \right)^{\frac{1}{3}} A_2 - A_1 \right] p + \left[\left(\frac{K_2}{K_1} \frac{S_1}{S_2} \right)^{\frac{1}{3}} B_2 - B_1 \right] q = \left(\frac{K_2}{K_1} \frac{S_1}{S_2} \right)^{\frac{1}{3}} - 1$$

The above equation represents a line in the p-q space. So, considering any two regions in the image, we constrain (p,q) to lie on a line in the gradient space (see Figure 3.16).

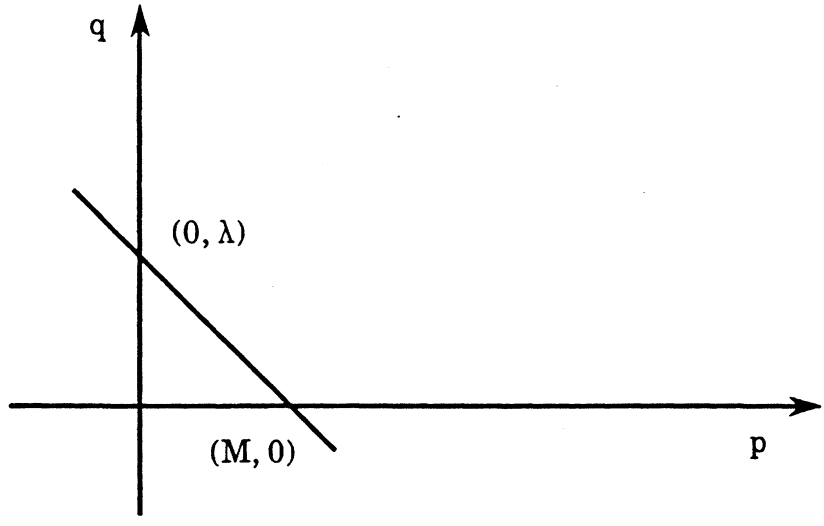


Figure 3.16: The constraint in gradient space

In Figure 3.16, the uniform density assumption, taken in two image regions s_1, s_2 with areas S_1 and S_2, and K_1 and K_2 texels respectively, constrains the gradient of the plane to lie on the above drawn line, where

$$\lambda = \frac{\left[\left(\frac{K_2}{K_1}\frac{S_1}{S_2}\right)^{\frac{1}{3}} - 1\right]}{\left[\left(\frac{K_2}{K_1}\frac{S_1}{S_2}\right)^{\frac{1}{3}} B_2 - B_1\right]} \quad \textit{and} \quad M = \frac{\left[\left(\frac{K_2}{K_1}\frac{S_1}{S_2}\right)^{\frac{1}{3}} - 1\right]}{\left[\left(\frac{K_2}{K_1}\frac{S_1}{S_2}\right)^{\frac{1}{3}} A_2 - A_1\right]}$$

with (A_1, B_1) and (A_2, B_2) the centers of gravity of the regions S_1 and S_2 respectively.

It is now clear that taking two pairs of image regions we can solve for p and q. But because of the errors introduced by the sampling process (image digitization and density fluctuations of the regions), we may get inaccurate results. To overcome this problem, we employ the least-square-fit mechanism. We consider many pairs of image regions, each one of them gives us one line in the gradient space. The desired solution is estimated by the point whose sum of distances from all the lines is minimum (see figure 3.17). If these sampling errors are normally distributed, this estimator gives the best estimation

The desired solution (as seen in the above figure) is estimated by the point whose sum of distances from all the lines is minimum.

3.2.6.2 Determining shape provided that the edges can be located

In the previous sections we developed a method to recover the orientation of a textured plane from its image, based on the assumption of uniform density. By uniform density, we meant that the number of texels per unit area of the world plane is about the same. Application of this method in natural images did not seem to work very well because no good methods have been developed up to now that can identify texels in an image, and our algorithm depends critically on the number of texels in an unit area, as we have already emphasized. Perhaps the most serious objection against Gibson's assumption [Witkin, 1981], is the fact that it has not been demonstrated up to now that

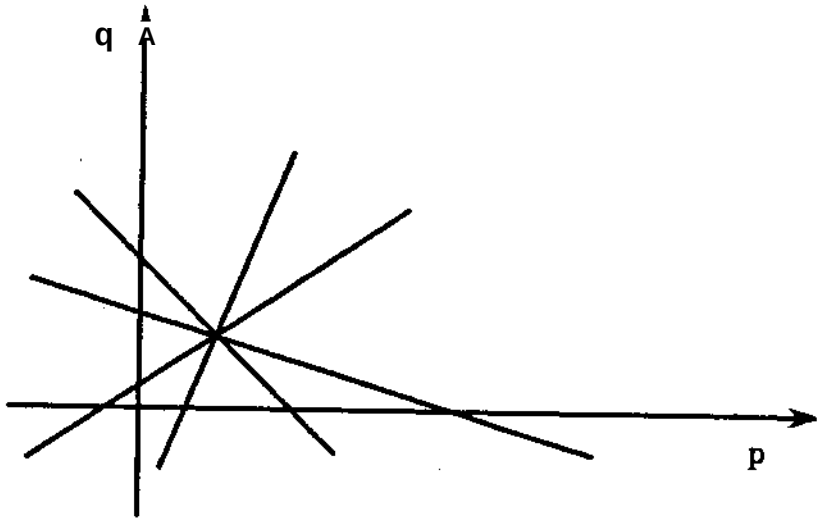


Figure 3.17: The solution as constraint intersection

texels can be reasonably found in a natural image. We believe that indeed it is very hard to find texels in a natural image, and our experiments to date indicate that this is indeed the case. On the other hand, recent literature [Marr, 1979a, b; Bandopadhyay, 1984; Canny, 1984; Nalwa, 1985; Sher, 1986] provides many robust methods for the computation and identification of the boundaries of the texels (edges) everywhere in an image with texture. Therefore, we slightly modify our uniform density assumption to a criterion that is sensitive to projective distortion and is computable on natural images.

If indeed Gibson's assumption is true for a large subset of natural images, then given that the size of the texels will also be uniformly distributed, it follows that the sum of the legths of the edges will also be uniformly distributed. We now define *density* in the world plane as the *total length of the texel boundaries per unit area,* and our uniform density assumption states that this new density is the same everywhere in the world plane. This new assumption is not far from the previous one; it seems to be true for a large subset of natural images. Of course, it cannot be proved that such an assumption is the appropriate one to be used for the recovery of shape from texture. An empirical analysis is needed for such a thing. We have found experimentally that this modified uniform density assumption (sum of the lengths of the contours per unit area is about the same

everywhere) is true for many natural and man-made textured planes (in particular, it has been found true for 50 different textured planes - grass fields, gravel paths, leaves on walls, sea waves, brick walls, carpets, cloth designs, aerial views of towns and parking lots, books on shelves, text, textured floors, ceilings, and many other cases). We now utilize this new assumption to devise algorithms for the recovery of surface orientation from texture. Again, the analysis is done for the cases of paraperspective and perspective projection.

3,2.6.2.1 The case of perspective projection

In this section we develop an algorithm for the recovery of surface shape using the assumption introduced in the previous section, under perspective projection.

Let the image be divided into small regions s; , .., s^ (squares, as in fig. 3.15). Suppose that we find in the area s_i the line segments L_a , $L\{_2$,.., L^$_i$., for i= 1,.., n . Then, the new uniform density assumption states that the quantities:

$$d_i = \frac{\sum_{i_1}^{J} \int_{L_j} \sqrt{(Eds^{\%}+2Fdxdy+Gdy^2)}}{\int_{s_i} \int \sqrt{EG-F^2}dxdy},$$

with *E, F, G* the first fundamental coefficients, should be about the same, for *i = l, .., n.* Again, in the expressions d_i we can get rid of the constant c and we follow the same procedure as in the previous section , i.e. we choose this orientation that minimizes the variance of the sample $d_{lf}..,d_n$.

The next section studies the same problem, but under the paraperspective projection assumption.

3.2.6.2.2 The case of paraperspective projection

Let the image be divided into small regions $\$_t$, $\$_2$,.., s^. For each region s$_i$., there is an imaging function *f.* (that depends on the center of mass of *Si)* in the way that was described in section 3.2.3.5. Let also *Si* and *Ri* be the area and area change ratio of the region s; respectively. So, the world area that has s; as an image is Sj. *Ri.*

In a region s; with center of mass *(AJJ),di* line segment

(lco$Q_f I sinQ) with length / is due to the projection of a line segment in the world plane withlength/-ft) = {. II/fell.

Let the total length of line segments in direction 9 in the region S; be *Ifi* . Then, the new uniform density assumption states that:

$$\frac{\int_{\theta} l_i^{\theta} \, \| f_{i\theta} \| \, d\theta}{S_i, R_i} = \text{constant for all regions } L$$

Since the area change ratio and the length change ratio for each direction are fixed within every region for each quantized orientation, tables of those ratios can be precomputed. The edges in the image are broken into line segments and the values :

$$c_i = \frac{\sum_{\theta} l_i^{\theta} \, \| f_{i\theta} \|}{S.R.}$$

for all the regions i, can be computed simply by a table look-up method for each orientation in the solution space (with a replacement of the unbounded p-q coordinates with the bounded azimuth-elevation of the Gaussian sphere formalism). The solution for the orientation of the world plane can be estimated by the orientation which minimizes the variance of the sample $\{c_t, .., cJ$, in the same exactly way as in the previous sections.

3.2.7 A comparison between perspective and paraperspective

In this section we compare the algorithms developed for shape from texture, under the perspective and paraperspective projections. Obviously, the paraperspective projection is an approximation of the perspective, and it is an area-to-area projection. When the area under consideration becomes very small, then the paraperspective projection becomes equivalent to perspective (as it can be seen from sections 3.2.4.1 and 3.2.4.2). But if we keep the areas very small, then the algorithms in sections 3.2.6.1.2 and 3.2.6.2.2 do not perform well, since not enough information is contained in this small area; from the other hand, if we make the areas large, again the algorithms do not give satisfactory results since the error introduced by the paraperspective projection is high .

We observed that if the areas are kept in between 25*25 and 50*50 pixels, then the results were very satisfactory.

Comparing the performance of the algorithms in 3.2.6.1. (given that the texels have been located) we observed that the algorithm based on the perspective projection performed a little better than the one based on the paraperspective projection. The algorithms in 3.6.2.2. (based on edges--partial texel boundaries) do not have any considerable difference in their results. This fact should not be surprising, since both the algorithms basically minimize a function, and despite the fact that these functions are not the same for the case of perspective and paraperspective, our experiments showed that their minimum seems to be the same. Actually, we can say that the algorithm based on the paraperspective projection performed better than the one based on perspective ; we think that this is due to the numerical errors introduced by the numerical approximation of the integrals. The final section describes relevant experiments. The next section describes an error analysis of the paraperspective projection.

3.2.8 Error analysis

Some of the methods developed in this paper that use the paraperspective projection depend critically on the promise that the introduced two-step projection process is a very good approximation of the perspective projection. In this section, we present a theoretical analysis of the error introduced under the paraperspective projection and we show that perspective projection and our two step projection process are indeed very close to each other. A more thorough error analysis can be found in [Aloimonos et al, 1986]. An error analysis of the two step process that is used in this paper is complicated, since the error depends on *how big is the area that is considered*, on *the orientation of the world plane* and on *the depth*. But before we proceed with our analysis, we challenge the reader to distinguish between the perspective projection and the paraperspective projection from the figures below (Fig. 3.18). Figures 3.18a and 3.18b are the images of a textured plane. One of them is produced using perspective projection and the other one using paraperspective .

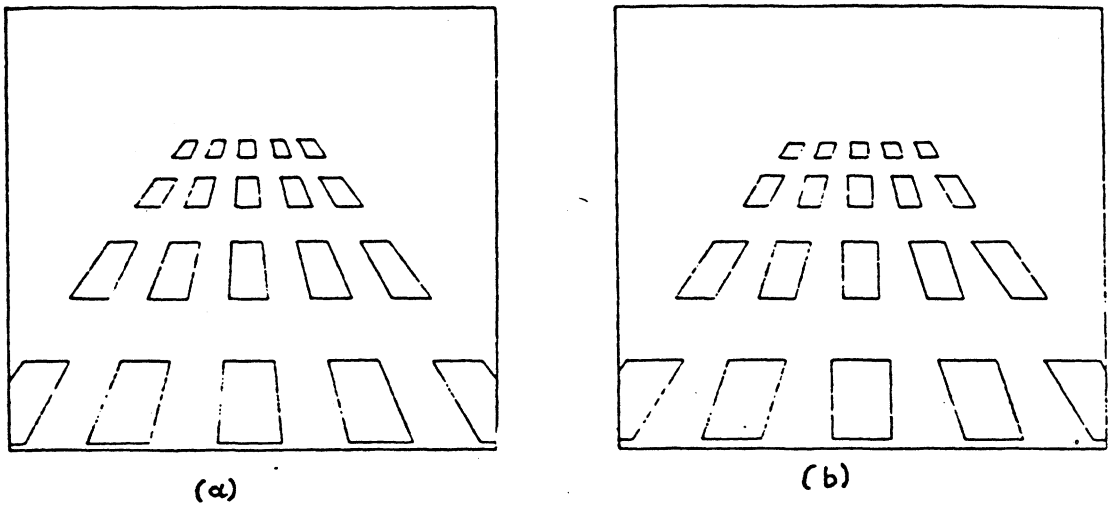


Figure 3.18: (a) the image of a textured plane using perspective projection and figure (b) the image of the same plane using the paraperspective projection.

The way we approach the error analysis is the following: We compute the image of a point under perspective projection and then we compute the image of the same point under our two step projection (paraperspective). The distance of these two images is the the *error*, which is a function of many parameters and we study its behaviour.

We use the terminology introduced in the previous sections. Consider a region S_w in the world plane and let G be its center of gravity. Let the perspective image of G be the point $P = (A, B, -1)$ on the image plane. Note that the image of G under our two step process is also the point P. It is easy to see now that $G = d.(A,B,-1)$. Consider now a point

M on the world plane, such that

$$\overline{OM} - \overline{OP} = (\Delta x, \Delta y, \Delta z)$$

This means that $M = d\,.\,(A, B, -1) + (\Delta x, \Delta y, \Delta z)$. But we can easily prove, since $z = px + qY + c$, that $\Delta z = -p\,\Delta x - q\,\Delta y$. So, we conclude that:

$$M = (d\,.\,A + \Delta x, d\,.\,B + \Delta y, -d - p\,\Delta x - q\,\Delta y)\ .$$

We are now going to compute the image of the point M under perspective projection and under our two step projection (paraperspective) process. It is clear that the point M involves the *orientation of the plane*, the *distance from the center of gravity to M* and *the distance d*. Let the images of the point M be M_P and M_O under perspective and paraperspective respectively. Then, we have:

$$M_P = \left(\frac{d\,.\,A + \Delta x}{d + p\Delta x + q\Delta y}\ ,\ \frac{d\,.\,B + \Delta y}{d + p\Delta x + q\Delta y} \right)$$

and

$$M_O = \left(\frac{d\,.\,A + \Delta x + (p\Delta x + q\Delta y)\,.\,A}{d}\ ,\ \frac{d\,.\,B + \Delta y + (pDx + q\Delta y\,.\,B}{d} \right)$$

From these equations, we conclude that the difference of the two projections, i.e. the length of the vector $M_O\,M_P$, or in other words the introduced error, is:

$$Error = \| \overline{M_O M_P} \| = \frac{z1\,.\,\sqrt{(2dA + \Delta x + z1A)^2 + (2dB + \Delta y + z1B)^2}}{d(d + z1)}$$

where $z_1 = p\Delta x + q\Delta y$. It is clear from the above formula that the error depends on many parameters. Using this formula the calculated error was very small. The following figures show the dependence of the error on some of the parameters when the rest of the parameters were fixed. So, it is clear from this analysis that indeed the perspective projection and our approximation are very close.

Figure 3.19 below shows the dependence of the error on the depth d. The slant and tilt of the world plane were 54.5 and 45 degrees respectively. The area under consideration in the image had center of gravity the point (5,5). The difference of the x and y coordinates of the world point, whose projections are considered, from the center of gravity of the world area, were $\Delta x = 2$ and $\Delta y = 2$. Figure 3.20 is the same as Figure 3.19 with the difference that $\Delta x = 2$- and $\Delta y = 20$.

Figure 3.21 shows the dependence of the error on how large is the area under consideration. All the quantities were the same as before, with the addition that the depth $d=300$ and the exception that the distance Δx was the independent variable. Finally Figure 3.22 shows the dependence of the error on the orientation. The fixed values were $(A,B)=(5,5)$, $d=300$, $(\Delta x, \Delta y) = (20, 20)$ and the quantity $q=10$. The independent variable was p.

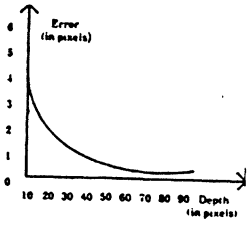


Figure 3.19

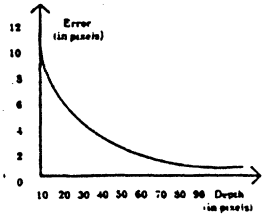


Figure 3.20

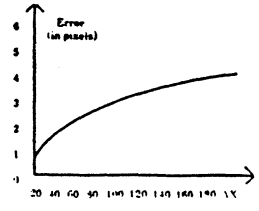


Figure 3.21

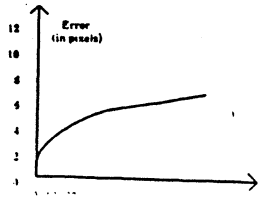


Figure 3.22

We have experimented a great deal in order to discover the relation between paraperspective and perspective projection, and in the rest of this section we will describe some more experiments that describe the error of the paraperspective projection (with respect to perspective). In the previous figures we showed the error in the projection of a point. But, for the shape from texture problem, we are primarily interested on the error in the ratio of areas. We chose two triangles on a world plane, and we computed the ratio of their images, under paraperspective and perspective projection. From this, the error of the paraperspective projection was computed for this case. Figure 3.22.1 shows the percent error as the slant of the world plane varies. Figure 3.22.2 shows the error as the tilt of theworld plane varies. Figure 3.22.3 shows the error as the depth of the world plane varies (parameter c).

3.2.9 Implementation and experiments

We have tested our algorithms on many synthetic and natural images. The results were very satisfactory. In this section, we use slant($\tan^{-1}$ $\sqrt{(p^2+q^2)}$) and tilt ($\tan^{-1}(q/p)$) to represent the orientation of a plane, since they are more intuitive.

3.2.9.1 Synthetic images

In our experiments, we used all the algorithms of 3.2.6.1 and 3.2.6.2, *i.e.* for the algorithms in section 3.2.6.1. (3.2.6.1.1-perspective and 3.2.6.1.2-paraperspective) we used the number of texels per unit area as the density and for the algorithm in the second part of 3.2.6.2. we used the total length of the boundaries per unit area as the density, to get a solution. Figure 3.23 is the image of a plane covered with random dots parallel to the image plane. Figure 3.24 is the image of the previous plane after rotated and translated, with tilt = 135 and slant = 30 degrees. Algorithm 3.2.6.1.2 (paraperspective) recovered tilt = 134.4 and slant = 29.75 degrees. Algorithm 3.2.6.1.1 (perspective) recovered tilt = 134.6 and slant = 29.70 degrees. Figure 3.25 presents the image of a plane parallel to the image plane covered with random line segments. Figure 3.26 presents the image of this plane rotated with tilt = 135 and slant = 30 degrees. Algorithm 3.2.6.1.2 (paraperspective) recovered tilt = 133.77 and slant = 30.40. Algorithm 3.2.6.1.1 (perspective) recovered tilt = 134.1 and slant = 29.80 degrees. Figure 3.27 presents the

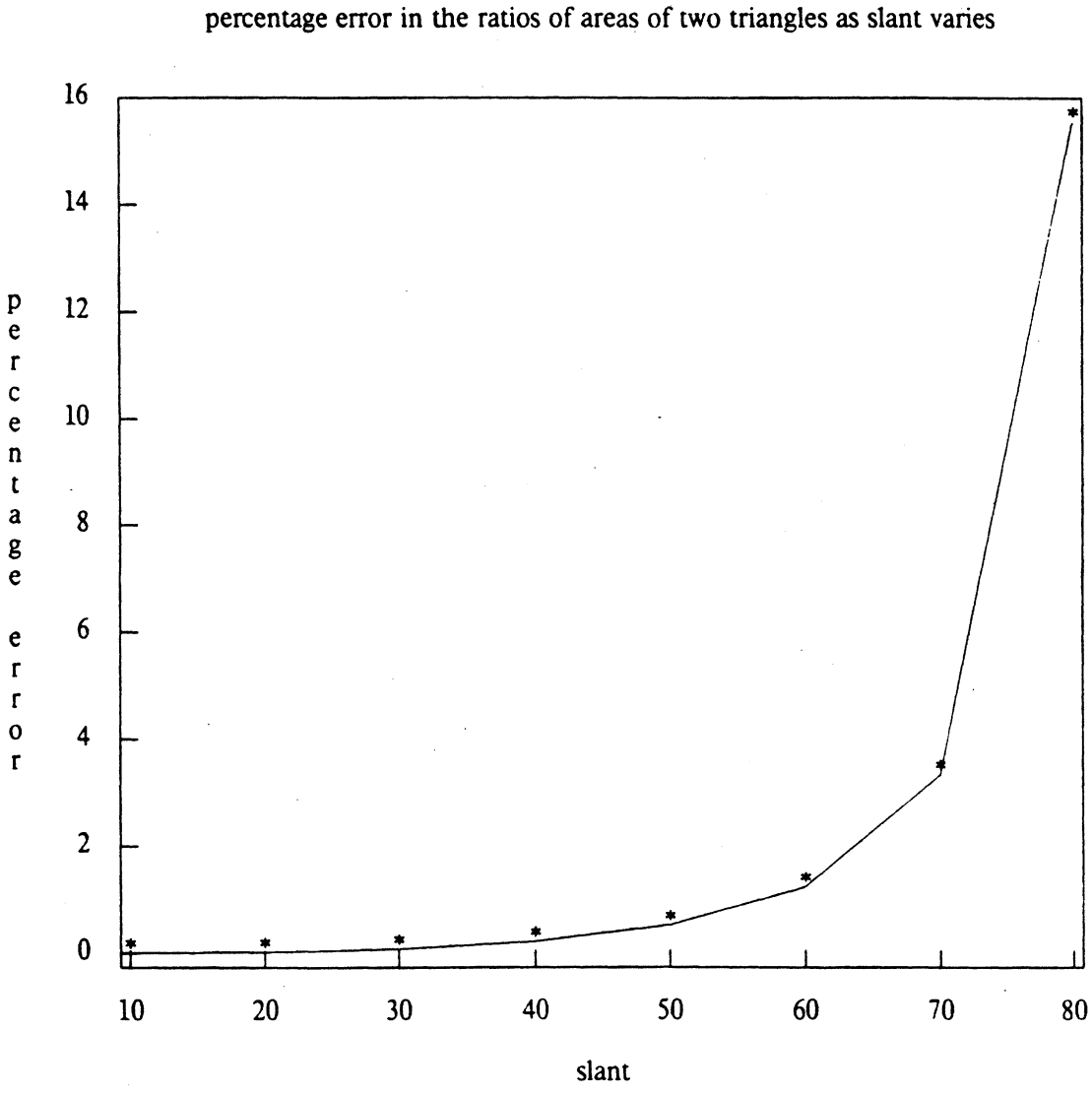
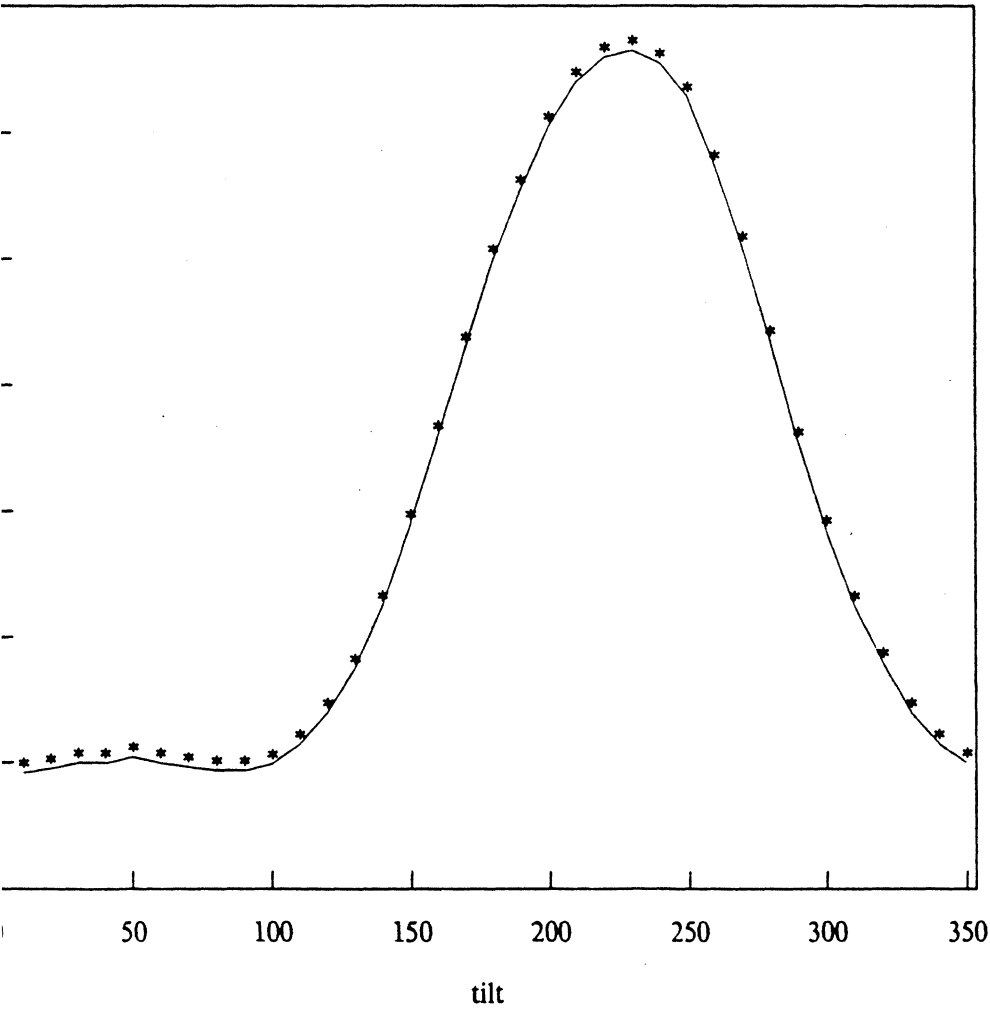

percentage error in the ratios of areas of two triangles as slant varies
percentage error
16
14
12
10
8
6
4
2
0
10
20
30
40
50
60
70
80
slant

Figure 3.22.1

percentage error in the ratios of areas of two triangles as tilt varies



percentage error in ratios of areas for two triangles as c varies

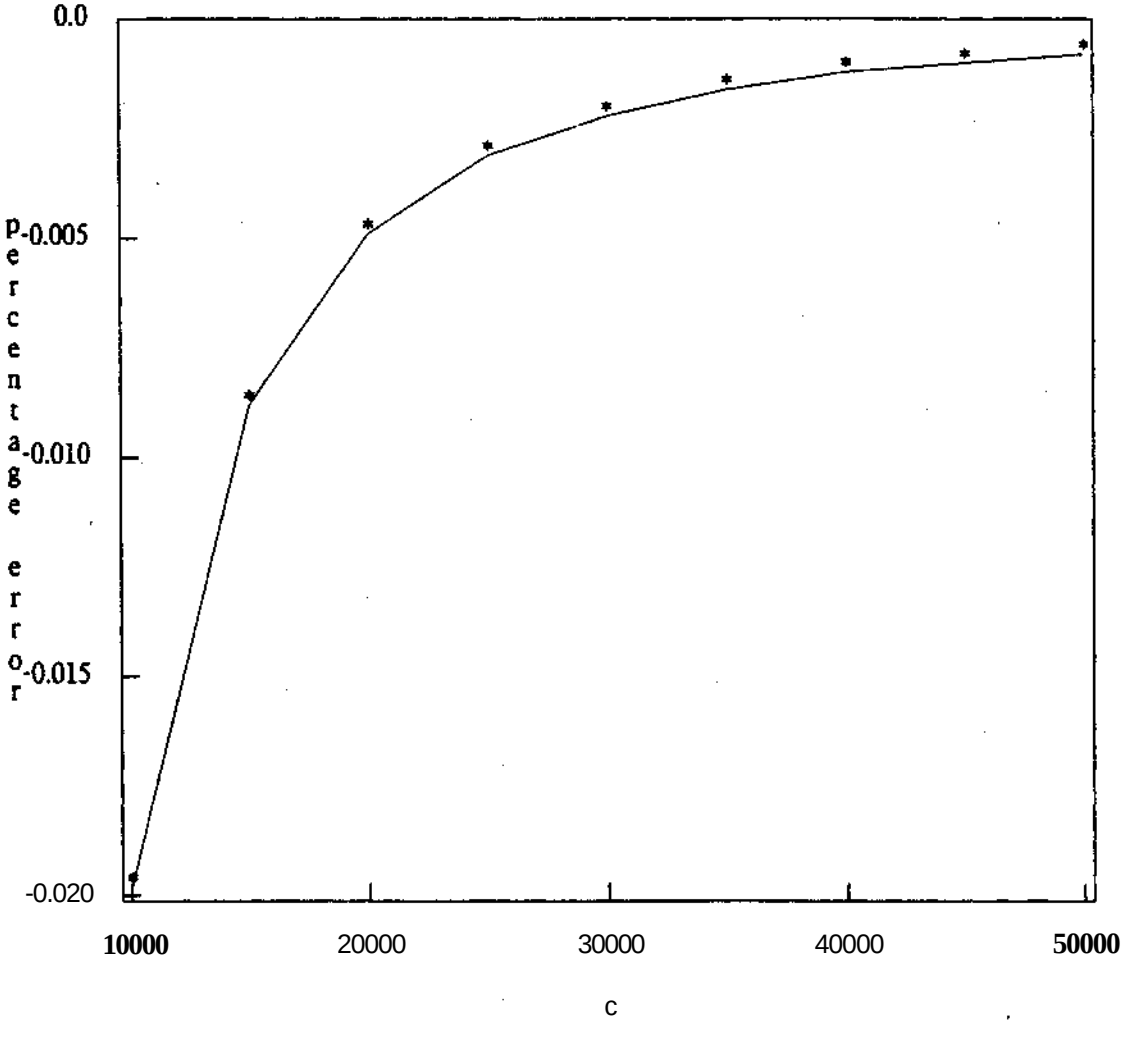


image of a plane covered with randomly generated small circles parallel to the image plane. Figure 3.28 presents the image of the plane rotated with tilt = 135 and slant = 30 degrees. Algorithm 3.2.6.1.2 (paraperspective) recovered tilt = 135.54 and slant = 29.77. Algorithm 3.2.6.1.1 (perspective) recovered tilt = 134.70 and slant = 29.85.

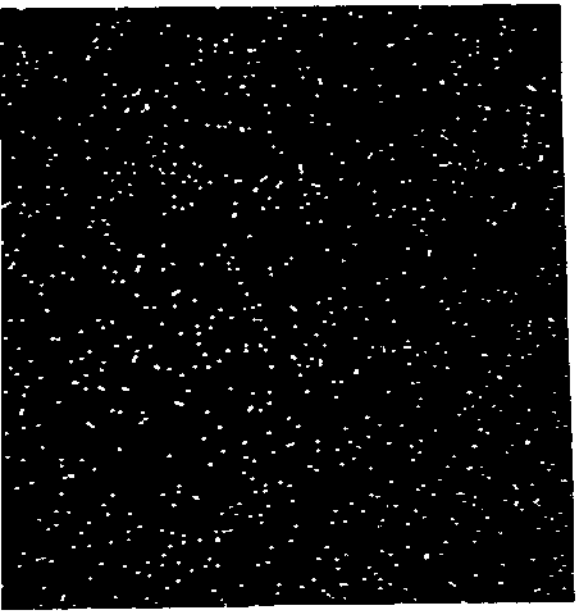


Figure 3,23:

Random dots frontal plane

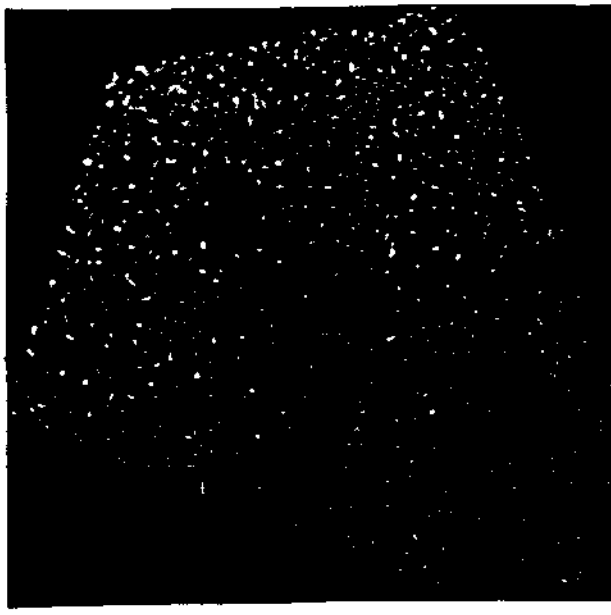


Figure 3.24:

translated and rotated

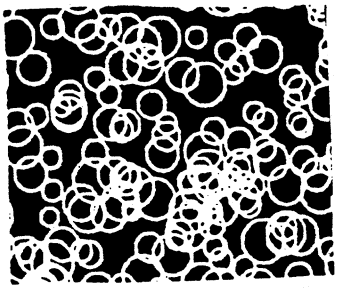


Figure 3.25

Random circles frontal plane

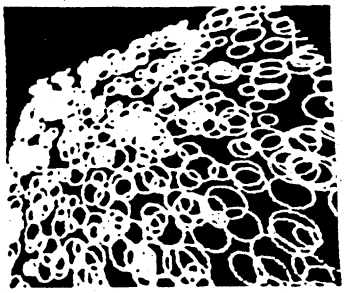


Figure 3.26

Translated and rotated

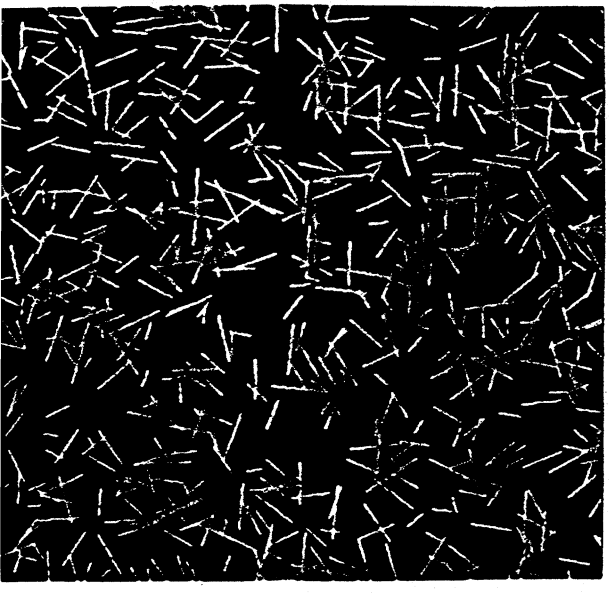


Figure 3.27

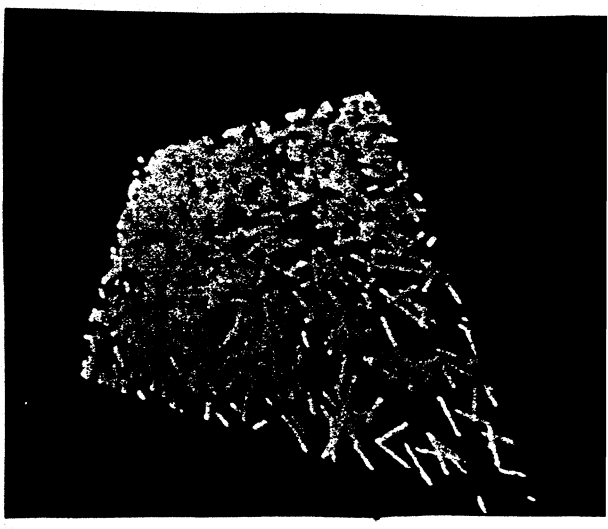


Figure 3.28

Random line segments frontal plane Translated and rotated

3.2.9.2 Natural Images

In the computation of the actual orientations when the pictures were taken, we estimate an error of about $\pm$ 2 degrees . Our results were in these bounds. The natural images used here were first preprocessed to find the boundaries of texels (edges) by applying modified Frei-Chen's operators introduced by Bandopadhyay [Bandopadhyay, 1984]. Figure 3.29 shows a photograph of a textured floor with slant = 45 and tilt = -108 degrees. The test result shows slant = 45.87 and tilt = -109.43 degrees (from the algorithm in 3.2.6.2.2-paraperspective) and slant = 46.10 and tilt = -110.5 degrees (from the algorithm in 3.2.6.2.1-perspective). Figure 3.30 shows the photograph of a part of a grass field with slant = 60 and tilt = 0 degrees. Figure 3.31 shows the image of its edges after the preprocessing. Algorithm 3.2.6.2.2 (paraperspective) recovered slant = 63.057 and tilt = -1.076 degrees. Algorithm 3.2.6.2.1 (perspective) recovered slant = 57.7 and tilt = -1.55 degrees. Figure 3.32 shows the image of a part of a brick wall, with slant = 40 and tilt = 90 degrees. Figure 3.33 shows the image of its edges. Algorithm 3.2.6.2.2 (paraperspective) recovered slant = 42.6 and tilt = 89 degrees. Algorithm 3.2.6.2.1. (perspective) recovered slant = 37.1 and tilt = 87.5 degrees.

Figure 3.34 shows the image of another brick wall , with slant = 30 and tilt = 0 degrees. Figure 3.35 shows the image of its edges. Algorithm 3.2.6.2.2 (paraperspective) recovered slant = 28 and tilt = 1.2 degrees. Algorithm 3.2.6.2.1 (perspective) recovered slant = 27.5 and tilt = 2.75 degrees.

Figure 3.36 shows the image of a part of a gravel path, with $p=0$ and $q=0$. Figure 3.37 shows the image of its edges. Algorithm 3.2.6.2.2 (paraperspective)recovered $p=0.25$ and $q=0.1$. Algorithm 3.2.6.2.1 (perspective) recovered $p=0.15$ and $q=0.12$.

Figure 3.38 shows shows the image of ivy leaves on a wall, with slant = 20 and tilt = 0 degrees. Figure 3.39 shows the image of its edges. Algorithm 3.2.6.2.2 (paraperspective) recovered slant = 24.5 and tilt = 5.6. Algorithm 3.2.6.2.1 (perspective) recovered slant = 17.35 and tilt = 4.7 degrees.

Figure 3.40 shows the image of ivy leaves on a wall, with p=0 and q=0 .Figure 3.41 shows the image of its edges. Algorithm 3.2.6.2.2 (paraperspective) recovered $p=0.012$ and $q=0.024$. Algorithm 3.2.6.2.1 (perspective) recovered $p=0.05$ and $q=0.015$.

All the above pictures in this section have been taken directly from a TV display (after digitization) because we wanted to show the reader the quality of the images that we were working with (discretization effects). The following figures 3.42, 3.43 and 3.44 show the actual pictures of some of the images that we used in our experiments. The results were again in the bounds of 0 to 5 degrees from the actual values.

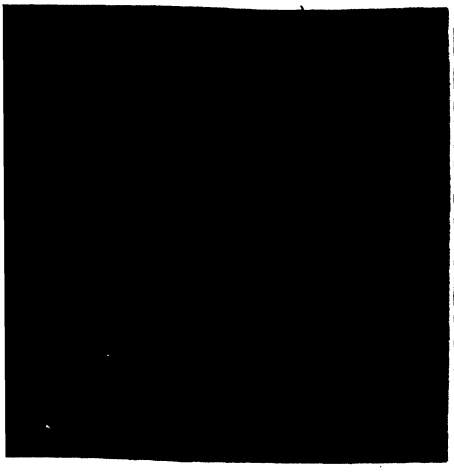
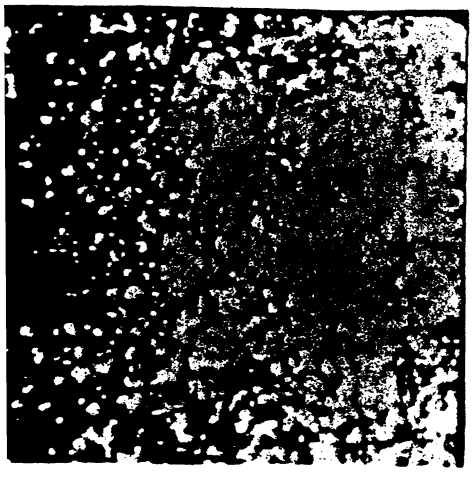


Figure 3.29 : Floor & edge image

Figure 3.30: grass field



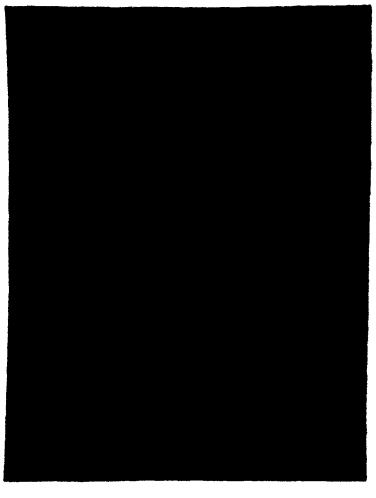
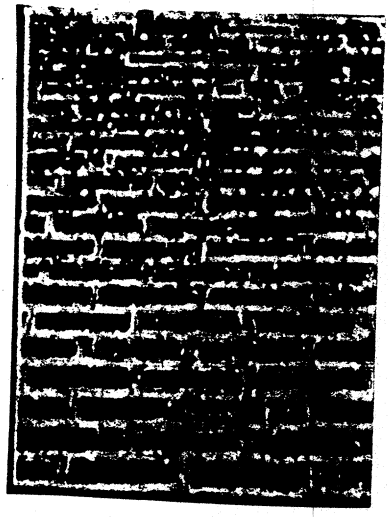


Figure 3.31:edge image

Figure 3.32: brick wall



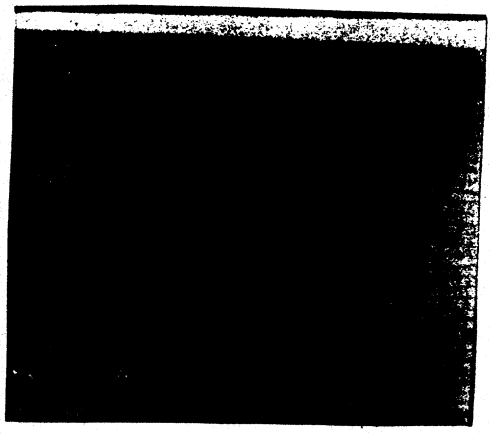


Figure 3.33: Edge image

Figure 3.34: brick wall

Figure 3.35

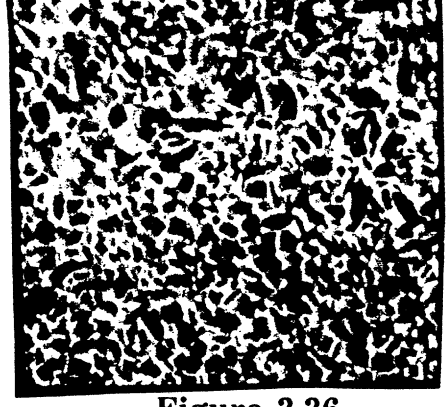


Figure 3.36

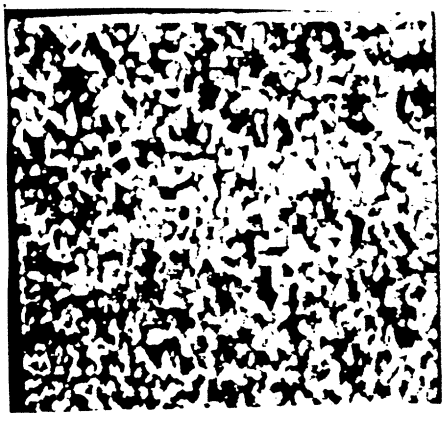


Figure 3.37

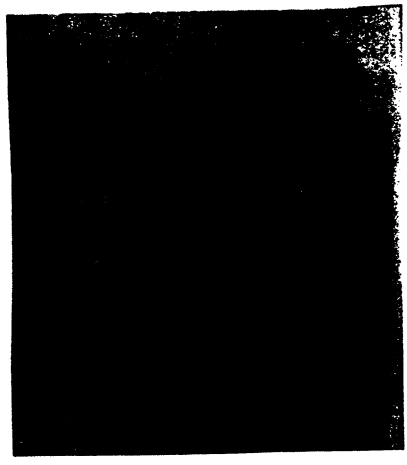


Figure 3.38

Figure 3.39

Figure 3.40

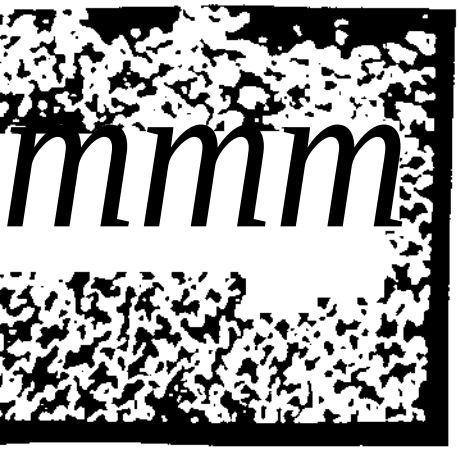


Figure 3.41

Figure 3.42

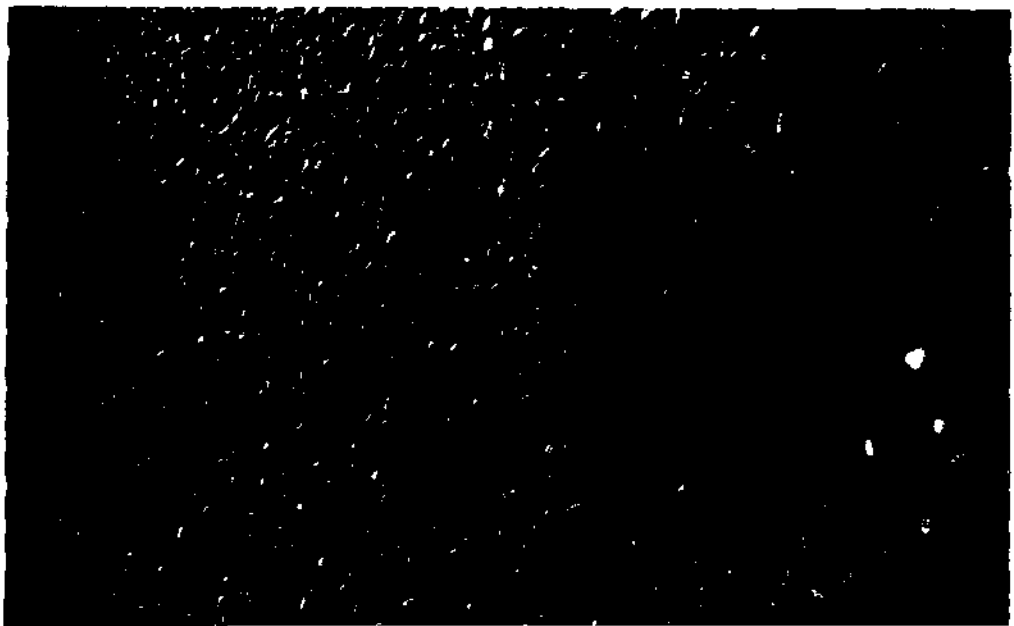


Figure 3.43 **Figure 3-44**

3.2.10 Conclusions and future directions

We have developed a simple and effective method to obtain the orientation of a textured plane from its image. Our method is based on the idea of uniform density introduced by Gibson. Our algorithm works very well in a large subset of artificial and natural images. We realize that its success depends heavily on the ability to identify the texels or the texel boundaries. We presented algorithms that employ perspective projection and an approximation of the perspective, which we called paraperspective. Overall, the performance of the algorithms under paraperspective projection was better. The basic assumption is that the segmentation problem has been solved and so we are applying the developed algorithms to the image of a single plane. But if we have the image of a set of textured planes, then segmentation can be helped by the our theories, since we can apply them to small windows all over the image, and from the different results that we will get for the orientation, to discover and segment the different planes in the image. We are currently working towards texture segmentation using the algorithms presented in this paper *(i.e.* separate the different textured planes in the image, by finding orientation). Preliminary psychological experiments indicate that this direction seems to be promising. Finally, we are working towards the extension of our theory to curved surfaces. There is an immediate extension of our theory for curved surfaces, but under the assumption that the form of the equation of the surface is given. Another way is to apply a local analysis, assuming that the surface in view is locally planar.

4

Shape from Shading and Motion: Combining information

Results

In this Chapter we prove that if we combine shading with motion, then we can uniquely compute the direction of the light source and the shape of the object in view. In particular:
1) We develop a constraint between retinal motion displacements, local shape and the direction of the light source. It is worth noting, that this constraint does not involve the albedo of the imaged surface. This constraint is of importance by its own, and it can be used in related research in computer or human vision.
2) We develop a constraint between retinal displacements and local shape. Again, this constraint is important on its own, and it is the heart of the algorithms presented later in this Chapter.
3) We present algorithms for the unique computation of the lighting direction and the shape of the object in view.
4) and we present several experimental results that test the theory.

The basic assumption in this chapter is that the retinal motion is computed everywhere in the image, in the case of a moving observer and a stationary scene, or a stationary observer and a moving object. If several objects are moving in the scene, then a segmentation is required first, i.e. the algorithms developed here can be applied to one rigidly moving object.

Introduction

Shading is important for the estimation of three-dimensional shape from two dimensional images, for instance for distinguishing between the smooth occluding contour generated by the edge of a sphere and the sharp occluding contour generated by the edge of a disc. In order to successfully use shading, one must know the illuminant direction l. This is because variations in image intensity (shading) are caused by changes in surface orientation relative to the illuminant. This chapter reviews previous approaches to the solution of the determination of the illuminant direction problem and presents a new method for the unambiguous determination of the single lighting direction, and from that of the shape information. In particular, this part of the thesis shows that if we combine information from shading and motion, then we can uniquely compute shape and the illuminant direction. Since many concepts from motion will be used, the non-advanced reader is advised to skip this chapter in the first reading and study chapter 5 first. Finally, in this chapter we are making the assumption of orthographic projection, since reflectance equation models are not known up to this point under perspective projection.

4.1 Prerequisites

The ability to obtain three dimensional shape from two dimensional intensity images, is an important part of vision. The human visual system in particular is able to use shading cues to infer changes in surface orientation fairly accurately, with or without the aid of texture of surface markings. An example in which shading information is important, is the change in luminance that distinguishes a smooth occluding contour (such as that generated by the edge of a sphere) from a sharp occluding contour (such as that generated by the edge of a disc).

The direction of illumination is required to be known in order to obtain accurate three-dimensional surface shape from two dimensional shading because changes in image intensity are primarily a function of changes in surface orientation relative to the illuminant. For example, small changes in surface orientation parallel to the illuminant direction can cause large changes in image intensity , whereas large changes in surface orientation that occur in a direction perpendicular to the direction of illumination will not change image intensity at all. So, the illuminant direction must be known before one can determine what a particular change in image intensity implies about changes in surface orientation. In this chapter, we develop a computational theory for the determination of the illuminant direction, and the shape of the object in view, from two images of a moving

object (or from two images taken by a moving observer). Before we proceed, we should discuss a little about image formation, even though this was discussed in chapter 2, in general terms.

4.2 Process of image formation

In order to be able to make quantitative statements about the world and the image and specifically to estimate the illuminant direction, we must use a mathematical model for the image formation. A great deal of work has been done in this area (Horn, 1975, 1979) and many models have been developed. For the purposes of this chapter, we use the following simple and universally accepted model (See figure 4.1).

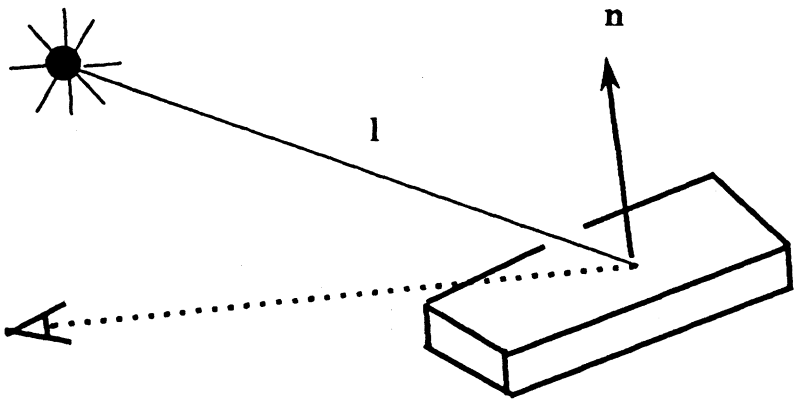


Figure 4.1: Process of image formation

Assuming orthographic projection, if $\mathbf{n}$ is the surface normal at a point on the imaged surface, $\mathbf{l}$ is the illuminant direction and f is the flux emitted towards the surface and we assume a Lambertian reflectance function for the surface (Horn 1975, 1979), the the image intensity is given by:

$$I = \rho f(\mathbf{n} \cdot \mathbf{l})$$

where ρ is the albedo of the surface, a constant depending on the surface.

4.3 Motivation and previous work

Despite the fact that the problem of determining the illuminant direction is important for computer vision (shape from shading), not too much work has been done* towards its solution.

We stress here the fact that the problem of the determination of the illuminant direction is important. Most of the work in shape from shading (Horn, 1975, Strat, 1979, Ikeuchi, 1981) assumes that the albedo of the surface in view and the illuminant direction are known *a priori;* in other words, this work assumes that the reflectance map specifies how the brightness of a surface patch depends on its orientation, under given circumstances. It therefore encodes information about the reflecting properties of the surface and information about the distribution and intensity of the light sources. In fact, the reflectance map can be computed from the bidirectional reflectance-distribution function and the light source arrangement, as shown by Horn and Sjoberg (1979).

When encountering a new scene, we usually do not have the information required to determine the reflectance map. Yet, without this information, we are unable to formulate the Shape from Shading problem, much less solve it.

The dilemma may be resolved if a calibration object of known shape appears in the scene, since the reflectance map can be computed from its image. But what happens when we are not that fortunate? It is evident from the above discussion that at least the knowledge of the illuminant direction is required. The only work done for the illuminant direction determination, is due to Pentland (1982), Brooks (1985) and Brown and Ballard(1983). Pentland's method is based on the assumption that surface orientation, when considered as random variable over all possible scenes, is isotropically distributed. A consequence of this assumption, is that the change of surface normals is also isotropically distributed. Pentland's method, that uses the same model of image formation that we do, is valid for some objects. Under his assumptions, Pentland solves the problem uniquely, but his assumptions are very restrictive.

On the other hand, Brooks and Horn (1985) presented a method in the general framework of the ill-posed problems and regularization in early vision. Their theory proposes to solve the shape from shading problem and at the same time to compute the illuminant direction, by minimization of an appropriate functional. They did not present any uniqueness or convergence proofs of their iterative methods, but their experimental results for synthetic images were reasonable.

Finally, Ballard and Brown presented a method that based on Lambertian reflectance and a Hough transform technique, recovered the direction of the light source.

In the sequel, we prove that the illuminant direction cannot be recovered from only one intensity image of a lambertian surface. After this, we will prove that two intensity images (moving object or moving observer) with the correspondence between them established, can uniquely recover the illuminant direction, and from that the shape.

4L4 A uniqueness result

Here we prove the following theorem.

THEOREM 1: Given an image (i.e. an intensity function I(x,y)), there are an infinite number of surfaces and an infinite number of positions of the light source, that will produce the same image, under the process of image formation described in section 4.2.

PROOF : Suppose that for a shape n ^ y), *(x,y)* € Q (Q is the domain where the image function is deffned) and a light source position s_t we have:

I(x, y) = p ni(x, y)" Sj, where p is the albedo of the surface in view (considered constant everywhere).

Define a shape n_2(x,y) over Q and a light source position s_2 , as follows:

»2(x, y) = 2m(ni(x, y)-m) - ni(x, y) , S2 =2m(si«m)- s_x

for any vector m, with j| m || = 1.

Then, considering a surface with the same albedo as before and with shape n_2(x,y) and illuminated from a point source in the direction $s_{2\,f}$ we have :

p n_2(x, y)-s_2 = p (2m (n_2(x, y)-m) - **ni(x,** y))-(2 m (si m)- s_t) =

p [4 (m • m)(ni(x, y)-m)(srm) - 2 **(ni(x,** y)-m)-(si - m) - 2 **(si-m)** +ni(x, y)-si]= p **ni(x,** y)si

So, *I(x,y)* = p n_2(x,y) · s_2 .

This means that the image *I(x,y)* could be due to an infinite number of surfaces illuminated from the one of an infinite number of light sources, since the vector m can be arbitrary, *(q.e.d)*

The importance of the above theorem is that no correct and robust method can exist that will find the illuminant directionfrom one intensity image of a lambertian surface illuminated from a point source.

We now move to the main part of this Chapter, that is a theorem that states that given two images of a moving object (or two images of the same object taken by a moving observer), with the correspondence between the two images established, the position of the light source can be uniquely determined. But before that, we need some technical prerequisites that are presented in the following section.

4.5 Technical Prerequisites

In this section we develop two technical results, one concerning the relationship between shape, intensity, displacements and the lighting direction and the other concerning the parameters of a small motion (small rotation) with the shape. We proceed with the following theorem.

THEOREM 2: Suppose that two views (rigid motion) of the same (Lambertian) surface (locally planar) are given and let /j and j2 be the two intensity functions. Suppose also that the displacement vector field (u(x,y),v(x,y)), (x,y) € ^ is known, where Ω is the domain of the image, i.e. a point (x,y) in the first image will move to the point (x + u(x,y), y + v(x,y)) in the second image. If the lighting direction is 1 = Gi,l2^3) *and the gradient of a surface point whose image is the point (x,y) is (p,q), the the following relation holds:*

$$p^2[(l_2\Delta^y u - l_1(1+\Delta^y v))^2 - r^2 l_1^2] +$$
$$2pq\ l(h\ A^x v - l_2(1+A^x u))(h\ (1+A^y u) - l_2\ A^y u) - 2x^2 l\ l_2] +$$
$$q2\ [(li\ \&^x v - l_2\ (1+A^x u)\ P - r^2 l^2_2 -$$
$$2\ ph\ l_3 r(r-((l + A^x u)(l+\Delta^y v) - \Delta^x u\ \Delta^x v)) -$$
$$2ql_2\ l3r(r-((l + A^x u)(1 + A^y v) \bullet A^y u\ A^x v)) -$$
$$((1 + A^x u)(l + Ayv) - Ayu\ A'v)^2 +$$
$$l^2 l\ ((A'v)^2 + (1 + A>u)^2) + l^2_2\ ((Ayu)^2 + (1 + A^x u)^2) +$$
$$+ llll_2\ ((1 + A^x u)\ A^x v + A^y u\ (1 + A^y v)) -$$
$$-r2\ l^2_3 - 2\ r\ l^2_3\ ((1 + A^x u)\ (1 + Ayv) - Ayu\ A^x v) = 0 \qquad (4.1)$$

where,

$$r = \frac{I_0(x + u(x,y), y + u(x,y))}{I_1(x,y)}$$

and

$\Delta^x u = u(x + l,y) - u(x,y)$
$\Delta^y u = u(x,y + l) - irfx,y)$
$\Delta^x v = v(x + l,y) - v(x,y)$
$\Delta^y v = v(x_9y+l) - v(x,y)$

It is clear that the above equation (4.1) is local, i.e. it involves the gradient at a point $(x_y y)_t$ the displacements around the point (x,y) along with the global direction of lighting.

PROOF: **To exploit the rigid motion assumption, we represent the surface normal by two vectors and note that their length, angular separation and hence their dot and cross product are preserved by rigid motion. Consider the surface S, a point A on S, the vector n = pi+qj + k perpendicular to S at**

the point A, and the plane II that is tangent to the surface S at the point A (see figure

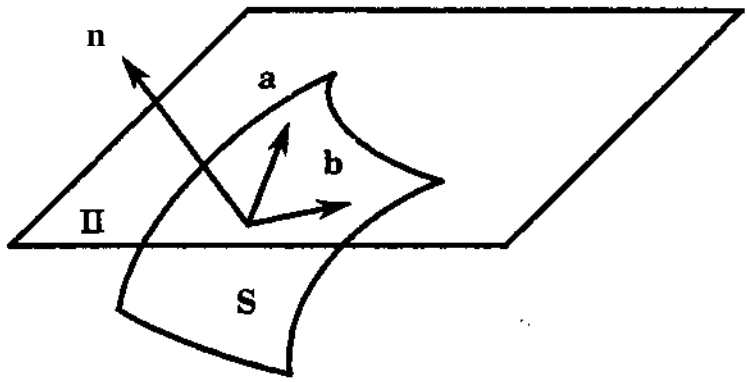


Figure 4*2: Shape vectors

4.2).

The vectors a=(1,0,-p) and b = (0,1,-q) lie in II and:
a X b = pi + qj + k = n

We shall use vectors a and b as the shape (surface normal) representation.

We use the following traditional camera model. Let O be the position of the nodal point of the eye, let OXYZ be a coordinate system that is fixed with respect to the eye, and let OZ be the line of sight. Finally, let the image plane be perpendicular to the Z-axis at the point (0,0,1).

Consider a point *A(x9y)* on the surface *S* at time *t* whose shape vectors are a=vec(AB) = i-pk and b = vec(AC)=j-qk (vec(AB) means the vector from the point A to

the point B). Since the projection of $\mathbf{a} = \text{vec(AB)}$ on the image plane is $\mathbf{i}$, we conclude that if $I_A = (x,y)$ is the projection of A then the projection of B will be $I_B = (x+1,y)$. Similarly, the projection of C will be the point $I_C = (x,y+1)$ on the image plane. Consider now the object at the next frame where the point A will become A', with shape vectors $\mathbf{a'} = \text{A'B'}$ and $\mathbf{b'} = \text{A'C'}$

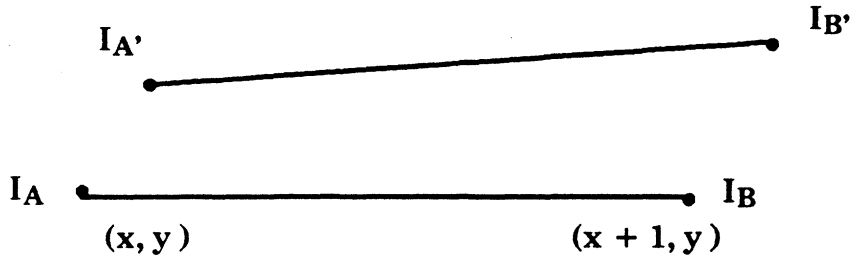


Figure 4.3: Displacement vectors

Let $I_{A'}$ be the projection of A'. $I_{A'}$ is the position to which I_A moves, which can be determined from the displacements. Thus, $I_{A'} = (x+u(x,y), y+v(x,y))$. Similarly, if $I_{B'}$ and $I_{C'}$ are the projections of B' and C', then $I_{B'}$ is the position to which I_B will move. Of course, displacement at I_B is due to the motion of the surface's point that has the same x,y coordinates as B (orthography), but because of the assumed local planarity, this point is the same as B. (The planarity constraint of course fails at boundaries). So, we have: $I_{B'} = (x+1+u(x+1,y), y+v(x+1,y))$, and for the same reasons $I_{C'} = (x+u(x,y+1), y+1+v(x,y+1))$. The projection of $\mathbf{a'} = \text{vec(A'B')}$ on the image plane is thus:

$I_{A'}I_{B'} = [1+u(x+1,y)-u(x,y)]\mathbf{i} + [v(x+1,y)-v(x,y)]\mathbf{j}$

But according to our hypotheses, the above relation can be written:

$I_{A'}I_{B'} = (1+\Delta^x u)\mathbf{i} + \Delta^x v\,\mathbf{j}$

Similarly, we get:

$I_{A'}I_{C'} = \Delta^y u\,\mathbf{i} + (1+\Delta^y v)\mathbf{j}$

The above two equations give us the expressions for $I_{A'}I_{B'}$ and $I_{A'}I_{C'}$ which are the projections of the shape vectors $\mathbf{a'}$ and $\mathbf{b'}$ respectively. But then,

$\mathbf{a'} = (1+\Delta^x u)\mathbf{i} + \Delta^x v\,\mathbf{j} + \lambda\mathbf{k}$ and

$\mathbf{b'} = \Delta^y u\,\mathbf{i} + (1+\Delta^y v)\,\mathbf{j} + \mu\mathbf{k}$ where λ, μ are to be determined.

But since rigid body motion preserves the vector length, we have that

$$\|a'\|^2 = \|a\|^2$$

or

$$A = \pm(p2 - A^x u - A^x v - 2*A^x u)^{1/2}$$

Similarly, we get

$$ji = \pm(q2.\ A^y u - A^y v - 2*A^y u)^{1/2}$$

Assuming that neither region is in shadow, we have that:

$$I_1(x,y) = Pi\ n_A\ 1 \qquad (4,2)$$

and

$$h(x + u(x,y), y + v(x_f y)) = p_2\ n \wedge l \qquad (4.3)$$

Equation (4.2) above, gives the intensity of the point $A(x_f y)$ in the first frame. Note that n^ is the surface normal at the point A (in the first view). Equation (4.3) gives the intensity at the point $A'(x + u(x_f y), y + v(x_y y))$ in the second frame. Note that n^» is the surface normal at the new position of the point A .

Dividing equations (4.2) and (4.3) and setting $l2(A')IIi(A) = r$ and taking into account that Pi = P2 (surface markings do not change), we get:

$$r n_A.l = n_A>.l \qquad (4.4)$$

But

$$n_A = \frac{a \times b}{\| a \times b \|}$$

(4.5) and from the rigidity of the motion it follows that:

$$n_{A'} = \pm \frac{a' \times b'}{\| a' \times b' \|} \qquad (4.6)$$

where the sign is chosen such that $\mathbf{n}_{A'} \cdot \mathbf{k} > 0$.

But since

$\|\mathbf{a}'\| = \|\mathbf{a}\|$, $\|\mathbf{b}'\| = \|\mathbf{b}\|$ and $\mathbf{a}\cdot\mathbf{b} = \mathbf{a}'\cdot\mathbf{b}'$, it follows that

$$\|\mathbf{a} \times \mathbf{b}\| = \|\mathbf{a}' \times \mathbf{b}'\| \qquad (4.7)$$

Using equation (4.7), equation (4.4) becomes:

$$\mathbf{r}\,[\,\mathbf{a}, \mathbf{b}, \mathbf{l}\,] = \pm\,[\,\mathbf{a}', \mathbf{b}', \mathbf{l}'] \qquad (4.8)$$

where the sign chosen is the sign of $[\mathbf{a}',\mathbf{b}',\mathbf{k}]$, and $[\ ,\ ,\ ]$ is the triple scalar product of vectors. But $[\mathbf{a}',\mathbf{b}',\mathbf{k}] = (1+\Delta^x u)(1+\Delta^y v) - \Delta^y u\ \Delta^x v$ It is obvious that $[\mathbf{a},\mathbf{b},\mathbf{k}]>0$; if $[\mathbf{a}',\mathbf{b}',\mathbf{k}]<0$, then we don't have a valid motion, because $[\mathbf{a}',\mathbf{b}',\mathbf{k}]<0$ means that we have reversed orientation so that the the texture in the image is viewed as if seen in a mirror.

So, $[\mathbf{a}',\mathbf{b}',\mathbf{k}]>0$, and substituting in equation (4.8) the values of $\mathbf{a},\mathbf{b},\mathbf{a}',\mathbf{b}'$ after algebraic manipulations and using the fact that :

$$\lambda^2 = p^2 + 1 - (1+\Delta^x u)^2 - (\Delta^x v)^2$$

$$\mu^2 = q^2 + 1 - (1+\Delta^y v)^2 - (\Delta^y u)^2$$

$$\lambda * \mu = p * q - (1+\Delta^x u)\Delta^y u - (1+\Delta^y v)(\Delta^x v)$$

we get equation (4.1).

It is obvious that equation (4.1) involves the lighting direction , but it also involves the shape (p,q). We would like to find the direction $\mathbf{l}$ without knowing the shape (p,q), otherwise the problem is of no importance. Theorem 2 is very important , in the sense that it has established a constraint between lighting direction, shape and displacement vectors.

We now proceed with the following theorem.

THEOREM 3. Suppose that the surface S (locally planar) is moving with a rigid motion, and the camera model is the one described in the previous theorem. Let the gradient of the surface (with respect to the first frame) be $(p(x,y),q(x,y))$ and the displacement vector field be $(u(x,y),v(x,y))$. It is known that this motion can be considered as a translation (dx,dy,dz) plus a rotation of an angle θ about an axis (n_1,n_2,n_3) passing through the origin $(n^2_1 + n^2_2 + n^2_3 = 1)$. If the rotation angle θ is small, then the following relations hold:

(a): The displacement vector field $(u(x,y),v(x,y))$ is given by :

$$u(x,y) = dx + Bz(x,y) - Cy$$

$$v(x,y) = dy + Cx - Az(x,y),$$

where $A = n_1\theta, B = n_2\theta, C = n_3\theta$ *and* $z(x,y)$ *the depth of the surface point whose projection is the image point* (x,y).

(b):

$$p(x,y) = \frac{1}{B}\,\Delta^x u$$

$$q(x,y) = \frac{1}{A}\,\Delta^y v$$

$$\frac{A}{B} = \frac{\Delta^y v}{\Delta^x u} \cdot \frac{\dfrac{\Delta^x v - \Delta^y u + \sqrt{(\Delta^y u + \Delta^x v)^2 - 4 * \Delta^x u * \Delta^y v}}{2} - \Delta^x v}{\Delta^y v}$$

PROOF:

a) Trivial.

b) Using the two equations in (a) and the assumption about local planarity, we get:

$\Delta^x u = Bp$

$\Delta^x v = C - Ap$

$\Delta^y u = Bq - A$

$\Delta^y v = -Aq$

where the p and q are considered at the point (x,y). From the above equations we get:

$$p(x,y) = \frac{1}{B}\,\Delta^x u$$

$$q(x,y) = \frac{1}{A}\,\Delta^y v$$

$$\frac{A}{B} = \frac{A^y v}{A^x u} \cdot \frac{\Delta^x v - \Delta^y u + \sqrt{(\Delta^y u + A^x v)^2 - 4 * A^x u * A^y v}}{2 A^y v} - \&^x v$$

(q.e.d).

4.6 Development of the lighting direction constraint

In this section we develop the constraint that will be used as the heart of the algorithm that will solve the problem of the determination of of the illuminant direction. If we let $1 / A = \mathbf{a}$ $1/B = \mathbf{P}$ $\mathbf{B / A} = K$ and use part (b) of theorem 3, to subsitute in equation (4.1) for p and q, we get the following equation.

$$(\Delta^x u)^2 \beta^2 [(l_2 A^y u - I_1 + \Delta^y v)^2 - r^2 l_1] +$$
$$+ 2 A^x u A^y v K\$^2 [(h A^x u - l_2)(1 + A^x u))(h (1 + A^y v) - l_2 A^y u) - 2$$
$$r^2 l_1 l_2] + (A^y v)2 K2 \$2 [(^ A^x v - 1_2 (1 + \&^x u))2 - r 2 l^2_2] -$$
$$-2 (k^x u) \mathbf{p} li I3 r (r-(l + A^x u + A^* v + \&^x u \&>v - A\hat{}u \&^x v)) -$$
$$-2 (\&v) K \$ l_2 I3 r (r-(l + A^x u + \&^y v + A^x u A'u - A'u \&^x v)) -$$
$$- (1 + A^x u A\hat{}i; + L^x u + A\hat{}-A>u \&^x v)) +$$
$$+ l^2 l ((A^* v)2 + (1 + A y_U)2) + l^2_2 ((A^x u)2 + (1 + A^* u)2) +$$
$$+ h h ((1 + A^x u) A^x v + A^y u (1 + A^y v)) - r2 l^2_3 +$$
$$+ 2r l^2_3 (1 + A^x u + A^y v + A^x u A^y v - A^y u A^x v) = 0 \qquad (4.9).$$

The above equation is a polynomial in l_1, l_2, l_3, β.

Considering equation (4.9) in four points we get a polynomial system of four equations in four unknowns. A simple but tedious calculation of the Jacobian of this system gives us the fact that the Jacobian has rank four (except for the degenerate cases whose set has measure zero). This means (inverse function theorem) that the function defined by the equations of the system is locally an isomorphism, which means that its zeros are isolated. But, from Whitney's theorem, the set of the zeros of this algebraic system is an algebraic set and it has finite components.

The conclusion of this is that the solutions of the system are finite (uniqueness). If we now consider equation (4.9) in five points, then we get a system of five equations in four unknowns which, barring degeneracy, will have at most one solution.

It is clear from the above discussion, that two intensity images of a Lambertian surface, with the correspondence between them established, gives the lighting direction uniquely. In the next section, we present a practical way to recover the lighting direction based on the constraint developed in this section.

4.7 The algorithm for finding illuminant direction

First of all we choose the Gaussian sphere formalism (azimuth-elevation) to represent the vector that denotes the lighting direction. More specifically, we set:

$$l_1 = \cos\phi \cos\theta$$

$$l_2 = \sin\theta$$

$$l_3 = \cos\phi \cos\theta$$

where 8 and ϕ are the azimuth and elevation (See figure 4.4).

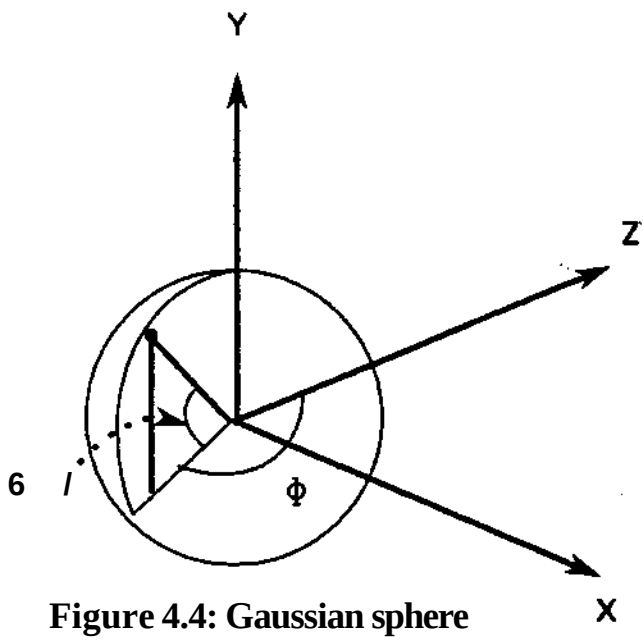


Figure 4.4: Gaussian sphere

Now we consider equation (4.9) in n points in the images, and we get n equations *eql, eq2, ..9eqn* in the three unknowns f$,8,4>. Then, the following algorithm solves the problem:

```
for all 6
    for all <J>
        {
            get n quadratic equations in {3
            Check if they have a common solution
            If yes, output 8<|>.
        }
```

4.8 Applying the algorithm to natural images

When one is experimenting with natural images, it is sometimes difficult to compute the displacement field for every point in the image. In that case, one can compute the parameters of the correspondence of small regions. In other words, if we have a small planar region Si in the first image that corresponds to a small region *S2* in the second image, then the parameters of an afiine transformation $f(x,y)=(ax + by+c,dx+cy+fi$ between the two patterns (see figure 4.5)

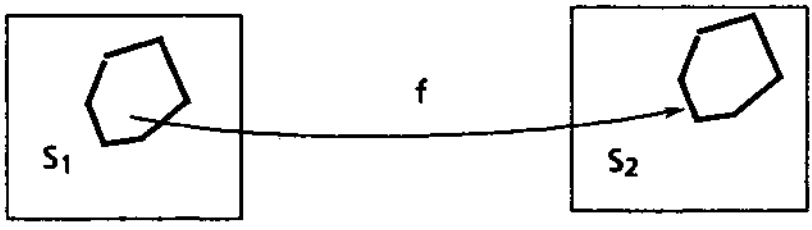


Figure 4.5:Corresponding regions

can be computed using a variation of a least-squares method introduced by Lucas and Kanade that is described in Webb[1981]. In that case, the essential constraint (equation (4.9)), has a similar form and the whole analysis proceeds as previously.

4.9 Implementation and experiments

We have implemented the abovementioned algorithm, and it works succesfully for synthetic images. Figure 4.6 represents the displacements vector field that was obtained from the motion (rotational with $\omega_x = 1, \omega_y = 2, \omega_z = 3$) of a sphere. Figure 4.7 represents the image of the sphere before the motion. The surface of the sphere is supposed to be Lambertian, the albedo $\rho = 1$ and the lighting direction with gradient $(ps,qs) = (0.7,0.3)$, i.e. to the right and a little above the horizon. The computed lighting direction was (0.65, 0.33). The observed inaccuracy is due to discretization effects. In our synthetic experiment, we did not compute the displacements; instead , we calculated them since we knew the motion and the exact position of the sphere. If the Lambertian reflectance model is not adequate for capturing natural images (which it is not, of course), there probably exists a model (not discovered yet) that captures natural shading. This model, should of course depend on the shape and the lighting direction. The approach that we took here, can be taken with any other model of reflectance, and it is one of our future goals to apply the method in natural images and employ general reflectance models, that consider the illumination from the sun and the sky.

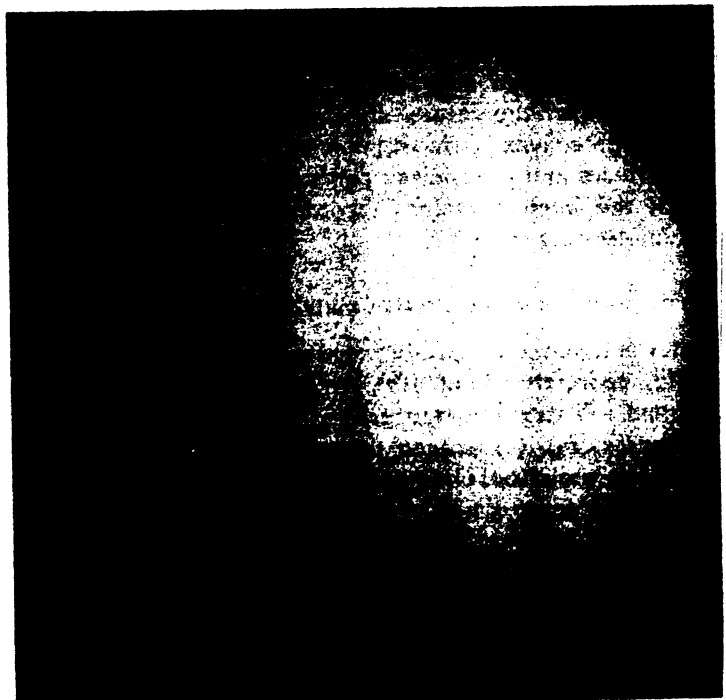


Figure 4.6:
Intensity image

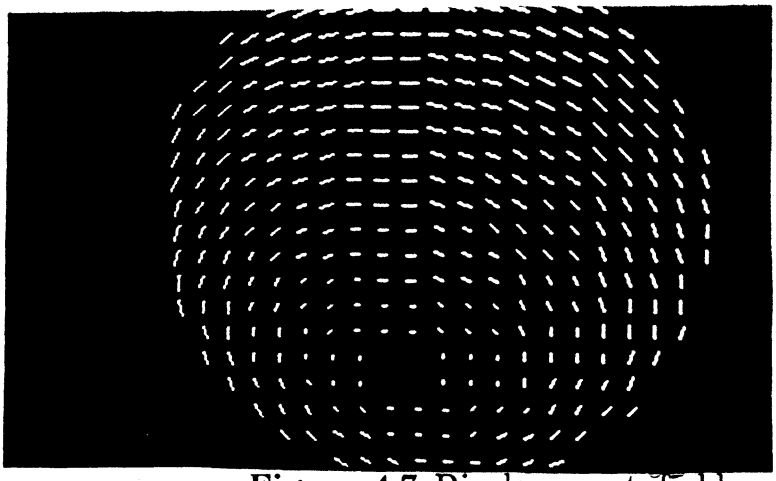


Figure 4.7: Displacements field

The next section discusses the problem of determining shape from shading and motion, in a unique way. Again the findings of the next section cannot be applied at this point to natural images, for the reasons that we mentioned above. The treatment again, is of theoretical value, and the method could be applied to natural imagery, if better reflectance models (i.e. models that capture the reality) were known.

4.10 Computing shape from Shading and motion

In this section we discuss the problem of determining Shape from Shading and Motion. Before we proceed we need some technical prerequisites , that are introduced in the next sections.

4.11 The constraint between shape and displacements

THEOREM 4: With the assumptions and notation of THEOREM 2, the gradient (p,q) of a surface point whose image is the point (x,y), with displacement vector (u,v), satisfies the constraint :

$Ap^2+Bq^2-2Cpq+D=0$, with

$A=(\Delta^y u(x,y))^2+(\Delta^y v(x,y))^2+2\Delta^y v(x,y)$

$B=(\Delta^x u(x,y))2+(\Delta^x v(x,y))2+2\ \Delta^x u(x,y)$

$C=\Delta^y u\ (x,y)+\ \Delta^y u\ (x,y)\Delta^x u(x,y)+\Delta^x v(x,y)+\Delta^x v(x,y)\Delta^y v(x,y)$

$D=C^2-AB$,where the coefficients

$\Delta^x u$, $\Delta^y u$, $\Delta^x v$, $\Delta^y v$ are defined in theorem 2.

PROOF: From the proof of theorem 2, because of the rigid motion assumption, we have the preservation of the dot product. So,

$$Ap^2+Bq^2-2Cpq+D=0 \qquad (4.10)$$

where (p,q) is the gr adient at the point (x,y), and

$$A=(\Delta^y u(x,y))^2+(\Delta^y v(x,y))^2+2\Delta^y v(x,y)$$

$$B=(\Delta^x u(x,y))2+(\Delta^x v(x,y))2+2\ \Delta^x u(x,y)$$

$$C=\Delta^y u(x,y)+\Delta^y u(x,y)\Delta^x u(x,y)+\Delta^x v(x,y)+\Delta^x v(x,y)\Delta^y v(x,y)$$

$$D=C^2-AB$$

Equation (4.10) gives the constraint between displacements and shape and represents a conic section in p-q space. This conic section is a hyperbola or parabola depending on the values of the coefficients A,B,C. The constraint (4.10) is a constraint between shape and displacements. Constraint (4.1) is a constraint between shape, displacements and the lighting direction. For the purposes of the rest of this Chapter, and to avoid confusion, we will refer to constraint (4.1) as the *lighting constraint*, and to constraint (4.10) as *shape-motion* constraint.

4.12 How to utilize the constraints

Here we show how to utilize the constraints to recover the three-dimensional shape of the object in view, using shading and motion. It is assumed that the lighting direction has already been computed with the algorithm described in section 4.7. Up to now we have developed two constraints on shape, that also involve retinal motion displacements and the lighting direction. The lighting constraint is a conic on p,q, with coefficients that depend on intensities (relative), displacements and lighting direction. The shape-motion constraint is again a conic on p and q, with coefficients that depend on the displacement vectors. Finally, the image irradiance equation:

$$I = \rho\, f(\mathbf{n}\cdot\mathbf{l}) \qquad (4.11)$$

that determines the intensity $I(x,y)$ at a point (x,y) of the image as a function of the shape $\mathbf{n}$ of the world point whose image is the point (x,y) and the lighting direction $\mathbf{l}$, is another constraint on p and q that is also a conic. This constraint we will be callling *image-irradiance* constraint. The next subsections will describe algorithms for the unique computation of shape, under a variety of situations.

Figure 4.7.1 below gives a geometrical description of the constraints.

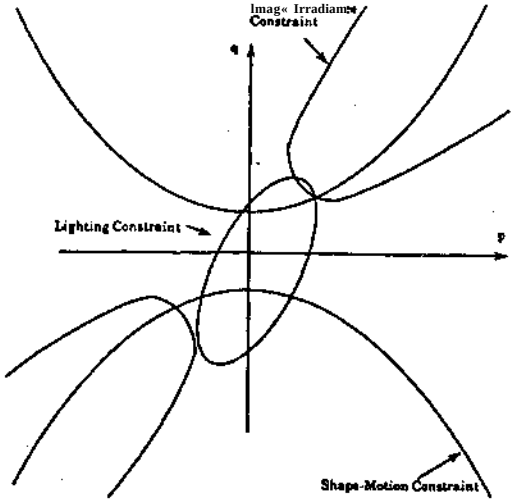


Figure 4.7,1: Pictorial description of the constraints

4,12.1 Computing shape when the albedo is known

When the albedo is known, then we have at our disposal three constraints at every image point for the computation of shape. The lighting constraint, let it be *Fx(p,q) =0,* the shape-motion constraint, let it be *F2(p,q)=0,* and the image-irradiance constraint, let it be *F^(p,q)=0.* These are three equations, all of degree two, and their system, barring degeneracy, will have at most one solution. Several algebraic or geometrical techniques exist for solving a system of algebraic equations, each of degree two. Direct methods result in solving equations of a high degree, and so we prefer to use an iterative technique; and even though we do not have theoretical results about the convergence of the technique, in pactice it has shown to converge very fast, to the right solution.

The function $E(p,q)= A_1(F_1(p,q))2+ A_2(F_2(p,q))2 + X_3(F_3(p,q))2$, should be minimized everywhere in the image, where $Xi_t A_{2f}$ A3 constants with their sum equal to one. If we add one more term in the error function that accounts for smoothness and by setting the partial derivatives of E(p,q) equal to zero and solving for p and q, we get equations of the following form:

$p=Gi(p,q,p_{av})$, $q = G_2(p,q,q_{av})$, where G_x and G_2 are polynomials of p,q,p_{av} and p,q,q_{av} respectively. These equations can be solved iteratively, provided that we have an approximate initial solution. If the values of p and q at the boundaries are known, then p,q are propagated throughout the image using a general smoothness criterion, by an algorithm similar to the Gauss-Seidel algorithm of Ikeuchi and Horn [Ikeuchi and Horn,1981]. At this point we should emphasize that we do not depend on smoothness to

achieve uniqueness in our methods. Smoothness is used to achieve an approximate initial solution.

4.12.2 Computing shape when the albedo is not known

a) Iteratively.

If the albedo is not known, then we cannot utilize the image-irradiance constraint, because it contains the albedo as a coefficient. We have to use the lighting constraint and the shape-motion constraint. An algorithm similar to the one in the above section can be easily obtained. At this point, the uniqueness of this problem has to discussed. The lighting constraint and the shape-motion constraint are two conics in p and q. The Jacobian of the system that they form, is non-zero. So, the function defined by the system is locally an isomorphism (from the inverse function theorem), which means that that its zeros are isolated. But fromWhitney's theorem, the set of zeros of this algebraic system is an algebraic set and it has finite components. The conclusion of this is that the sytstem has *finite solutions*. In this case, the solutions can be restricted to a unique solution, if a local smothness constraint is used. This, being in the paradigm of the regularization theory (which we do not follow), has been observed from experiments, and up to this point we do not have a formal proof.

b) Directly

In a minimization scheme based on the Lagrange multiplier technique, the solution is obtained without propagating the boundary conditions. If $F_1(p,q)=0$ is the lighting constraint and $F_2(p,q)=0$ the shape motion constraint, the error function $E(p,q)=(F1(p,q))2$ is to be minimized subject to the constraint $F_2(p,q)=0$. The Lagrange multiplier scheme says that the p,q that minimize E are one of the solutions of the following system:

$$\nabla E=\lambda\nabla F_2,\ F_2=0 \text{ where } \lambda \text{ the Lagrange multiplier.}$$

4.12.3 Implementation and experiments

Figures 4.8 and 4.9 represent exactly the same entities as figures 4.6 and 4.7. From this input, our algorithm (4.12.1) recovered the shape shown in figure 4.10. A local

smoothing scheme has been used at the end of the program to smooth out the results. The error in the resulting shape is very low, basically due to discretization effects. If we introduce noise in the input, then the results get very much corrupted, if we don't apply a smoothing scheme, because of the locality of the method.. If a local smoothness constraint is utilized, then the results are very good. At this point, we should emphasize that a local smoothness constraint is not restrictive, and almost every natural surface obeys this constraint.

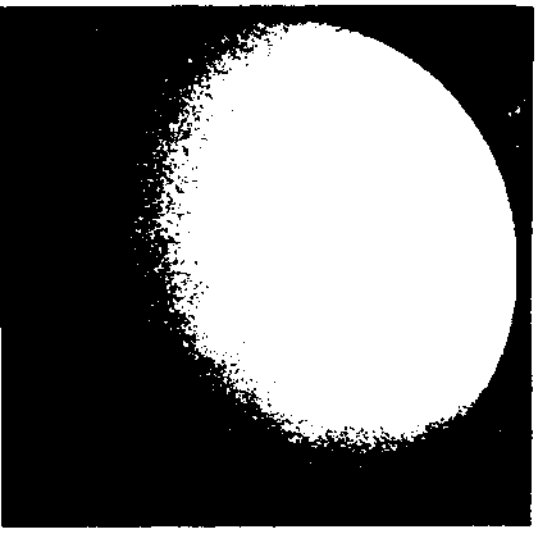


Figure 4.8: Intensity image for a sphere

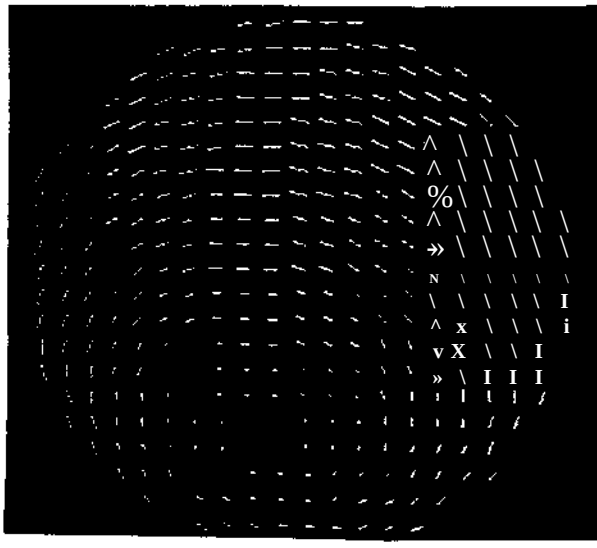


Figure 4.9: Displacements field

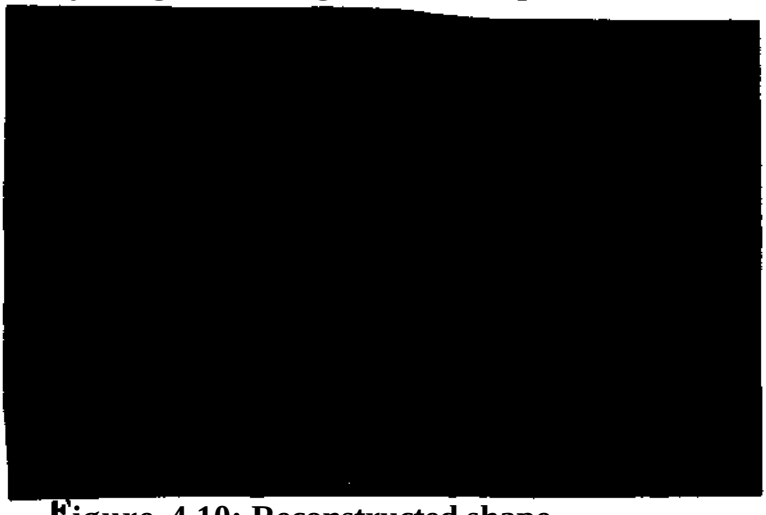


Figure 4,10: Reconstructed shape

4.13 Conclusions and future directions

In this chapter we have presented a theory on how to compute in a unique way shape and the direction of the light source, from shading and motion. Our input, is the intensity of two images in a dynamic sequence, with the correspondence between the two frames established. We proved that in this case the light source direction and the shape of the object in view, can be uniquely determined in contrast with existing theories that are based on heuristics and restrictive assumptions [Pentland, 1981, Horn, 1979, Ikeuchi, 1981]]. Our results have theoretical value, since they demonstrate that if we combine information from different sources then we can obtain unique results for intrinsic images. In the past, there has been in this framework only the work of Grimson [Grimson, 1983], that combined shading with stereo with very good results. It is one of our future goals, to extend this theory to capture a very general reflectance maps [Brooks and Horn, 1986], that model the illumination due to the sun and the sky.

5

Visual Motion Analysis

Results

In this Chapter we study the problem of interpreting Visual Motion. We derive several results of both theoretical and practical importance. In particular, our results are the following:

1) The orthographic velocity field does not provide enough information to recover the structure of the object in view.

2) The orthographic velocity, if the surface in view is nonplanar, determines the structure of the inducing object up to a depth-scaling, or in other words the tilt at every point of the surface in view can be uniquely recovered. For planar objects, the orthographic velocity field admits two interpretations for the structure of the inducing object, up to a depth scaling, or in other words there are two distinct solutions for the tilt of every point of the surface in view.

3) For the discrete case, three orthographic projections of three points in space can uniquely recover the structure of the points, when a testable condition holds. Furthermore, when this condition does not hold, the number of structures compatible with the motion is at most two.

4) The perspective optic flow field uniquely defines the three-dimensional motion parameters except for the case of planar surfaces and some special kind of quadric surfaces (Hyperboloids).

5) Shape and three-dimensional motion are equivalent in the sense that the one greatly simplifies the computation of the other.

6) In the case of differential motion and using only one camera, the spatiotemporal derivatives of the intensity function are enough to detect some kinds of motion. In

particular, only rotational or only translational motion. The general case is reduced to the solution of a nonlinear system. So, motion can be detected without the need of optical flow.
7) If a binocular observer is used instead of a monocular one, then the problem of detecting three-dimensional motion becomes easier in the sense that nonlinear motion equations of the monocular case, now become linear. Of course, since a binocular observer is used, it appears that we need to solve the correspondence problem between the left and right images. But we present a theory *on* how to find depth without correspondence for the case of planar surfaces. For the case of general (curved) surfaces, the work of finding depth without correspondence is one of our future goals.
8) Finally we show how to recover three-dimensional motion in the discrete perspective case, without point correspondences.

The basic assumption in this chapter is that we only consider rigid motion, i.e. we either consider the camera moving in a static environment or only one rigid object moving in front of a stationary camera. All of the results, exactly as in the case of shape from texture, if there are more than one object moving in the visual field, in order to be applied, a segmentation is first required. And again, if some of the results are applied locally everywhere in the image, they can contribute a great deal to segmentation.

Prolegomena

A lot of useful information can be extracted from time varying images. At first, it might seem foolish to consider processing sequences of images, given the difficulty of interpreting even a single image. Curiously though, some information seems to be easier to obtain from a time sequence. When the camera is moving relative to the objects being imaged, or equivalently the imaged objects are moving, then the brightness pattens in the image are moving. This motion, i.e. the motion of the image is called *image* or *retinal motion*. Several theories have been proposed for measuring and interpreting this retinal motion. In this chapter, we consider the problem of interpreting image motion, i.e. to recover the structure and three-dimensional motion of a moving object from a sequence of its images, and we suggest several computational mechanisms for the motion interpretation process. This problem is known in the literature as the *structure from motion* problem. Despite the fact that we don't propose any theories for the measurement of the image motion, we will criticize previous approaches and we suggest some ideas that might prove to be fruitful for the computation of retinal motion.

5.1 **Introduction**

The perception of motion and structure from temporally varying two dimensional retinal stimulus is important to computer vision as well as the cognitive and perceptual sciences. In this thesis, this problem is addressed from the standpoint of formulating a computational theory underlying the motion interpretation process. For the purpose of this chapter, the input (stimulus) to this process is a two dimensional map of the intensity of reflected illumination from the imaged scene. In other words the starting point is cine or video imagery. The *first* step in the computation, according to all the existing theories, involves the estimation of two dimensional motion in the image plane. The second stage involves the computation of the three dimensional scene intrinsics like *structure* and *motion parameters* from the two dimensional image motion. Here we are basically concerned with the latter stage in the perceptual process. There is a lot of research on the estimation of image motion. Unfortunately, the problem of estimating image motion has proved to be extremely hard, if not impossible to solve. Later in this chapter we shall discuss the difficulties involved in this process. The subsequent analysis is based on the following imaging conditions and assumptions:

1. The image is monochromatic.
2. The imaged surface moves rigidly.
3. There is no ambiguities due to occlusion.

Even though almost all the existing theories are based on the hierarchical model of motion perception (computation of retinal motion first, then interpretation), we prove later that this is not necessary, and the interpretation process can be done in some cases without having to compute retinal motion.

5.2 Technical Prerequisites

The problem of estimating image motion from time varying image intensity distributions is by no means a simple one. There are essentially *two ways* in which this problem has been tackled.

(i) The first assumes the dynamic image to be a three dimensional function of two spatial arguments and a temporal argument. Then, if this function is locally well behaved and its spatio temporal gradients are computable, the *image velocity* or *optical flow* may be computed [Horn &Schunck 1982, Ullman &Hildreth 1984, Bandopadhay 1984, Nagel 1984, Davis et al 1983, Haralick&Lee 1984]. From now on, we will call this kind of retinal motion *differential*, or *continuous* or *short range*, or *small motion*.

(ii) The second method for measuring image motion considers the cases when the motion is "large" and the first technique is not applicable. In these instances the measurement

technique relies upon isolating and tracking highlights and feature points in the image through time. This entails tackling the correspondence problem that can be difficult in many situations. From now on, we will call this kind of motion *discrete*, or *long range*, or *large motion*.

Thus there is a situation where the motion is differential or "small" as opposed to the case of "large" or discrete motion. The mathematical relations that hold between the image motion and the three dimensional scene parameters are quite different in these two cases. It will be seen later that the information recoverable from these two types of motion is sometimes different. This fact indicates that there is a need for analyzing these two motion cases separately. Furthermore, due to the difference in the computational theory underlying these two motion types, perceptual processes for motion analysis must also be organized in cognizance of this dichotomy. The input to the perceptual process is a two dimensional intensity function which changes with time (image). The image contains two kinds of information, *photometric* and *geometric*. For the purpose of this chapter, only the geometric information is important.

5.2.1 Motion equations under perspective projection

Here we analyze the relation between the retinal motion and the corresponding three-dimensional motion for the case of perspective, under both *small* amd *large* motion.

Considering the familiar perspective projection model and some point P in space whose coordinates are (X,Y,Z) with respect to a fixed inertial frame XYZ (see fig. 5.1). The image of this point is p = (x , y) whose coordinates are given with respect to a xy frame located on the image plane, as it was explained in Chapter 2. The relation between the world point P and the image point p is given by the familiar equation $(x,y)=(FX/Z,FY/Z)$ (5.1),

where 'F' is the focal length of the imaging system. The focal length is assumed to be unity in the following analysis.

Now if a rigid surface moves with a translational velocity $V_T = (U, V, W)$ and a rotational velocity $\Omega = (\alpha, \beta, \gamma)$, then from kinematics, the three- dimensional velocity of any point on the surface can be written as

$$(\frac{dX}{dt}, \frac{dY}{dt}, \frac{dY}{dt}) = V_T + \Omega \times (X,Y,Z) \qquad (5.2)$$

where 't' is the time variable and 'x' denotes vector product.

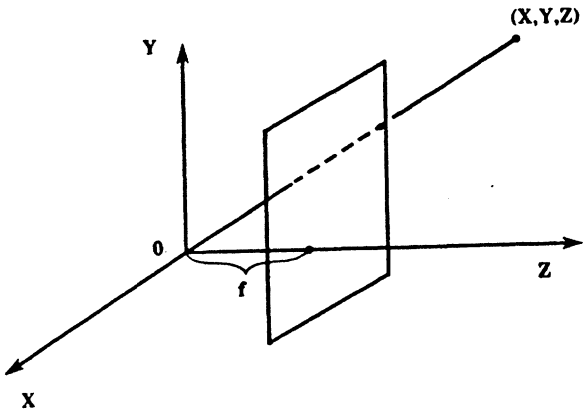


Figure 5.1: Perspective projection

In the differential motion case the image motion or optical flow at the point (x,y) is denoted by (u,v) = (dx/dt , dy/dt) . Differentiating equation (5.1) and substituting from equation (5.2) we have the following relations

$$u = \frac{U - xW}{Z} - \alpha xy + \beta(x^2+1) - \gamma y \qquad (5.3.1)$$

$$v = \frac{V - yW}{Z} - \alpha(y^2+1) + \beta xy + \gamma x \qquad (5.3.2)$$

Eliminating the unknown depth variable from the above we get

$$\frac{u + \alpha xy - \beta(x^2+1) + \gamma y}{v + \alpha(y^2+1) - \beta xy - \gamma x} = \frac{U - xW}{V - yW} \qquad (5.4)$$

The above equation describes the constraint imposed by the measured value of the optical flow (u,v) at an image point (x,y) on the six motion parameters (U,V,W,α,β,γ).

The discrete analogs of equations (5.3) and (5.4) are more complex in form, and they follow in the rest of this section.

Again we consider the pinhole camera model that was described in the former part of this section, and consider one point $P=(X,Y,Z)$ before the motion with image (x,y). Suppose that the point moves with a general motion, and goes to the position $P'=(X',Y',Z')$ with

image (x',y'). It is well known that any three-dimensional rigid body motion is equivalent to a rotation by an angle θ around an axis through the origin with directional cosines n_1,n_2,n_3, followed by a translation $T=(\Delta X,\Delta Y,\Delta Z)^T$. The relation between the coordinates of the point before and after the transformation is given by:

$(X',Y',Z')^T=R(X,Y,Z)^T+T$,

where R is a 3X3 orthonormal matrix of the first kind (i.e. $det(R)=1$)

$$R = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \quad \text{with}$$

$r_1=n_1^2+(1-n_1^2)\cos\theta$, $r_2=n_1n_2(1-\cos\theta)-n_3\sin\theta$, $r_3=n_1n_3(1-\cos\theta)+n_2\sin\theta$

$r_4=n_1n_2(1-\cos\theta)+n_3\sin\theta$, $r_5=n_2^2+(1-n_2^2)\cos\theta$, $r_6=n_1n_3(1-\cos\theta)-n_1\sin\theta$

$r_7=n_1n_3(1-\cos\theta)-n_2\sin\theta$, $r_8=n_2n_3(1-\cos\theta)+n_1\sin\theta$, $r_9=n_3^2+(1-n_3^2)\cos\theta$

Although the elements of R $r_1,r_2,..,r_9$ are complicated functions of the rotation parameters n_1,n_2,n_3,θ, the latter can be easily determined without ambiguity from the former [Tsai and Huang, 1984].Therefore, we can freely talk about the uniqueness and computation of the matrix R, rather than n_1,n_2,n_3, θ.

Taking now into account the perspective projection equations that relate the coordinates X,Y,Z to x,y and the coordinates X',Y',Z' to x',y', we get (assuming that the focal length is unity):

$$x'=\frac{(r_1x+r_2y+r_3)Z+\Delta X}{(r_7x+r_8y+r_9)Z+\Delta Z}$$

and

$$y'=\frac{(r_4x+r_5y+r_6)Z+\Delta X}{(r_7x+r_8y+r_9)Z+\Delta Z}$$

By eliminating the depth Z from the above two equations, we get:

$$[x', y', 1] = E \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where

$$E = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \quad \text{with}$$

$e_1 = \Delta Z r_4 - \Delta Y r_7$, $e_2 = \Delta Z r_5 - \Delta Y r_8$, $e_3 = \Delta Z r_6 - \Delta Y r_9$
$e_4 = \Delta X r_7 - \Delta Z r_1$, $e_5 = \Delta X r_8 - \Delta Z r_2$, $e_6 = \Delta X r_9 - \Delta Z r_3$
$e_7 = \Delta Y r_1 - \Delta X r_4$, $e_8 = \Delta Y r_2 - \Delta X r_5$, $e_9 = \Delta Y r_3 - \Delta X r_6$

The above equation describes the constraint between retinal and three-dimensional motion in the discrete case, and it represents all the information we can get from the motion of one point.

We now develop the same equations for the case of orthographic projection.

5.2.2 Motion equations under orthographic projection.

The projection equation for the case of the orthographic projection, becomes:

$(x,y) = (X,Y)$ (5.6)

Again, following the same method as in the previous section, we get that the optical flow field is given by the following equations:

$u = U + \beta Z - \gamma y$ (5.7.1)

$v = V - \alpha Z + \gamma x$ (5.7.2)

It is obvious that the translation in depth does not affect at all the image motion in this case.

The equations for the discrete case, if developed, basically solve the problem

(structure from motion). For this reason, they will be developed in section 5.6 that discusses the problem of determining structure from motion in the discrete case under orthographic projection.

5.3 Previous work

Several researchers have worked in this area and most of the published papers presented satisfactory results in keeping with assumptions and restrictions made. It is clear by now, that the problem of *Motion Analysis* concerns the *recovery* of the *structure* of the scene (or objects) in view and the *rigid motion parameters* (of the moving viewed object or of the moving sensor), from the perceived changing retinal image. The problem has been studied for both the cases of *differential retinal motion* or *short range motion* (optical flow) and *long range motion* (discrete displacements), under both *orthography* and *perspective*.

In the case of *discrete* motion under *orthography* the pioneering work of Ullman [Ullman 1979,] stands out for being highly precise. In his classical paper on the interpretation of structure from motion, Ullman showed how structure was determined uniquely (up to a reflection) *from the orhographically projected locations of four non-coplanar points, obtained at three different instances of time.* At this point, we should mention in passing that the problem of the interpretation of non-rigid motion (Johansson's "biological motion" [Johansson, 1974]) which we don't address here, was analysed by Hoffman and Flinchbaugh [1980], Hoffman and Bennett [1981], Bennett and Hoffman [1985] and Webb and Aggarwal [1982]. Their analysis is again for orthographic projection and discrete motion, with the additional assumption that the axis of rotation is fixed for the entire period of observation (*i.e.* equivalently, the motion is planar).

In the case of differential motion (optical flow) under orthography, published research is confined to Hoffman [1980] and Sugihara [1985]. Hoffman develops a relationship between optical flow derivatives and local surface orientation and illustrates that if the acceleration of the optical flow field is known, then the computation of shape is feasible. On the other hand, Sugihara presented an involved proof that optical flow under orthography cannot recover local surface orientation, and he developed a method that *using two optical flow fields, the structure of the object in view can be recovered (at most four solutions).* Sugihara, observed that despite the fact that his theoretical analysis predicted four solutions for the structure, given two optical flow fields, in his experiments the solution recovered was unique. So, he developed a conjecture which states that: *two*

optic flow fields uniquely define structure. In this chapter, we prove that *optical flow cannot recover local surface orientation* in a much simpler way, and we give an explanation of Sugihara's conjecture for the discrete case (*i.e.* we prove that *three frames (two optical flow fields means at least three frames) of three points uniquely recover the structure*, except some degenerate cases whose set is of measure zero).

In the case of *differential* motion under *perspective* the relation between the motion parameters and the retinal motion (eq. 5.3) is a nonlinear equation in the direction of the translation and the rotational parameters. The effort of almost all the researchers in this area concentrates on how to solve this non-linear equation, or to formulate the problem in such a way that it becomes tractable. In this case (differential motion under perspective projection), which is a case of great interest, there is a lot of recent work concering the recovery of the *structure* and *motion parameters* of a moving object from its changing retinal image (optical flow). In this area, there is the work of Longuet-Higgins and Prazdny [1982] that developed the relation between the optical flow and the motion parameters as well as the relationship between the gradient of the surface in view and the derivatives of the optical flow. Their method does not guarantee uniqueness of the motion parameters and the algorithm relies on the solution of a non-linear system whose coefficients involve second-order derivatives of the optical flow. Furthermore their method is local, and as we point out in section 5.6.2 every local method is bound to be unstable, because a small patch of optic flow under noisy situations will resemble ambiguous flow fields, and so it cannot recover surface orientation.

Soon researchers in the area realized that Eq. (5.3) (relation of flow to motion parameters), is bilinear in the *direction of translation* and the *rotational parameters*; this means, that Eq. (5.3) is nonlinear in the motion parameters, but if the direction of translation is known, then it becomes linear with respect to the rotational parameters, and *vice versa*. So, efforts have been made to separate the optical flow in its *rotational* and *translational* component, and make the problem of the determination of the motion parameters easy. The work of Prazdny [1984] and Lawton and Rieger [1984] are examples of this type of effort. Both these papers are based essentially on the property of a translational field, that all the flow vectors pass through the same point on the image plane (FOE or FOC); based on that, they develop statistical methods for the computation of the motion parameters. But this is not true in general, i.e. there are cases where the optical flow field vectors pass through the same point,with the motion consisting of both translation and rotation [Bandyopadhyay, 1985]. At this point, we should mention in

passing the work of Bruss and Horn [1984], that considered the cases of only translational or only rotational motion and developed algorithms for the determination of the direction of translation or the rotational parameters.

The work of Huang,Tsai and Fang stands out for being mathematically precise and for being the first to address the uniqueness problem in this area, *i.e.* how many values for the motion parameters are compatible with a flow field -- or two discrete frames. Despite the fact that this work is done in the discrete case under perspective, the results are not far from the differential case under perspective, because they made the assumption that the rotation used is very small. (If small rotation is considered, then the equations that relate the displacements to the motion parameters, are very similar to the analog ones for the differential case.

Fang and Huang [1985], proved that the nonlinear system developed using five points has a unique solution which may be found using iterative methods. They also presented a nine points method, but the condition that has to be satisfied for this method to work, cannot be tested from the image data. Also, Tsai, Huang and Zhu [1984], studied the problem for the case of a moving plane, and they concluded that the number of solutions is either one or two depending on the multiplicities of the singular values of a matrix that contains eight essential parameters. Finally, Tsai and Huang [1985] proved that three views, in the case where the moving surface is planar, guarantee the uniqueness of the motion parameters. Also, in this area there is the work of Kanade [1985] , who inspired by the work of Yen and Huang [1985], developed a method, that using line correspondences in three frames (small rotation), can uniquely find the motion parameters. Experimental results, not known to us at this point, will determine how immune to noise the method is or how well the employed assumptions fit practical situations.

At this point we should mention the work of Waxman [Waxman and Sinha,1985], who, motivated by important psychological and neurobiological experiments by Regan and Beverley [1984] presented a method, termed *Dynamic Stereo*. This method is based on the comparison of image flow fields obtained from two cameras in known relative motion. For stationary objects this technique reduces to conventional motion stereo. Finally, in the case of *discrete motion* under *perspective*, there is little work (arbitrary rotation), only that of Ullman [1977] and Tsai and Huang [1984]. Ullman studied the problem when the rotation is around the z-axis and, using two views of three points he developed a *polar equation* (a fourth degree equation on the sin of the rotation angle) from

which he could determine the motion and structure, but the solution was not unique.

On the other hand, Tsai and Huang, developed a method for the determination of motion parameters in the case of curved surfaces. Using seven points, they prove that the five motion parameters can be uniquely recovered (from the other eight essential parameters), provided that *the seven points do not belong on two planes with one passing through the origin or one a cone containing the origin.* After the eight essential parameters have been computed, the five motion parameters can be uniquely computed using the Singular Value Decomposition technique [Stewart, 1980]

5.4 Criticism of previous work

Before we criticize previous work on the problem of the interpretation of image motion fields, we should classify previous work in some broad categories . There are basically two categories:

1) The first assumes the dynamic image to be a three-dimensional function of two spatial arguments and a temporal argument. Then if this function is locally well - behaved and its spatiotemporal derivatives are computable, the image velocity or optical flow may be computed.

2) The second method for measuring image motion considers the cases where the motion is "large" and the previous technique is not applicable. In these instances the measurement technique relies upon isolating and tracking highlights or feature points in the image through time. In other words operators are applied on both dynamic frames which output a set of points in both images, and then the correspondence problem between these two sets of points has to be solved (i.e. finding which points on both dynamic frames are due to the projection of the same world point).

In both the above approaches, after the optical flow field or the discrete displacements field (which can be sparse) are computed, then algorithms are constructed for the determination of the three-dimensional motion , based on the optical flow or discrete displacements values.

As the problem has been formulated over the years, one camera is used and so the three dimensional motion parameters that have to be computed and can be computed, are five (two for the direction of translation and three for the rotation).

The basic motivation for this research is on one hand the fact that optical flow (or discrete displacement) fields produced from real images by existing techniques are corrupted by noise and are partially incorrect [Ullman, 1983]. So, it is doudtful if retinal motion can be used as input to a three-dimensional motion analysis process. Furthermore, the uniqueness properties of the motion interpretation process, have not yet been examined in detail. As far as computations from retinal motion are concerned, all the algorithms in the litterature that use the retinal motion field to recover three-dimensional motion fail when the input (retinal motion) is noisy. We will address the uniqueness issues at the end of the section. Here we proceed with our criticism of the previous work.

Some researchers [Roach and Aggarwal 1980, Prazdny 19880, Nagel 1981, Nagel and Neumann 1981, Fang and Huang 1983, Fang and Huang, 1984] developed sets of nonlinear equations with the three-dimensional motion parameters as unknowns, which are solved by iterations and initial guessing. These methods are very sensitive to noise, as it is reported in [Roach and Aggarwal 1980, Nagel 1981, Fang and Huang 1984, Fang and Huang 1981]. On the other hand, other researchers [Longuet-Higgins 1981, Tsai and Huang, 1984] developed methods that do not require the solution of nonlinear systems, but the solution of linear ones. Despite that, under the presence of noise, the results are not satisfactory [Longuet-Higgins 1981, Tsai and Huang, 1984].

Bruss and Horn [1984] presented a least-squares formalism that tried to compute the motion parameters by minimizing a measure of the difference between the input optic flow and the predicted one from the motion parameters. The method, in the general case, results in solving a system of nonlinear equations with all the inherent difficulties in such a task, and it seems to have good behavior with respect to noise only when the noise in the optical flow field has a particular distribution. Prazdny, Rieger, and Lawton presented methods based on the separation of the optical flow field in its translational and rotational components, under different assumptions [Prazdny 1981, Rieger and Lawton 1983]. But difficulties are reported with the approach of Prazdny in the present of noise [Jerian and Jain 1983], while the methods of Rieger and Lawton require the presence of occluding boundaries in the scene, something which cannot be guaranteed. Finally, Ullman in his

pioneering work [Ullman, 1977] presented a local analysis, but his approach is sensitive to noise, because of its local nature.

Several other authors [Longuet Higgins and Prazdny 1980, Waxman and Ullman 1983] use the optical flow field and its first and second spatial derivatives at corresponding points to obtain the motion parameters. But these derivatives seem to be unreliable with noise, and there is no known algorithm which can determine them reasonably in real images. Others [Adiv 1984] follow an approach based partially on local interpretation of the flow field, but it can be proved [Ullman 1983] that any local interpretation of the flow field is unstable.

At this point it is worth noting that all the aforementioned methods assume an unrestricted motion (translation and rotation). In the case where prior assumptions are employed or in the case of restricted motion (only translation), there is some good published work. Ballard and Kimball [1983] report a method for measuring three-dimensional motion based on three-dimensional flow. For the case of translational motion, a robust algorithm has been reported by Lawton [1982], which was successfully applied to some real images. His method is based on a global sampling of an error measure that corresponds to the potential position of the focus of expansion (FOE); finally, a local search is required to determine the exact location of the minimum value. However, the method is time-consuming, and is likely to be very sensitive to small rotations. Also the inherent problems of correspondence, in the sense that there may be drop-ins or drop-outs in the two dynamic frames, is not taken into account. All in all, most of the methods presented up to now for the computation of three-dimensional motion depend on the value of flow or retinal displacements. Probably there is no algorithm until now that can compute retinal motion reasonably (for example, 10% accuracy) in real images.

Even if we had some way, however, to compute retinal motion in a reasonable (acceptable) fashion, i.e., with at most an error of 10%, for example, all the algorithms proposed to date that use retinal motion as input, would still produce non-robust results. The reason for this is the fact that the motion constraint (i.e., the relation between three-dimensional motion and retinal displacements) is very sensitive to small perturbations . Table 1 shows how the error of motion parameters grows as the error in image point correspondence increases when 8-point correspondence is used, and Table 2 shows the same relationship when 20-point correspondence is used with 2.5% error on point correspondences based on a recent algorithm of great mathematical elegance [Tsai and Huang, 1984].

(Tables 1 and 2 are from [Tsai and Huang, 1984].)

Table 1: Error of motion parameters for 8-point correspondence for 2.5% error in point correspondence.

Error of E (essential parameters)	**73.91 %**
Error of rotation parameters	**38.70%**
Error of translations	**103.60%**

Table 2: Error of motion parameters for 20-point correspondence for 2.5% error in point correspondence.

Error of E (essential parameters)	19.49%
Error of rotation parameters	2.40%
Error of translations	**29.66%**

It is clear from the above tables that the sensitivity of the algorithm in [Tsai and Huang, 1984] to small errors is very high. It is worth noting at this point that the algorithm in [Tsai and Huang, 1984] is solving linear equations, but the sensitivity to error in point correspondences is not improved with respect to algorithms that solve non-linear equations. Also, it is worth mentioning at this point that the same behaviour is present in the algotithms that compute 3-D motion in the case of planar surfaces .

So, as the problem has been formulated (monocular observer), it seems to have a great deal of difficulty, because of the correspondence problem. This is again not surprising, and the same problem is encountered in many other problems in computer vision (shape from shading, structure from motion, stereo, etc.).

5.5 Motivation for this research and an outline of what is to come

It is by now clear that there are many difficulties with the structure from motion problem. The uniqueness properties of the problem have not yet been discovered, i.e. it is not yet known what kinds of surfaces and motions are amenable to multiple interpretations when our only input is the image motion field.

The next section does the feasibility evaluation of the structure from motion problem. The problem is studied under orthography and perspective for both small (differential) and large (discrete) motion. Theorems are developed concerning what can be computed from the differential or discrete motion field, and under what assumptions. At this point, we should say that we use the image motion field only as an abstraction, i.e. as the information we have about motion, in order to achieve uniqueness proofs. This does not mean that we will develop algorithms for the interpretation of retinal motion that are based on optical flow, because we feel that the retinal motion field cannot be computed in a robust way without prior assumptions.

The section after next, develops algorithms for the solution of the structure from motion problem, without trying to solve the correspondence problem as an intermediate step. Instead it uses novel techniques that do not require correspondence. There has recently been an approach to combine information from different sources in order to achieve uniqueness and robustness of low-level visual computations . With regard to the three-dimensional motion parameters determination problem, why not combine motion information with some other kind of information? It is clear that in this case the constraints won't be the same, and there is some hope for robustness in the computed parameters. As the other kind of information that should be combined with motion, we choose stereo. There are more deep theoretical reasons for combining motion with another cue (depth). The reason is that the constraint between three-dimensional motion and retinal motion when one camera is used is very sensitive to small perturbations; and so, even if we could compute retinal motion with a reasonable accuracy, it wouldn't be enough for computing three-dimensional motion.

The need for combining stereo with motion has recently been appreciated by a number of researchers [Jenkin and Tsotsos 1986, Huang and Blonstein 1985, Waxman and Sinha 1985, Richards 1985]. Jenkin and Tsotsos used stereo information for the computation of retinal motion, and they presented good results for their images. Waxman *et al* presented a promising method for dynamic stereo, which is based on the comparison of image flow fields obtained from cameras in known relative motion, with passive ranging as goal. Whitman Richards is combining stereo disparity with motion in order to recover correct three-dimensional configurations from two-dimensional images (othography-vergence). Finally, Huang and Blonstein presented a method for three-dimensional motion estimation that is based on stereo information. In their work, the static stereo problem as well as the three-dimensional matching problem have to be

solved before the motion estimation problem. The emphasis is placed on the error analysis, since the amount of noise (in typical image resolutions) in the input of the motion estimation algorithm is very large.

So a natural question arises: is it possible to recover three-dimensional motion from images without having to go through the very difficult correspondence problem? And if such a thing is possible, how immune to noise will the algorithm be? In this Chapter, we prove that if we combine stereo and motion in some sense and we avoid any static or dynamic correspondence, then we can compute the three-dimensional motion of a moving object. At this point, it is worth noting recent results by Kanatani [Kanatani 1985] that deal with finding the three-dimensional motion of planar contours in small motion, without point correspondences. These methods seem to suffer from numerical instability a great deal, but they have a great mathematical elegance.

As the problem has been formulated over the years, usually one camera is used and so the 3-D motion parameters that can be computed are five : two for the direction of translation and three for the rotation. If we assume a binocular observer then we can recover six motion parameters : three for the translation and three for the rotation.

With the traditional one camera approach for the estimation of the 3-D motion parameters of a rigid planar patch, it was just mentioned [Roach and Aggarwal, 1980],that one should use the image point correspondences for object points not on a single planar patch when estimating 3-D motions of rigid objects. But it was not known, how many solutions there were, what was the minimum number of points and views needed to assure uniqueness and how could those solutions be computed without using any iterative search (i.e. without having to solve non-linear systems). It was proved [Tsai and Huang 1984] that there are exactly two solutions for the 3-D motion parameters and plane orientations, as we will see later in section 5.6. However, the solutions are unique if three views of the planar patch are given or two views with at least two planar patches. In our approach, the duality problem does not exist for two views, since two cameras are used (and so the analysis is done in 3-D).

The outline of the chapter is as follows: Section 5.6 does a feasibility evaluation of the problem of structure from motion under orthography and perspective. Section 5.7 examines the problem of detecting the three-dimensional motion in the differential case, without using optical flow, but the spatiotemporal derivatives of the flow. Finally, the last section deals with the problem of recovering three-dimensional motion without

correspondence for the case of discrete motion under perspective projection.

5.6 Structure from motion : A feasibility evaluation

Here we study what the constraints are between three-dimensional motion and image motion, as well as what can we compute from two dimensional motion. We first analyze the case where the projection is orthographic.

5.6.1 Structure from motion: the case of orthography

Here we investigate lower bounds in relation to the structure from motion problem , i.e. the minimal number of points from an ensemble of points that move in a rigid configuration and the minimal number of projections that are required to uniquely recover the structure. We show that it is possible to uniquely recover structure from three orthographic projections of three points in space, when a certain condition holds. Furthermore, when this condition does not hold, the number of structures compatible with the motion is at most two.

The interpretation of visual motion by humans and other biological organisms is an exciting field in the study of perception. An issue here is what kinds of mathematical analysis are adequate and lead to a biologically plausible model of computation for the task. Here we examine ways and means by which a perceptual system may be organized to detect the three dimensional structure of rigid objects from their projected motion. The ability of the human visual system to discern structure from motion stimulus was demonstrated by experiments by Wallach and O'Connell in the 1950's. Subsequently Gunnar Johansson discovered our ability to recognize the human form from the projected motion of as few as ten points on the body, such as the various joints like elbows, shoulders and knees.

It would seem that the perception of rigid structure from motion should not require the detection of the projected trajectory of too many points. One of the first rigorous mathematical treatments of this problem was done by S. Ullman [Ullman 1977]. In his classical paper on the computation of structure from motion, Ullman showed how structure was determined uniquely (up to a reflection) from the projected locations of four noncoplanar points, obtained at three different instants of time. His analysis is based on the orthographic projection model. The treatment also considered the correspondence of

the four projected points between the three frames, as available. In our analysis we too work with orthographic projection and assume the point correspondences already given.

While it is true that the perspective or central projection model is more appropriate for image formation in the human visual system, we will argue that orthographic projection is a realistic simplification for this specific problem. One reason is that at small retinal eccentricities perspective effects are small. Another reason is that in Ullman's scheme, as well as ours, only a small number of points are considered at a time and so orthography will serve as an adequate model, because of the locality of the approach.

We should mention in passing that the problem of interpretation of Johansson's "biological motion" was analysed by Hoffman and Flinchbaugh [19811, Hoffman and Bennett [1984] and Webb and Aggarwal [1982]. Their analysis is for orthographic projection with the additional assumption that the axis of rotation is fixed for the entire period of observation (i.e. equivalently, the motion is planar). From the other hand, our analysis does not require the fixed axis assumption.

5.6.1.2 Mathematical formulation **and lower bound arguments**

Consider the Cartesian representation of a point in 3-D space. This is the vector *(X,YJZ)* . A quartet of four such points can be written as $(X_i,Y_{if}ZJ,1=1,2,3,4$. Let these points move and take up new positions $(X'_{i,9}Y'_{if}Zy$. Considering rigidity, we have the fact that the motion can be represented by an affine transformation:

$$(X\backslash,Y\backslash,Z\backslash)^T = R\ (X_x?.,Z)^T + (AXAY,AZ)^T$$

where JR is a 3 by 3 rotation matrix and $(AX,AY,AZ)^r$ is a translation vector. Taking the orthographic projection of the above we have:

$$r=rI.frT.fr,\ Z_i+\Delta X$$
$$Y'=rX.+rY+rZ+t\rangle Y$$

where the elements r^{ij} of the rotation matrix depend upon three independent parameters - the axis of rotation and the angle of rotation about this axis.

Now if we take two views of three points, we obtain six equations in the seven variables - three for the rotation, two depth variables (we have three depths but only relative depth can be recovered) and two for the translation. Thus we cannot solve the problem in this case. A similar argument holds for three views of two points and two

views of four coplanar points. So, according to the above argument, the following theorem has been proved.

THEOREM 5.1: In general it is impossible to recover the structure of

1) Three points, given two orthographic projections of these points,

2) Two points, given three orthographic projections of these points, and

3) Four coplanar points, given two orthographic projections of these points.

In the sequel we are going to prove that *three orthographic projections of three points uniquely recover the structure of these points* . So, given theorem 1, the above results will constitute lower bounds for the problem at hand. Before we proceed, we need constraints between the structure of rigidly moving points and their image displacements. In the next section, we develop these constraints, in *lemmas 5.1* and *5.2* .

5.6.1.3 Mathematical preliminaries.

In this section we develop the constraint that was mentioned in the previous section, in two forms, in *lemmas 1* and *2* .

LEMMA 1 : Given two distinct orthographic projections of three points in a rigid configuration, the gradient (p,q) of the plane that the three points define (with respect to the coordinate system of the first frame), lies on a conic section in the gradient space. The coefficients of this conic section depend entirely on the interframe displacements of the above points. PROOF: Let the three points in space be O, A, B in their first position and O', A', B' in their second position and their projections in the two frames be O_1, A_1, B_1 and O_2, A_2, B_2, respectively (See figure 5.2).

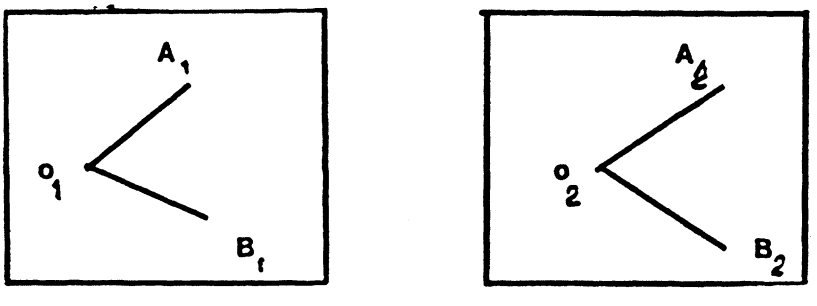


Figure 5.2: Two orthographic projections of three points

Let also the gradient of the plane *OAB* be $G = (p,q)$. Furthermore, let

$\mathbf{O_1A_1 = a_1 = (x_1, y_1)}$

$\mathbf{O_1B_1 = b_1 = (c_1, d_1)}$

$O,A_1 = a, = (x_2, y_2)$

$\mathbf{O_2B_2 = b_2 = (c_2, d_2)}$

Considering the geometry of the first projection (OAB to $O_1A_1B_1$), we have that:

$OA = (x_1, y_l, Ga_1)$ and $OB \wedge c \wedge G \wedge)$, (5.9)

with $^{n}{}''$ the inner vector product operator.

Similarly, considering the second projection (OAB to $O_2A_2B_2$), we get:

$O'A^* = (x_2, y_2, A)$ and $O'B' = (c_2, d_2, p)$ (5.10)

where A and p are to be determined.

But, because of the rigid motion, the vectors **OA** and O^fA^1 have the same length. The same holds for the vectors **OB** and O^fB^f. From these requirements we get:

$$\lambda = \pm(a_1^2 + (Ga_1)^2 - a_2^2)^{1/2}$$

and (5.11)

$$\mu = \pm(b_1^2 + (Gb_1)^2 - b_2^2)^{1/2}$$

Finally, again because of the rigidity, the angles between the vectors OA , **OB** and O^fA^1 , O^fB^f are the same. From this, we get:

$\mathbf{0AOB = 0'A'0'B'}$ (5.12)

where "•" denotes the dot product operation.

Substituting to equation 5.12 from equations 5.9,5.10,5.11 we get:

$$\mathbf{a_1 \cdot a_2 + (Ga_1)(Gb_1) = a_2 \cdot b_2 \pm \lambda\mu}$$

and substituting the values for A and p and squaring appropriately , we get the following equation 5.13:

$$(\mathbf{b}_1^2-\mathbf{b}_2^2)(\mathbf{Ga}_1)^2+(\mathbf{a}_1^2-\mathbf{a}_2^2)(\mathbf{Gb}_1)^2-2(\mathbf{Ga}_1)(\mathbf{Gb}_1)(\mathbf{a}_1\mathbf{b}_1-\mathbf{a}_2\mathbf{b}_2)+(\mathbf{a}_1^2-\mathbf{a}_2^2)(\mathbf{b}_1^2-\mathbf{b}_2^2)-(\mathbf{a}_1\mathbf{b}_1-\mathbf{a}_2b_2$$

Given that

$\mathbf{Ga}_1=\mathbf{p}x_1+\mathbf{q}y_1$ and $\mathbf{Gb}_1=\mathbf{p}c_1+\mathbf{q}d_1$

the above equation is of the form:

$Ap^2+Bq^2+Cpq+D=0$

where the coefficients A,B,C,D depend on the image vectors $\mathbf{a}_1,\mathbf{a}_2,\mathbf{b}_1,\mathbf{b}_2$. (q.e.d.).

We now state and prove a second lemma, that relates the depth differences of the world points with the interframe displacements.

LEMMA 2: Given two distinct orthographic projections of three points O,A,B with depths z_O, z_A, z_B *(with respect to the coordinate system of the first frame), the tuple (* z_1 , z_2 *), with* $z_1 = z_O - z_A$ *and* $z_2 = z_O - z_B$ *, lies on a conic section on the plane* (z_1, z_2) *. The coefficients of this conic depend entirely on the interframe displacements of the above points.*

PROOF: It is obvious that this statement is equivalent to the previous lemma. The reason that we state it, is that we will use this form of the constraint in our subsequent analysis. Using the nomenclature of the previous lemma, we observe that :

$\mathbf{Ga}_1=z_1$ and $\mathbf{Gb}_1=z_2$

and so equation 5.13 becomes:

$$(\mathbf{b}_1^2-\mathbf{b}_2^2){z_1}^2+(\mathbf{a}_1^2-\mathbf{a}_2^2){z_2}^2-2z_1z_2(\mathbf{a}_1\mathbf{b}_1-\mathbf{a}_2\mathbf{b}_2)+(\mathbf{a}_1^2-\mathbf{a}_2^2)(\mathbf{b}_1^2-\mathbf{b}_2^2)-(\mathbf{a}_1\mathbf{b}_1-\mathbf{a}_2\mathbf{b}_2)^2=0 \quad (5.14)$$

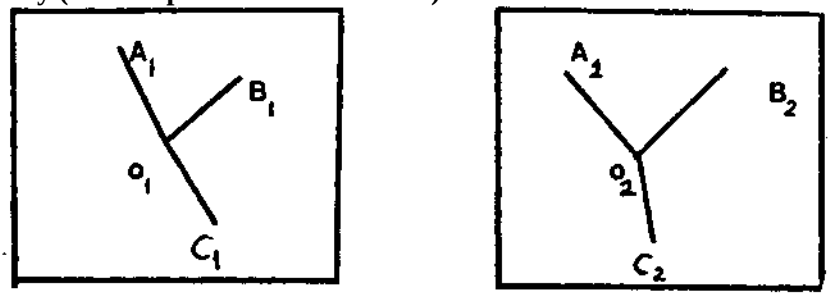
The above equation (5.14) proves the claim. The above lemmas relate the structure (shape) of three points with their two distinct orthographic projections. Whether the points move or the projection plane moves (moving observer) or both of them move, the analysis remains the same. We will now state and prove the theorems pertaining to lower bound results in the recovery of structure from motion.

5.6.1.4 Lower bound results

So far, we have established the fact that two orthographic views of less than four points cannot recover the structure of these points. We now show that if the number of points is four, structure can be determined.

THEOREM 3: Two orthographic projections of four rigidly linked noncoplanar points are compatible with infinite interpretations of their relative 3-D positions, in general. Adding a third view yields a unique interpretation of the structure of the four points.

PROOF: Let the four points in space be O,A,B_9C. Let also the projections of the four points in the two frames be $O_{lf}A_{1\$}B_{lt}$ C_2 and O_2,A_2,B_2, C_2 respectively (See Fig. 5.3), and the gradients of the planes OAB, OBC and OCA be $G_t=(p_{l9}q_t)_9$ $G_f=(p_{2f}q_2)$ and $G_3=(p_3,qJ$ respectively (with respect to the first frame).



frame 1 frame 2

Figure 5.3: Two projections of four noncoplanar points

Using the projections O_tA_t ,O_JB_l and their corresponding ones O_2A_2 , O_2B_2 and utilizing *lemma 1* we get :

$$A_1p_1^2+B_1q_1^2+C_1p_1q_1+D_1=0 \quad (5.15)$$

where the coefficients depend entirely on the image vectors. Similarly, considering the projections O_JB_l and O_1C_1 and their corresponding ones in the second frame and the projections O_2C_1 and O_JA_1 and their corresponding ones in the second frame, we get:

$$A_2p_2^2+B_2q_2^2+C_2p_2q_2+D_2=0 \quad (5.16)$$

$$A_3p_3^2+B_3q_3^2+C_3p_3q_3+D_3=0 \quad (5.17)$$

At this point we should say that the above equations seem independent because they come from the rigidity of the three rods OA, OB, OC. In other words the fact that the three lengths OA, OB, and OC in space remain constant and the two angles AOB and BOC in space remain constant between the two frames, does not imply that the third angle COA will remain the same.

Proceeding, we note that we have more information about the gradients G_1 , G_2 , G_3 from the well known Mackworth constraints that they state:

$$\mathbf{G_1 \cdot O_1B_1 = G_2 \cdot O_1B_1}$$
$$\mathbf{G_2 \cdot O_1C_1 = G_3 \cdot O_1C_1} \quad (5.18)$$
$$\mathbf{G_3 \cdot O_1A_1 = G_1 \cdot O_1A_1}$$

The above equations 5.15-5.18 constitute a system of six equations in the six uknowns $p_1, q_1, p_2, q_2, p_3, q_3$. Before we proceed with a rigorous proof, we shed some light on the form and information content of the equations 5.15-5.18. Equations 5.18 simply express the fact that the gradients $\mathbf{G_1}$,$\mathbf{G_2}$,$\mathbf{G_3}$ of the three planes make a triangle the direction of whose sides are known, but we don't know its position and its scaling. On the other hand, equations 5.15, 5.16 and 5.17 state that each of the gradients $\mathbf{G_1}$,$\mathbf{G_2}$,$\mathbf{G_3}$ lies on a conic section in gradient space. So, in order to solve the problem (i.e. to find the three gradients) we have to put a triangle on gradient space, such that its sides have the orientation defined by the Mackworth constraints (equations 5.18) and each one of its vertices lies on each one of the three conic sections (defined by equations 5.15, 5.16 and 5.17). At this point we should say, that several important problems in Vision Processing have been solved in a very similar way. Horn (Horn, 1977) solved the problem of determining the shape of a polyhedral object from intensity information and the Mackworth constraints, and Kanade (1982) solved the same problem (shape of polyhedral objects) but using skewed symmetry and the Mackworth constraints .

The simple fact that we have six equations in six unknowns here does not mean that this system will have a finite number of solutions. To find out if there are a finite number of solutions we apply the inverse function theorem. This theorem allows us to conclude that whenever the Jacobian of these equations is nonsingular, the mapping defined by these equations is locally one to one and onto. Hence, any roots at points where the Jacobian is nonsingular are isolated and not part of a continuum of solutions.

It is a simple exercise to compute the Jacobian of the above system and prove that in general it has rank less than six . (*One has to be careful when determining the rank of the Jacobian; all the coefficients have to be expressed in the image coordinates, otherwise hidden dependencies may cause problems. The degenerate cases can be easily found by factoring the determinant of the Jacobian*). Consequently we can assert that the system has an ifinite number of solutions.

To conclude the proof of the theorem, if we add one more view, then the solution is unique, and the proof is immediate from the "Structure from Motion" theorem, by S. Ullman (Ullman, S.,1979). (q.e.d).

We now proceed with our second theorem.

THEOREM 4: Three orthographic projections of three rigidly linked points are compatible with at most one interpretation (plus reflection) of their relative 3-D positions, in general. Furthermore, when a certain testable condition holds then there are at most two interpretations (plus reflections). Adding a fourth view yields a unique interpretation of the structure of the four points.

PROOF : Let the three points in space be O,A,B with depths z_O , z_A , z_B (with respect to the coordinate system of the first view), and their projections on the three frames be O_i , A_i , B_i for $i=$ *1,2,3* respectively (See Fig. 5.5).

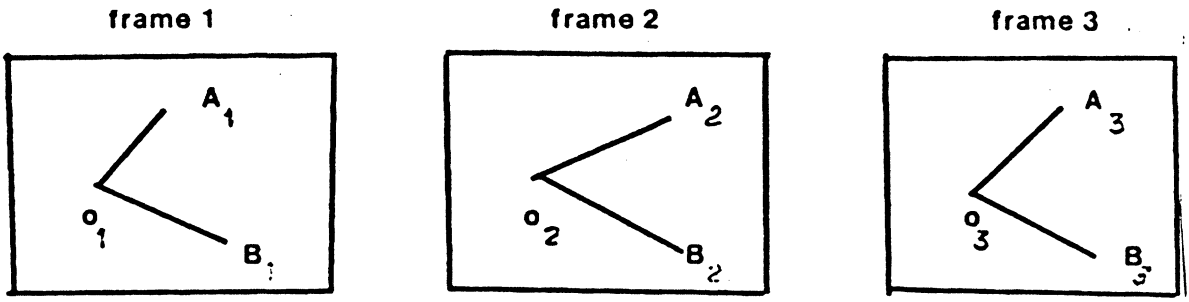


Figure 5.5: Three projections of three points

Let also :

$z_1 = z_O - z_A$

$z_2 = z_O - z_B$

$O_iA_i = a_i$

$O_iB_i = b_i$

Now applying *lemma 2* for frames 1 and 2 and then for frames 1 and 3 we get the following system Σ of equations :

$$(b_1^2 - b_2^2)z_1^2 + (a_1^2 - a_2^2)z_2^2 - 2z_1z_2(a_1b_1 - a_2b_2) + (a_1^2 - a_2^2)(b_1^2 - b_2^2) - (a_1b_1 - a_2b_2)^2 = 0$$

$$(b_1^2 - b_3^2)z_1^2 + (a_1^2 - a_3^2)z_2^2 - 2z_1z_2(a_1b_1 - a_3b_3) + (a_1^2 - a_3^2)(b_1^2 - b_3^2) - (a_1b_1 - a_3b_3)^2 = 0$$

The above equations constitute a system of two equations in the two uknowns z_1 , z_2 . The Jacobian of this system has rank two in general, and so by applying the inverse function theorem we conclude that the system has finite solutions. Using Bezout's theorem we conclude that the system has at most four solutions. (Actually two solutions, plus the Necker reflections). In the sequel we prove that in general the above system has a unique solution (plus reflection).

After eliminating the constant terms from the above equations we get :

$(K_2N_1 - K_1N_2)Z_1^2 + (M_2N_1 - M_1N_2)z_1z_2 + (L_2N_1 - L_1N_2)z_2^2 = 0$ *(5.22)*

with

$K_1 = b_1^2 - b_2^2$

$K_2 = b_1^2 - b_3^2$

$L_1 = a_1^2 - a_2^2$

$L_2 = a_1^2 - a_3^2$

$M_1 = -2(a_1b_1 - a_2b_2)$

$M_2 = -2(a_1b_1 - a_3b_3)$

$$N_1 = K_1L_1 - \frac{M_1^2}{4}$$

$$N_2 = K_2L_2 - \frac{M_2^2}{4}$$

Equation (5.22) is homogeneous in z_1 , z_2 and by dividing with z_2^2 and setting $z_1/z_2 = x$ we get the following equation:

$$(K_2N_1-K_1N_2)x^2+(M_2N_1-M_1N_2)x+(L_2N_1-L_1N_2)=O \qquad (5.23)$$

The solution of the above equation (30) is given by:

$$x=\frac{-(M_2N_1 - \text{MJTVJJ}) \pm (Disc)^m}{K_2N_1-K_1N_2}$$

where *Disc* is the discriminant of equation (5.23).

From the other hand, if the length of the vectors $OA_f OB$ is p and p respectively, then from the geometry of the projection on the first frame, it is obvious that:

$$z_1=\pm\sqrt{\rho^2-a_1^2}$$

and

$$2_2=\pm \text{V}\,\text{u}^2 - 6_1^2$$

Consequently,

$$x=\pm\frac{\text{V}\,\text{u}^2 - a_1^2}{\sqrt{\rho^2-b_1^2}}$$

Thus, if x has two solutions then these solutions must have the same absolute value and opposite sign if both are to be valid. From (5.23) we conclude that x will have two valid solutions if:

$$M_1N_2=M_2N_1 \qquad (5.24)$$

Obviously the above condition (5.24) is a testable condition in the image data. So far, we have concluded that if condition (5.24) holds, then the problem has two solutions (plus reflections), because then there will be two solutions for x^$zjz_{\frac{1}{2}}$ and so four solutions for $(z_{l9}z$^ (actually two solutions, plus reflections). If condition (5.24) does not hold, then there is only one solution for x and consequently two solutions for (z_t, z_2) (actually one solution, plus reflection).

In addition, the above description can be used to actually find the structure of three points from three projections, by developing equation (5.23), solve for x and then use this value in conjunction with the equations of the abovementioned system E, to solve for z_L, z_2 rejecting the imaginary roots.

Finally,to conclude the proof we have to prove that if we add one more view, then we get a unique result. If we call O_4, A_4, B_4 the projections in the fourth view, and let O_4A_4

$= \mathbf{a}_4$ and $\mathbf{O}_4\mathbf{B}_4 = \mathbf{b}_4$, then considering the first and the fourth frame we get the equation:

$$(\mathbf{b}_1^2 - \mathbf{b}_4^2)z_1^2 + (\mathbf{a}_1^2 - \mathbf{a}_4^2)z_2^2 - 2z_1z_2(\mathbf{a}_1\mathbf{b}_1 - \mathbf{a}_4\mathbf{b}_4) + (\mathbf{a}_1^2 - \mathbf{a}_4^2)(\mathbf{b}_1^2 - \mathbf{b}_4^2) - (\mathbf{a}_1\mathbf{b}_1 - \mathbf{a}_4\mathbf{b}_4)^2 = 0 \quad (5.24)$$

Equations of the system Σ and (5.25) constitute a system of three equations with two uknowns. So, this system, barring degeneracy will have at most one solution. This concludes the proof of the theorem.

The rest of this Chapter discusses the problem in the differential case.

5.6.1.5 The Constraint Induced by the Discrete Displacements Field

Consider a moving surface $z = z(x, y)$ and let $(\Delta u(x, y), \Delta v(x, y))$, for all (x, y) on the image, be the discrete displacements field for two time instances t_1 and t_2 with $t_1 < t_2$, i.e., if an image point is at the position (x, y) at time t_1, then at time t_2 it will be at the position $(x + \Delta u(x, y), y + \Delta v(x, y))$. Then, from the previous theorems (see also 8) it can be proved (and actually it has been proved in Chapter 4) that the gradient (p, q), at a surface point whose projection on the image plane is the point (x, y), satisfies the following conic constraint:

$$k_1 p^2 + k_2 q^2 - 2k_3 pq + k_4 = 0 \quad (5.26)$$

with

$$k_1 = [\Delta^y u(x, y)]^2 + [\Delta^y v(x, y)]^2 + 2\Delta^y u(x, y)$$

$$k_2 = [\Delta^x u(x, y)]^2 + [\Delta^x v(x, y)]^2 + 2\Delta^x u(x, y)$$

$$k_3 = \Delta^y u(x, y) + \Delta^y u(x, y)\,\Delta^x u(x, y) + \Delta^x v(x, y) + \Delta^x v(x, y)\Delta^y v(x, y)$$

$$k_4 = k_3^2 - k_1 k_2$$

where

$$\Delta^x u(x, y) = \Delta u(x+1, y) - \Delta u(x, y)$$

$$\Delta^y u(x, y) = \Delta u(x, y+1) - \Delta u(x, y)$$

$$\Delta^x v(x,y) = \Delta v(x+1,y) - \Delta v(x,y)$$

$$\Delta^y v(x,y) = \Delta v(x,y+1) - \Delta v(x,y)$$

Equation (5.26) has been used with smoothness and boundary conditions to orientation from a dense discrete displacements field. The point that we want to stress in this section is that the discrete displacements oblige the gradient (p, q) to lie on a conic section in general (Eq. 5.26).

We now proceed with the analysis in the differential case.

6.6.1.6 The Differential Case.

Here we treat the case where the optic flow field is given, i.e., the 2-D velocities of the image points. Let a surface z = z(x, y) translate with translation (u, v, w) and rotate with rotation (A, B, C) around an axis passing through the origin of a fixed coordinate system, whose z-axis serves as the optical axis and image plane is perpendicular to the z-axis. Then, the optic flow field induced by the motion of the surface z = z(x, y) is given by:

$$u(x,y) = u + Bz - Cy \quad (5.27)$$

and

$$v(x,y) = v + Cx - Az \quad (5.28)$$

with (x, y) the image coordinates (same as the world coordinates). If we consider a surface z' = λz(x, y) moving with translation (u, v, w) and rotation (A/λ, B/λ, C), then the optical flow field induced by this motion of z' is identical to the previous one (Eqs. 35, 36) induced by the motion of the surface z. But the shapes of the surfaces z and z' are different under orthography. (Knowledge of the shape under orthography means knowledge of the depth difference corresponding to any two image points. In other words, under orthography, if we know the shape of an object in view, then we know exactly the object, but we don't know its depth.) Clearly, the surfaces z and z' have different shapes, but under the above described motions they induce the same optical flow. Since the choice of λ was arbitrary, we conclude that there are infinite surfaces with different shapes that induce the same optic flow field when moving with appropriate motions. So, we have proved the following theorem

THEOREM 4: Under orthography, optical flow cannot recover surface orientation. We now move to discover the relationship between the flow field and the surface gradient. Differentiating the flow field (Eqs. 5.27, 5.28) with respect to x and y, we get:

$$\frac{\partial u}{\partial x} = Bp \qquad (5.29.1)$$

$$\frac{\partial v}{\partial x} = C - Ap \qquad (5.29.2)$$

$$\frac{\partial u}{\partial y} = Bq - C \qquad (5.29.3)$$

$$\frac{\partial v}{\partial x} = Aq \qquad (5.29.4)$$

with(p, q) the gradient of the surface in view at the point (x, y) (i.e., $p = \partial z/\partial x$, $q = \partial z/\partial y$).

It turns out from the system of Equations (5.29) that the quantities C, p/q, and A/B can be computed. In particular, we get:

$$c_{1,2} = \frac{\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \pm \sqrt{\left(\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2 - 4\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}\right)}}{2} \qquad (5.30)$$

and

$$\frac{p}{q} = \frac{\frac{\partial u}{\partial x}}{\frac{\partial u}{\partial y} + c_{1,2}} \qquad (5.31)$$

So, we have proved the following theorem.

THEOREM 5: The optic flow field, at every point (x, y) of an image under orthography, constrains the gradient (p, q) of the surface point whose image is the point (x, y) to lie on one of two straight lines that pass through the origin of the gradient space (e.g., (5.31)).

Under closer examination, we can prove (and it has actually been proved by Ullman) that for nonplanar surfaces the value p/q has a unique solution, whereas for planar surfaces, the value p/q has two distinct solutions. For a proof different than Ullman's, see [Aloimonos, 1985,]. So, the following theorem has been established.

THEOREM 6: The orthographic optic flow at every point (x, y) of an image constrains the gradient (p, q) of the surface (nonplanar) point whose image is the point (x, y) to lie on a straight line passing through the origin of the gradient space.

Up to this point we have established that the discrete displacements field constrains the gradient (p, q) of the surface in view at every point to lie on a conic section (hyperbola, parabola, ellipse) in general, whereas the optic flow field constrains the gradient (p, q) to lie on a degenerate conic section (two straight lines, or one straight line). Despite the difference in the constraints, the content in terms of the structure from motion problem, is the same.

5.6.1.8. Discussion and Conclusion (Motion under orthography)

The perception of rigid structure from motion stimulus is well within the competence level of the human visual system.

Our results fill an important gap in the study of the perception of structure from motion--showing the limitation of the approach.

We believe that our work forms an important extension to Ullman's theory and, in conjunction with interpretation schemes for recovering structure in the case of biological motion using the planarity (or fixed axis) assumption, constitutes a significant advance in the problem of the interpretation of structure from motion.

Concluding, we would like to propose a research problem, with which we have had some success up to now. Given the constraint in section 5, surface smoothness and boundary conditions, under what assumptions do we have a unique solution for the surface structure? Our investigation shows that methods similar to the ones used by A. Bruss [Bruss, 1982] for the shape-from-shading problem are very fruitful.

At this point we have concluded our analysis of the structure from motion under orthographic projection.

5.6.2 Structure from motion: the case of perspective

Here we do a feasibility evaluation of the computation of three-dimensional motion, under perspective projection. We will only study the case of differential motion. The analysis in the case of discrete motion has been done fully by Longuet-Higgins [Longuet-

Hoggins, 1981] and Tsai and Huang [Tsai and Huang, 1984], and can be summarized in the following propositions:

Proposition 1: Given the image correspondences of four points that lie on a plane that moves rigidly, the motion parameters are computable and there can be at most two solutions. In paricular, if the motion can be realized by rotationg the object around the origin and then translating it along the normal direction of the plane's surface, then the motion parameters are unique, otherwise there are exactly two solutions.

Proposition 2: Given the image point correspondences of two planes not passing through the origin (lens center), the motion is unique.

Proposition 3: The image point correspondences of six points, with four points on one plane not containing the origin, and two points common to the above two groups of four points on the intersection of the two planes, ensure unique solutions for the motion parameters.

Proposition 4: The image correspondences of four points on a plane not passing through the origin and two other points not on this plane, determine the motion parameters uniquely.

Proposition 5: Given the image correspondences of seven or more points not traversable by two planes with one plane containing the origin, nor by a cone containing the origin, the motion parameters are unique.

We now move to the differential case that has drawn a lot of attention during the past few years.

5.6.2.1 Introduction

Following the model introduced in section 5.2.1, we have that if the camera is moving with translation T = (U,V,W) and rotation Q = (A,B,C), then the optical flow (u,v) at a point (x,y) is given by:

$$u = \frac{U - xW}{Z} - ary + \beta(x^2+1) - \gamma y$$

$$v = \frac{V - yW}{Z} - a(y^2+1) + \{3xy + \backslash x$$

where Z is the depth of the imaged surface point, whose image is the point (x,y).The question that arises then is whether we can compute the three-dimensional motion from the flow field. Is there place for ambiguity? In other words, are there different surfaces

and corresponding motions that willl produce the same optic flow field and so no matter what algorithm we use, we will never be able to recover the actual three-dimensional motion in this case?

Before we proceed with our analysis, we must say that all the published approaches that a purely local analysis of the flow field will never succeed. In a sufficiently small patch, given the noise in the real data, the estimated motion field will not be distinguishable from one resulting from surfaces for which there is no unique solution, as we will see in the rest of this section.

5.6.2.2 Uniqueness analysis of flow fields

It is an important question in motion research whether a given optic flow field could be due to the different motions of different surfaces. Research in the field has shown that this is true for planar surfaces [Maybank, 1984, Subbarao and Waxman, 1985]. It is very important to discover what kinds of surfaces are bound to ambiguity, because if their set is very rich, then we should reconsider many of the published theories. In what follows,

because of the fact that the equations become very complicated, we will use vector notation, to keep the equations neat. Considering the traditional camera model, with focal length =1, a world point is denoted by $\mathbf{P}=(X,Y,Z)^T$ and its projection on the image plane by

$\mathbf{p}=(x,y,1)^T$. Then the perspective projection equation becomes

$$\mathbf{p}=\frac{1}{\mathbf{P}.\mathbf{k}}\mathbf{P}$$

where **k** the unit vector along the Z axis. If the camera is moving with translational velocity $T=(U,V,W)$ and rotational $\Omega=(A,B,C)$. then in order to find the flow field we must differentiate the above equation with respect to time t, and take into account that

$$\frac{d\mathbf{P}}{dt}=-T-\mathbf{P}X\Omega$$

and from that we get:

$$\frac{d\mathbf{P}}{dt}=\frac{1}{\mathbf{P}.k}\left((T.k)p-T\right)+(pX\Omega).p-pX\Omega$$

where "X" is the cross product vector operation. This is a vector equation, which when expanded to its components, will yield equations (5.3).

5.6.2.3 Finding surfaces that yield identical motion fields

Suppose that we have a surface $Z_1(x,y)$ moving with motion (T_1,Ω_1) and a surface $Z_2(x,y)$ moving with motion (T_2,Ω_2) . Suppose further that these surfaces yield identical motion fields, i.e.

$$\frac{1}{z_1}\left((T_1.k)p-T_1\right)+(pX\Omega_1).kp-pX\Omega_1=\frac{1}{z_2}\left((T_2.k)p-T_2\right)+(pX\Omega_2).kp-pX\Omega_2$$

or

$$\frac{1}{z_1}\left((T_1.k)p-T_1\right)-\frac{1}{z_2}\left((T_2.k)p-T_2\right)=(pX\Delta\Omega).kp-pX\Delta\Omega$$

with $\Delta\Omega=\Omega_1-\Omega_2$. So we see that only the difference of the rotational velocities matters. This has been observed by Bandyopadyhay too [Bandyoapadhay, 1986].

Using the above equation we can discover the surfaces that give rise to ambiguous flow fields . If the motion is only translational or only rotational, or it is general but we know either the rotation or the translation, then we can uniquely recover three-dimensional motion from flow fields, i.e. the motion fields cannot be ambiguous. A proof of this is easy, and the interested reader is referred to [Bruss and Horn, 1984, Bandyopadhyay, 1986].

Assuming that the translations are nonzero, we proceed with the general case:
If we solve the above vector equation for Z_t and Z_2 we get:

$$\frac{1}{{}^{2}l}(T_2 X T_1)\mathbf{p} + (\mathbf{p} X \Delta\Omega)(T_2 X \mathbf{p}) = 0$$

and

$$\frac{1}{z2}(T_2 X T_1)\mathbf{p} + (\mathbf{p} X \Delta\Omega)(T1 X \mathbf{p}) = 0$$

These equations give the surfaces in terms of retinal coordinates. From these, it is clear that the depth function of surfaces with ambiguous motion fields, when expressed in retinal coordinates it is the ratio of a first order polynomial over a second order polynomial. This was known to Bandyopadhyay [Bandyopadhyay, 1985]. Here we go further to get the description in terms of three-dimensional coordinates. For this to happen, we must express the retinal coordinates in terms of the three-dimensional coordinates from the perspective projection equation. We have that,

$$\mathbf{p} = \frac{1}{z_1}\mathbf{P}_1$$

and

$$\mathbf{p} = \frac{1}{z_2^2}\mathbf{p}$$

If we substitute in the above equations we get:

$$(T_2 X T)P + (P X \Delta\Omega)(T) = 0$$

and

$$(TJCTJP_2 + (P_2 X \Delta\Omega)(T_2 X P_2) = 0$$

Obviously the above equations, when expressed in coordinates X,Y, Z are of second order, So, up to now we know that only planes and quadric surfaces can create motion fields that are ambiguous. It remains to be investigated what kinds of second order surfaces are ambiguous.

5.6*2.4 What kinds of quadrics are ambiguous

In the previous section we proved that only quadric surfaces (with the exception of planes) are candidates for ambiguous flow fields. The question that arises then is : is any quadric surface problematic in this matter, or only paricular kinds of second order surfaces.

We know that second order surfaces are ellipsoids, hyperboloids, paraboloids and quadric cones. Here we prove that only hyperboloid surfaces are candidates for ambiguous interpretation.

From the equation of a quadric, we know of two methods that can be used to determine the kind of the quadric. One has to do with the signs of several expressions such as the determinant of the matrix of the coefficients of the quadric, and the other with the eigenvalues of the matrix. Here we choose the second, because it is simpler to implement.

The equation of a quadric that passes through the origin, as in our case, can be written in the form $\mathbf{x}^T\mathbf{A}\mathbf{x}+\mathbf{B}\mathbf{x}=\mathbf{0}$, where $\mathbf{x}=(x,y,z)^T$ and A and **B** 3X3 matrices. If we change coordinate systems and we move the origin to the center of the quadric, then we get rid of the linear terms and the equation becomes: $\mathbf{x}^T\mathbf{A}\mathbf{x}=b$, with b a constant and **x** the new coordinates. Then from the eigenvalues of the matrix A we may decide about the kind of the quadric.

Recalling from the previous section, the equation of the pathological quadric is :

$$(T_2 X T_1)P_1+(P_1 X \Delta\Omega)(T_1 X P_1)=0$$

or

$$(T_2 X T_1)P_1+(P_1 T_2)(\Delta\Omega P_1)-(r_2\ \Delta\Omega)(P_1 P_1)=0$$

or

$$(T_2 X T_1)P_1+P_1^T(T_2\Delta\Omega^T)P_1-(T_2\ \Delta\Omega)P_1^T P_1=0$$

or

$$2(T_2XT_1)P_1 + P_1^T(T_2\Delta\Omega^T + \Delta\Omega T_2^T - 2(T_2\Delta\Omega)I)P_1 = 0$$

with I the 3X3 identity matrix. So, the quadric equation can be written as

$P_1^T AP_1 + 2(T_2XT_1)P_1 = 0$

with

$$A = T_2\Delta\Omega^T + \Delta\Omega T_2^T - 2(T_2\Delta\Omega)I$$

If we transform the coordinate system center to the center of the quadric, we get as coefficient matrix the matrix A. Its eigenvalues are:

$\lambda_1 = -T_2.\Delta\Omega + \|T_2\|\|\Delta\Omega\|$

$\lambda_2 = -T_2.\Delta\Omega - \|T_2\|\|\Delta\Omega\|$

$\lambda_3 = -2(T_2.\Delta\Omega)$

Obviously λ_1 is positive and λ_2 is negative.So, the quadric is a hyperboloid.

At this point we should mention that research in this area has been done by Bandyopadhyay, who obtained the result that the problematic surface is the ratio of two polynomials of degree one and two [Bandyopadhyay, 1986]. In the sequel we describe two theorems concerning the equivalence of stucture and motion.

Proposition: If the three-dimensional motion parameters are known, then the structure of the object in view is uniquely determined by the optic flow field.

Proof: Immediate from equations 5.3.1 and 5.3.2

Proposition: If the structure of the object in view is known, then the three-dimensional motion parameters are uniquely determined from the optic flow field.

Proof: See [Bandyopadhyay and Aloimonos, 1985]

5.7 Algorithms for Motion perception

Here we study ways and means for computing the three-dimensional structure and motion of a moving object from a sequence of its images. Because of the fact that the problem of finding structure and the problem of finding three-dimensional motion are related as we already stated in section 5.6.2, in the sense that the knowledge of the one greatly simplifies the other, in this section we will only study the problem of determining three-dimensional motion. We will present algorithms that do not depend on finding first the correspondence between points in the sequence of images, but they recover the three-dimensional motion without using any correspondence. We will study the problem under

both the differential (continuous or small motion) and the discrete (apparent or large motion) case.

The next section describes the problems with any approach that utilizes local motion (point correspondences or optical flow).

5.7.1 Optical flow or discrete displacements: Can we compute them?

Extensive research in dynamic scene analysis has shown that the computation of retinal motion is very hard. Let us first address the problem of finding optic flow (retinal motion in the differential case). Suppose that the camera is moving (or the imaged object is moving). Then, the image intensity function f is a function of three arguments (space - position in the image (x,y), and time - t). If at time t the velocity of an image point (x,y) is (u,v), then it can be easily proved [Horn and Schunck, 1982], that it obeys the following relation:

$f_x u + f_y v + f_t = 0$, where f_x f_y f_t the spatiotemoral derivatives of the intensity function. From now on we will call the above equation, *image flow equation.* This relation is the only information we can have about the image velocity at the point *(x,y)*. Obviously, we need two parameters (u,v), but we only have one equation. So, without other assumptions, we cannot compute the optic flow (u,v). Despite that, several methods have been proposed for the computation of flow, which belong basically in the regularization paradigm. In other words, the optic flow field is assumed to be smooth, and this introduces additional constraints that may reduce the solution space to a unique point. It is obvious however, that the optical flow fields are not smooth in most of the situations, and it is the discontinuities of the flow field that are of some interest, since they contain information about the structure discontinuities of the surface in view. Restrictive assumptions about the flow filed (smooth), cannot lead to methods that will work satisfactorily in a variety of situations. From this discussion, we conclude that the optical flow field cannot be measured, and eventhough a large part of todays research is devoted to the computation of this optical flow field, leading researchers in the field are starting to realize that computation of optical flow is a *utopia* [Horn, 1986].

So, in the case of differential motion, the only information that we can have about retinal motion, is the spatiotemporal derivatives of the image intensity function, f_x, f_y, f_t. At this point, we conclude our discussion about the feasibility of the computation of optical flow.

Moving now to the problem of the feasibility of the computation of discrete displacements, we have to say that this problem can probably be solved, in contrast with the problem of computing optical flow. This case has to do with apparent (large motion). Suppose that the point (x,y) is the image of the three-dimensional point (X,Y,Z). Suppose also that a general motion occurs and the point (X,Y,Z) moves to the position (X',Y',Z') with the new image (x\y'). The retinal motion that we observe, and which can be our only input, is the motion of the point (x,y) to the position (x',y*). Let us call the point (x,y) ***point before the motion*** **and point (x\y')** ***point after the motion.*** **The discrete displacement from this motion is the vector (x'-x,y'-y). Now, suppose that we have many point on the image plane before the motion, say (x_i,y_i), i=l,..n. These points are the projections of texture markings on the three-dimensional object. If a motion occurs, then the three-dimensional markings on the object move rigidly, and after the motion their new projections are the points (x'_i,y'_i), i = l,2,..,n. Now, in order to find the retinal motion, we have to find to which point after the motion every point before the motion corresponds. This is known in the literatture as the** ***Correspondence problem.*** **There have been several approaches towards the solution of the correspondence problem. These can basically be classified in the following categories.**

1) ***Minimum distance criterion*** **[Ullman, 1977, Nagel, 1984]**

2) ***Matching contours*** **[Hildreth, 1984, Waxman and Wohn, 1985]**

3) ***Similarity measures and relaxation*** **[Prager and Arbib, 1984, Horn and Schunck, 1982, Barnard and Thompson, 1981]**

4) Clustering [Bandyopadhyay, 1986].

In what follows we will discuss and criticize each approach .

The minimum distance criterion methods, are basically based on the heuristic that a point before the motion will be matched with the point after the motion that is nearest to it. This approach, that Ullman uses in a global criterion, would work if the points before and after the motion were very sparse and the motion was relatively not large. Then we could say that the nearest point will be the corresponding one. Unfortunately, the points before and after the motion about which we are discussing, do not come automatically. They have to be extracted from the sequence of the intensity images. And eventhough there exist several methods for their extraction, no one of them is perfect, in the sense that there will exist points in the first dynamic frame whose corresponding one will no be there in the second dynamic frame and vice versa. This of course is due to the

inaccuracies of the methods, and to the unpredictability of the natural images. But even if we could extract the points in a accurate way, no one guarantees that they will be sparse so that the minimum distance criterion can be applied. The points that are extracted are interesting points (corners, high curvature points, etc), and their sparseness depends solely on the imaged scene. From the other hand, the minimum distance criterion is a heuristic that will be true only under certain kinds of motions and surfaces.

The method of matching contour points seems very promising and there is good work in this area by Hildreth and Waxman. Points along contours are matched, and the apperture problem is addressed by relaxing the results along the contours. Of course the results will be reasonable if the contours under consideration are smooth. And even though several contours in natural images are smooth, we cannot know this apriori. Furthermore, to extract the contours seems a hard problem, eventhough there are some new results that show great promise. Finally, what do we do if there are no contours in the image?

The methods that are based on the similarity measures, follow the heuristic that nearby points will have similar displacements, i.e. the difference of their displacement vectors will be in some small interval. Several methods have been proposed in this line of thought, and they are basically based on iterative relaxation methods, which work with the hope that the system will converge to a correct solution. Of course, there are no results on the convergence and the uniqueness of the relaxation computations and more importantly the heuristic of the similarity of the nearby displacement vectors is correct only for smooth surfaces, something that the surfaces of our visual world do not follow.

Finally, recently clustering methods have been proposed. The clustering methods are again based on the fact that nearby displacements will have similar values, and so they form a cluster. The clustering is done in a two-dimensional space (only displacement values), instead of a four-dimensional (dispacement values and position on the image plane), for efficiency reasons. Even though the clustering methods are based on similarity measures and so our criticism in the previous paragraph applies here too, the method is new and more experimental and theoretical work is required in order to get more information about the method.

Up to this point, we have discussed all the methods for computing image motion displacements. Here we will talk about a very important constraint that has not been used by research in retinal motion computation.

5.7.2 Should we want to compute retinal displacements, we should rely on constraints

The previous section reviewed previous and current work in the area of the computation of retinal motion displacements. As we saw, all the methods are based on heuristics about the similarity of the displacements, minimal distance, and the formation of clusters. No method has taken into account the very strong constraint that exists among the displacements. The points in the two dynamic frames (before and after the motion), are *related by the very strong constraint of rigid motion.* Indeed, the points before and after the motion are the projections of three-dimensional points that move rigidly. If we ever hope to recover retinal motion displacements, we must take into account the existing constraints. If we don't, we will never be able to solve the problem, since we will always be obliged to rely on heuristics and restrictive assumptions. So, we must investigate the rigid motion constraints. Recalling from section 5.2.1 the relation between two corresponding retinal points we have:

If (x,y) is the retinal point before the motion and (x',y') is the point after the motion, then:

$$[x', y', 1] = E \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where

$$e_1 = \Delta Zr_4 - \Delta Yr_7 \;,\; e_2 = \Delta Zr_5 - \Delta Yr_8,\; e_3 = \Delta Zr_6 - \Delta Yr_9$$
$$e_4 = \Delta Xr_7 - \Delta Zr_1,\; e_5 = \Delta Xr_8 - \Delta Zr_2 \;,\; e_6 = \Delta Xr_9 - \Delta Zr_3$$
$$e_7 = \Delta Yr_1 - \Delta Xr_4,\; e_8 = \Delta Yr_2 - \Delta Xr_5 \;,\; e_9 = \Delta Yr_3 - \Delta Xr_6$$

So, we have here the following mathematical problem:

$$E = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix} \quad \text{with}$$

Given the set of points $A=\{(x_i,y_i),i=1,..,n\}$ *before the motion and the set* $A'=\{(x'_i,y'_i),i=1,..,n\}$ *after the motion, correspond the points of these two sets , such that* $\exists$ *matrix E, with the property*:

$$[x', y', 1] = E \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

for all the corresponding points $(x,y)\in A$ *and* $(x',y')\in A'$.

This is the correspondence motion problem, phrased in a mathematical way, utilizing the available constraints.

The question that arises now then, is : Can we solve this problem?

It would be nice to solve this problem, without having to compute the matrix E, because then we would know the three-dimensional motion parameters, and then the problem of computing retinal displacements would be very easy. Unfortunately, we haven't been able to solve this problem up to now, without first finding the matrix E. Later, in section 5.7.4 we will show how to find matrix E without correspondences, and then come back and compute the displacements. It remains a problem of our future reseach, to address this question.

At this point, we conclude the criticism of previous research, and we move to algorithms for the computation of three-dimensional motion parameters.

5.7*3 Algorithms for 3-D motion perception

Here we study how to recover three-dimensional motion without i correspondences (optical flow or discrete displacements). We will study the differ and discrete cases separately.

5.7.3.1 The differential case

In these sections we study the problem of determining three-dimensional moti< the case of nonplanar surfaces. The case of planar surfaces, is easier, and it has been analyzed recently [Negadharipur and Horn, 1985].

We approach the problem of motion estimation from a least squares point of Given the motion and the spatial brightness gradient one can predict the time deri> of brightness at each point in the image. We find the motion that minimizes the int of the square of the difference between this predicted value and the observed derivative. The integral is taken over the image region of interest, which may b whole (in the egomotion case). Before we apply the method to real images, care mi taken in filtering and sampling. The estimate of the spatial gradient and time derive is sensitive to effects of aliasing that comes from inadequate low-pass filtering t sampling. It can be considered a mistake to simply pick every nth frame out of an i sequence. At the least, all the frames should be averaged before sampling in ore reduce the high frequency components. Of course, in this way we might have smearing but a series of widely separated snap-shots do not obey the conditions < sampling theorem, and the estimates of the derivatives will have large errors.

5*7.3*2 The relation between 3-D motion and retinal motion

A camera is assumed to move in a static environment. Let a coordinate system X,l fixed with respect to the camera, with the Z-axis pointing along the optical axis, rigid body B be stationary in the environment, from the surface of which a closed sui is visible. Any rigid body motion, as we have already seen, can be resolved in two fa a translation and a rotation. We shall denote by T =(U,V,W) the translational comp of the motion and by Q = (A,B,C) its angular velocity. We also consider the image perpendicular to the Z axis at the point (0,0,1) (i.e. focal length = 1). and we dene (x,y) the coordinates of a point on the image plane. We have already seen that the o flow equations are given by:

$$u=\frac{-U+xW}{Z}+Axy-B(x^2+1)+Cy \qquad 5.32$$

$$v=\frac{-V+yW}{Z}+A(y^2+1)-Bxy-Cx \qquad 533$$

It is clear that there is translational part in the flow and a rotational one. In other words, the flow equations can be written as:

$u=u_t+u_r$, $v=v_t+v_r$ **where** u_t, v_t **the translational parts and** u_r, v_r **the rotational parts, with**

$$u_t=\frac{-U+xW}{Z}$$

$$u_r=+Axy-B(x^2+1)+Cy$$

$$v_t=\frac{-V+yW}{Z}$$

$$v_r=+A(y^2+1)-Bxy-Cx$$

But, we know that the optical flow at every point in the image satisfies the following equation:

$f_x u+f_y v+f_t=0.$

If we substitute the values of the optical flow field from the equations 5.32, 5.33 into this equation, then we get the following equation:

$$f_x\left(\frac{-U+xW}{Z}+Axy-B(x^2+1)+Cy\right)+f_y\left(\frac{-V+yW}{Z}+A(y^2+1)-Bxy-Cx\right)+f_t=0$$

We call this equation *image brightness motion equation,* **and it will be the basis of the forthcoming analysis.**

Determining the motion of a moving camera from successive images is much easier if we are told that the motion is purely translational or purely rotational.

5.7.3.3 Rotational case

In this section we discuss the case where the motion of the camera is assumed purely rotational. In that case, the optical flow is:

$$u = Axy - B(x^2+1) + Cy$$

and

$$v = A(y^2+1) - Bxy - Cx$$

For the following we assume that the image plane is the rectangle $x \in [-\lambda,\lambda]$, $y \in [-\mu,\mu]$. The same method applies if the image has some other shape. matter of fact, it can be used on subimages corresponding to individual objects in th that the environment contains objects that may move relative to one another; of c this case is much harder, and we do not consider it in this thesis).

The image britghness motion equation now becomes :

$$f_x(Axy - B(x^2+1) + Cy) + f_y(A(y^2+1) - Bxy - Cx) + f_t = 0$$

or

$$B(-f_x(x^2+1) - f_y xy) + A(f_x xy + f_y(y^2+1)) + C(f_x y - f_y x) + f_t = 0 \quad (5.34)$$

The above linear equation contains the desired parameters A,B,C and everywhere in the image. It has to be understood that the coefficients of this equ involve measurable parameters, the spatiotemporal derivatives of the image int function.

If we use equation 5.34 at three points in the image, then we obtain a linear syst three equations with three unknowns A,B,C, from which the unknowns are recovered. But if we take into account the noise in the image (introduced b digitization process and other factors) a well as the errors introduced by the num approximation of the image derivatives f_x f_y and f_t then we may get very undes results. So, seeking a global method we wish to minimize the expression:

$$\int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} [(-f_x(x^2+1) - fyxy)]B + A(fxxy + fy(y2+1)) + C(fxy - fyx) + ft]^2 dxdy$$

In this case we determine the best fit with respect to the L_2 norm, which is defined as

$$\|g(x,y)= \int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} [g(x,y)]^2 dxdy$$

So, we differentiate equation 5.34 with respect to A,B, and C and we set the resulting expressions equal to zero.

Let us introduce the following abbreviations.

$K(x,y)= -f_x(x^2+1)-f_y xy,$

$L(x,y)=f_x xy+f_y(y^2+1),$

$M(x,y)=f_x y-f_y x.$

Expression 5.34 thus becomes:

$$\int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} [K(x,y)B+L(x,y)A+M(x,y)C+f_t]^2 dxdy$$

After we differentiate equation 5.34 with respect to A,B,C we obtain the following three equations:

$$\begin{aligned} &a_{11}A+a_{12}B+a_{13}C=b_1 \\ &a_{21}A+a_{22}B+a_{23}C=b_2 \quad \Sigma_1 \\ &a_{31}A+a_{32}B+a_{33}C=b_3 \end{aligned}$$

with

$$a_{11}= \int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} L^2(x,y)dxdy$$

$$a_{12}= \int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} L(x,y)K(x,y)dxdy$$

$$a_{13}= \int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} L(x,y)M(x,y)dxdy$$

$$a_{21}= \int_{-\lambda}^{\lambda} \int_{-\mu}^{\mu} L(x,y)K(x,y)dxdy$$

$$a_{22}=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}K^2(x,y)dxdy$$

$$a_{23}=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}K(x,y)M(x,y)dxdy$$

$$a_{31}=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}L(x,y)M(x,y)dxdy$$

$$a_{32}=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}K(x,y)M(x,y)dxdy$$

$$a_{33}=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}M^2(x,y)dxdy$$

$$b_1=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}L(x,y)f_t dxdy$$

and

$$b_2=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}K(x,y)f_t dxdy$$

$$b_3=\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}M(x,y)f_t(x,y)dxdy$$

The system Σ_1 determines uniquely the parameters A,B and C.

5.7.3.4 Translational case

In this section we discuss the case where the motion of the camera is assumed purely translational. In that case, the optical flow is:

$$u = \frac{-U + xW}{Z}$$

and

$$v = \frac{-V + yW}{Z}$$

and the image brightness motion equation becomes:

$$f_x\left(\frac{-U + xW}{Z}\right) + f_y\left(\frac{-V + yW}{Z}\right) + f_t = 0$$

or

$$f_x(-U + xW) + f_y(-V + yW) + f_t Z = 0$$

We should note that the depth is involved now in the image brightness motion equation and so our method will be different.

By differentiating the above equation with respect to x and y, we obtain:

$$f_{xx}(-U + xW) + f_x W + f_{yx}(-V + yW) + f_{tx} Z + f_t(\partial Z/\partial x) = 0$$

and

$$f_{xy}(-U + xW) + f_y W + f_{yy}(-V + yW) + f_{ty} Z + f_t(\partial Z/\partial y) = 0$$

or (by dividing the above two equations and assuming that there is motion in depth, i.e. $W \neq 0$),

$$\frac{f_{xx}((-U/W) + x) + f_x + f_{yx}((-V/W) + y)}{f_{xy}((-U/W) + x) + f_y + f_{yy}((-V/W) + y)} = \frac{f_{tx} Z + f_t(\partial Z/\partial x)}{f_{ty} Z + f_t(\partial Z/\partial y)}$$

In the above equation the depth and the derivatives of the depth with respect to the image coordinates are involved. But the derivatives of the depth function with respect to the retinal coordinates (x,y), are related to the derivatives of the depth function with respect to the world coordinates (X,Y,Z), in the following way:

$$\partial Z/\partial x = \frac{Z \partial Z/\partial X}{1 - x \partial Z/\partial X - y \partial Z/\partial Y}$$

$$aZ/dy=\frac{ZdZ/dY}{1-xdZ/aX-ydZ/dY}$$

Near the origin of the image plane the denominator of the above equations becon
(The above equations are true under the assumption that the focal length of the cam*
1). So, near the origin of the image plane, equation 5.35 becomes:

$$\frac{/_{xx}((-um+x) + f+fji-v/m+y)}{f_{xy}((-U/W)+x)+f_y+f_{yy}((-V/W)+y)} = \frac{f_{tx}+f_t(\partial Z/\partial X)}{f_{ty}+f_t(dZ/dY)} \qquad (5.36)$$

The above equation, linear in the unknowns U/W, V/W can be used in a least sq
formulation to give us the direction of translation. The obvious price we have *U*
though, is that we have to compute the shape of the object (3Z/8X.3Z/6Y).
Equation 5.36 after some algebraic manipulations becomes:

$K(xj)a+L(x,y)b=M(x,y)$

where we have introduced the abbreviations:

$a = U/W,$

$b=V/W,$

$K(x,y) = -fjf_{ty}+f_t(\partial Z/\partial y))+f_{xy}(f_{tx}+f_t(\partial Z/\partial x))$

$L(x,y) = -f_{yx}(f_{ty}+f_t(\partial Z/\partial y))+f_{yy}(f_{tx}+f_t(\partial Z/\partial x))$

$M(x,y,\quad v)(f+f(\partial Z/\partial v))-(f\ x+f+f_{yy}y)(f_{tx}+f_t(\partial Z/\partial x))$

Finally using equation 5.36 in a least squares scheme, we derive the system:

$$[I\ I\ K\backslash xj)dxdy]a+l\backslash\ j\ K(x,y)L(x,y)dxdy]b=\ I\ I\ KbjWM$$

$$[J\ j\ K(x_9y)L(x_tl)dxdy]aH\backslash\ j\ L(x,y)dxdy]b = J\ J\ L(x,y)M(xj)d$$

whose solution gives the desired direction of translation.
To study the general case, we can follow the same approach (i.e. differentiate the i
brightness motion equation with respect to x and y) and follow the same least sq
method, with the use of shape information.
In what follows, because the image brigthness motion equation is complicated, we
use a vector notation. This equation can be written as:

$f_t+ V.Q + (1/Z)K.T = O,$

where Ω the rotational velocity, T the translational velocity, and

$\mathbf{V}=(f_y+y(xf_x+yf_y),\ -f_x-x(xf_x+yf_y),\ yf_x-xf_y)^T$

$\mathbf{K}=(-f_x,\ -f_y,\ xf_x+yf_y)^T$

5.7.3.5 Motion known

The image brigthness motion equation can be used to find depth if the motion is known. Indeed, from this equation we get:

$$Z=-\frac{\mathbf{K}.\mathbf{T}}{f_t+\mathbf{V}.\Omega}$$

All the quantities of the right hand side of this equation are computable from the image gradients or known (motion). Obviously this method may produce inaccurate estimates of the depth Z, because the numerator and denominator may be small, if the britghness gradients are small or the vectors K and T are nearly orthogonal.

5.7.3.6 Depth known

Suppose that the depth of the surface in view is known. Then, it is trivial to recover motion without correspondences in this differential case. In order to avoid errors from noise, we minimize the quantity:

$$\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}[f_t+V.\Omega+(1/Z)K.T]^2$$

Differentiating with respect to Ω and T, we get the following vector equations:

$$[\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}(1/Z)^2\mathbf{K}\mathbf{K}^T]\mathbf{T} \quad + \quad [\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}(1/Z)\mathbf{K}\mathbf{V}^T]\Omega = \ -\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}f_t(1/Z)\mathbf{K}$$

$$[\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}(1/Z)\mathbf{V}\mathbf{K}^T]T \quad + \quad [\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}\mathbf{V}\mathbf{V}^T]\Omega = \quad -\int_{-\lambda}^{\lambda}\int_{-\mu}^{\mu}f_t\mathbf{V}$$

These equations represent six linear equations in the six motion parameters unknowns.

5.7.3.7 Stability of the method in the case of rotational motion

Recalling from section 5.7.3.3 the method that recovers rotation, we had to soh linear system (in vector notation):

$$\int^{\mu} \mathbf{V}\mathbf{V}^T]\Omega = -\int^{\lambda} \int^{\mu} f_t \mathbf{V}$$

In order to study the properties of thge algorithm that solves this linear system, we study the matrix

$$\mathbf{rr}^{T}_{-\lambda\ -\mu}$$

But such a thing seems very difficult at this point, because the vector V depends o intensity function and its derivatives. So, the only thing that we can hope for, is to < approximate analysis, by assuming that the values of vectors V are unifo distributed.

Instead, we give some intuitive reasons for the stability that have been confirmed b experiments. Rotations about the x and y axis are computed with high accuracy, wh rotation about the Z-axis is corrupted with small amounts of noise. The reason for tt that rotation about the x and y axes produce motion fields (spatiotemoral variations] vary a little over the image, and in that case a small field of view can be used to esti these components. From the other hand, rotation about the z axis produces a field varies a lot over the image. So, the maximum velocity depends on the size of the fi< view.

5.7.3.8 The translational case revisited

In section 5.7.3.4 we studied the translational case. In our analysis, we neede< shape of the object in view in order to obtain a solution. This is rather weak in some s So, the question that arises then, is : can we recover the direction of translation wi1 correspondence in the discrete case, and without using shape information? Prelimi investigations show that this is possible if a binocular observer is used. Another pos approach is the following:

The image brightness motion equation under the assumption that the moti< translational, becomes:

$Z = -(1/f)K \cdot T$.

Obviously the depth has to be positive. So, we must find what are the numbers T=(U,V,W) that make the depth Z from the above equation, positive, at every image **point.** Clearly, the **problem as** posed might not have a unique solution, but we might be **able** to find a set of solutions, which can be satisfactory. Finally, it has to be understood, **that** the depth has an upper bound in such a situation. Indeed, from the above equation, we have that:

Z=-fl/#V·T , or \Z] = IKTI/l/;i or \Z I/||T||£ \\K\\/\f_t\ . This is an upper bound under the assumption that T is parallel to K. Actually, the depth will be much smaller if K is nearly orthogonal to T.

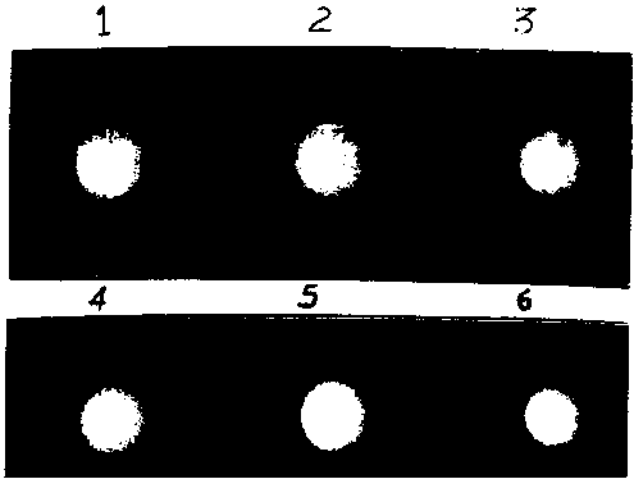
5.7.3.9 The general case

In the general case (rotation plus translation) the method that we propose is the same as in section 5.7.3.4, with the difference that the resulting system will be nonlinear in the five parameters (U/.W,V/W, A,B,C). In order for this system to be solved, we must start with an approximate solution near the actual solution. Otherwise, it will not converge to the actual solution.

5.7.3.10 Implementation and experiments

The purely translational and purely rotational cases have been implemented. Figure 5.5.1 shows pictures of a sphere taken from a moving camera (synthesized motion). The camera was moving with velocity U=-7, V=-7, W=-7 and took pictures every unit of time. The actual direction of translation, was (U/W=l, V/W=l), and our algorithm (5.7.3.4), from this sequence of images, yielded:

(U/W= 1.052632, V/W=0.925926).



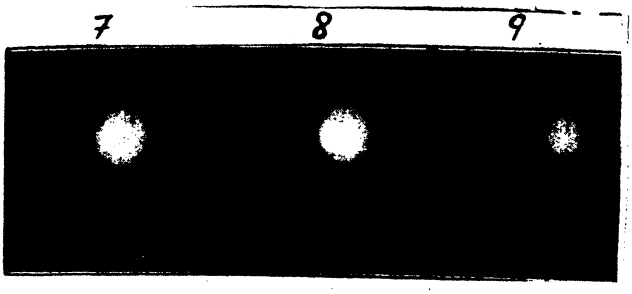
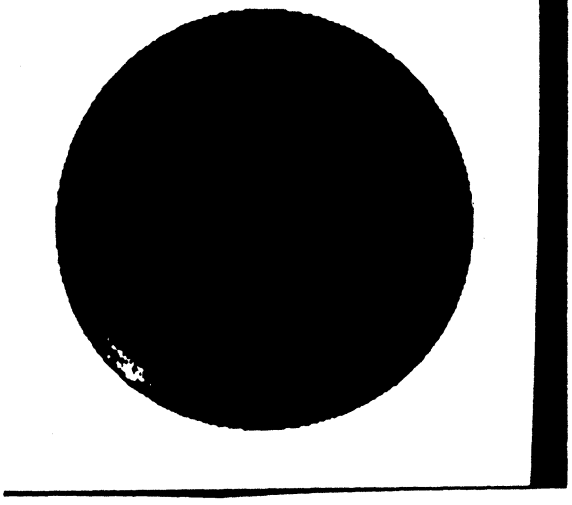


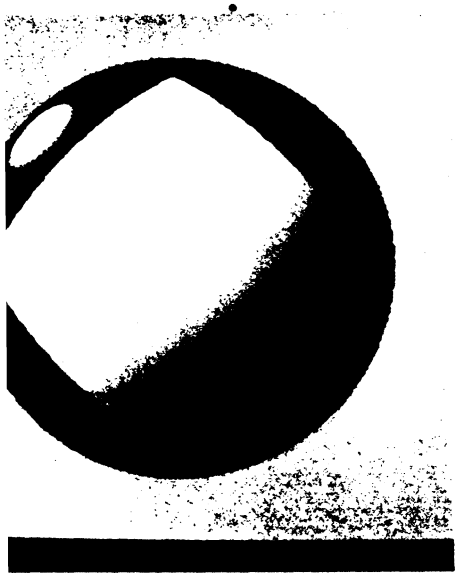
Figure 5.5.1: Nine snapshots of a translating sphere

For the purely rotational case, we experimented with several motions and several different reflectance functions. In general, if the image intensity function is smooth, then the results are very accurate. If the image intensity function is not smooth enough, then the results get corrupted, because, even with sufficient smoothing, the image spatiotemporal derivatives are very inaccurate. In the sequel, we will present experiments for the case where the motion was (A=C=0, B=0.001), for spherical surfaces (since they are the worst for this kind of experiment), with many different reflectances. The following table presents the results obtained with the least squares method, for the different surfaces that are presented in figure 5.5.2.

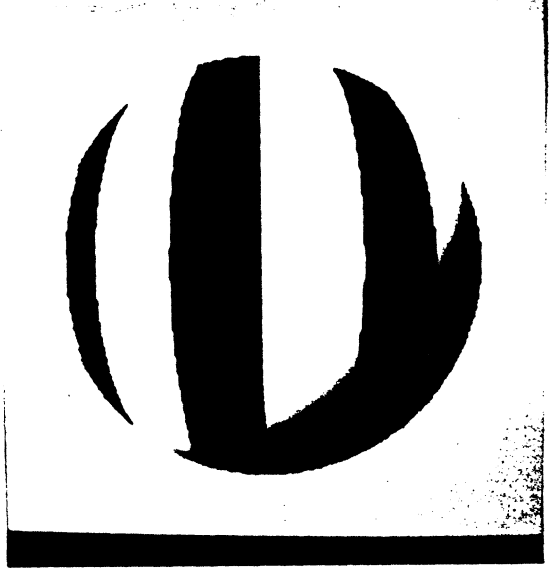
ii. Specular



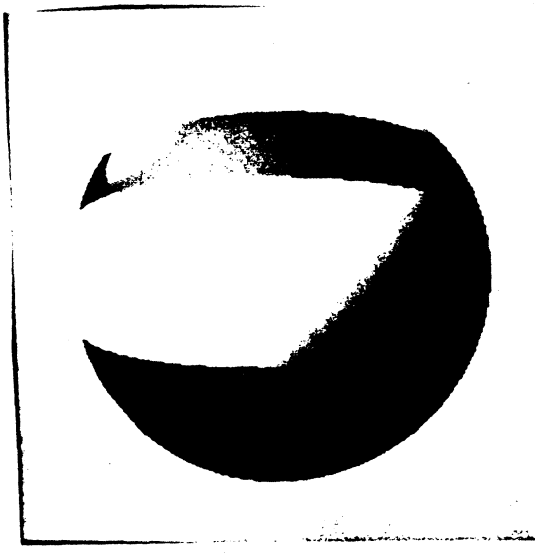
iv. Specular x Random



Specular x Diamond



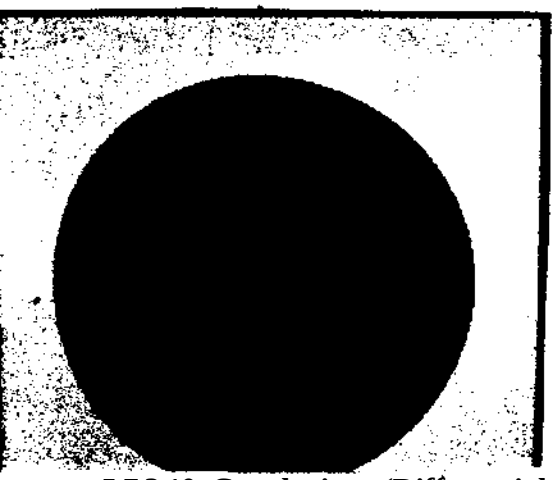
vi. Highly Specular x Vertical Stripes



vii. Specular x Stri

Figure 5.5.2

i.	Lambertian	-0.000048	0.001025	-0.000019
ii.	Random	-0.002594	0.002572	-0.000088
iii.	Specular	-0.000191	0.001205	0.000384
iv.	Specular x Random	-0.002588	0.002220	-0.001412
v.	Specular x Diamond	0.000009	0.000856	-0.000580
vi.	Highly Specular x Vertical Stripes	-0.000970	0.000991	-0.000262
vii.	Specular x Stripes	0.000092	0.001932	-0.000790



5,7.3,10 Conclusions (Differential motion without correspondence)

In this section we presented algorithms for the computation of the three-dimensional motion, in the case of continuous motion, without using the intermediate stage of computing optical flow. Our algorithms work well for the rotational and translational case. The general case needs more investigation. Our future work in this area, is to work out the details of computing translational motion without the need of the shape of the surface in view, even though the use of surface shape agrees with our general framework of combining information from different sources. Also, to work for the development of linear equations for the general case (translation & rotation). Our treatment did not use any correspondences (optical flow). This by no means attempts to indicate that correspondence is not implemented in some way in the human visual system. This is something that we don't know, even though there is some indication (from various psychological and psychophysical experiments) that the human (animal) visual system is **engaged** in some kind of visual correspondence. Our analysis did not try to establish the fact that correspondence is useless. On the contrary, correspondence is very powerful, but it has not yet been demonstrated that it is feasible. We simply demonstrated that three-dimensional motion **can be obtained** without using correspondences. Our theory is not based on **an** input that we don't know if it is computable, as optical flow for example. It is based on the spatiotemporal derivatives of the intensity function, something that is very

well defined and measurable. Also, it is highly parallel and easily implementable in neuronal hardware. Finally it is worth saying that our analysis employed the use of a single camera. It is one of our future goals to study the motion perception problem using a binocular observer. The reason for this is, as the next section will explain, the fact that using a binocular observer, the constraints between retinal and three-dimensional motion change, and the complexity of the problem changes too. The highly nonlinear equations for the case of a monocular observer, become linear in the case of a binocular observer.

5.7.4 The discrete case

In this section we study how to recover three-dimensional motion from retinal motion in the case of discrete motion. The problem is the following:

Consider a set $A=\{(X_i,Y_i,Z_i),i=1,..,n\}$ of three-dimensional points, that move rigidly and they come to a new position, such that they constitute the set $A'=\{(X'_i,Y'_i,Z'_i),i=1,..,n\}$. The points are imaged by a camera (traditional model, as described in Chapter 2), and their projections before the motion make the set $A_1=\{(x_i,y_i),i=1,..,n\}$ and after the motion the set $A'_1=\{(x'_i,y'_i),i=1,..,n\}$. With only input the sets A_1 and A'_1 we want to recover the three-dimensional motion that transformed set A to set A'. All the traditional approaches that are based on the correspondence approach, first try to find out the correspondence between the points of the two sets A and A',i.e. to find out for every point $(x,y)\in A$, what point $(x',y')\in A'$, is the image of the same three-dimensional point. From the association of point (x,y) to (x',y'), we have a displacement vector, and from several displacement vectors the three-dimensional motion may be obtained, as it has been shown by several published algorithms (Section 5.3 contains several references). Sections 5.3 and 5.4 criticized the approaches that use correspondence, from the point of view that correspondence is very difficult. So, we would like to solve this problem, without having to go first through the solution of the correspondence problem. Our only input is the sets A and A', i.e. the perspective projections of a cloud of 3-D points *before* and *after* the motion.

Our analysis is done for the case of a binocular observer. Of course, now we should address the problem of finding depth, which requires the solution of the correspondence problem between the left and right image. But we show in our analysis, that it is possible to recover depth without correspondence, at least for the case of planar surfaces. For the purposes of this section, we will assume that in the case of nonplanar surfaces, the depth is known.

In the sequel, we will address the problem of finding motion without correspondence, in the case of discrete motion, for both planar and nonplanar surfaces. These cases will be treated differently.

5.7.4.1. Stereo without correspondence for planar surfaces

In this section we present a method for the recovery of the 3-D parameters for the set of 3-D planar points from their left and right images without using any point-to-point correspondence; instead we consider all point correspondences at once and so there is no need to solve the difficult correspondence problem in the case of the static stereo.

Let an orthogonal cartesian coordinate system OXYZ be fixed with respect to the left camera, with O at the origin (O being also the nodal point of the left eye) and the Z-axis pointing along the optical axis.

Let the image plane of the left camera be perpendicular to the Z-axis at the point (0,0,f), (focal length = f).

Let the nodal point of the right camera be at the point (d,0,0) and its image plane be identical to the left one; the optical axis of the right camera (eye) points also along the Z-axis and passes through point (d,0,0).

Consider a set of 3-D points $A = \{ (X_i, Y_i, Z_i) / i=1,2,3 \dots n \}$ lying on the same plane, the latter being described by the equation :

$$Z = pX + qY + c$$

Let O_l, O_r be the origins of the two-dimensional orthogonal coordinate systems on each image plane; these origins are located on the left and right optical axes while the corresponding coordinate systems have their y-axes parallel to the axis OY, and their x-axes parallel to OX.Finally let $\{(x_{li}, y_{li}) / i=1,2,3 \dots n\}$ and $\{(x_{ri}, y_{ri}) / i=1,2,3 \dots n\}$ be the projections of the points of set A on the left and right retinae, respectively, i.e.

$$x_{li} = \frac{fX_i}{Z_i} \qquad (5.37) \qquad y_{li} = \frac{fY_i}{Z_i} \qquad (5.38) \qquad / \ i=1,2,3 \dots n$$

$$x_{ri} = \frac{f(X_i - d)}{Z_i} \qquad (5.39) \qquad y_{ri} = \frac{fY_i}{Z_i} \qquad (5.40) \qquad / \ i=1,2,3 \dots n$$

Let (x_{li},y_{li}) and (x_{ri},y_{ri}) be corresponding points in the two frames. Then we have that.

$$x_{li}-x_{ri}=\frac{fd}{Z_i} \qquad (5.41)$$

$$y_{li}=y_{ri} \qquad (5.42)$$

where Z_i, the depth of the 3-D point having those projections.

In the sequel, we prove that the quantity

$$\sum_{i=1}^{n}\frac{y_{li}^{k}}{Z_i}$$

is directly computable without using any point correspondence between the left and right frames. We proceed with the following propositions:

5.7.4.2 Proposition : Using the aforementioned nomenclature the quantity

$$\sum_{i=1}^{n}\frac{y_{li}^{k}}{Z_i}$$

where

$$k\geq 0 \ \wedge \ k\neq\frac{m}{2^{*}n}, \qquad m,n\in Z-\{0\},$$

is directly computable.

Proof: We have that

$$\sum_{i=1}^{n} \frac{y_{li}^{*}}{Z_i} = (\text{from equation } (5.41)) = \sum_{i=1}^{n} y_{li} \frac{(x_{li} - x_{ri})}{7TZ} =$$

$$= \sum_{i=1}^{n} \text{"TT"} \text{"} \sum_{i=1}^{n} \text{"TT"} \text{ (from equation (5.42)) =}$$

$$= \sum_{i=1}^{n} \frac{x_{li}\, y_{li}^{k}}{f\, d} - \sum_{i=1}^{n} \frac{x_{ri}\, y_{ri}^{k}}{f\, d} .$$

Thus,

$$\sum^{n} \frac{y_{li}^{k}}{} = \sum^{n} \frac{x_{li}\, y_{li}^{k}}{} - \sum^{n} \qquad (5.43)$$

From equation (5.43) the claim is obvious.

5*7.4.3 Proposition : Using the aforementioned nomenclature, the parameters p, q and c of the plane in view are directly computable without using any point-to-point correspondence between the two frames.

Proof: The equation of the world plane when expressed in terms of the coordinates of the left frame, becomes:

$$\frac{1}{Z} = (f - px_l - qy_l)\, \frac{1}{cf} \qquad (5.44)$$

So, from equation (8) it follows that:

$$\frac{-L}{Z_i} = (f - p_{Xu} - q y_u)\, \frac{-L}{cf} \qquad i=1,2,3\ldots n \qquad (5.45)$$

Now, we have:

$$\sum_{i=1}^{n} \frac{1}{Z_i} = \sum_{i=1}^{n} (f - px_{li} - qy_{li}) \frac{y_{li}^k}{cf}$$

or

$$\sum_{i=1}^{n} \frac{1}{Z_i} = \frac{1}{c} \sum_{i=1}^{n} y_{li} - \frac{1}{cf} \left[\sum_{i=1}^{n} px_{li} y_{li}^k + \sum_{i=1}^{n} qy_{li} y_{li}^k \right] \qquad (5.46)$$

The left-hand side of equation (10) has been shown to be computable without using any point-to-point correspondence (see Proposition 5.7.4.1).

If we write equation (10) for three different values of k, we obtain the following linear system in the unknowns p,q,c which in general has a unique solution (except for the case where the projection of all points of set A, have the same y-coordinate in both frames):

$$\sum_{i=1}^{n} \frac{x_{li} y_{li}^{k1}}{f d} - \sum_{i=1}^{n} \frac{x_{ri} y_{ri}^{k1}}{d} = \frac{1}{c} \sum_{i=1}^{n} y_{li}^{k1} - \frac{1}{cf} \left[\sum_{i=1}^{n} px_{li} y_{li}^{k1} + \sum_{i=1}^{n} qy_{li} y_{li}^{k1} \right] \quad (5.47)$$

$$\sum_{i=1}^{n} \frac{x_{li} y_{li}^{k2}}{f d} - \sum_{i=1}^{n} \frac{x_{ri} y_{ri}^{k2}}{d} = \frac{1}{c} \sum_{i=1}^{n} y_{li}^{k2} - \frac{1}{c^* f} \left[\sum_{i=1}^{n} px_{li} y_{li}^{k2} + \sum_{i=1}^{n} qy_{li} y_{li}^{k2} \right] \quad (5.48)$$

$$\sum_{i=1}^{n} \frac{x_{li} y_{li}^{k3}}{f d} - \sum_{i=1}^{n} \frac{x_{ri} y_{ri}^{k3}}{d} = \frac{1}{c} \sum_{i=1}^{n} y_{li}^{k3} - \frac{1}{cf} \left[\sum_{i=1}^{n} px_{li} y_{li}^{k3} + \sum_{i=1}^{n} qy_{li} y_{li}^{k3} \right] \quad (5.49)$$

where we used equation (5.43) to the left hand sides.

The solution of the above system recovers the structure and the depth of the points of set A without any correspondence and this is the conclusion of Proposition 5.7.4.2.

5.7.4.4. Practical Considerations

We have implemented the above method for different values of ki,k2,k3 and especially for the cases:

a) $k_r = 0$ $k_2 = 1/3$ $k_3 = 2/3$

b) $k_1 = 0$ $k_2 = 1/3$ $k_3 = 1/5$

The noiseless cases give extremely accurate results.

Before we proceed, we must explain what we mean by noise introduced in images. When we say that one frame (left or right) has noise of *a%*, we mean that L plane contains N projection points we added [(N*a)/100] randomly distributed poir *Note: []* denotes the integer part of its argument).

When the noise in both frames is kept below 2% then the results are still satisfactory. When the noise exceeds 5% then only the value of p gets corrupted, bui values of q and c remain very satisfactory. To correct this and get satisfactory res for high noise percentages, we devised the following method that uses three cameras

" We consider the three camera configuration system as in Figure 5.6, where th< camera has only vertical displacement with respect to the left one. If all three im; are corrupted by noise (ranging from 5% to 20%) then application of the algoritl *Proposition 3.2*) to the left and top frames will give very reasonable va-lues for p a and corrupt q, which q, as well as c, are accurately computed from the applicatk the same algorithm to the right and left frames ".

So, by applying our stereo (without correspondence) algorithm to tr, camera configuration vision system, we obtain accurate results for the parame describing the 3-D planar patch, even for noise percentages of 20% or slightly more, for different amounts of noise in the different frames.

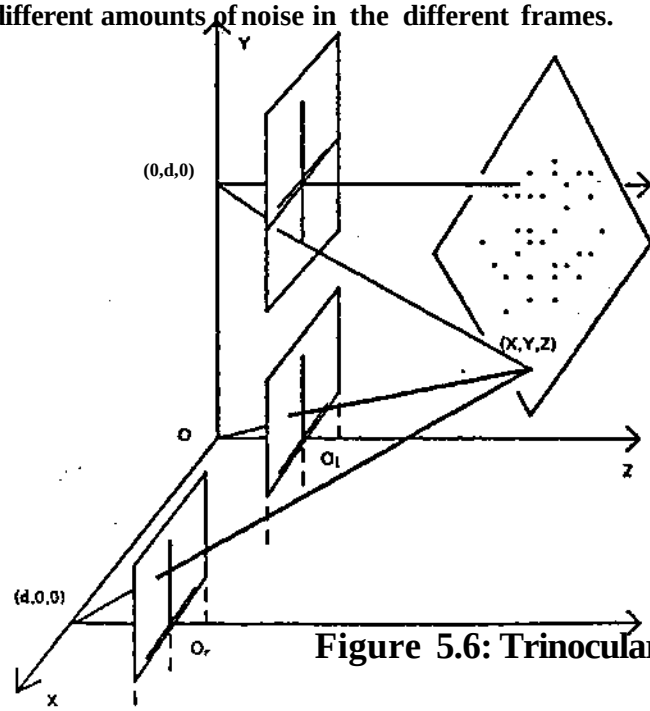


Figure 5.6: Trinocular system

&7,4.5. Recovering the direction of translation.

Here we treat the case where the points of set A just rigidly translate, and we wish to recover the direction of the translation. In this case, the depth is not needed but the orientation of the plane is required. The general case is treated in the next section.

5.7.4.5.1 Technical prerequisites*

Consider a coordinate system OXYZ fixed with respect to the camera; O coincides with the nodal point of the eye, while the image plane is perpendicular to the Z-axis (focal length=f), that is pointing along the optical axis (see Figure 5.7.).

Let us represent points on the image plane with small letters (e.g (x,y)) and points in the world with capital ones (e.g. (X,Y,Z)).

Let us consider a point $P=(Xi_f Yi_t Zi)$ in the world, with perspective image (x_1,y_1), $vfherex_1=(fX_1)/Zandy_1=(fY_1)/Z$.

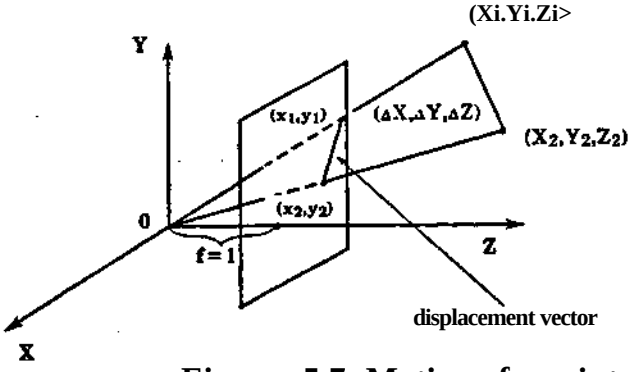


Figure 5.7: Motion of a point

If the point P moves to the position $P^f=(X2,Y2_9Z2)$ with

$$X_2=X_1+\Delta X \quad (5.50)$$

$$Y_2=Y_1+\Delta Y \quad (5.51)$$

$$Z_2=Z_t+LZ \quad (5.52)$$

then we desire to find the direction of the translation (AX/AZ,AY/AZ).

If the perspective image of P' is (X2,y2), then the observed motion of the world point in the image plane is given by the displacement vector : *(X2-xi, y2-yi)* (which in the case of very small motion is also known as "optical flow").

We can easily prove that :

$$x_2 - x_1 = \frac{f\,\Delta X - x_1\,\Delta Z}{Z_1 + \Delta Z} \qquad (5.53)$$

$$y_2 - y_1 = \frac{f\,\Delta Y - y_1\,\Delta Z}{Z_1 + \Delta Z} \qquad (5.54)$$

Under the assumption that the motion in depth is small with respect to the d
the equations above become :

$$x_2 - x_1 = \frac{f\,\Delta X - x_1\,\Delta Z}{Z_1} \qquad (5.55)$$

$$y_2 - y_1 = \frac{f\,\Delta Y - y_1\,\Delta Z}{Z_1} \qquad (5.56)$$

The above equations relate the retinal motion (left-hand sides) to the v
motion $\Delta X, \Delta Y, \Delta Z$.

5.7.4.5.2 Detecting 3-D direction of translation without correspondence.

Consider again a coordinate system OXYZ fixed with respect to the camera
Figure 5.8, and let $A = \{ (X_i, Y_i, Z_i) / i = 1,2,3 \ldots n \}$, such that

$$Z_i = pX_i + qY_i + c \qquad , i = 1,2,3 \ldots n$$

that is the points are planar. Let the points translate rigidly with transl
$(\Delta X, \Delta Y, \Delta Z)$, and let $\{ (x_i, y_i) / i = 1,2,3 \ldots n \}$ and $\{ (x_i', y_i') / i = 1,2,3, \ldots n \}$ be
projections of the set A before and after the translation, respectively.

Consider a point (x_i,y_i) in the first frame which has a corresponding one (x_i',y_i') in the second (dynamic) frame.

For the moment we do not worry about where the point (x_i', y_i') is, but we do know that the following relations hold between these two points

$$x_i' - x_i = \frac{f\Delta X - x_i \Delta Z}{Z_i} \tag{5.57}$$

$$y_i' - y_i = \frac{f\Delta Y - y_i \Delta Z}{Z_i} \tag{5.58}$$

where Z_i is the depth of the 3-D point whose projection (on the first dynamic frame) is the point (x_i,y_i). Taking now into account that

$$\frac{1}{Z_i} = \frac{f - p\, x_i - q\, y_i}{c\, f} \tag{5.59}$$

the above equations become :

$$x_i' - x_i = (f\,\Delta X - x_i\, \Delta Z)\, \frac{f - p\, x_i - q\, y_i}{c\, f} \tag{5.60}$$

$$y_i' - y_i = (f\,\Delta Y - y_i\, \Delta Z)\, \frac{f - p\, x_i - q\, y_i}{c\, f} \tag{5.61}$$

If we now write equation (24) for all the points in the two dynamic frames and sum the resulting equations up, we take :

$$\sum_{i=1}^{n} (x_i' - x_i) = \sum_{i=1}^{n} [(f\,\Delta X - x_i\, \Delta Z)\, \frac{f - p\, x_i - q\, y_i}{c\, f}\,]$$

or

$$\sum_{i=1}^{n} (x_i' - x_i) = \sum_{i=1}^{n} [\, \frac{f(f - p\,x_i - q\,y_i)\,\Delta X - x_i\,(f - p\,x_i - q\,y_i)\,\Delta Z}{c^* f}\,] \tag{5.62}$$

Similarly, if we do the same for equation (25), we take :

$$\sum_{i=1}^{n} (y_i' - y_i) = \sum_{i=1}^{n} [(f\,\Delta Y - y_i\,\Delta Z)\,\frac{f - p\,x_i - q\,y_i}{c\,f}]$$

or

$$\sum_{i=1}^{n} (y_i' - y_i) = \sum_{i=1}^{n} [\frac{f(f - p\,x_i - q\,y_i)\,\Delta Y - y_i\,(f - p\,x_i - q\,y_i)\,\Delta Z}{c\,f}] \quad (5.63)$$

At this point it has to be understood that equations (5.62) and (5.63) d require our finding of any correspondence.

By dividing equation (5.62) by equation (5.63), we get :

$$\frac{\sum_{i=1}^{n} x_i' - \sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} y_i' - \sum_{i=1}^{n} y_i} = \frac{\sum_{i=1}^{n} [\frac{\Delta X}{\Delta Z} f\,(f - p\,x_{li} - q\,y_{li}) - (f - p\,x_{li} - q\,y_{li})\,x_{li}]}{\sum_{i=1}^{n} [\frac{\Delta Y}{\Delta Z} f\,(f - p\,x_{li} - q\,y_{li}) - (f - p\,x_{li} - q\,y_{li})\,y_{li}]} \quad (5.64)$$

Equation (5.64) is a linear equation in the unknowns $\Delta X/\Delta Z$, $\Delta Y/\Delta Z$ and coefficients consist of expressions involving summations of point coordinates in dynamic frames; for the computation of the latter no establishment of any p correspondences is required.

So, if we consider a binocular observer, applying the above procedure in both and right "eyes", we get two linear equations (of the form of equation (5.64)) in the unknowns $\Delta X/\Delta Z$, $\Delta Y/\Delta Z$, which constitute a linear system that in general has a un solution.

5.7.4.5.3 What the previous method is not about, an unexpected bonus some problems

If one is not careful when analyzing the previous method, then he might think all the method does, is to correspond the center of mass of the image points before motion with the center of mass of the image points after the motion, and then base

that retinal motion to recover three dimensional motion. But this is wrong, because perspective projection does not preserve simple ratios, and so the center of mass of the image points before the motion does not correspond to the center of mass of the image points after the motion. All the above method does, is aggregation of of the motion constraints; it does not correspond centers of mass. At this point we should mention that the method we presented, in order to be valid, needs the set of the world planar points to be visible by both eyes; otherwise, the mathematics is not valid. But, on the other hand, this method would work with a textured region, for which we wouldn't be able to find corresponding interest points.

5.7.4.5.4 Practical considerations.

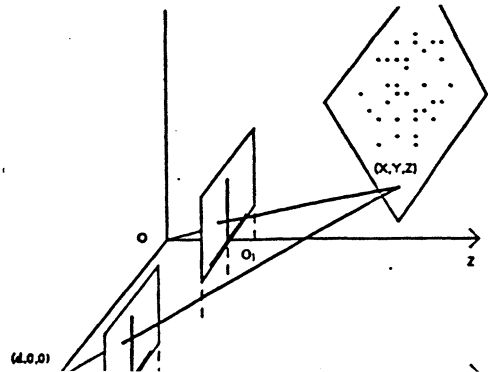
We have implemented the above method with a variety of planes as well as displacements; noiseless cases give exremely accurate results, while cases with noise percentages up to 20% (even with different amounts of noise in all four frames (first left and right - second left and right)) give very satisfactory results (an error of at most 5%) . We now proceed considering the general case.

5.7.4.6 Determining unrestricted 3-D motion of a rigid planar patch without point correspondences.

Consider again the imaging system (binocular) of Figure 5.9, as well as the set A= { $(X_i,Y_i,Z_i) / i=1,2,3 \ldots n$ } such that :

$$Z_i = pX_i + qY_i + c \qquad i=1,2,3 \ldots n$$

i.e. the points are planar; let B be the plane on which they lie. Suppose that the points of the set A move rigidly in space (translation plus rotation) and they become members of a set A' = { $(X_i', Y_i', Z_i') / i=1,2,3 \ldots n$ }. Since all o f the points of set A move rigidly, it follows that the points of set A' are also planar; let B' be the (new) plane on which these points lie.



Figures 5.8: A stereo imaging system

In other words the set A becomes A' after the rigid motion transformation. We wii recover the parameters of this transformation . From the projection of sets A and *i* the left and right image planes and using the method described in Section 5.7.4.3 the A and A' can be computed. In other words, we know exactly the positions in 3-D of al points of the sets A and A' (and this has been found without using any point correspondences).

So, the problem of recovering the 3-D motion has been transformed to the followi

*"Given the set A of planar points in 3D and the set A[9] of new planar points, which has been produced by applying to the points of set A a rigid motion transformation, recover that transformation.[9]**

Any rigid body motion can be analyzed to a rotation plus a translation; the rot* axis can be considered as passing through any point in the space, but after this poi chosen, everything else is fixed.

If we consider the rotation axis as passing through the center of mass (CM) o points of set A, then the vector which has as its two endpoints the centers of mass < **and $CM_{A'}$** of sets A and A' respectively, represents the *exact* 3-D translation.

So, for the translation we can write

$$\textbf{translation - T} = (X,Y,Z) = \mathbf{CM_{A'} - CM_A}$$

It remains to recover the rotation matrix.

Let, therefore, nj and 112 be the surface normals of the planes B and B\ Then, the an; between ni and **n**2 , where

$$\cos 6 = \frac{\mathbf{n}_r \quad \mathbf{n}_2}{|n_1| * |n_s|}$$

, with * • [f] the inner—product operator represents the rotation around an axis O1O2 perpendicular to the plane

defined by ni and 112, where

$$O_1O_2 = \frac{n_1 \times n_2}{| n_1 \times n_2 |}, \text{ with } ' \times ' \text{ the cross—product operator}$$

From the axis O1O2 and the angle G we develop a rotation matrix B_i. The matrix Ri does not represent the final rotation matrix since we are still missing the rotation around the surface normal. Indeed, if we apply the rotation matrix Ri and the translation T to the set A, we will get a set A" of points, which is different than A', because the rotation matrix Ri does not include the rotation around the surface normal 112.

So we now have a matching problem : on the plane B' we have two sets of points A^1 and A" respectively, and we want to recover the angle ϕ by which we must rotate the points of set A" (with respect to the surface normal 112) in order to coincide with those of set A'.

Suppose that we can find angle ϕ. From ϕ and n2 we construct a new rotation matrix R2. The final rotation matrix R can be expressed in terms of Ri , R2 as follows:

$$R = Ri\ R_2$$

It therefore remains to explain how we can compute the angle ϕ. For this we need the statistical definition of the mean direction.

Definition .

Consider a set A = {(Xi,Yi)/i= 1,2,3 ... n} of points ail of which lie on the same plane. Consider the center of mass, CM, of these points to have coordinates (Xc^Ycn). Let also circle (CM,1) be the circle having its center at (Xcn^Ycm) and radius of length equal to l.Let Pi be the intersections of the vectors CMAj with the circumference of the circle (CM,1), i= 1,2,3 ... n. Then the "mean direction" of the points of the set A, is defined to be the vector MD, where

$$MDs \wedge \sum_{j=1}^{n} CMP_j$$

It is clear that the vector of the mean direction is intrinsically connected wit set of points considered each time, and if the set of points is rotated around an perpendicular to the plane and passing through CM, by an angle ω, the new direction vector is the previous one rotated by the same angle ω.

So, returning to the analysis of our approach, the angle ϕ is the angle betwee vectors of mean directions of the sets A' and A" (which have obviously, common CM

Moreover, it is obvious that the angle ϕ, and therefore the rotation matri cannot be computed in the case the mean direction is **0** (i.e. in the case the set of poi characterized by a point symmetry).

5.7.4.7 Determining unrestricted 3-D motion of a rigid surface without point correspondences

In this section we consider the problem of the recovery of unrestricted 3-D moti non-planar surfaces. Again, we consider a set of rigidly moving points, and we as that the depth information is available. In another work [Aloimonos et al, 1986 describe how to recover the depth of a set of non-planar points from their stereo im without having to go through the correspondence problem. So consider a bino imaging system, and a set $A=\{ P_i = (X_i, Y_i, Z_i), i = 1,2,3 \ldots n \}$ of 3-D non-planar p The coordinates are with respect to a fixed coordinate system that will be used throu this section (we can consider as this system either the system of the left or right ca or the head frame coordinate system). Applying the method described in [Aloimonos 1986] , from the left and right images of the points of set A, we can recover the memb A themselves, i.e. their 3-D coordinates. Suppose now that the points of the set A rigidly in space (translation plus rotation) and that they become members of the set A $P'_i = (X'_i, Y'_i, Z'_i) / i = 1,2,3 \ldots n \}$. It is evident that the set A' can be recovered exact the set A with the method described in [Aloimonos et al, 1986] . In other words, the becomes A' after the rigid motion transformation. We wish to recover the paramet this transformation. We have already stated that from the projection of the sets A a on the left and right image planes and using the method described in [Aloimonos 1986] , the sets A and A' can be computed. Hence we know exactly the positions o points of the sets A and A' (and we came up with this result whithout relying to any p

to-point correspondence). So, for the purposes of this section we will assume that the depth information is available.

From the above discussion, we see that the problem of recovering the 3-D motion has been transformed to the following:

** Given the set A ofnonplanar points and the set A* corresponding to the new positions of the initial points after they have experienced a rigid motion transformation, recover that transformation, without any point-to-point correspondences! "*

Any rigid motion can be analyzed to a rotation plus a translation; the rotation axis can be considered as passing through the any point in space, but after this point is chosen, everything else is fixed.

If we consider the rotation axis as passing through the origin of the coordinate system, then if the point (Xj, Y{, Z_x) € A moves to a new position (X'i, Y*i, Z'i) € A', the following relation holds:

$$(X'i, T_{it} Z'i)^T = R (X_{it} Yi, Zi)^T + T \quad / i = 1,2,3 \ldots n \quad (5.65)$$

where R is the 3x3 rotation matrix and Ts(AX, AX, AZ)^T is the translation vector. We wish to recover the parameters R and T, without using any point-to-point correspondences.

Let,

$$(X_{i(} Yj, Zi)' \; m \; Pj \text{ and } (X'i, Y'_{i(} Z\backslash)' = P'i \quad / i = 1,2,3 \ldots n$$

Then, equation (5.65) becomes:

$$Pi = RP'i + T \quad / i = 1,2,3 \ldots n$$

Summing up the above n equations and dividing by the total number of points, n, we get:

$$\frac{\sum_{i=1}^{n} Z'}{n} = R \frac{\sum_{i=1}^{n} I}{n} + T \qquad (5.66)$$

From equation (5.66) it is clear that if the rotation matrix R is known, then the translation vector T can be computed. So, in the sequel, we will describe how to recover the rotation matrix R. In order to get rid of the translational part of the motion we shall transform the 3-D points to " free " vectors by subtracting the center-of-mass vector.

Let, therefore, CM_A and $CM_{A'}$ be the center-of-mass vectors of the sets of poi
and A' respectively; i.e. $CM_A = \Sigma (P_i / n)$ and $CM_{A'} = \Sigma (P'_i / n)$. We further
define:

$$v_i = P_i - CM_A \quad / \; i=1,2,3 \ldots n$$

$$v'_i = P'_i - CM_{A'} \quad / \; i=1,2,3 \ldots n$$

With these definitions, the motion equation (5.65), becomes :

$$v'_i = R\, v_i \quad / \; i=1,2,3 \ldots n$$

where R is the (**orthogonal**) rotation matrix.

If we know the correspondences of some points (at least three) then the matrix R c
principle be recovered, and such efforts have been published [Huang and Blonstein, 1
. But we would like to recover matrix R without using any point correspondences.

Let,

$$v_i = (v_{x_i}, v_{y_i}, v_{z_i}) \quad / \; i=1,2,3 \ldots n$$
$$v'_i = (v'_{x_i}, v'_{y_i}, v'_{z_i}) \quad / \; i=1,2,3 \ldots n$$

Note that v_i and v'_i are the position vectors of the members of sets A and A' respecti
with respect to their center-of-mass coordinate systems.

We wish to find a quantity that will uniquely characterize the whole sets A and
terms of their " relationship " (rigid motion transformation). We have found tha
matrix consisting of the second order moments of the vectors v_i and v'_i has t
properties. In particular, let

$$V \equiv \begin{bmatrix} \sum_{i=1}^{n} v^2_{x_i} & \sum_{i=1}^{n} v_{x_i} v_{y_i} & \sum_{i=1}^{n} v_{x_i} v_{z_i} \\ \sum_{i=1}^{n} v_{y_i} v_{x_i} & \sum_{i=1}^{n} v^2_{y_i} & \sum_{i=1}^{n} v_{y_i} v_{z_i} \\ \sum_{i=1}^{n} v_{x_i} v_{z_i} & \sum_{i=1}^{n} v_{y_i} v_{z_i} & \sum_{i=1}^{n} v^2_{z_i} \end{bmatrix}$$

$$V' \equiv \begin{bmatrix} \sum_{i=1}^{n} v'^{2}_{x_i} & \sum_{i=1}^{n} v'_{x_i} v'_{y_i} & \sum_{i=1}^{n} v'_{x_i} v'_{z_i} \\ \sum_{i=1}^{n} v'_{y_i} v'_{x_i} & \sum_{i=1}^{n} v'^{2}_{y_i} & \sum_{i=1}^{n} v'_{y_i} v'_{z_i} \\ \sum_{i=1}^{n} v'_{x_i} v'_{z_i} & \sum_{i=1}^{n} v'_{y_i} v'_{z_i} & \sum_{i=1}^{n} v'^{2}_{z_i} \end{bmatrix}$$

From these relations, we have that:

$$V \overset{n}{_{i=1}} = E(v'_{Ii}, v'_{yi}, v'_{z_i})^{1}(v^{>}_{x_i}, v'_{yi}, v'_{z_i}) =$$

$$= \sum_{i=1}^{n} R(V^{*}_{i}, Vy_i, Vz_i)' (Vj_i, Vy_i, Vz_i) \; R' =$$

$$= RVR^{1}$$

So, $$V' = RVR' \qquad (5.67)$$

At this point it should be mentioned that equation (5.67) represents an invariance between the two sets of 3-D points A and A', since the matrices V and V are similar. In other words we have discovered that matrix V remains invariant under rigid motion transformation. From now on, the recovery of the rotation matrix R is simple and comes from basic Linear Algebra.Furthermore equation (5.67) implies that the matrices V and V* have the same set of eigenvalues [Stewart, 1980].

But sinceV and V* are symmetric matrices, they can be expanded in their eigenvalue decomposition, i.e. thcro ovict motrirofl S T eiiv> that•

$$V = SDS^t \quad (5.68)$$

$$V' = TDT^t \quad (5.69)$$

where S, T are orthogonal matrices having as columns the eigenvectors of the matri and V' respectively (e.g. i-th column corresponding to the i-th eigenvalue) an diagonal matrix consisting of the eigenvalues of the matrices V and V'. We ha mention at this point that in order to make the decomposition unique we require tha eigenvectors in the columns of matrices S and T be orthonormal.

From equations (5.67), (5.68), (5.69) we derive that matrices T and R S both co of the orthonormal eigenvectors of matrix V'. In other words, the columns of matrice and T must be the same, with a possible change of sign. So, the matrix RS is equal t of eight possible matrices, T_i , $i=1,..,8$. Thus, $R=T_iS^T$, $i=1,..,8$. But the rotation m is orthogonal and it has determinant equal to one. Furthermore, if we apply matrix the set of vectors v_i then we should get the set of vectors v_i'. So, given the above t conditions and Chasles theorem, the matrix R can be computed uniquely.

There is something to be said about the uniqueness properties of the algori When all the eigenvalues of the matrix V have multiplicity one then the problem h unique solution. When there are eigenvalues with multiplicity more than one, then t is some inherent symmetry in the problem that exhibits some degeneracy properties. example, if the surface in view (i.e. the surface on which the points lie) is a soli revolution, then there is an eigenvalue (of the matrix V) with multiplicity 2, and only eigenvector corresponding to the axis of revolution can be found. The other eigenvectors define a plane vertical to the axis of revolution. So, in this case there i inherent degeneracy. We are currently working towards a complete mathemat characterization of the degenerate cases of the problem. We are also develo experiments to test the robustness of the method as well as setting up the equipment experimentation in natural images. The study of the sensitivity of the algorithm respect to different number of points in the successive dynamic frames, is one of future goals. The algorithm is not sensitive in small perturbations of the po [Aloimonos et al, 1986].

5.7.4.7.1. Experiments.

We will describe experiments for both the detection of structure and depth without correspondence and the detection of 3-D motion without correspondence for the case of planar surfaces. Experiments for the case of curved (general) surfaces are under development.

In our experiments, we considered a set of three-dimensional planar points, which we projected perspectively in both the left and right frames. From the projections we recover the structure and depth of the 3-D plane using the alogrithm described in Section 3 , or using the projections in three frames . It is clear, that the equations that are used to develop the linear system described in Section 5.7.4.3, are based on the assumption that the number of points on (left and right frames), is the same. But in noisy situations, this is not the case. In particular, in real images operators have first to be applied on all four frames (two before the motion and two after the motion) that will produce points of interest, and then the theory developed in this-paper is applied to these points.

But any method that will produce points of interest from intensity images is bound to have errors due to the noise in the images and the unpredictable behavior of the intensity function in natural scenes. When we say that the methods that find interesting points in intensity images are bound to errors, we mean that there will be points in the left frame whose corresponding ones have not been found in the right stereo frame, and also there will be points in the first dynamic frame whose cor- responding ones have not been found in the second dynamic frame, and vice-versa. So, the number of points will not be the same in the different images . Because of that, our method is bound to have an error, since it is based on the assumption that the number of points is everywhere the same. To reduce this error we do the following: Equations (5.47), (5.48), (5.49) are not affected if both sides are divided by the number of points in all the frames (under the assumption that the number of points is the same in all frames). If now the numbers of points in the left and right fram e are different, say $n\hat{}ft$ and n ^ ^ , in the static stereo case, then we divide the summations resulting from each of the frames, by the number of points of the corresponding frame, and the resulting equations are (for the static stereo case):

$$\sum_{i=1}^{n} \frac{x_{li}\, y_{li}^{k1}}{f dn_{left}} - \sum_{i=1}^{n} \frac{x_{ri}\, y_{ri}^{k1}}{f dn_{right}} = \frac{1}{cn_{left}} \sum_{i=1}^{n} y_{li}^{k1} - \frac{1}{cfn_{left}} [\sum_{i=1}^{n} p x_{li} y_{li}^{k1} + \sum_{i=1}^{n} q y_{li} {}^{*} y_{li}^{k1}$$

$$\sum_{i=1}^{n} \frac{x_{li}\, y_{li}^{k2}}{f dn_{left}} - \sum_{i=1}^{n} \frac{x_{ri}\, y_{ri}^{k2}}{f dn_{right}} = \frac{1}{cn_{left}} \sum_{i=1}^{n} y_{li}^{k2} - \frac{1}{cfn_{left}} [\sum_{i=1}^{n} p x_{li} y_{li}^{k2} + \sum_{i=1}^{n} q y_{li} {}^{*} y_{li}^{k2}$$

$$\sum_{i=1}^{n} \frac{x_{li}\, y_{li}^{k3}}{f dn_{left}} - \sum_{i=1}^{n} \frac{x_{ri}\, y_{ri}^{k3}}{f dn_{right}} = \frac{1}{cn_{left}} \sum_{i=1}^{n} y_{li}^{k3} - \frac{1}{cfn_{left}} [\sum_{i=1}^{n} p x_{li} y_{li}^{k3} + \sum_{i=1}^{n} q y_{li} y_{li}^{k3}]$$

where n_{left} and n_{right} represent the numbers of points in the left and right frames respectively. It is clear that the resulting equations are approximate, but our experiir show that the introduced error is very small. It has to be mentioned , however, tha intrinsic difficulty, appearing in the traditional methods (i.e. stereo, optical flow), o being able to find corresponding points, exists even in our algorithm but under the foi different numbers of points in the different frames, because of the globality of approach. However, even considerable differences in the numbers of points among different frames hardly affects the results. Furthermore, the same technique is appli the case of motion as well.

Figure 5.10 shows the projections of a set of planar points on both the left and i frames. The frame on top is the superposition of the left and right frames. The a< parameters of the plane were:

p = o.O, q = 0.0, c = 10000, while the number of points was equal to 1000.

We did not include any noise to our pictures.

The computed ones were: P=-0.0 , Q = -0.0, C = 10000.0

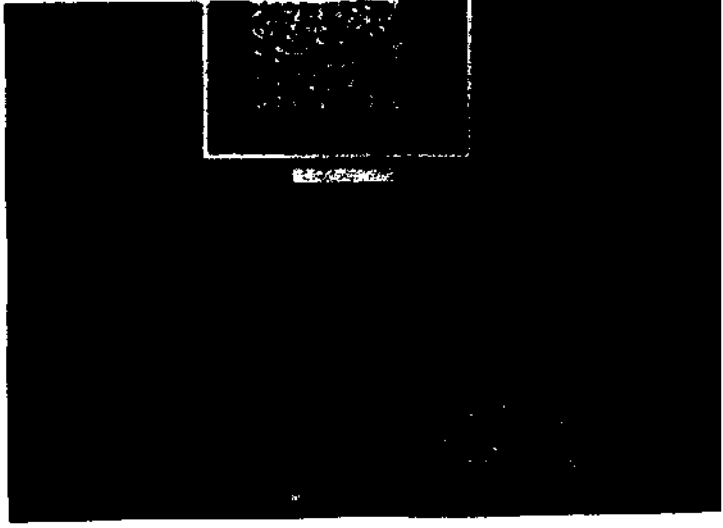


Figure 5.1(h

Stereo without correspondence

Figure 5.11 shows the projections of a set of planar points on both the left and right frames. The frame on top is the superposition of the left and right frames. The actual parameters of the plane were:

p = 1.0, q = 1.0, c =-1*0000, while the number of points was equal to 1000.

We did not include any noise to our pictures.

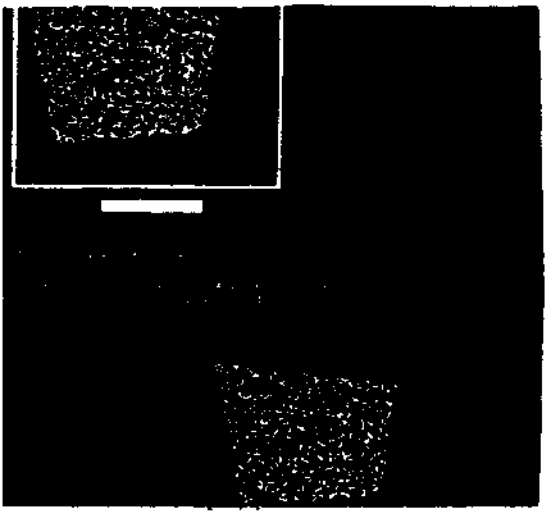
The computed ones were: P = 0.98, Q = 1.00, C = 9809.8

Figure 5.12. shows the projections of a set of planar points on both the left and right frames. The frame on top is the superposition of the left and right frames. The actual parameters of the plane were:

p = 1.0, q = 1.0, c = 10000, while the number of points was equal to 1000.

We included *5%* noise to the left frame and 7% to the right one.

The computed ones were: P = 1.7, Q = 1.2, C = 10266.7



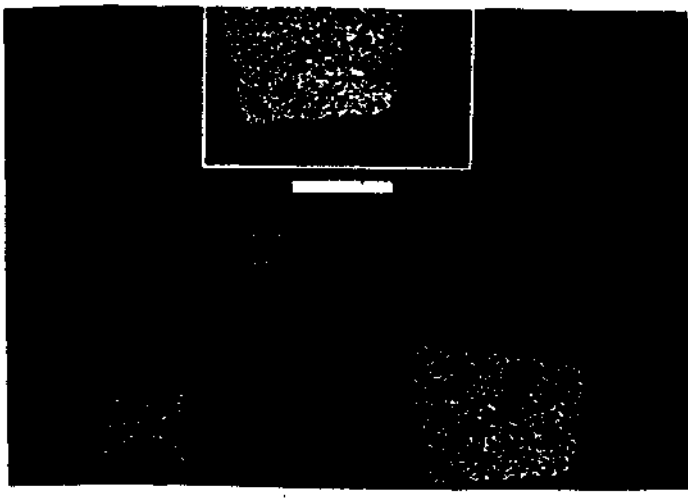


Figure 5.11: **Figure 5.12**

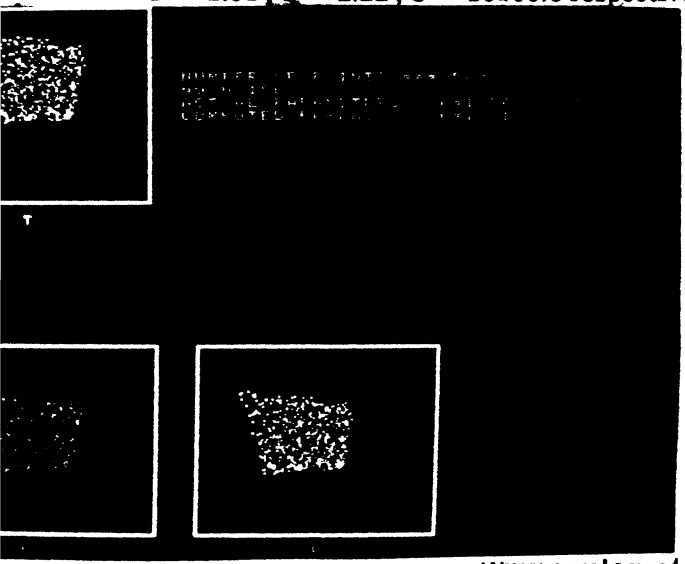
Figures 5.13a., 5.13b. show the results from the *3-eye method.* Here the projections of a set of 3-D planar points on all the three frames are considered. The actual parameters were:

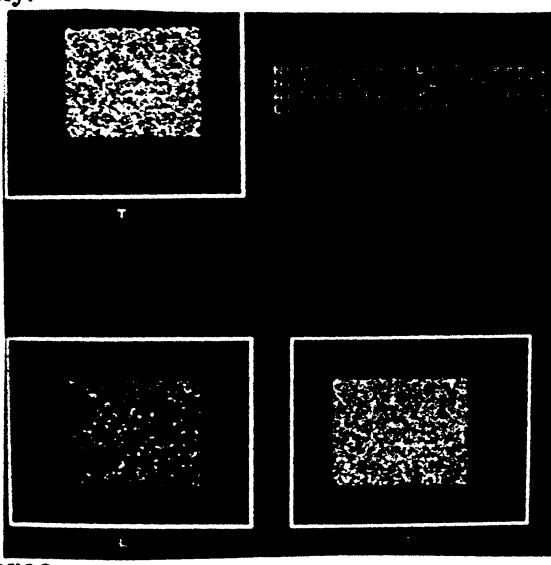
p = 0.0 , q = 0.0 , c = 10000 (Figure 5.13a.) and p = 1.50 , q = 2.30 , c = 10000 (Figure 5.13b.) respectively. The number of points was equal to 1000, in both pictures.

Picture 5.13b. did not have any noise, whereas Figure 5.13a. had 5% noise in the left frame and 7% noise in the right and top frames.

The computed ones were: P = 0.10 , Q = 0.05 , C = 10197.0 and

P = 1.51 , Q = 2.22 , C = 10000.0 respectively.





Trinocular stereo

Figure 5.13a. **Figure 5.13b**

Figures 5.14,5.15,5.16,5.17,5.18, show the 3-D motion determination results. In figure 5.5., the two frames at the bottom represent the projections of a set of 3-D planar points on the left and right eyes respectively. The two frames at the top, represent the projections of the same set of points, after it has been translated. The actual direction of translation was equal to (-2.0 , 2.0), and the computed one was (-1.9, 2.0).

The noise percentage was equal to 10% in all four frames while the number of points was equal to 1000. At this point it has to be mentioned that the parameters p,q were also computed computed,since the latter are used in the determination of the direction of translation. Figures 5.15 and 5.16, represent similar experiments.

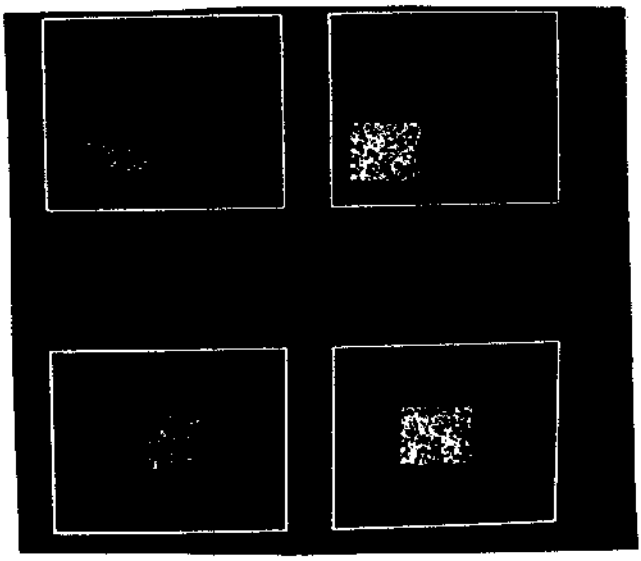
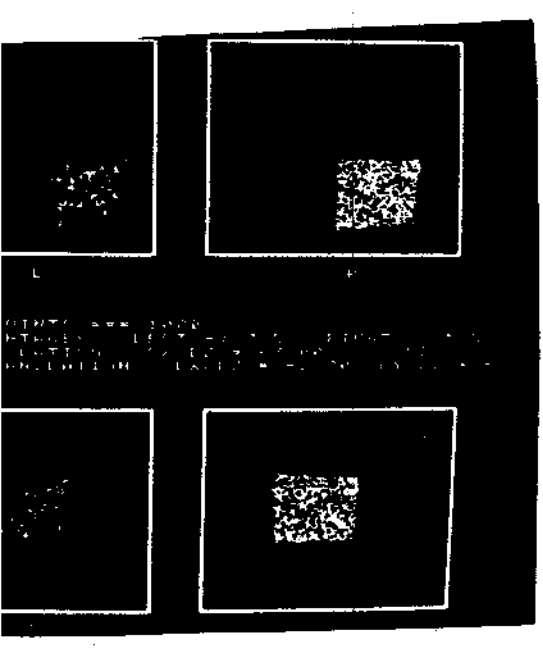


Figure 5.14: Direction of translation without Correspondence



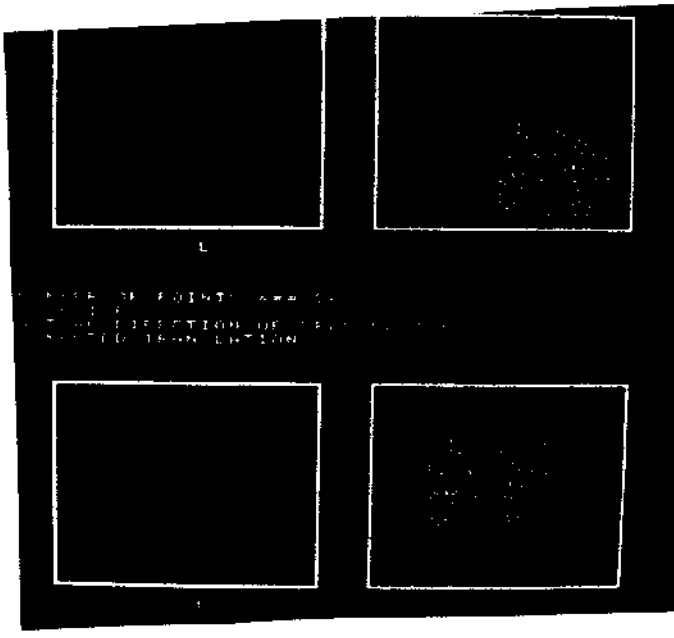
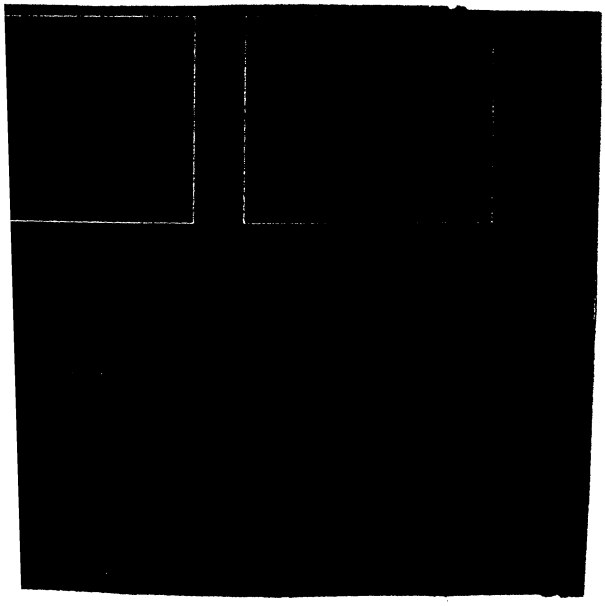


Figure 5.15 **Figure 5.16**

Figures 5.17 and 5.18, show experiments determining the general motion . The results were computed according to the method presented in section 5.7.4.6, and the results were recalculated with respect to the left-camera coordinate system.



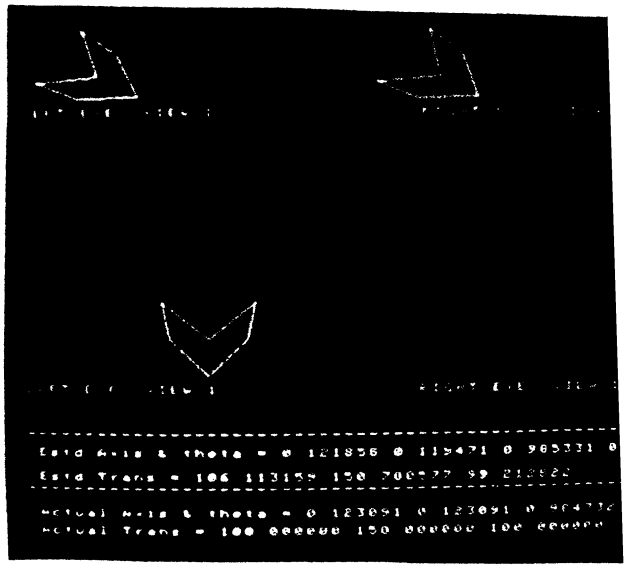


Figure 5.17 **Figure 5.18**

NOTE: All the parameters involved in the above experiments that have a dimension of length ($L^1 M^0 T^0$) a re calculated in pixels , where 1 pixel = 100μm.

5.8. Conclusion and future work.

We have presented a method on how a binocular (or trinocular) observer can recover the structure, depth, and 3-D motion of rigidly moving surface patch without using any static or dynamic point correspondences. It is one of our future goals to experiment for

the application of the method in natural images. We will also work towards the analysis of nonrigid motion and occluded scenes.

6

Shape and 3-D Motion from Contour

Results

In this chapter we study the detection of surface shape and three-dimensional motion from the perception of a planar contour. We prove that a binocular observer can compute the orientation and the 3-D motion of a moving contour without using point to point correspondences.In particular:

1) We develop constraints between the coordinates of the points that constitute the contours in the left and right retina of a binocular observer that enable him to detect the structure and the depth of the plane in view without using any point to point correspondences.

2) We develop constraints between the lengths and the areas of the contours in the left and the right retina of a binocular observer that enable him to compute the structure and the depth of the plane in view without any point correspondences. These constraints are of significant value by their own, and they can be successfully used in many related areas, as object recognition and identification.

3) We discover constraints between the retinal motions of the contour and its three-dimensional motion that make it possible to recover 3-D motion without any correspondences,

4) and finally we generalize some of the above results for a monocular observer. In particular, a translating monocular observer can recover the shape of an imaged contour without using any point to point crspondences.

The basic assumption here is that the contours in the left and right images have been found and the correspondence between them has been established.

6*1 Introduction

The human perceiver is able to derive enormous amounts of information from the **contours** in **a** scene. As part of this **capacity,** we **are** able to use the shapes of image contours **(as** they are seen by both eyes) to infer **the shapes** and dispositions in space of the surfaces they lie on, as well as their motion. To the extent the inferences we draw are accurate, our strategies for drawing them must have some basis in the character of the visual world, just as the efficacy of stereopsis as a source for depth information has a basis in the geometry of projection and triangulation. The aim of the research described here is (1) *to discover constraints on the visual world that allow surface shape and motion to be reliably inferred from contours in images,* (2) *to derive methods of inference from these constraints.* The interpretation of contours by a binocular observer falls into four subproblems (following Witkin, [Witkin 1981]). In particular these four subproblems are the following:

a) *Locating contours in the images.*

If contours are to be used to infer anything, they must be found. The human perceiver has little difficulty deciding what is and is not a contour, yet the automatic detection of edges has proved very difficult. Perhaps this fact should not be surprising; the contours that we see in natural images usually correspond to definite physical events, such as shadows, depth discontinuities, color differences and the like. Our ability to detect these events may say more about their significance for image interpretation than about their ease of detection. Why should we expect events that have simple descriptions in terms of the structure of the scene to have simple descriptions in terms of the image intensity as well? If the physical significance of contours is taken as their primary feature, then at least we know what is being detected, even if we don't know how. But recent research [Nalwa, 1985] shows that we are in pretty good state as far as detection of contours goes. Actually, we can say that we can fairly well detect the contours in an image, even if there are some inaccuracies.

b) *Labeling contours (i.e. distinguishing contours which are due to different physical events)*

If contours correspond to different physical events, then an essential component of their interpretation must be to decide which contours denote which event, since each kind

of contour imparts a different meaning. Recent work has shown that strong struc constraints can be applied to distinguish one kind of contour from another.

c) *Corresponding contours (i.e. finding which contours in the left and right images a image of the same 3-D contour).*

Before we apply some interpretation method to the images of the contours (lef right), we should know which contours in both images correspond to each other, *i.e.* are the images of the same three-dimensional contour.

d) *Interpreting contours.*

Even after contours have been found, labeled and the corresponding ones in th and right images have been identified, not much is known about the physical structu the scene, if we don't wish to resolve in a point-to-point correspondence between th and right images. It is clear that contours play an important role in the hu perceiver's ability to decide how things are shaped and where they are, apart fror application of specific "higher level" knowledge to objects of known shape. This res addresses this fourth problem, i.e., given the left and right image of a moving plana contour, to recover its orientation, depth and 3-D motion, without using any point-to- correspondence neither between the left and right images nor between the dyn frames. The reason that we want to solve the problem without using p correspondences is that correspondence is a very hard problem and it does not tractable with the available tools. So, we would like to address the problem in such a that we avoid the correspondence problem.

6.2 Motivation

This research is motivated by the inherent difficulties of the conventional s problem as well as the difficulty of the dynamic correspondence problem (to recover flow or discrete displacements, that will be used for the recovery of 3-D motion). criticism about the difficulty of dynamic correspondence was presented in the pre chapter.

Passive ranging by triangulation methods, which is employed successfully by hu under certain conditions, has received much attention in computer vision literatu recent years [Jarvis, 1983]. It is obvious that the ability to recover absolute ran objects in a scene would be important in a variety of robotic applications. To date, two basic methods of passive ranging have been reported, the "static stereo," i.e. the u two cameras separated by a known baseline and "motion stereo," i.e., the use of a s

camera moving in a known way through a stationary scene. Recently, a new concept has been introduced for passive ranging to moving objects, termed "dynamic stereo," which is based on the comparison of multiple image flows [Waxman *et* a/., 1984]. In the sequel, we will only deal with the criticism of the first method (static stereo). Most of the literature on passive ranging has been concerned with the difficult "correspondence" problem associated with the assignment of stereo disparities (for the static stereo method). Beside the traditional method of intensity correlation between images, much attention has been paid to the theory of Marr and Poggio [1979], with implementation by Grimson [1981], The use of more than two camera locations, to aid in solving the correspondence between images, has been approached in different ways by Tsai [1983] and Moravec [1981]. Nevertheless, solution of this correspondence problem remains a computationally expensive and slow process, with partial success in a variety of input images. Moreover, a maximum ranging distance is implied by the finite resolution of the cameras and the statically configured baseline between cameras. Most of the work needed to solve the correspondence problem deals with the matching of microfeatures, such as points of interest (corners, high curvature points), and edges. A natural question that arises then, is: Is it possible to recover structure and depth, given that we have matched a macrofeature (i.e., a planar contour) instead of a microfeature? We prove that it is. Of course in this study we don't deal with "how to match the planar contours in the two stereo frames," i.e., to find in both images the contours which are due to the projection of the same three-dimensional planar contour (size, color, texture, fractal dimension could be used for the solution of this problem). Also, it has to be realized that the constraints for the static stereo problem are unique. The constraints cannot change. But the method we propose, which is based on a global approach, can be considered as immune to noise, since it gives very good results when the images are corrupted with noise up to 7%.

We also show that it is possible to solve the 3-D motion determination problem without using point-to-point correpondence for the case where the imaged object is a planar contour. In the previous Chapter we showed that this is possible for a collection of points. Here,we show that it is possible for the case of a planar contour, i.e., a binocular observer can understand the 3-D motion of a contour, from two temporally close positions of the contour, without using any point-to-point correspondence. Of course there are still difficulties with this new approach and the inherent problems of the dynamic imagery appear in another form, different from the one of the traditional methods (one camera —>

retinal motion --> 3-D motion); but it turns out that these problems, in the prese small noise percentages, hardly affect the results.

The organization of this Chapter is as follows. Section 6.3 describes previous Section 6.4 introduces the concept of "aggregate stereo," a method that compute structure and depth of a 3-D planar contour from its images on the left and righ retina, and that was basically presented in the previous chapter.. Section 6.5 intro new constraints for the stereo problem, which are not based on triangulation, but o change of area and perimeter in the left and right images of the contour. Sectio introduces the concept of determining the direction of the translation of a transl planar contour, without using any point-to-point correspondence, and introduce reader to Section 6.7 which deals with the solution of the general problem (the case v the 3-D planar contour is translating and rotating).

In what follows, because of the discrete nature of images, we will consider a co either as a collection of points (which it actually is) or as a continuous curve, when n to establish the mathematical rigorousness of a proof.

6.3 Previous Work

The idea of using more than one camera to recover the shape of a contour seems new.

The recovery of three-dimensional shape and surface orientation from a dimensional contour is a fundamental process in any visual system. Recently, a nu of methods have been proposed for computing this shape from contour. For the most previous techniques have concentrated on trying to identify a few simple, ge constraints and assumptions that are consistent with the nature of all possible object imaging geometries in order to recover a single "best" interpretation, from amon many possible for a given image. For example, Kanade [1981] defines shape constr in terms of image space regularities such as parallel lines and skew symmetries u orthographic projection. Witkin [1981] looks for the most uniform distributi tangents to a contour over a set of possible inverse projections in object space u orthography. Similarly, Brady and Yuille [1984] search for the most compact s (using the measure of area over perimeter squared) in the object space of inverse proj planar contours.

Rather than attempting to maximize some general shape-based evaluation fun over the space of possible inverse projective transforms of a given image contour

keeping in our framework of attempting unique solutions without employing any restrictive assumptions and heuristics, we propose to find a unique solution by using more than one camera, since it can be easily proved that only one image (under orthography or perspective) of a planar contour admits infinite intepretations of the structure of the world plane on which the contour lies, if no other information is known. Finally, the need for a unique solution, which is guaranteed in our approach, comes also from the fact that there exist many real world counterexamples to the evaluation functions that have been developed to date. For example, Kanade's and Witkin's measures incorrectly estimate surface orientation for regular shapes such as ellipses (which are often interpreted as slanted circles). Brady's compactness measure does not correctly interpret non-compact figures such as rectangles since he will compute it to be a rotated square (e.g. if we view a rectangular table top, we do not see it as a rotated square surface, but as a rotated rectangle.)

Finally, the need for the solution of the 3-D motion parameters determination problem without using point-to-point correspondence has recently been appreciated by Kanatani [1985]. But the proposed methods, despite their mathematical elegance, are quite artificial and subject to numerical errors. The methods that we will propose in the following sections are quite intuitive and can be considered immune to small noise percentages.

6.4. Aggregate Stereo

In this section we present a theory for the recovery of the three-dimensional parameters of a planar contour, from its left and right images, without using any point-to-point correspondence. Instead, we consider all the point correspondences at once; thus, there is no need for the solution of the correspondence problem of points. Correspondence of the contours as a whole is required.

Let a coordinate system OXYZ be fixed with respect to the left camera, with the Z axis pointing along the optical axis. We consider that the image plane I_{mi} is perpendicular to the Z axis at the point (0,0,1). Let the nodal point of the right camera be the point (d,0,0), and its image plane I_{mr} identical to the previous one. Consider also a plane P in the world with equation $Z = pX + qY + c$, which contains a contour C and consider the images (perspective) C/ and C_r of the contour on the left and right image planes respectively (See Fig. 6.0). From this point we will denote the coordinates on the left and right image planes by (x/,y/) and (x_r,y_r) respectively. We consider every contour on each image plane

as a collection of points. So,

$$C_l = \left\{ (x_{l_i}, y_{l_i}) \;\middle|\; i = 1, \ldots n \right\} \text{ and } C_r = \left\{ (x_{r_i}, y_{r_i}) \;\middle|\; i = 1, \ldots, n \right\}$$

Then with the method that was analyzed in section 5.7.4.1 we can recover orientation anddeptitof the contour.

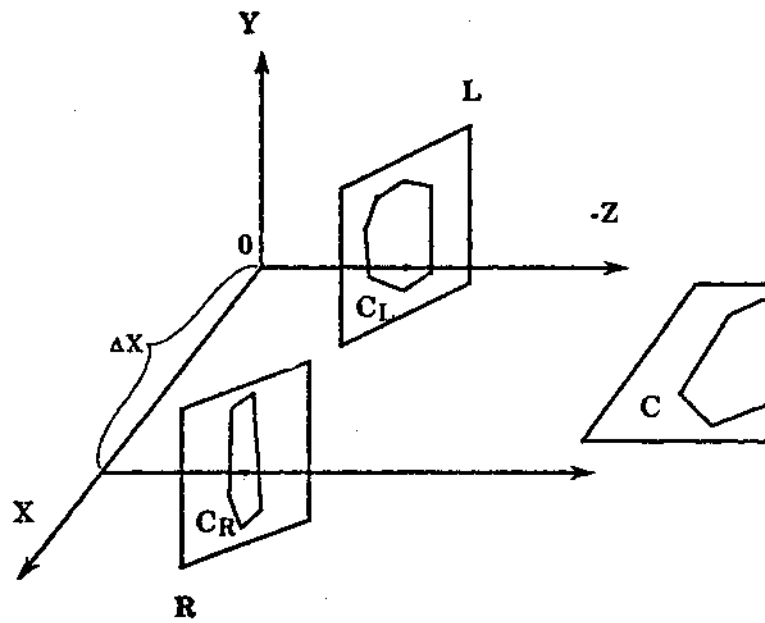


Figure 6.0

The algorithm, that is the same as in section 5.7.4.3, is not sensitive to small i percentages, as it was observed from experiments.

It is obvious that in this case the triangulation constraint has been aggregated « the three-dimensional surface (plane) can be represented with few (3) parame Simulations on synthetic data with different percentages of uniform noise (up to 7< both frames (left and right) indicate that the algorithm is immune to noise, since the« in the resulting plane parameters (p,q,c) is negligible.

At this point we should also explain what we mean by noise. When we hav. images of a contour, and we say that the images are corrupted by noise a«. we mean

we randomly drop from both frames (left and right contour images) a_l% and a_r% of the points that constitute the left and right contour respectively, with $a = (a_l + a_r)/2$. Such a noisy situation is te be expected in real images, due to perspective effects and bad behaving intensity functions. (We should remember that the contour points will be extracted from intensity images.) Finally, in the case where we are imaging a textured planar surface, we first preprocess the left and right images to extract points [Bandyopadhay, 1984; Kitchen & Rosenfeld, 1980; Moravec, 1977], and on these points we apply the algorithm of Section 2.2. But this algorithm expects the same number of points in both frames, something that will not be the case in actual situations, because of the following two problems:

a) Any method that finds interesting points from intensity images is bound to errors, i.e., there will be points in the left frame for which there will not exist corresponding ones in the right frame, and vice versa.

b) There are points seen by the left camera which are not seen by the right camera, and vice versa.

To simulate the effects of the noise due to the above reasons, we add random points to both frames. When we say that there is noise a%, we mean that we have added a_l% and a_r% random points in the left and right frames respectively, with $a = (a_l + a_r)/2$. In a later section we describe relevant experiments, and explain some techniques that have been used in the actual implementations in order to reduce the error in the computed parameters.

6.5. Orientation of a contour without correspondence

In this section, we show how to recover the orientation of a planar contour without using any correspondence between the left and right images of the contour and without basing our approach on the triangulation procedure. To do this, we need some technical prerequisites, which are introduced in the next section. In particular, we will describe the co-called paraperspective projection, which is an approximation of the perspective. The results that we will get can be generalized for the case of the perspective projection. But we present the results first for the case of the paraperspective projection because of the intuition behind it and because of the natural extension of the results for the perspective projection. The paraperspective projection has been already analyzed in Chapters 2 and 3.

The area ratio constraint

We have seen that the paraperspective projection is an affine transformation (see 3.1.2). The determinant of the matrix of an affine transformation is equal to the rat the areas of the two patterns before and after the transformation. Specifically, if S_W i area of a world contour that lies on a plane with gradient (p,q) and S_I is the area c image that has mass center (A,B), then we have:

$$\frac{S_I}{S_W} = \frac{1}{\beta^2} \, det \begin{bmatrix} \frac{-1+pA}{\sqrt{(1+p^2)}} & \frac{pB}{\sqrt{(1+p^2)}} \\ \frac{q(p+A)}{\sqrt{(1+p^2)(1+p^2+q^2)}} & \frac{qB-p^2-1}{\sqrt{(1+p^2)(1+p^2+q^2)}} \end{bmatrix}$$

or

$$\frac{S_I}{S_W} = \frac{1}{\beta^2} \cdot \frac{1-Ap-Bq}{\sqrt{(1+p^2+q^2)}}$$

or

$$S_I = \frac{S_W}{\beta^2} \cdot \frac{1-Ap-Bq}{\sqrt{1+p^2+q^2)}} \quad (6.8)$$

Equation (6.8) relates the area of a world contour S_W, its gradient (p,q), the area its image and its mass center (A,B). If we call the quantity S_I "textural intensity," an quantity S_W/β^2 "textural albedo," then equation (6.8) is very similar to the in irradiance equation for Lambertian surfaces:

$$I = \lambda \frac{1+Ap+Bq}{\sqrt{(1+p^2+q^2)}}$$

where (p,q) is the gradient of the surface point whose image has intensity I, λ is the al at that point and (A,B,1) the direction of the light source [Horn, 1977; Ikeuchi, 19 Thus equation (6.8) can be used to recover surface orientation.

In the sequel we present a theory for the recovery of shape from contour. Our ana is based on three views or on two views. We proceed with the following proposition.

6.5.1 Shape from change in the area of a contour in three frames

6.5.1.1 Proposition

Let a coordinate system OXYZ be fixed, with the -Z axis pointing along the optical axis. We consider that the image plane Imi is perpendicular to the Z axis at the point (0.0,-1). Consider a plane II with equation -Z = pX + qY 4- c in the world, where (p,q) is the gradient of the plane that contains a contour C. Furthermore, we consider two more cameras with image planes Im_2 and Im_3, whose coordinate systems (nodal points) are such that any world point has the same depth with respect to any of the cameras . Then assuming paraperspective projection of the contour C on the image planes, the images C_x, C_2, and C_3 of the contour on the three cameras are enough to determine uniquely the orientation of the plane II, without having to solve the point-to-point correspondence between Cl, C2 and C3.

Proof

Let S_x, S_2, and S_3 be the areas of the contours C_x, C_2 and C_3 respectively. Let also the depth of the center of gravity of the contour C be 0. If S_w is the area of the contour C on the plane II, and $(A_1$ JB^t (A_2,B_2) and (A_3,B_3) the centers of gravity of the image contours C_1, C_2 and C_3 respectively, then by dividing appropriately the area ratio constraints (previous section), we get:

$$\frac{S_1}{F_z} \bullet \sqrt{1-A_1p-B_1q} \quad (612)$$

$$\frac{S_2}{S_3} = \frac{1-A_2p-B_2q}{1-A_3p-B_3q} \quad (613)$$

Equations (6.12) and (6.13) constitute a linear system with unknowns p and q, which in general has a unique solution (q.e.d.).

A degenerate case in the solution of the above system arises when the centers of all three image planes are collinear. Experiments using the above method on perspective images computed the orientation of the world contour with great accuracy. This is due to the fact that equations (6.12) and (6.13), despite the fact that they were developed under the paraperspective projection assumption, are true under perspective too, as we prove in the Appendix.

We now proceed to solve the same problem, but given two images of the contour.

6.5.2 Solving the problem with two frames

In the previous section, we used three frames for the recovery of shape from cor
But the information we used from the image contours was only their area, a
particular how the area was changing from view to view. A useful piece of inform
that we have not yet utilized is the length of the contour (which is of course independ
its area in general). Using this information, we can solve the shape from contour pr
with two projections (binocular observer) but in a computationally much harder
involving nonlinear equations.

Consider a coordinate system OXYZ to be fixed with respect to the left camera,
the -Z axis again pointing along the optical axis. We consider that the image plane
left camera is perpendicular to the Z axis at the point (0.0,-1). The nodal point of the
camera is the point (Δx,0,0) and the image plane of the right camera is identical to th
of the left camera. C is a contour on the world plane Π with equation $-Z = pX+qY+$
C_L and C_R are the projections of the contour C on the left and right image respect
using the paraperspective projection. We can easily prove, assuming paraperspe
projection, [Aloimonos et al., 1985], that a small line segment (l cos θ. l sin θ) o
image plane is due to the projection of a line segment on the world plane, with length
$l L_\theta$, with

$$L_\theta = \frac{c}{(1-Ap-Bq)^2} \sqrt{(k_1 cos^2\theta + k_2 sin^2\theta + k_3 sin\theta cos\theta)}$$

where:

$k_1 = (1\text{-}qB)^2 + (pB)^2 + p^2$

$k_2 = (1\text{-}pA)^2 + (qA)^2 + q^2$

$k_3 = 2((1\text{-}qB)qA + (1\text{-}pA)pB + pq)$,

and (A,B) is the center of gravity of the area under consideration. So, given a conto
an image, if we break the contour into small line segments (edges) ($l_i \cos\theta_i$, $l_i \sin\theta_i$), i
..., n, then the length of the contour in the world plane is given by:

$$L = \sum_{i=1}^{n} l_i L_i$$

with

$$L_i = \frac{\beta}{1-AP-Bq}\sqrt{(k_1 cos^2\theta_i + k_2 sin^2\theta_i + k_3 cos\theta_i sin\theta_i)}$$

where k_1, k_2, k_3 are as above, and β is the depth of the center of gravity of the world contour. If we consider now the left and right images of the contour C , and we compute the length of the world contour from each one, we should find the same answer. In other words, if L_L and L_R are the length of the world contour that we compute from the left and right image, respectively, we must have

$$L_L = L_R \quad (6.14)$$

Equation (6.14) is an equation in the unknowns p,q, but it is in a complicated form that does not permit easy algebraic manipulations.

On the other hand, if S_W, S_L, S_R are the areas of the world contour, the left image contour and the right image contour respectively, then we have

$$\frac{S_L}{S_W} = \frac{1}{\beta^2}\,\frac{1-A_L p-B_L q}{\sqrt{(1+p^2+q^2)}} \quad (6.15)$$

and

$$\frac{S_R}{S_W} = \frac{1}{\beta^2}\,\frac{1-A_R p-B_R q}{\sqrt{(1+p^2+q^2)}} \quad (6.16)$$

where (A_L,B_L) and (A_R,B_R) are the centers of gravity of the left and right image contour respectively. From (6.15) and (6.16) we conclude

$$\frac{S_L}{S_R} = \frac{1-A_L p-B_L q}{1-A_L p-B_L q} \quad (6.17)$$

Equation (6.17) represents a straight line in gradient space, or a great circle in the (equivalent) Gaussian sphere formalism. Equations (6.14) and (6.17) constitute a nonlinear system in the unknowns p and q. Experimental results, based on the following discrete method, indicate that there exists a unique solution. The discrete method we used is as follows: Equation (6.14) represents a great circle in the Gaussian sphere (constant azimuth, varying elevation). By taking different values for the elevation angle

(180 values, if the different values are 1 degree apart) we solve for the gradient p,q an choose the p,q that makes the function
$(L_L - L_R)^2$ minimum.

6.5.3 Solving the problem with two frames and without the paraperspec approximation

In the previous section we presented a method for the recovery of the shape contour from two images (binocular observer) under the paraperspective proje assumption. In this section we show that the problem can be solved by assu perspective projection, but the solution is the same, with the method in the pre section being better for its simplicity. The method presented in the previous section based on Equations (6.14) (lengths) and (6.17) (areas). Equation (6.17), despite the that it was developed under the paraperspective projection model, is exact. Wha mean is that equation (6.17) is true under perspective projection and a proof of this c is given in the Appendix. So, in this Section we shall show that an equation analogo (6.14) can be developed if we assume perspective projection.

For that, we need to develop the first fundamental form of the world plane a function of the retinal coordinates, in order to be able to compute the length of the w contour (up to a constant factor, of course), and use it in an equation analogous to (6.1 we fix a coordinate system OXYZ with the Z axis as the optical axis and focal length F we consider a plane $\Pi : Z = pX + qY + c$ in the world with a contour C on it, an denote by (x,y) the coordinates on the image plane, then a point (X,Y,Z) in the w planar contour C is projected onto the point:

$$x = \frac{XF}{Z} \; ; \; y = \frac{YF}{Z} \quad (6.18)$$

The inverse imaging function, call it f, is the function that maps the image plane the world plane; so, if (x,y) is an image point, the 3-D world point on the plane Z = p qY + c that has (x,y) as its image, is given by

$$f(x,y) = \left(\frac{cx}{(F - px - qy}, \frac{cy}{F - px - qy}, \frac{cF}{F - px - qy} \right)$$

The first fundamental form of f [Lipschutz, 1969] is the quadratic form

$$Edx^2 + 2Fdxdy + E\,dy^2 ,$$

with

$E = f_s f_x$
$F = f_f f_y$ **and**
$G = f_y \cdot f_y$.

If we consider two points *(xj)* and *(x+dx, y+dy)* on the image plane, then the three-dimensional distance dC of the corresponding points on the world plane is given by:

$$dC = \sqrt{(Edx^2 + 2F\,dx\,dy + G\,dy^2} \quad (6.19)$$

Consequently, if we have a contour C on the image plane, then the 3-D planar contour has length:

$$\int_{Jc} V(Edx^2+2Fdxdy + Gdy^2) \quad (6.20)$$

The above expression (6.20) can be used to compute the quantities L_L and L_R , so that the equation $L_L = L_R$ can be developed. It has to be realized that this equation can be developed only in terms of p,q (the constants of the plane, which are different for the two frames, are eliminated).

6.5.4 A comparison between paraperspective and perspective projection

In the previous section, we showed how to develop an equation analogous to (6.14) which in conjunction with equation (6.17) would result in the recovery of the orientation (p,q) with exactly the same method presented in previous section. It is clear that in the method presented here the desired p,q are the values that minimize the function $(L_L - L_R)^2$ while satisfying equation (6.17). The difference between the method using paraperspective projection and using perspective projection is that the functions $(L_L - L_R)^2$ are different. But despite this fact, our experiments showed that the values of (p,q) that minimize $(L_L - L_R)^2$ while satisfying equation (6.17) are about the same in both the paraperspective and perspective cases. So, we find the paraperspective method more appealing, for the simple reason that it gives the same results with the perspective one

and is computationally simpler, since it does not have to approximate numerical integral (6.27), as the perspective method requires.

6.6 Finding the depth without triangulation

In the previous sections , we presented two methods on how to recover the shape of a planar contour, without correspondence, and without any triangulation. In section we show how to compute the depth of the 3-D planar contour (i.e., the param of the world plane). From equations (6.15) and (6.16) we get:

$$\frac{S_L}{S_R} = \frac{c-pd}{c} \left(\frac{1-A_L p - B_L q}{1-A_R p - B_R q} \right)^2$$

which is a linear equation in the unknown c. Of course, in the above equatio paraperspective projection is assumed, but the introduced error is negligible, a experiments at the end of the paper indicate.

So far, we have presented methods for the recovery of shape and depth from co using three or two frames (binocular observer). We now proceed to a method fo motion determination without having to find point-to-point correspondence betwee successive dynamic frames.

6.7. Determining 3-D motion without correspondence

Here we only treat the case of pure translation. The general case is treated in the section. The treatment in this section presumes real perspective projection paraperspective.

Consider a coordinate system OXYZ fixed with respect to the camera, O the point of the eye and the image plane perpendicular to the Z axis, (focal length 1) t pointing along the optical axis . Let us represent points on the image plane with letters ((x,y)) and points in the world with capital letters ((X,Y,Z)).

Let a point $P = (X_1, Y_1, Z_1)$ in the world have perspective image (x_1, y_1) where $x_1 =$ *and* $y_1 = Y_1/Z_1$. If the point P moves to the position $P' = (X_2, Y_2, Z_2)$ with

$$X_2 = X_1 + \Delta X$$
$$Y_2 = Y_1 + \Delta Y$$
$$Z_2 = Z_1 + \Delta Z ,$$

then we desire to find the direction of the translation ($\Delta X/\Delta Z$, $\Delta Y/\Delta Z$). If the image o (x_2, y_2), then the observed motion of the world point in the image plane is given b

displacement vector (x_2-x_1, y_2-y_1) (which in the case of very small motion is also known as optic flow).

We can easily prove that

$$x_2 - x_1 = \frac{\Delta X - x_1 \Delta Z}{z_1 + \Delta Z}$$

$$y_2 - y_1 = \frac{\Delta Y - y_1 \Delta Z}{z_1 + \Delta Z}$$

Under the assumption that the depth is large (and the motion in depth small), the equations above become:

$$x_2 - x_1 = \frac{\Delta X - x_1 \Delta Z}{Z} \quad (6.28)$$

$$y_2 - y_1 = \frac{\Delta Y - y_1 \Delta Z}{Z} \quad (6.29)$$

All the published methods for the recovery of the direction ($\Delta X/\Delta Z$, $\Delta Y/\Delta Z$) are based on equations (6.28) and (6.29) (see [Ullman 1979; Longuet-Higgins, 1981; Tsai & Huang, 1984; Bandyopadhyay & Aloimonos, 1985]), which of course require the knowledge of the correspondence between points in the successive frames. In the next section, we present a method for the recovery of the translational direction of a moving planar contour ($\Delta X/\Delta Z$, $\Delta Y/\Delta Z$), without having to solve the point-to-point correspondence problem.

6.7.1 Detecting 3-D direction of translation without correspondence

This case is exactly the same as the one described in section 5.7.4.5.2.

Experimental results based on this method are accurate and robust. A recent method presented by Kanatani [1985a, 1985b] has numerical instabilities that affect the desired result a great deal.

6.7.2 The aperture problem in the "large"

It seems, from the analysis in Section 6.7.1 (which is equivalent to 5.7.5.4.2), that the perspective effects are not taken into account. In other words, it is assumed that the

contour points are the same in number, before and after the motion. Of course, this i true in general, because of the perspective effects; so, in general, the number of poin the contours before and after the motion will not be the same. We call this prol "apperture problem in the large." The inherent difficulties of the point-to-point dyn correspondence problem are present in this method too, but in another form (diffe number of points in the two dynamic positions of the contour) because of the globali the approach. This fact should not be surprising, because the "constraints that relat retinal motion to the 3-D motion" have not changed. These constraints cannot cha and the algorithm in Section 6.7.1 is just "aggregating" the motion constraints. In o words, the method in Section 6.7.1 aggregates the motion constraints that have been in all the approaches that employ point-to-point correspondences. But the point tha raise is that despite this fact, if the motion is not large (so that the difference of number of points in the two dynamic frames is kept small), then the results are accurate. Later we describe relevant experiments and explain some techniques that used in actual implementations in order to reduce the error in the computed paramete

The next section deals with the general problem (unrestricted motion) and in this the "apperture problem in the large" is not present, since the analysis is done in 3-D. method in Section 6.7.1 is completely different from the method in Section 6.7.3 since latter uses more sources of information (depth) and in a way that does not require p correspondences.

6.7.3. Detecting 3-D motion without correspondence: General case

In the previous section we presented a method to recover the direction of translation of a translating planar contour, from the motion of its left and right image is clear that we did not use any depth information. In this section we present a metho how to recover the motion parameters of a rigidly moving planar contour. Any r motion can be represented as a rotation around an axis that we can freely choose to through any point of our choice, plus a translation. The problem then is reduce finding the translation and the rotation matrix.

So, suppose that we have four images of a moving planar contour (left and right be the motion, left and right after the motion). With the already presented methods , we recover the orientation and depth of the 3-D contour before and after the motion, an (p_1, q_1, c_1) and (p_2, q_2, c_2) be the parameters of the 3-D contour before and after the mo

respectively. But since we know the contour in 3-D, we will do our analysis of the motion in 3-D, instead of the image plane.

So, let

$$C_1=\{(jr_{.f}r._1,z.) \mid i=1,\ldots,n\}$$

and

$$C_2=\left\{(X_j^{'},Y_j^{'},Z_j^{'}) \mid j=1,\ldots,m\right\}$$

the two positions of the 3-D contour.

We assume that the rotation axis passes through the center of gravity of Ci. This has as an immediate consequence that the translation is given by the displacement of the center of the gravity between the two positons of the contour. So,

Translation = (AX, AY, AZ) = T =

= center of mass of C2-center of mass of Ci =

$$\frac{flK}{\backslash m}\quad\frac{l^{x}t}{n}\quad\frac{1^{Y}\backslash}{m}\qquad\frac{l^{Y}i}{n}\quad\frac{l^{z}\backslash}{m}\qquad\frac{l^{z}i\backslash}{n\ /}$$

It is obvious that we used different points in the two positions of the contour. Obviously, we did not need to do this. The methods that find the 3-D position of the contour do not address any "apperture in the large" problem. But the 3-D points are found from their projections and discretization effects may cause a small difference in the number of points of the two positions of the contour. We found that equation (6.29) gave very good results as we will see in the section on experiments.

What remains to be found is the rotation matrix. But since we know the surface normals *ni = (pi, qi, -1) , r%2 =(P2, <i2> mD* of the two positions of the contour, we can immediately find the rotation around an axis parallel to the plane of the contour Ci. Indeed, the angle 9 between ni and n2

$$\left(_{oos9=}\ \frac{n_2}{\|n_1\|\,\|n_2\|}\right)$$

gives the rotation angle around the axis

$$\frac{n_1 \times n_2}{\|n_1 \times n_2\|} = l \ .$$

The angle θ along with the axis l constitute a rotation matrix, R_1. It is obvious that not the final rotation matrix because it misses rotation around an axis perpendicul the world plane. In other words, if we apply to contour C_1 the rotation matrix R_1 an translation T, then the result will not be contour C_2, but a contour C_1' which lies o same plane as C_2, and has the same center of gravity of C_2. To find the missing rota we must find the angle that we have to rotate contour C_1' around an axis n which p through the center of gravity of contour C_2 and is perpendicular to C_2 .

To do that, we start rotating the contour C_1' until it coincides with contour C_2. T done with small increments and the coincidence of the two contours (C_1' and C signaled by the maximization of their common area. The resulting angle φ along wit axis n constitute a new rotation matrix R_2. Obviously, the final rotation matrix is g by $R = R_1 \cdot R_2$.

Finally, it is clear that the method described above will not work (rotation matr will not be found) for some symmetric contours. If, for example, the 3-D contour is a c matrix R_2 cannot be found, since C_1' and C_2 coincide; or if the 3-D contour is a squar the rotation angle $\phi = \pi/2$, then again matrix R_2 cannot be found. This simple fact is obviously true for human observers who observe apparent motion and are aske estimate the 3-D motion parameters.

6.8 Using a monocular observer

Extension of the above results can obviously be trivially generalized for a mono observer who is translating with known motion.

We proceed now with the final section which describes experimental results bas the previous methods for the recovery of structure, depth and 3-D motion of a m planar contour by a binocular or trinocular observer.

6.9 Experiments

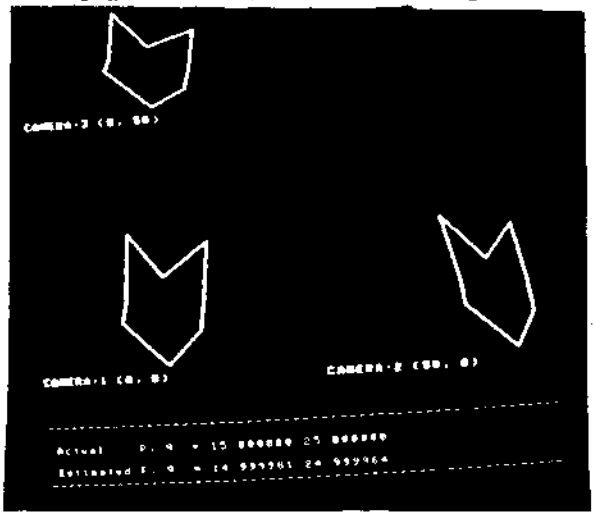
Here we present experimental results from the implementation of the algori developed in this chapter. Figures 6.1-6.5 show results of binocular and trino experiments. We did not add any noise, since we already have the problem of diff number of points in the different images.

Figure 6.1 shows the perspective images of a planar contour taken by three cameras at the positions (0,0), (0,50) and (50,0) respectively. The actual orientation of the contour in space was given by the gradient (p,q) = (15,25). The computed orientation, was (p,q) = (14.99, 24.99). Figure 6.2 shows again the perspective images of a planar contour taken by three cameras at the positions (0,0), (0,50) and (50,0) respectively. The actual orientation of the contour in space was (p,q) = (30,5) and the estimated orientation was (p,q) = (30, 4.99). Figure 6.3 shows the images of a translating planar contour (human figure) taken by a binocular system at two different time instants. The actual orientation of the contour in space was (p,q) = (10,5) and the actual direction of translation (dx/xz, dy/dz) = (-4,6). , our program recovered orientation (p,q)= (10,00007, 5.000297) and direction of translation (dx/dz, dy/dz) = (-4.000309, 6.00463). Figure 6.4 shows again the perspective images of a translating planar contour taken by a binocular system at two different time instances. The actual orientation of the contour was (p,q) = (-25,30) and the direction of translation (dx/dz, dy/dz) = (50,60). The computed orientation from these images was (p,q) = (-24.99, 30.000021) and the computed direction of translation (dx/dz, dy/dz) = (49.858421, 59.830266). Figure 6.5 shows the perspective images of a translating planar contour taken by a binocular system at two different times. The actual orientation of the contour was (p,q) = (10,-11) and the direction of translation (dx/dz, dy/dz) = (1.66, 3.33). The estimated parameters from these images were (p,q) = (9,99, -11.000383) and (dx/dz, dy/dz) = (1.66, 3.33).

The experiments to determine the general motion parameters are shown in 6.6 - 6.10. The actual and computed parameters are recalculated with respect to the coordinate system of the left camera. In figure 6.6 the actual translation was (100,100,100) and actual rotation was 0.2 radians around the axis (0.707, 0.707, 0); the estimated values were translation = (100.4, 99.6, 99.8) and rotation = 0.1997 radians around the axis (0.707, 0.707, 0). The results for the next figure were as follows: actual translation (50, 60, 40) and actual rotation = 0.2 radians around the axis (0.707, 0, 0.707). The estimated translation was (44.25, 54.94, 39.53) and the estimated rotation was 0.1980 radians around the axis (0.704, 0.014, 0.711). Figure 6.8 shows the actual translation as (100, 150, 100) and rotation of 0.9 radians around the axis (0.123, 0.123, 0.985); the estimates were translation = (106.11, 150.7, 99.21) and rotation = 0.902 radians around the axis (0.121, 0.119, 0.985). The ship in figure 6.9 was translated by (100, 150, 80) and rotated by 1.5

(95.30, 145.98, 80.04) and rotation = 1.49 radians around the axis (0.124, 0, 0.91 Figure 6.10 shows the actual parameters as translation = (100, 50,40) and rotation = radians around the axis (0.577,0.577,0.577). The estimated parameters were translat = (102.75,49,59.49) and rotation = 0.199 radians around the axis (0.577,0.573,0.582

***NOTE: All the parameters involved in the above experiments that have a dimensioi length (L* M° TO) are calculated in pixels, where 1 pixel = 100* p m.**



Figures/[1]

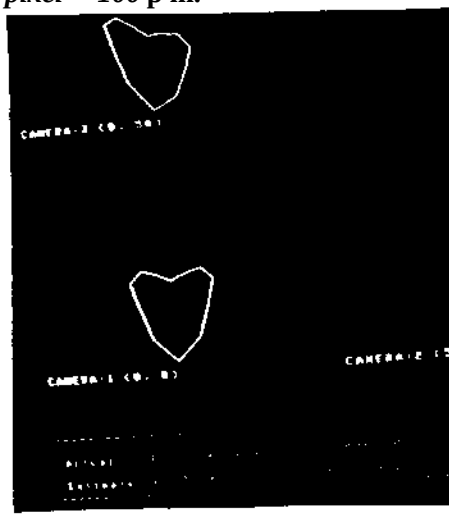


Figure 6.2

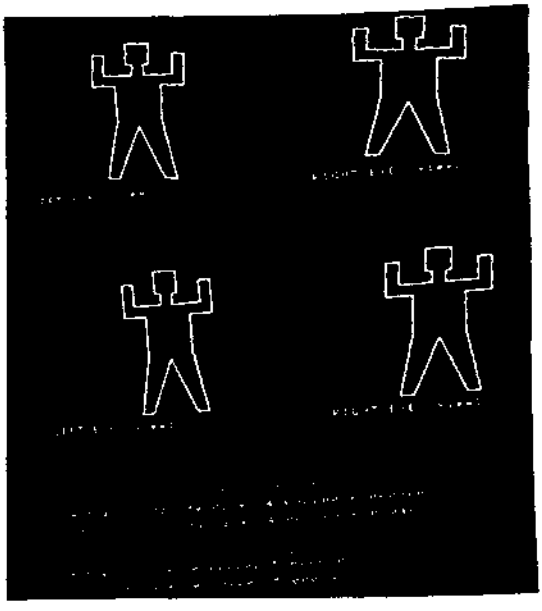


Figure 6.3

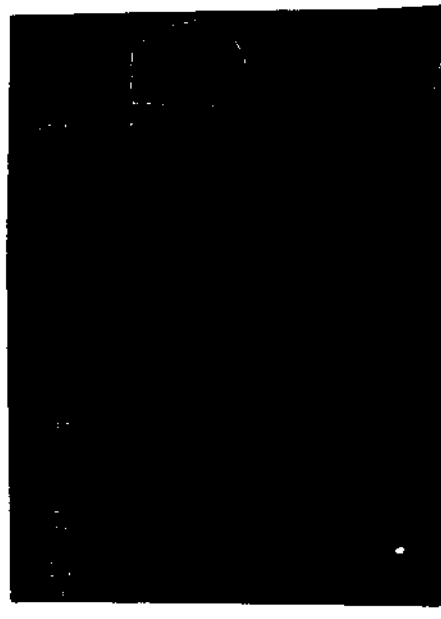


Figure 6.4

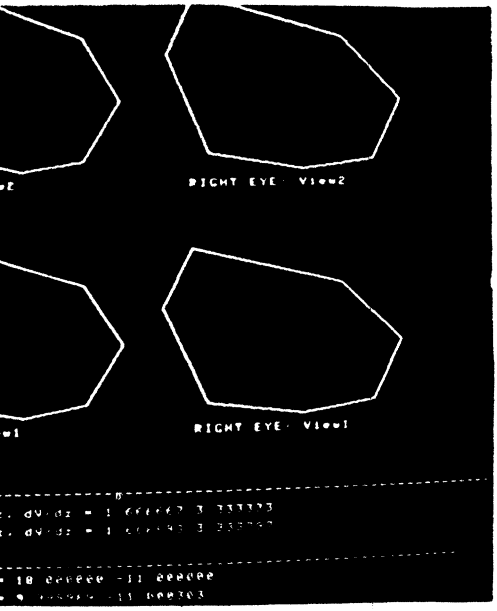


Figure 6.5

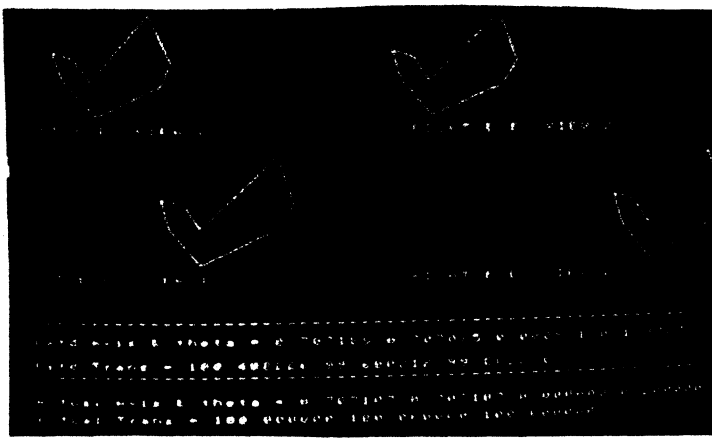


Figure 6.6

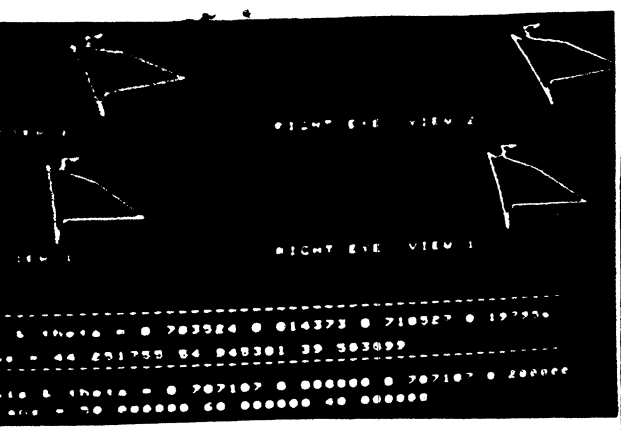


Figure 6.7

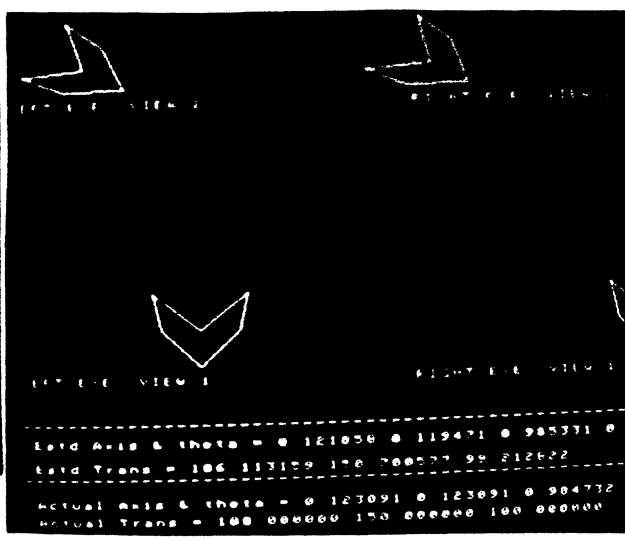
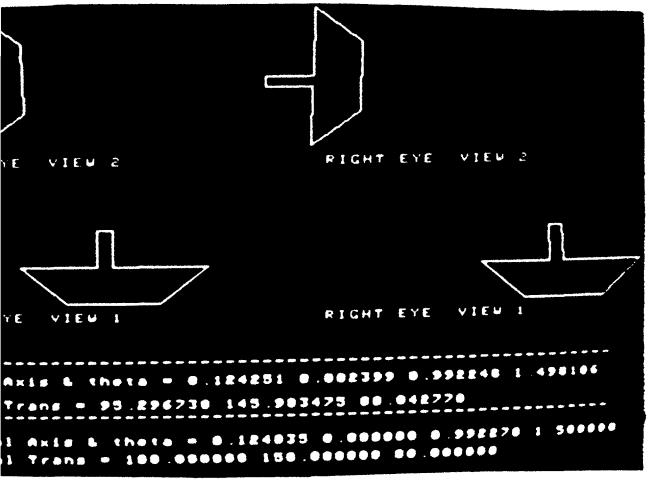


Figure 6.8



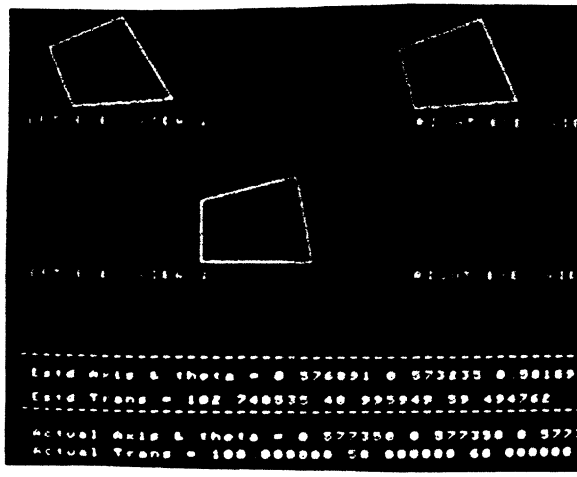


Figure 6.9

Figure 6.10

6.10. Conclusions and future directions

We presented a theory for the recovery of the structure, depth and three-dimensional motion of a moving planar contour by a binocular or trinocular observer. The method seems promising and does not use any static or dynamic point-to-point correspondences. It is one of our future goals to extend this theory to nonplanar contours. In particular we are working towards the characterization of a non-planar contour , as well the detection of its three-dimensional motion without point to point correspondences. Methods that are based on the invariance of the 3-D length of the contour over multiple frames seem very fruitfull, and preliminary results are very promising.

7

Conclusions and future directions

In this thesis we claimed that low-level visual computations should be done in such a way so that uniqueness and robustness of the computations is guaranteed and that visual computations can be done in this way. We justified our claims by examining several problems, such as shape from texture, shape from shading, structure from motion and visual motion analysis, shape and motion from contour and some cases of stereo.

The problem of understanding vision and building intelligent machines with a visual sense is very hard and by no means solved. We have argued that a very large part of today's research is analyzing visual capabilities, i.e. research is concentrating on topics that correspond to identifiable modules in the human visual system. And even though it is not at all clear what are the topics that correspond to identifiable modules in the human visual system, research has shown that shading, texture, motion, contours and stereo are areas that help to understand the extrapersonal space. Existing theories for the analysis of these cues fall basically into the regualization paradigm.

We showed that the regularization school suffers from three basic problems:

(a) the employed assumptions do not capture the real world;

(b) the available constraints are not sufficient to guarantee the uniqueness of visual computations. So restrictive assumptions such as smoothness are usually employed. The resulting algorithms work poorly in synthetic imagery and do not work at all in natural imagery; and

(c) the resulting algorithms, even if uniqueness from the constraints is guarai are non robust, in the sense that a small error in the input is enough to dc completely the results.

There is no doubt that vision is full of redundancy and there is a lot of informati the image which if used correctly will give rise to constraints which will guar; uniqueness and robustness of the visual computations. We have demonstrated this f< case of the problems that appeared in Chapters 3, 4, 5, and 6. Obviously, we need to c robust and unique visual computations if we ever want to advance our understand] vision.

There is a standard way to design large and complex information systems as res in computational fields has shown [Feldman 1985].

(1) First we divide the system into functional components which break the o^ task into autonomous parts, and analyze these components.

(2) Then we must choose the representation of information within the sybsyj and the language of communication among them.

(3) After this, the details of the systems are tested individually, in pairs ar together.

In this thesis, in order to analyze and understand a visual system (machi: biological), we started with the first two steps and a part of the third, and we did th some subsystems (texture, shading, motion, contours, stereo). Our technical results c found at the beginning of Chapers 3, 4, 5, and 6. Our results can be summarized *b* Figure 2.2 of Chapter 2.

There are more subsystems to be analyzed such as color, nonplanar cont recognition of objects, navigation modules, and many others. The analysis of all of constitutes our future research. More importantly, our immediate future research v devoted to the third stepp, where we have to test the subsystem all together

7.1 Future Research

Our future research will investigate more subsystems, in the introduced paradigm (minimal assumptions, uniqueness, robustness). In particular,

(a) We will work for the analysis of nonplanar contours, i.e. a characterization of their shape as well as finding their three-dimensional motion and structure without point-correspondences. We have found the invariance of the length of the contour is very fruitful as a tool for successfully combining the three-dimensional motion of the contour with the lengths of the projections of the contours in the different frames.

(b) There is no doubt that retinal motion is very important, if it can be found. In Section 5.7.2 we showed that there exists a very strong constraint between coresponding points (from rigid motion) and that this constraint has not been utilized for the recovery of retinal motion. We will recover retinal motion from this constraint without having to compute the matrix E first. We believe that a connectionist architecture might prove very useful in solving this problem.

(c) Recent psychological experiments by Todd *et al.* [1986] have indicated that the ability of humans to recover shape from shading is not at all correlated with their ability to recover the illuminant direction. Of course there exist other psychological experiments [Pentland 1983] which support the opposite. There is a little work in this area by Koenderick and Van Doorn [1979] which does not propose any computational mechanisms for the perception of solid shape from shading. We will follow this line of thought to investigate if global methods vs. local methods are possible for the solution of the problem at hand (shape from shading).

(d) We have demonstrated that shading and pattern texture have a strong relationship in terms of constraints. Shading can be viewed as a differential case of "pattern texture," where the patterns become very small. So, shape from shading could probably be obtained with a method similar to the one presented in Section 3.17, if we can transform the shading to a pattern texture. Up to this point, we know how to do this for the case where the light source is in the direction

of the optical axis. We will work towards generalizing this for any ligh direction.

(e) Lines in an image are very important for understanding the three-dimensi structure. In parallel with our approaches without correspondences, we willl v towards extracting three-dimensional motion from lines (ρ, θ representat without correspondences. There is current work in this area by Huang & Mit [Huang *et al.* 1986; Mitiche & Aggarwal 1986], which results in nonlir equations from corresponding lines. We will work toward extracting lir equations for 3-D motion determination, from lines without correspond individual lines.

(f) We will work towards extracting depth information for the case of nonpla surfaces given a set of points in the left and right images. If we know the for the equation of the depth function, then the problem is not complicated. But i do not, then the situation is much different. The problem might be approac from the point of view of three-dimensional motion, since a vergence stereo sys is a camera and the same camera translated and rotated by a fixed amount. S know the matrix E of Section 5.7.2 and so the correspondence may be obtai The stability of the method is up to investigation.

(g) It is our ambition to work towards the recognition of objects. Recognition of obj consisting of line segments is an easier task compared to solid objects. Recogni of objects will follow after the analysis of nonplanar contours. Recognitio objects consisting of line segments can be done by camera rotation. This ena us to compute the structure of the object in view [Kanatani 1986]. In tur search for the space of models encoded in a parallel activation network may gi solution.

(h) Finally, we plan to work towards the coupling of visual computations. To m this clear, let us take a simple example. Suppose that we have three processes p_2, p_3, which compute intrinsic parameters β_1, β_2, β_3 (Figure 7.1) but as we h seen, several intrinsic parameters are connected among them through all kinc functions. So we will have

$$\overline{\beta_1} = f_1(\overline{\beta_2}, \overline{\beta_3}), \quad \overline{\beta_2} = f_2(\overline{\beta_1}, \overline{\beta_3}) \text{ , } etc.$$

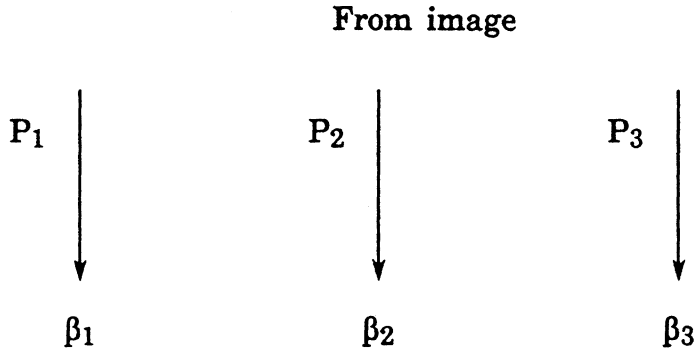


Figure 7.1

This tells us that the computation of an intrinsic parameter will greatly simplify the computation of the others. But, for example, how can β_1 be used in the computation of β_2 when β_1 is not known yet? We must find a way that will enable computations to interact before their completion (Figure 7.2).

In this problem, there are computational as well as empirical issues. By computational issues, we mean problems such as determining the point where process p_1 will interact with p_2, or vice versa, given some constraints, or, if the computations are of an iterative fashion, when do we know when to stop computing and interacting given that we want to compute all parameters β_1, β_2, β_3, for example. At this point, the connectionist architectures [Feldman 1980] show great promise for a solution to this problem, given that we can develop some "goodness" functions as well as some stopping criteria. Empirical issues have to do with what kinds of cues are more important in natural images from others. For example, shading seems to be a weak cue when compared with contour or texture. This (what cues are stronger than others) will enable us to decide what computations should have more weight when computations are interacting.

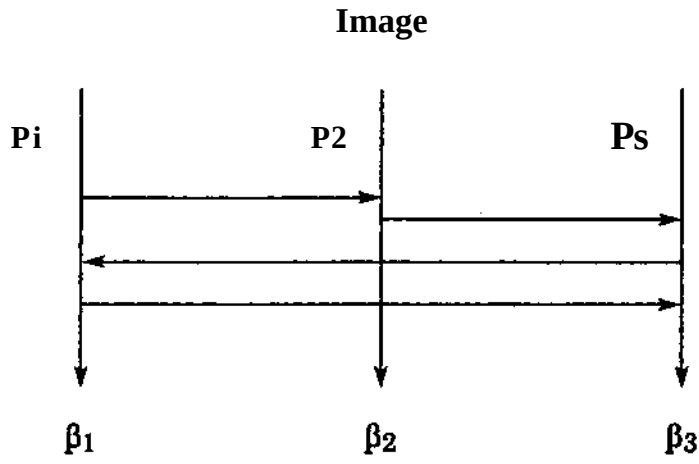


Figure 7.2

We believe that what we have presented in this thesis will advance our understan* of low-level visual computations and we hope that several researchers in the discip will follow our paradigm to enrich it with new ideas which will contribute to understanding of vision.

APPENDIX

Here we prove that equation 6.17 is true under perspective projection. We prove this for a more general case, i.e. the case where the two cameras do not only have horizontal but also vertical displacements.

Theorem:

Let a coordinate system $OXYZ$ be fixed with respect to the left camera, with the Z axis pointing along the optical axis. We consider that the image plane Im_1 is perpendicular to the Z axis at the point (0,0,1) and O the nodal point of the left camera. Let the nodal point of the right camera, be the point $(R,L,0)$ and its image plane identical to the previous one, i.e. $Im_1=Im_2$. Consider a polygon P on the world plane $Z=pX+qY+c$, defined by the points (X_i,Y_i,Z_i), $i=1,..,n$, and having area S_W. Let S_1, S_2 the areas of the paraperspective projections of P on the left and right cameras respectively and S'_1, S'_2 the areas of the perspective projections of the polygon P on the left and right cameras respectively. Then,

$$\frac{S_1}{S_2}=\frac{S'_1}{S'_2}$$

Proof:

The proof is given in several parts.

Let (A_1,B_1) and (A_2,B_2) the centers of mass of the projections of the contour P on the left and right image planes respectively (it has to be noted that (A_1,B_1) and (A_2,B_2) are the centers of mass of the actual left and right images as opposed to the projections of the center of mass of P onto the left and right image planes). Then, we have:

$$\frac{S_2}{S_1}=\frac{1-A_2p-B_2q}{1-A_1p-B_1q} \quad (1)$$

The above equation is the equation (6.17), that we will prove to be exact under perspective projection.

But,

$$A_1=\frac{1}{n}\sum\left(\frac{X_i}{Z_i}\right), \quad B_1=\frac{1}{n}\sum\left(\frac{Y_i}{Z_i}\right), \quad and \quad A_2=\frac{1}{n}\sum\left(\frac{X_i-R}{Z_i}\right), \quad B_2=\frac{1}{n}\sum\left(\frac{Y_i-L}{Z_i}\right)$$

Substituting in (1) we get after some tedious manipulations:

$$\frac{S_2}{c} = 1 + \frac{pR + qL}{c} \qquad (2)$$

From the other hand, we can easily prove that:

$$\frac{S_2}{S_1} = 1 + R \frac{\mathrm{V}\left(\frac{Y_i - Y_{i+1}}{Z_{i+1}}\right)}{M} - I.T \frac{\sum\left(\frac{X_i - X_{i+1}}{ZZ_{i+1}}\right)}{M} \qquad (3)$$

with

$$M = \sum \left(\frac{X_i Y_i \wedge . - X_i \wedge . Y_i}{Z_i Z_{i+1}}\right) \qquad (4)$$

We can also easily prove that:

$$\frac{\sum\left(\frac{Y_i - Y_{i+1}}{Z Z_{i+1}}\right)}{\text{Af}} = \frac{p}{c} \qquad (5)$$

and

$$\frac{\sum\left(\frac{X_i - X_{i+1}}{Z}\right)}{M} \frac{}{i+1} = \frac{}{c} \qquad (6)$$

From equations (2),(3),(4),(5) and (6) the proof of the theorem is immediate.

Bibliography

[Adiv 1984] Adiv, G., "Determining 3-D motion and structure from optical flow generated by several moving objects," COINS-TR 84-07, Univ. Massachusetts at Amherst, 1984.

[Aho, Hopcroft & Ullman 1974] Aho, A.G., J.E. Hopcroft and J.D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley Publishing Co., Reading, MA, 1974.

[Aloimonos 1986b] Aloimonos, J., "Determining the illuminant direction," submitted for publication, 1986.

[Aloimonos 1986c] Aloimonos, J., "Shape and motion from contour, without point to point correspondence: General principles," TR173, Computer Science Dept., Univ. Rochester, 1986; also *Proceedings*, CVPR, 1986.

[Aloimonos 1986d] Aloimonos, J., "Solving the correspondence problem," in preparation, 1986.

[Aloimonos 1984] Aloimonos, J. "One eye suffices: A computational model of monocular depth perception," TR160, Computer Science Dept., Univ. Rochester, December 1984.

[Aloimonos and Basu 1986] Aloimonos, J. and A. Basu, "Determining the translation of a rigidly moving surface, without correspondence," TR176, Computer Science Dept., Univ Rochester; also *Proceedings*, IEEE CVPR, 1986.

[Aloimonos and Brown 1984a] Aloimonos, J. and C.M. Brown, "The relationship between optical flow and surface orientation," Proceedings, 7th ICPR, 1, 542-46, Montreal, August 1984.

[Aloimonos and Brown 1984b] Aloimonos, J. and C.M. Brown, "Direct processing of

curvilinear sensor motion from a sequence of perspective images," *Proceedi* Workshop in Computer Vision: Representation and Control, 72-77, Annapolis, 1984.

[Aloimonos and Chou 1985a] Aloimonos, J. and P. Chou "Detection of surface orienta and motion from texture: The case of planes," TR161, Computer Science D Univ. Rochester, January 1985.

[Aloimonos and Chou 1985b] Aloimonos, J. and P. Chou, "Detection of surface orienta from texture," *Optics News*, September 1985.

[Aloimonos and Rigoutsos, 1986] Aloimonos, J. and I. Rigoutsos, "Determining the th dimensional motion of a rigid planar pathch without correspondence, u persopective projection," TR178, Computer Science Dept., Univ. Rochester; *Proceedings*, Canadian Artificial Intelligence Conf., Montreal, QUE., 1986.

[Aloimonos and Swain 1985] Aloimonos, J. and M. Swain, "Shape from textu *Proceedings*, 9th Int'l. Joint Conf. on Artificial Intelligence, Los Angeles, CA, 931, August 1985.

[Aloimonos, Bandyopadhyay and Chou 1985] Aloimonos, J., A. Bandyopadhyay an Chou, "On the foundations of trinocular machine vision," *Proceedings*, An Meeting. Optical Soc. Amer., Lake Tahoe, NV., March 1985.

[Aloimonos, Basu and Brown 1985] Aloimonos, J., A. Basu and C.M. Brown, "Cont shape and motion," *Proceedings*, DARPA Image Understanding Workshop, Mi FL., December 1985.

[Bandyopadhyay and Aloimonos 1985] Bandyopadhyay, A. and J. Aloimonos, "Percep of rigid motion from spatio-temporal derivatives of optical flow," TR157, Comp Science Dept., Univ. Rochester, March 1985.

[Ballard 1984] Ballard, D.H., "Parameter networks," *Articifial Intelligence*, 22, 235- 1984.

[Ballard 1981] Ballard, D.H., "On shapes," *Proceedings*, 7th IJCAI, Vancouver, Can 1981.

[Ballard and Bandyopadhyay 1982] Ballard, D.H. and A. Bandyopadhyay, "Space-time computations of visual motion," Internal working paper, Computer Science Department, University of Rochester, 1982.

[Ballard and Brown 1982] Ballard, D.H. and C.M. Brown, *Computer Vision*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[Ballard and Kimball 1983] Ballard, D.H. and O.A. Kimball, "Rigid body motion from depth and optical flow," *CVGIP*, 22, 95-115, 1983.

[Bajcsy and Lieberman 1976] Bajcsy, R. and L. Lieberman, "Texture gradients as a depth cue," *Comp. Graphics and Image Processing*, **5**, 52-67, 1976.

[Bandyopadhyay 1985] Bandyopadhyay, A., "Interest points, disparities and correspondence," *Proceedings*, DARPA Image Understanding Workshop, 1985.

[Bandyopadhyay 1984] Bandyopadhyay, A., "A multiple channel model for the perception of optical flow," *Proceedings*, IEEE Workshop on Computer Vision: Representation and Control, Annapolis, MD., 78-82, 1984.

[Bandyopadhyay and Aloimonos 1985] Bandyopadhyay, A. and J. Aloimonos, "Perception of rigid motion from spatio-temporal derivatives of optical flow," submitted to Cognitive Science Conf., Univ. of Calif. - Irvine, 1985.

[Barnard 1985] Barnard, S., "Choosing a basis for perceptual space," *CVGIP*, 29, 87-99, 1985.

[Barnard 1983] Barnard, S., "Interpreting perspective images," *Artificial Intelligence*, **21**, 435-462, 1983.

[Barnard and Thompson 1980] Barnard, S.T. and W.B. Thompson, "Disparity analysis of images," *IEEE Trans. PAMI*, **2**, 333-340, 1980.

[Barrow and Tenenbaum 1981] Barrow, H.G. and J.M. Tenenbaum, "Interpreting line drawings as three-dimensional surfaces," *Artificial Intelligence*, **17**, 75-116, 1981.

[Barrow and Tenenbaum 1978a] Barrow H.G. and J.M. Tenenbaum, "Recove intrinsic scene characteristics from images," in *Computer Visual Systemi* Hansen and E. Riseman (eds.), Academic Press, New York, 1978.

[Barrow and Tenenbaum 1978b]Barrow,H.G. and Tanembaum, J.M., Experimeni ionterpretation guided semantics, in:Hanson and Riseman, Eds., Computer V Systems (Academic Press, New York, 1978).

[Barrow and Popplestone 1971] Barrow, H.G., and Popplestone, R.J., Relati descriptions in picture processing, Machine Intelligence 6,1971.

[Bennett and Hoffman 1985] Bennett, J. and Hoffman, D.D.,"Computation of structur from fixed axis:nonrigid structures", *Biological Cybernetics* ,51,293-300,198

[Bobrow and Winograd] 1977 Bobrow, D.G., and T. Winograd, "An overview of KR knowledge representation language," *Cognitive Science*$_t$ 1,1977.

[Braddick 1980] Braddick, D., "Low level and high level processes in apparent mot *Phil Trans. R. Soc. Lond.*, B 290,137-151, 1980.

[Braddick 1974] Braddick, D., "A short range process in apparent motion," Vi Research, 14, 519-527,1974.

[Brady 1979] Brady,J.,The development of a computer vision system, Reche Psicologica, (1979).

[Brady and Yuille 1983] Brady, M. and A. Yuille, "An Extremun Principle for Shape i Contour," 1983.

[Braunstein and Andersen 1984] Braunstein, M.L. and G.L. Andersen, "A cou example to the rigidity assumption in the visual perception of structure I motion," Perception, 13, 2, 213-217,1984.

[Brice and Fennema 1970] Brice, C.R., and Fennema,C.L., Scene analusis using reg Artificial Intelligence 1, (1970), 205-226.

[Brooks 1979] Brooks, M.J., "Surface normals from closed paths," Proceedings, 6th IJCAI, Tokyo, Japan, 98-101,1979.

[Brooks **1985]** Brooks, M. and Horn, B.K.P,"Shape and source from shading", *Proc. IJCAI* 1985, Los Angeles, CA.

[Brown, Curtiss and Sher 1983] Brown, CM., M.B. Curtiss, and D.B. Sher, "Advanced Hough transform implementations," Proceedings, 7th IJCAI, Vancouver, Canada, 1081,1983.

[Brown et al 1983], Brown, CM., Ballard, D.H and Rainero, E.,"Contsraint propagation in shape from shading", *Proc. Image Understanding Workshop,* 1983.

[Bruss 1980] Bruss, A.R., "The image-irradiance equation, its solution and application," Ph.D. thesis, Dept. of Electrical Engineering and Computer Science, MIT, 1980.

[Bruss 1979] Bruss, A.R., "Some properties of discontinuities in the image irradiance equation," AI Memo 517, Artificial Intelligence Lab., MIT, 1979.

[Bruss and Horn 1983] Bruss, A. and B.K.P. Horn, "Passive nagivation," *CVGIP,* 21, 3-20,1983.

[Chetverikov 1984] Chetverikov, D., "Measuring the degree of texture regularity," *Proceedings,* 7th ICPR, 1984.

[Chomsky 1965] Chomsky, N., *Aspects of the Theory of Syntax,* Cambridge, Mass, MIT Press, 1965.

[Clocksin 1978] Clocksin, W., "Determining the orientation of surfaces from optical flow," *Proceedings,* 3rd AISB Conf., Hamburg, 93-102,1978.

[Clowes 1971] Clowes, M.B., "On seeing things," *Artificial Intelligence,* 2,1,79-116,1971.

[Conners 1980] Conners, H., "A theoretical comparison of the texture algorithms," *IEEE Trans. PAMI,* **PAMI-2,** 204-222, May, 1980.

[Conners and Harlow 1980] Conners, H. and Harlow, "A theoretical compariso texture algorithms," *IEEE Trans. PAMI,* **PAMI-2,**, May 1980.

[Conte and C. deBoor 1972] Conte, S.D. and C. deBoor, *Elementary Numerical Anal* New York, McGraw-Hill, 1972.

[Coxeter] Coxeter, H.S.M., *Introduction to Geometry,* John Wiley & Sons.

[Davis and Rosenfeld 1981] Davis, L.S. and A. Rosenfeld, "Cooperative processes for level vision: A survey," *Artificial Intelligence,* 1981.

[Davis, Wu and Sun 1983] Davis, L.S., Z. Wu, and H. Sun, "Contour based mo estimation," *CVGIP,* 23, 313-326,1983.

[Draper 1980] Draper, S.W., "Reasoning about depth in line-drawing interpretati Ph.D. thesis, University of Sussex, 1980.

[Dunn, Davis and Janos] Dunn, Davis and Janos, "Efficient recovery of shape from texture," *IEEE Trans. PAMI,* **PAMI-5,** 5, 485-492, September 1983.

[Fang and Huang 1984] Fang, J.Q. and T.S. Huang, "Solving three-dimensional small rotation motion equations: Uniqueness, algorithms, and numerical results," *CV(* 26,183-206,1984.

[Fang and Huang 1983] Fang, J.Q. and Huang, T.S.,"Experiments in the determinate of rigid motion", *IJCAI*1983.

[Feldman 1985] Feldman, J.A., "Four frames suffice: A provisional model of vision space," *Behavioral and Brain Sciences,* June 1985.

[Fennema and Thompson 1979] Fennema, C.L. and W.B. Thompson, "Velo determination in scenes containing several moving objects," *CVGIP,* 9, 301- 1979.

[Frisby 1980] Frisby, J.P., *Seeing (Illusion, Brain and Mind),* Oxford University Pi 1980.

[Freuder 1974] Freuder, E.C., "A computer vision system for visual recognition using active knowledge," TR 345, MIT AI Lab, 1974.

[Gibson 1950] Gibson, J.J., *The Perception of the Visual World,* Houghton Mifflin, Boston, 1950.

[Grimson 1984] Grimson, "Binocular shading and visual surface reconstruction," *CVGIP,* 19-44,1984.

[Haralick 1979] Haralick R., "Statistical and structural approaches to texture," *Proceedings of the IEEE,* 67, 5, 786-804, May 1979.

[Haralick and Lee 1983] Haralick R.M. and J.S. Lee, "The facet approach to optical flow," *Proceedings,* DARPA Image Understanding Workshop, Arlington, VA., 84-93, June, 1983.

[Hildreth 1984] Hildreth, E.C., "Computations underlying the measurement of visual motion," *Artificial Intelligence,* 23, 309-354,1984.

[Hoffman 1982] Hoffman, D.D., "Interpreting time-varying images: The planarity assumption," *Proceedings,* IEEE Workshop on Computer Vision, Representation and Control, 92-94, August, 1982.

[Hoffman 1982] Hoffman, D.D., "Inferring local surface orientation from motion fields," J. *Optical Soc. Amer.,* 72, 7, 888-892, July 1982

[Hoffman and Bennett 1985] Hoffman, D.D. and B.M. Bennett, "Inferring the relative three dimensional positions of two moving points," *J. Optical Soc. Amer.* A, 2, 2, 350-353, February 1985.

[Hoffman and Flinchbaugh 1982] Hoffman, D.D. and B.E. Flinchbaugh, "The interpretation of biological motion," BioZ. *Cybernetics,* 42,195-204, 1982.

[Horn 1986] Horn, B.K.P., *Robot Vision,* McGraw-Hill, 1986.

[Horn 1981] Horn, B.K.P., "Sequins & Quills -- representation for surface topography *Representation of 3-Dimensional Objects*, R. Bajcsy (ed.), Springer-Verlag, Be 1981.

[Horn 1979] Horn, B.K.P., "Hill-shading and the reflectance map," *Proceedings*, DA Image Understanding Workshop, Palo Alto, CA, 79-120, 1979.

[Horn 1977] Horn, B.K.P., "Understanding image intensities," *Artificial Intelligenc* 2, 201-231, 1977.

[Horn 1975] Horn, B.K.P., "Obtaining shape from shading information," in *Psychology of Computer Vision*, P.H. Winston (ed.), McGraw-Hill, New York, 155, 1975.

[Horn 1974] Horn, B.K.P., "Determining lightness from an image," *Computer Grap and Image Processing*, 3, 1, 111-299, 1974.

[Horn and Bachman 1979] Horn, B.K.P. and B.L. Bachman, "Using synthetic imag register real images with surface models," *Comm. ACM*, 21, 914-924, 1978.

[Horn and Schunck 1981] Horn, B.K.P. and B.G. Schunck, "Determining optical fl *Artificial Intelligence*, 17, 185-204, 1981.

[Horn and Sjoberg 1979] Horn, B.K.P. and R.W. Sjoberg, "Calculating the reflect map," *Applied Optics*, 18, 1770-1779, 1979.

[Horn, Woodham and Silver 1978] Horn, B.K.P., R.J. Woodham and W.M. Sil "Determining shape and reflectance using multiple images," AI Memo Artificial Intelligence Lab., MIT, 1978.

[Huang and Tsai 1981] Huang, T.S. and R.Y. Tsai, "Image sequence analysis: Mo estimation," in *Image Sequence Analysis*, T.S. Huang (ed.), Springer-Verlag, 19

[Huang and Blonstein], Huang, T.S. and Blonstein,M.,"Robust algorithms for computi three dimensional motion from image sequences", *IEEE CVPR*, 1985.

[Huffman 1971] Huffman,D.A., "Impossible objects as nonsense sentences," inMeltzer, B. and Michie, D., (eds.), *Machine Intelligence*, **6** Edinburgh University Press, Edinburgh 1971.

[Hummel and Zucker 1980] Hummel, R.A. and S.W. Zucker, "On the foundations of relaxation labeling processes," Report No 80-7, Computer Vision and Graphics Laboratory, Dept. of Electrical Engineering, McGill University, Montreal, Quebec, 1980.

[Ikeuchi 1984] Ikeuchi, K., "Shape from regular patterns," *Artificial Intelligence*, **22**, 49-75, 1984.

[Ikeuchi and Horn 1981] Ikeuchi, K. and B.K.P. Horn, "Numerical shape from shading and occluding boundaries," *Artificial Intelligence*, **17**, 141-184, 1981.

[Jerian and Jain 1983] Jerian, N. and Jain, R.,"Determining motion parameters for scenes with translation and rotation", Proc. Workshop on Motion, Toronto, CA 1983.

[Jenkin] Jenkin, J., "The stereopsis of the varying images", Technical Report, RBCV-TR-84-3, University of Toronto, Dept. of Computer Science.

[Johansson 1973] Johansson, G., "Visual perception of biological motion and a model for its analysis," *Perception & Psychophysics*, **14**, 2, 201-211, 1973.

[Julesz 1981] Julesz, B., "Textons, the elements of tecture percepture, and their interactions," *Nature*, **290**, 91-97, 12 March 1981.

[Julesz 1971] Julesz,B., *Foundations of Cyclopean Perception*, The University of Chicago Press, Chicago, 1971.

[Kanade 1985] Kanade, T., "Camera motion from image differentials," *Proceeding*, Conf. of the Optical Society of America, Lake Tahoe, 1985.

[Kanade 1980] Kanade, T., "A theory of Origami world," *Artificial Intelligence*, **13**, 1, 279-311, 1980.

[Kanade] Kanade, T., "Regions segmentation: Signal vs. semantics," *Comput. Grap Image Processing,* 13, 279-297,1980.

[Kanade 1979] Kanade, T., "Recovery of the three dimensional shape of an object fro single view," CMU-CS-79-153, Dept. of Comp. Science, Carnegie-Mellon Ui 1979.

[Kanade and Kender 1980] Kanade, T. and J. Kender, "Skewed symmetry: Map] image regularities into shape," Technical Report CMU-CS-80-133, Dept Computer Science, Carnegie-Mellon Univ. 1980.

[Kanatani 1984] Kanatani, K., "Detection of surface orientation and motion from tex by a stereological technique," *Artificial Intelligence,* 23, 213-237,1984.

[Kender 1983] Kender, J., "Surface constraints from linear extents," *Proceedings, A/* 83,187-190, 1983.

[Kender 1981] Kender, J.R., "Shape from texture," Technical Report CMU-CS-81- Dept. of Comp. Science, Carnegie-Mellon Univ., 1981.

[Kender 1981] Kender, J.R., "Shape from texture: An aggregation transform that ma class of textures into surface orientation," *Proceedings,* IJCAI, 475-480,1981.

[Kender 1980] Kender, J., Ph.D. thesis, Carnegie-Mellon Univ., Pittsburgh, 1980.

[Kender 1979] Kender, J.R. "Shape from texture: A computational paradig *Proceedings,* DARPA Image Understanding Workshop, 79-84, April 1979.

[Koenderink and van Doom 1977] Koenderink, J.J. and A.J. van Doom, "Invari properties of the motion parallax field due to the movement of rigid bodies rela to an observer," *OpticaAct,* 22, 9,773-791,1977.

[Koenderink and van Doom 1976] Koenderink, J.J. and A.J. van Doom, "Vis perception of rigidity of solid shape," J. *Mathm. Biol,* 3, 79, 79-85, 1976.

[Land and McCann 1971] Land, E.H. and L.D. McCann, "Lightness and retinex theo *J. Optical Society of America,* 61, 1-11, 1971.

[Lawton and Rieger 1983] Lawton, D. and J.H. Rieger, "The use of difference fields in processing sensor motion," *Proceedings*, DARPA Image Understanding Workshop, Arlington, VA., June 1983.

[Lee 1980] Lee, D.N., "The optic flow field: The foundation of vision," *Phil. Trans. Royal Soc. London* (Series B), **B 290**, 169-179, 1980.

[Lesser and Erman 1977] Lesser,V.R., and Erman, L.D., "A retrospective view of the Hearsay-II architecture," *Proceedings*, IJCAI, 2, 790-800, 1977.

[Longuet-Higgins and Prazdny 1984] Longuet-Higgins, H.C. and K. Prazdny, "The interpretation of a moving retinal image," *Proc. Royal Soc. London*, **B 208**, 385-397, 1984.

[Longuet-Higgins 1981] Longuet-Higgins, H.D., "A computer algorithm for reconstructing a scene from two projections," *Nature*, **293**, 133-135, September 1981.

[Mackworth 1973] Mackworth, A.,K., "Interpreting pictures of polyhedral scenes," *Artificial Intelligence*, **4**, 121-137, 1973.

[Mackworth 1976] Mackworth, A.K., "Model-driven interpretation in intelligent vision systems," *Perception*, **5**, 349-370, 1976.

[Mackworth 1973] Mackworth, A.K., "Interpreting pictures of polyhedral scenes," *Artificial Intelligence*, **4**, 2, 121-137, 1973.

[Marschall and Newcombe 1973] Marchall, J.C., and F. Newcombe, "Patterns of paralexia," *J. Psycholinguistic Research*, **2**, 175-199. 1973.

[Marr 1981] Marr, D., *Vision*, W.H. Freeman, San Francisco, 1981.

[Marr 1978] Marr, D., "Representing visual information," in *Computer Vision Systems*, Hanson and Riseman, (eds.), Academic Press, New York, 1978.

[Marr 1977] Marr, D., "Analysis of occluding contour," *Proceedings, Royal Soc. London*, **B 197**, 441-475, 1977.

[Marr 1976] Marr, D., "Early processing of visual information," *Phil. Trans of Royal London* (Series B), B 275,483-524,1976.

[Marr and Hildreth 1980] Marr, D. and E. Hildreth, "Theory of edge detection," *F Royal Soc. London,* B **207,**187-217,1980.

[Marr and Poggio 1979] Marr,D. and T. Poggio, "A theory of human stereo vision," *F Royal Society of London,* B **207,**301-328,1979.

[Marr and Poggio 1977] Marr, D., and Poggio, T., "From understanding computatio understanding neural circuitry," in *Neural Mechanisms in Visual Percept* Neuroscience Research Program Bulletin, Poppel, E., *et al*$_f$ (eds.), 15, 470-' 1977.

[Milencovich, et al] Milencovich V. and Kanade, T.," Trinocular stereo", *Proc. Image Understanding Workshop,* Miami FL, 1985.

[Minsky and Papert 1972] Minsky, M. and S. Papert, "Artificial Intelligence Prog Report," AI Memo 252, MIT, 1972.

[Nagel 1983] Nagel, H.H., "Displacement vectors derived from second order inter variations in image sequences," *CVGIP,* 21, 85-117,1983.

[Nagel 1983] Nagel, H.H., "Overview on image sequence analysis," in *Image Seqw Processing and Dynamic Scene Analysis,* T.S. Huang (ed.), 1983.

[Nagel and Neumann 1981] Nagel, H.H. and Neumann, B., "On 3-D reconstruction fro two perspective views", Proc. 7-th IJCAI, Vancouver, CA, 1981.

[Neghadaripur, S. and Horn, B.K.P.,"Direct passive navigation", *Proc. Image Understanding Workshop,* Miami FL, 1985.

[Ohta, Maenobu and Sakai 1981] Ohta, Y., K. Maenobu and T. Sakai, "Obtaining sur orientation from texels under perspective projection," *Proceedings,* 7th IJC Vancouver, Canada, 746-751,1981.

[Pentland 1984] Pentland, A.P., Shading into texture, *IEEE Trans. PAMI,* 1984.

[Pentland 1984] Pentland, A.P., "Local shading analaysis," *IEEE Trans. PAMI*, **PAMI-6**, 170-187, March 1984.

[Pentland 1982] Pentland, A.P., "Finding the illuminant direction," *Optical Society of America*, **72**, 4, 448-455, April 1982.

[Prager and Arbib 1983] Prager, J. and M. Arbib, "Computation of the optic flow: The MATCH algorithm and prediction," *CVGIP*, December 1983.

[Prazdny 1983] Prazdny, K., "Stereoscopic matching, eye position, and absolute depth," *Perception*, **12**, 151-160, 1983.

[Prazdny 1984] Prazdny, K., "On the information in optical flow," *CVGIP*, **22**, 239-259, 1983.

[Prazdny 1981] Prazdny, K., "Determining the instantaneous direction of motion from optical flow generated by a curvilinearly moving observer," *CVGIP*, **17**, 238-248, 1981.

[Prazdny 1980] Prazdny, K., "Egomotion and relative depth map from optical flow," *Biol. Cybernetics*, **26**, 87-102, 1980.

[Reddy 1978] Reddy, R., "Pragmatic aspects of of machine vision, "in *Computer Vision Systems*, Hanson and Riseman, Eds., Academic Press, New York, 1978).

[Reiger and Lawton 1983] Reiger, J.H. and D.T. Lawton, "Determining the instantaneous axis of translation from optic flow generated by arbitrary sensor motion," Technical Report 83-1, Computer and Information Science Department, Univ. Mass. at Amherst, January 1983.

[Regan and Beverly] Regan D. and Beverly, K.I.,"Binocular and monocular stimuli for motion in depth:Changing disparity and changing size feed the same motion in depth", *Vision Research*, 19:1331-1342.

[Roach and Aggarwal 1980] Roach D. and Aggarwal, J.K.,"Determining the movement of

objects from a sequence of images", *PAMI* 2, 1980.

[Rosenfeld, Hummel and Zucker 1976] Rosenfeld, A. , R. Hummel, and S.W. Zu "Scene labeling by relaxation operations," *IEEE Trans. Systems, Man Cybernetics*, **6**, 420-433, 1976.

[Richards] Richards, W.," Structure from stereo and motion", AI Memo 731, Cambrid MA, MIT AI Lab. Also, in *J. Opt. Soc. of America*, A2:343-349 (86).

[Shafer 1982], Shafer, S.,"Shadow geometry", Ph.D. thesis, Carnegie -Mellon Univers Dept. of Computer Science.

[Shirai 1973] Shirai, Y., "A context-sensitive line finder for recognition of polyhe *Artificial Intelligence*, 4, 95-119, 1973.

[Stevens 1980] Stevens, K.A., "Constraints on the visual interpretation of su contours," AI Memo 522, MIT, Cambridge, MA, 1979.

[Stevens 1980] Stevens, K.A., "Surface perception from local analysis of texture contour," Ph.D. thesis, Technical Report TR 512, MIT Cambridge, MA, 1979.

[Stevens 1980] Stevens, K.A., "Occlusion clues and subjective contours," AI Memo IT, 1976.

[Stewart 1973] Stewart, G.W., *Introduction to Matrix Computations*, Academic P 1973.

[Strat 1979] Strat, T.M., "A numerical method for shape from shading from a si image," M.S. thesis, Dept. of Electrical Engineering and Computer Science, 1979.

[Sugihara 1978] Sugihara, K., "Quantitative analysis of line drawings of polyhe scenes," *Proceedings*, 4th Intl. Joint Conf. on Pattern Recognition, Kyoto, 771 1978.

[Sugihara and Sugie] Sugihara, K. and Sugie, N., On orthographically projected optic flow", *CVGIP*, November 1984.

[Tichonov and Arsenin] Tichonov and Arsenin,"Solution of Ill-Posed problems", Winston and Wiley, Washington DC, 1977.

[Todd et al] Todd et al, " Perception of solid shape from shading", *Biological Cybernetics*, 1986.

[Terzopoulos] Terzopoulos, D.," Multivelel processes for visible surface reconstruction", *CVGIP*, 24, 1983.

[Tsai and Huang 1984] Tsai, R.Y. and T.S. Huang, "Uniqueness and estimation of three dimensional motion parametrters of rigid objects with curved surfaces," *IEEE Trans. PAMI*, **PAMI-6**, 13-27, January 1984.

[Ullman 1984a] Ullman, S., "Rigidity and misperceived motion," *Perception*, **13**, 2, 219-220, 1984.

[Ullman 1984b] Ullman, S., "Maximizing rigidity: The incremental recovery of 3D structure from rigid and nonrigid motion," *Perception*, **13**, 255-274, 1984.

[Ullman 1979a] Ullman, S., "The interpretation of structure from motion," *Proc. Royal Soc. London*, (Series B), **B 203**, 405-426, 1979.

[Ullman 1979b] Ullman, S., *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA., 1979.

[Ullman and HIldreth 1982] Ullman, S. and E. Hildreth, "The measurement of visual motion," in *Physical and Biological Processing of Images*, (Proc. Int. Symp. Rank Prize Funds, London), A.C. Sleigh (ed.), 154-176, Springer-Verlag, September 1982.

[Van Diggelen 1951] Van Diggelen, J., "A photometric investigation of the slopes and heights of the ranges and hills in the maria of the moon," *Bull. Astron. Inst. Netherlands*, **121**, 283-289, 1951.

[Van Essen, Maunsell and Bixby 1981] VanEssen, D.C., J.H.R. Maunsell and J.L. Bixby, "The middle temporal visual area in Macaque: Myeloarchitecture, connections, functional properties and topographical organization," *Journal of Comparative Neurology*, **199**, 293-326, 1981.

[Wallach and O'Connell 1953] Wallach, H. and D.N. O'Connell, "Kinetic depth effect," *J. Exp. Psychol.*, **45**, 4, 204-217, 1953.

[Waltz 1975a] Waltz, D., "Understanding line drawings of scenes with shadows," in *The Psychology of Computer Vision*, P.H. Winston, (ed.), McGraw-Hill, New York, 19-91, 1975.

[Waltz 1975b] Waltz, D., "Generating semantic descriptions from drawings of scenes with shadows," in *The Psycology of Computer Vision*, P.H. Winston, (ed.), McGraw-Hill, New York, 1975.

[Walker and Kanade], Walker and Kanade, T., Shape from patterns", Private communication.

[Watson and Ahumada 1985] Watson, A.B. and A.J. Ahumada, "Model of human visual-motion sensing," *J. Optical Soc. Amer. A*, **2**, 2, 322-341, Februarry 1985.

[Waxman 1984] Waxman, A., "Kinematics of image flows," *Proceedings*, DARPA Image Understanding Workshop, Arlington, VA., June 1983.

[Waxman and Ullman 1985] Waxman, A. and Ullman, S.,"Surface structure and 3-D motion parameters from image flow kinematics" , *Int. J. of Robotics Research*, 4, (3), 79-94, 1985.

[Waxman , Kamgar-Parsi and Subbarao] Waxman,A., Kamgar-Parsi and Subbarao, M.,"Closed form solutions to image flow equations, for structure and motion", TR-CAR-190, University of Maryland, Center for Automation Research, 1986.

[Waxman and Duncan] Waxman A. and Duncan, J.,"Binocular image flows", *Proc. Workshop on Motion*, Charleston SC, 1986.

[Waxman and Sinha] Waxman, A. and Sinha, S.,"Dynamic stereo: Passive ranging to moving objects from relative image flows", *Proc. Image Understanding Workshop*, New Orleans, SAIC, pp. 130-136.

[Waxman and Wohn] Waxman, A. and Wohn, K.,"Contour evolution, neighborhood deformation and dglobal image flow: planar surfaces in motion", *Intl. Journal of Robotics Research*, 4, (3), 95-108.

[Webb 1981] Webb, J.A., "Shape and structure from motion of objects," Ph.D. thesis, Univ. Texas at Austin, 1981.

[Webb and Aggarwal 1983] Webb, J.A. and J.K. Aggarwal, "Shape and correspondence," *CVGIP*, **21**, 145-160, 1983.

[Webb and Aggarwal 1982] Webb, J.A. and J.K. Aggarwal, "Structure from motion of rigid and jointed objects," *Artificial Intelligence*, **19**, 107-130, 1982.

[Webb and Aggarwal 1981] Webb, J.A. and J.K. Aggarwal, "Structure from motion of rigid and jointed objects," *Proceedings*, 7th IJCAI, 686-691, 1981.

[Weiskrantz, Warrington, Sanders and Marschall 1974] Weiskrantz, L., Warrington, E.K., Sanders, M.D., and Marschall, J., "Visual capacity in the hemianopic field following a restricted occipital ablation," *Brain* **97**, (709-728, 1974.

[Winston 1972] Winston, P.H., "The MIT Robot," in *Machine Intelligence*, **7**, B. Meltzer, and D. Michie, (eds.), Edinburgh University Press, Edinburgh 1972.

[Witkin 1981] Witkin, A., "Recovering surface shape and orientation form texture," *Artificial Intelligence*, **17**, 17-45, 1981.

[Witkin 1980] Witkin, A., "Shape from contour," Ph.D. thesis, Dept. of Psychology, MIT, 1980.

[Woodham 1980a] Woodham, R.J., "Using digital terrain data to model image formation in remote sensing," *Proceedings*, SPIE, **238**, 361-369, 1980.

[Woodham 1980b] Woodham, R.J., "Photometric method for determining surface orientation from multiple images," *Optical Engineering,* 19,139-144,1980.

[Woodham 1979] Woodham, R.M., "Relating properties of surface curvature to image intensity," *Proceedings,* 6th IJCAI, Tokyo, Japan, 971-977,1979.

[Woodham 1978] Woodham, R.J., "Photometric stereo," AI Memo 479, MIT, June 1978.

[Woodham 1977] Woodham, R.J., "A cooperative algorithm for determining surface orientation from a single view," *Proceedings IJCAI-77,* Cambridge, MA, 635-641, 1977.

[Yen and Huang] Yen Y.A. and Huang, T.S.," Motion determination from line correspondences using spherical projection", in T.S. Huang (eds.) Image Sequence Processing and Dynamic scene analysis, Proc. of NATO , Adv. Study Inst, Braunlage, West Germany, Springer Verlag, 1983.

[Yu 1983] Yu, S.H., "Implementation of shape-from-shading algorithms," *USC-ISG-104,* 85-99, October 1983.