Acknowledgements

A number of people have contributed in various ways to helping me develop and formulate the ideas presented in my dissertation.

My deepest thanks go to Prof. Yiannis Aloimonos who introduced me to the field of active, qualitative vision and to motion analysis, who taught me the secrets of pursuing scientific research, and who encouraged me throughout the years. Through his excellent teaching capabilities and the enthusiasm for the research he conducts, he made the years of my dissertation a very exciting time. I would like to thank Prof. Walter Kropatsch for introducing me to the field of Computer Vision and for his advice, encouragement, and constructive criticism, and Prof. Georg Gottlob for being a member of my dissertation committee. Prof. Azriel Rosenfeld deserves much credit for providing a stimulating environment at the Center for Automation Research and for greatly improving the presentation of my work. I also would like to thank Prof. Larry Davis for discussions that led to the improvement of the thesis and Prof. Rama Chellappa for organizing a seminar on visual motion analysis. I wish to express my thanks to my colleagues at the Computer Vision Laborartory, especially Gregory Baratoff, Peter Cucka, David Doermann, Daniel DeMenthon, Jean-Yves Hervé, Liuqing Huang, Radu Jasinschi, Ehud Rivlin, Rajeev Sharma, David Shulman, Bradley Stewart, and Cheolwoo Yoo, who contributed to a pleasant working atmosphere on a daily basis; to my colleagues at the Technical University of Vienna, Horst Bischof, Axel Pinz, and Dieter Willersinn; and to Wolfgang Pöltzleitner at Joanneum Research, Graz, who encouraged me to pursue Computer Vision research in the United States. Profound discussions with the participants of the Ruzenagaard Workshop on Active Vision and especially Prof. Ruzena Bajcsy, Henrik Christianssen, Prof. Jim Crowley, Prof. Randal Nelson and Prof. Giulio Sandini were most useful in the development of my ideas. The help of Kourosh Pahlavan and Prof. Jan-Olof Eklundh in gathering image data with the KTH-head is highly appreciated.

Especially I would like to thank my family, Willibald and Dietlinde, Barbara, Elke, Wolfgang and Magdalena for their love and support throughout the years.

This work would not have been possible without the generous support of the Österreichisches Bundesministerium für Wissenschaft und Forschung, the Österreichische Bundekammer der Gewerblichen Wirtschaft and the Directorate of Robotics and Machine Intelligence of the National Science Foundation.

Contents

List of Figures

Chapter 1

Introduction

The long sought goal of humans to understand the process of vision and to create anthropomorphic machines possessing visual capabilities has challenged philosophers and scientists over the centuries [Barnes, 1983; Descartes, 1978; Wittgenstein, 1953]. The large number of scientific investigations conducted in various fields indicate how complex the problem of visual perception is. Scientists concerned with disciplines such as Optics, Neurobiology, Ethology, Psychology, Mathematics and Engineering have addressed different aspects of this problem. While physicists and photogrammetrists [Horn, 1986; Wolf, 1983] were mainly concerned with the photometric and geometric issues of image formation, psychologists, biologists and neuroscientists [Edelman, 1989; Gregory, 1970; Kanizsa, 1979] studied how biological organisms perform visual perception, and engineers, mathematicians and computer scientists [Ballard and Brown, 1982; Aloimonos and Shulman, 1989; Marr, 1982] investigated the development and computation of representations of the visual environment such as shape, form and color. Since all biological and robot systems exist in space/ time and move in their environments, in order to capture all dimensions of objective reality, they must possess capabilities enabling them to interpret visual motion. The computation and interpretation of motion on a projection surface (retina or film) to derive a description of the 3D motion of an observer relative to the environment is the topic this thesis is concerned with.

It is commonly recognized that Gibson [1950] was the first one to set forth the idea of deriving 3D motion and shape of the scene from the measurements of feature displacement on the retina, which he termed "optical flow". Since Gibson's descriptive, but non-mathematical work many technical studies have been devoted to mathematical modeling of the process of 3D motion perception. Much more than any other problem in computer vision, visual motion analysis has been the subject of scientific investigations. This can be seen from the number of research papers, books, and conference sessions, which

have appeared increasingly since the early 1980's. One of the main reasons for this concentrated effort lies in societal and governmental demands for automation, especially for the construction of autonomous vehicles and smart weapon systems.

One can roughly divide the existing studies in two categories: application-oriented research devoted to the development of special-purpose machines possessing capabilities for interpreting motion in highly restricted environments, and theoretical work, mainly following the approach of recovering 3D models of the scene in view. The work described in this dissertation is of a theoretical nature, but it does not follow the approach of general recovery [Marr, 1982]. It is motivated by a new methodological framework for studying computer vision and building visual systems. Based on the premise that visual capabilities only make sense for systems (biological or artificial) that interact with their environment and engage in some kind of motion, this framework suggests a new way of approaching the study of visual systems. Before proceeding to explain the ideas underlying this new approach that I call the synthetic approach and the contribution of this work, it is necessary to critically examine the philosophical paradigm under which most existing computer vision research has taken place.

1.1 Classical computer vision

The desire to automate the processing of visual information can be traced back to the advent of computers. The beginning was made by computational studies which were concerned with processing imagery appearing in remote sensing and medicine and some industrial inspection tasks which were solved by visual means. As the new scientific field of computer vision evolved over the years, its pioneers defined its scope and established a field of independent study. According to books published in the field, computer vision is defined as the construction of explicit, meaningful descriptions of the structure and the properties of the three dimensional world from two dimensional images. It soon became clear that processing two dimensional visual information, which appears so easy to humans, is extremely difficult to perform computationally. As a consequence most effort in computer vision has been devoted to computational issues.

Computer vision is generally regarded as one of the main subfields of Artificial Intelligence (AI). As such it has been addressed using the general AI methodology. AI, which has as its goal the study of intelligent behavior through the use of computational models, was approached with very high aims. The desire was to find general concepts for simulating intelligence. Thus the main emphasis in AI research has been on the finding of general purpose methodologies and general purpose representations that preserve

as much information as possible. Such efforts led to the rapid development of various programs such as the General Problem Solver (GPS)[Ernst and Newell, 1969] which was intended to solve a large variety of problems; MACSYMA [Moses, 1976], a program to solve numerous types of mathematical problems; and a geometry program [Gelernter, 1959], which its creator claimed to perform better than himself.

A machine that possesses artificial intelligence must be able to interact with its environment. Through its sensors it has to gather information from its surrounding 3D world, which might be in the form of natural language, images, taction, or other modalities. This information is then processed to arrive at different forms of representation which are appropriate for again interacting with the environment in form of language or actions performed by a robot. Their work on such complex tasks has led artificial intelligence researchers to take the approach of breaking any problem into three autonomous parts and studying them independently. The conversion of external data (sensor data, actuator commands, decision making, etc.) into an internal representation and vice versa has been separated from the phase of developing algorithms to perform computations on internal data. Most research has been devoted to processing the internal data, and as a consequence different subfields such as planning, machine learning, knowledge-based data bases and many more have appeared. Therefore it is not surprising that the first influential theory of computational vision (Marr [1982]) mainly concentrated on the computational and representational aspects. In this theory, vision is described as a reconstruction process, that is a problem of creating representations of increasingly high levels of abstraction leading from 2D images through the primal sketch through the $2\frac{1}{2}$D sketch to object centered descriptions ("from pixels to predicates") [Pentland, 1986].

Other researchers prefer to divide the above mentioned processes of visual reconstruction, according to the level of abstraction of the employed descriptions, into low, middle, and high level modules. The low level modules operate directly on the image and deliver image descriptions in the form of features of the image, such as places of high frequency values, edges, and lines (Marr's primal sketch). These descriptions are utilized by the middle level modules in order to perform 3D recovery (Marr's $2\frac{1}{2}$ D sketch). Examples of such modules include the computation of shape from texture, contour, shading, etc. Finally, the high level modules use the results of recovery to reason about the world (e.g. recognize objects).

1.2 The state of the art

Although important results have been achieved at the level of computational theories and algorithms dealing with internal representations, it has become clear that the progress in developing computational theories has been hampered by fundamental difficulties. As a result more and more complicated physical models have been employed using sophisticated mathematics. Nevertheless, it has to be admitted that vision, and AI in general, are far from their ultimate goal of building machines possessing capabilities of living organisms. Critics of the general AI philosophy argue that the study of intelligence cannot be separated from the study of the system's interaction with its environment. Applied to machine vision, these ideas have led many researchers to advocate that computer vision should no longer be considered as a scientific field of independent study. The scope of computer vision should not be limited to the study of mappings of a given set of visual data into representations on a more abstract level. It is argued that image understanding has to be extended to include the process of selective acquisition of data in space and time. A good theory for vision should create the interface between perception and other cognitive abilities, such as reasoning, planning, learning, and manipulation. This has produced a paradigm that has established itself under the term active vision [Aloimonos *et al.*, 1988; Bajcsy, 1988], sometimes also referred to as purposive [Aloimonos, 1990], animate [Ballard, 1991], or behavioral vision.

The first advocates of this paradigm have elaborated the advantages of active vision from different viewpoints by means of specific applications. In a theoretical paper Aloimonos *et al.* [1988] showed for the first time the mathematical advantages an active observer has over a passive one. They proved that for an observer which has the capability to change visual parameters in a controlled way additional information can be supplied, and certain classical problems that were originally considered ill-posed, ill-conditioned and non-linear can be solved. In particular, they elaborate the technical advantages in the computation of shape and structure from cues such as texture, shading, contour and motion. This, however, was only the first contribution. It still was embedded within the old policy of reconstructive vision that argues for transforming data by keeping as much information as possible. Since most information, however, is never needed, such an approach entails much wasted effort. The paradigm of active vision has matured, and by now it has become clear that vision should be studied in connection with the behavior of the organism. Thus, the approach of complete scene recovery has to be dropped. Finding a general-purpose representation is not necessary for performing many visual tasks. The classical problems that have been addressed have to be reconsidered as well. The question, of course, still remains: What are these easier problems to be solved?

In order to illuminate the answers to this question we should consider the basic motivations that led to the study of computer vision. The main reason for performing vision research is the fact that we, humans, and many animals possess visual capabilities. There is no doubt that most ideas in machine vision are inspired by the abilities of biological organisms.

For most of their history artificial intelligence and cognitive modeling have focused almost exclusively on human abilities and capacities. As a result there has been much cross-fertilization between AI and psychology; therefore the study of the human visual system has received a lot of attention. Psychological studies mainly concentrated on the understanding of singularities in human perception, or visual illusions, as they are commonly called. This effort, probably, arises from the intriguing fact that since vision is so effortless for humans, the complexity of its underlying mechanisms has been largely underestimated. Such fallacies led to the assumption that the brain is designed in a modular, principled fashion, and thus from the study of its malfunctions (illusions [Gregory, 1963]) information about its design can be deduced. However, recent results on visual agnosia (a human condition exhibited by patients with partially damaged brains) [Farah, 1990] indicate that the human brain is not designed in a clean modular fashion, but consists of several processes working in a cooperative distributed manner. The findings from studies of illusions actually support this point, since a multitude of computational theories of different natures have been proposed for explaining the plethora of human visual illusions. The realization of the tremendous complexity of human visual perception has thus been the major insight gained from psychological studies.

Much simpler than the human visual system are the perceptual systems of lower animals, like medusae, worms, crustaceans, insects, spiders and molluscs. Researchers in neuroethology have been studying such systems and have by now gained a great deal of understanding. Horridge [1987; 1991], working on insect vision, studied the evolution of visual mechanisms and proposed hierarchical classifications of visual capabilities. He argued that the most basic capabilities found in animals are based on motion. Animals up to the complexity of insects perceive objects entirely by relative motion. His viewpoint concerning the evolution of vision is that objects are first separated by their motions, and with the evolution of a memory for shapes, form vision progressively evolves. The importance of these studies on lower animals becomes very clear when we take into account the commonly held view by leaders in this field, that the principles governing visual motor control are basically the same in lower animals and humans. Horridge, in recent work, argues for copying principles of animal visual mechanisms into artificial systems. For example, systems that open doors and detect burglars, according to Horridge, should

have the same level of complexity as the shell-closure reflex of barnacles. Machine vision, of course, does not have to copy animal vision, but the existence of biological organisms gives us at least some reason to believe that it is possible for a system to work in the same or a similar way.

1.3 A synthetic (evolutionary) approach

The main reason why the approach to vision suggested by Marr has not led to the development of successful artificial systems is that vision was studied in a vacuum, i.e. its utilization was completely ignored. However, in no system is vision purposeless; thus it needs to be studied in conjunction with the task the system is involved in. From this viewpoint, understanding vision means understanding a system possessing visual capabilities.

In general, if our goal is to study (or more precisely formulated, analyze in order to design) a system, we are advised by engineering considerations to follow some common principles in system design and address a set of basic questions: What is the functionality of the system? What are the autonomous subsystems (modules) the system is divided into? What are the relationships of the modules to each other? What are the representations of information within the subsystems, and how do the modules communicate with each other? Finally, we have to ask: What is the most efficient and effective way to design the individual modules?

Having these questions in mind, I propose a new approach for studying vision and developing artificial vision systems. This approach takes it for granted that the observer (the system) possesses an active visual apparatus. Since, furthermore, it is inspired by evolutionary, neuroethological considerations, I call it the synthetic (evolutionary) approach [Fermüller, 1993c]. This approach constitutes a philosophy of how to systematically study visual systems which live in environments as complex and multifarious as those of human beings. It is basically substantiated by two principles. The first principle is related to the overall structure of the system and how it is modularized, what are the problems to be solved and in which order they ought to be addressed. The second principle is concerned with the way the individual modules should be realized.

As a basis for its computations a system has to utilize mathematical models, which serve as abstractions of the representations employed. The first principle of the synthetic approach states that the study of visual systems should be performed in a hierarchical order according to the complexity of the mathematical models involved. Naturally, the

computations and models are related to the class of tasks the system is supposed to perform. A system possesses a set of capabilities which allow it to solve certain tasks. The synthetic approach calls for first studying capabilities whose development relies on only simple models and then going on to study capabilities requiring more complex models. Simple models do not refer to environment—or situation-specific models which are of use in only limited numbers of situations. On the contrary, each of the capabilities requiring a specified set of models can be used for solving a well-defined class of tasks in every environment and situation the system is exposed to. In other words, the assumptions used have to be general with regard to the environment. The motivation for this approach is to increasingly gain insight into the process of vision, which is of such high complexity. Therefore the capabilities which require more complex models should be based on "simpler", already developed capabilities. The complexity of a capability is thus given by the complexity of its assumptions; what has been considered a simple capability might require complex models, and vice versa. For example, as shown in this thesis, the celebrated capability of egomotion estimation does not require, as has been believed, complex models about the geometry of the scene in view or the time evolution of the motion, but only a simple rigid motion model.

The second principle, motivated by the need for robustness, is the quest for algorithms which are qualitative in nature. The synthetic approach does not have as its goal the reconstruction of the scene in view, but the development of a class of capabilities that recognize aspects of objective reality which are necessary to perform a set of tasks. The function of every module in the system constitutes an act of recognizing specific situations by means of primitives which are applicable in general environments. For example, a system, in order to avoid obstacles, does not have to reconstruct the depth of the scene in view. It merely has to recognize that the distance to a close-by object is decreasing at a rate beyond some threshold given by the system's reaction time. Recognition, of course, is much easier than general reconstruction of the scene, simply because the information necessary to perform a specific task can be represented in a space having only a few degrees of freedom [Fermüller and Aloimonos, 1993a]. Moreover, in order to speak of an algorithm as qualitative, the primitives to be computed do not have to rely on explicit unstable, quantitative models. Qualitativeness can be achieved for a number of reasons: The primitives might be expressible in qualitative terms, or their computation might be derived from inexact measurements and pattern recognition techniques, or the computational model itself might be proved stable and robust in all possible cases.

To elucidate the synthetic approach, the next section is devoted to a discussion of how navigation has to be studied under this new philosophy.

1.4 Navigation

Navigation, in general, refers to the performance of sensory mediated movement, and visual navigation is defined as the process of motion control based on an analysis of images. A system with navigational capabilities interacts adaptively with its environment. The movement of the system is governed by sensory feedback which allows it to adapt to variations in the environment and does not have to be limited to a small set of predefined motions as is the case, for instance, with cam-activated machinery. By this definition visual navigation comprises the problem of manipulation where a system controls its single components relative to the environment and relative to each other. Visual navigation encompasses a wide range of perceptual capabilities of different complexity. Examples include the capability of kinetic stabilization, which refers to the ability of a single compact system to understand and control its own motion, independent motion detection, obstacle avoidance, visual interception, prey catching or target pursuit, homing or docking, hand-eye-coordination, and many more. By means of a few capabilities, all of which are only concerned with the movement of a single compact sensor, the synthetic approach is explained here.

In the past research on navigation has mainly been based on the reconstruction paradigm described by Marr. All navigational tasks have been considered as applications of the general *structure from motion problem* . The idea was to recover the relative 3D-motion and the structure of the scene in view from a given sequence of images taken by an observer in motion relative to its environment. Of course, if structure and motion can be computed, then various subsets of the computed parameters provide sufficient information to solve many practical navigational tasks. However, although a great deal of effort has been spent on the subject, the problem of structure from motion still remains unsolved for all practical purposes. The main reason for this is that the problem is ill-posed, in the sense that its solution does not continuously depend on the input. An extensive discussion of the difficulties involved in the structure from motion problem will be presented in the next chapter.

As discussed before, visual capabilities should be developed in a chronological order starting with the ones that employ the simplest models. In this thesis it will be shown that the most simple navigational capability is the estimation of egomotion. The observer's sensory apparatus (eye/camera), independent of the observer's body motion, is compact and rigid and thus moves rigidly with respect to a static environment. According to the synthetic approach, the capability of egomotion must therefore be developed first, because it only requires one model, that of rigid motion. Furthermore, it will be shown that

egomotion estimation can be performed in a qualitative way by introducing new global constraints which take the form of patterns in the image plane. The qualitativeness of the approach is justified for two reasons: First, the method does not require exact quantitative measurements, since the employed patterns are determined by only the signs of image velocity measurements. Second, since the constraints are defined globally and thus data from all parts of the image plane is considered, the algorithm is stable.

Another capability of the same low complexity is the computation of an object's motion by using only local measurements and thus providing legitimacy to the rigid motion model. It will be shown that a subset of the motion parameters, namely the ones expressing the direction of an object's translation, can be computed by an active observer by using only well-defined image measurements. These two capabilities, the estimation of egomotion and object motion, thus form the bottom of the hierarchy of visual navigational tasks.

Next in the hierarchy follow the capabilities of independent motion detection and obstacle avoidance. Although the detection of independent motion seems to be a very primitive task, it can easily be shown by a counterexample that in the general case it cannot be solved without any knowledge of the system's own motion. Imagine a moving system that takes an image showing two areas of different rigid motion. From this image alone, it is not decidable which area corresponds to the static environment and which to an independently moving object. A motion model more complex than the rigid one has to be employed. In order to perform obstacle avoidance it is necessary to have some representation of space. This representation must capture in some form the change of distance between the observer and the scene points which have the potential of lying in the observer's path. An observer that wants to avoid obstacles must be able to change its motion in a controlled way and must therefore be able to determine its own motion and set it to known values. As can be seen, the capability of egomotion estimation is a prerequisite for developing general independent motion detection as well as obstacle avoidance mechanisms.

Even higher in the hierarchy are the capabilities of target pursuit and homing (the ability of a system to find a particular location in its environment). Obviously, a system that possesses these capabilities must be able to compute its egomotion and must be able to avoid obstacles and detect independent motion. Furthermore, homing requires knowledge of the space and models of the environment, whereas target pursuit relies on models for representing the operational space and the motion of the target. These examples should demonstrate the principles of the synthetic approach, which argues for studying increasingly complex visual capabilities and developing robust (qualitative)

modules in such a way that more complex capabilities require the existence of simpler ones.

This study is devoted to the analysis and development of the capabilities which are most basic for any navigational vision system: the estimation of egomotion and object motion. Motivated by the ideas of the synthetic approach, this thesis in many respects is fundamentally different from most existing work on motion estimation.

The model that has mostly been employed in previous research to relate 2D image measurements to 3D motion and structure is that of rigid motion. Consequently, egomotion recovery for an observer moving in a static world has been treated in the same way as the estimation of an object's 3D motion relative to an observer. The rigid motion model is appropriate if only the observer is moving, but it holds only for a restricted subset of moving objects—mainly man-made ones. Indeed, all objects in the natural world move non-rigidly. However, considering only a small patch in the image of a moving object, a rigid motion approximation is legitimate. For the case of egomotion, data from all parts of the image plane can be used, whereas for object motion only local information can be employed [Fermüller and Aloimonos, 1993b]. Hence, conceptually different techniques for explaining the mechanisms underlying the perceptual processes of egomotion estimation and 3D object motion estimation are developed here.

Usually the problem of motion estimation has been considered as a numerical analysis problem, where sophisticated techniques (such as singular value decomposition, simulated annealing, Kalman filtering, maximum likelihood estimation, etc.) have been employed in order to estimate 3D motion from the geometric and photometric constraints which relate local image motion to the 3D world. In this study a different approach is taken. Unlike most other techniques, the methods developed here do not require as input exact image motion measurements, whose computation is an underconstrained problem. They only utilize well defined measurements, the spatio-temporal derivatives of the image intensity function. As a matter of fact, for deriving the parameters defining a moving observer's motion only the signs of the spatiotemporal derivatives are employed. The approach is based on an analysis of the properties of the motion field [Fermüller, 1993d; Fermüller, 1993b]. Based on the rigidity assumption, new constraints are discovered which define global patterns in the image plane. These constraints are exploited to solve the problem of egomotion estimation in a qualitative way.

The synthetic approach advocates studying visual problems in the form of modules which are directly related to the visual tasks of active observers. If an object is rotating around itself and also translating in some direction, we are usually just interested in its

translation. For an active observer the problem of computing the direction of translation from well defined input is solved. The main advantage of the method is due to the employment of an active observer's abilities to fixate on an object and to track it over a sequence of images, which are shown to be advantageous in simplifying the computations.

1.5 Comparison of the synthetic approach with other proposals

After the realization that general vision is a chimera, in the sense that there is simply too much information in the visual signal for a system to build a task-independent description of it, the traditional views about the architecture of vision systems start to fade away. It becomes clear that a modular architecture built in a principled fashion, successively implementing low, middle, and high level routines, was too general and thus of little practical relevance. Repeatedly unsuccessful attempts to construct working systems conforming to such an architecture [Waxman *et al.*, 1987a] and the appearance of successful systems based on a purposive design [Dickmanns and Graefe, 1988b; Dickmanns and Graefe, 1988a] have strengthened the efforts devoted to the quest for new architectures implementing the marriage of perception and action. The most notable outcomes up to now are the subsumption architecture [Brooks, 1986] and the labyrinthic approach [Sloman, 1989; Aloimonos, 1990].

The approach of Brooks proposes to address the study of intelligent behavior through the construction of working mechanisms. Abilities of progressively increasing sophistication displayed by living organisms should be duplicated in artificial systems. Contrary to the conventional viewpoint that treats implementation as a tool for checking the practicality of a theoretical concept, Brooks views the construction of robots not as a goal, but as a means to achieve a goal. In proposing the subsumption architecture, he suggests a hierarchy of competences for autonomous robots. Listed in increasing order of competence, such capabilities include: avoiding contact with objects, wandering around without hitting things, exploring the world by "seeing places" and moving towards them, building a map of the environment and planning paths between different places, reasoning about the world in terms of identifiable objects and performing tasks related to certain objects, etc. To summarize the subsumption architecture, a controller that achieves some level of competence consists of a set of specialized modules running asynchronously, each of them performing a particular task. The system is structured as a hierarchy of layers of controllers. Each layer has direct access to the data processed by the layers below, and any layer can take over any lower layer when needed.

The approach has had some influence on studies devoted to reactive planning. Never-

theless, it suffers from several drawbacks. First, it is strongly constrained by the hierarchy of competences one tries to simulate. Although one could not argue against the competence of "reasoning about the world in terms of identifiable objects and performing tasks related to certain objects"—after all, this competence amounts to having solved a large part of the vision problem—it is doubtful that Brooks' list of basic competences makes sense from a perceptual and scientific point of view. As a matter of fact, this approach does not even allow one to classify navigational tasks hierarchically without taking into account the system's physiology and purpose. Systems with different mobility characteristics, different amounts of computational capacity, and different purposes and goals, possess different navigational capabilities. Brooks' approach suffers from the same curse of generality that weakened Marr's modular architecture. The subsumption architecture might be viewed as an effort to emulate evolution. However, design is, by definition, the exact opposite of natural evolution. In order for the subsumption architecture to have a solid scientific and engineering basis, it must provide a systematic way of creating a hierarchy of competences by taking into account the system's purpose and physiology.

Another criticism of the subsumption architecture was brought forward by Aloimonos [1990]. He claimed that one of the main disadvantages of this architecture is that it does not allow direct communication between any two or more levels. Furthermore, he dismissed the architecture as incomplete, because there does not exist a single complete ordering of visual systems. A simple system (organism) could have capabilities that a more sophisticated system lacks. Aloimonos then proposed the concept of the labyrinthic architecture, and described some aspects of it by means of an example (the Medusa machine). The idea behind this approach is that knowledge representation is the most important issue regarding the design of visual systems, an issue which Brooks totally left out, as he took the world itself to be the repository of knowledge. A system perceiving and acting on its environment should, according to Aloimonos, be viewed as a collection of behaviors which involve various cognitive processes—vision, planning, reasoning, memory, etc. The representation of knowledge within the behaviors could be achieved using Bayes nets, discrete event dynamic systems, I/O automata, or other formalisms used for modeling distributed processing systems. The name "labyrinthic" stresses the lack of principled design and the dependence of a system's architecture on its physiology and purpose. The concepts behind the labyrinthic architecture are based on the ideas of embodiment of categories [Lakoff, 1987].

Whereas the subsumption and labyrinthic architectures have the grand vision of explaining the big picture and setting the foundations for the design of intelligent systems, the synthetic approach described above has a much more modest goal. While it

draws ideas from both approaches, it recognizes that intelligence is an amalgam of many information-processing and information-representation abilities. The synthetic approach considers the goal of representing a global view of how an intelligent system functions, as premature from a technical standpoint. Even if we knew the global structures of intelligent vision systems, we would not be able to design them before developing their basic visual capabilities. The statement that navigational competences form a hierarchy is not very useful, if we do not provide a systematic way of developing these competences. The synthetic approach acknowledges the fact that perception is embodied [Lakoff, 1987], but this realization becomes helpful only when we find out exactly how it is embodied.

The most basic visual capabilities found in insects, arthropods, fish, birds, and primates are the ones based on motion. Many research studies on this topic have allowed the developments in this thesis, a set of robust algorithms for the analysis of visual motion. These algorithms can be shared, appropriately modified, by the whole spectrum of vision systems. Therefore, they constitute building blocks that can be used in later developments. As we then move up in the hierarchy which is defined by the components' model complexity, we will eventually find the best way to put the components together in order to create the anthropomorphic automata of the future.

1.6 Organization of the thesis

Chapter 2 presents a literature survey of existing motion analysis algorithms. In Chapter 3 a short description of the contribution of this thesis is given, the input is described, and the choice of the coordinate system used in the modeling of different capabilities is discussed. Chapter 4 is devoted to the computation of object motion for an active observer. Chapter 5 is concerned with the estimation of egomotion. First, new constraints relating 2D image measurements to 3D motion parameters are introduced. Then, these constraints are utilized to develop parameter estimation techniques for the most general case (a passive observer moving with three translational and three rotational degrees of freedom). In Chapter 6 it is shown that if the observer is active and supplies additional information the general results can be exploited to solve various navigational tasks with only a small computational effort. Chapter 7 concludes the thesis with a summary of its contributions.

Chapter 2

Visual Motion Analysis

2.1 Historical overview

The significance of motion perception in estimating depth relations in a scene has long been recognized [Helmholtz, 1896]. Numerous psychological studies have been devoted to understanding the power of the human visual system in interpreting the structure and motion of the 3D-scene. Experiments have been performed that demonstrate that humans are able to interpret images of moving scenes correctly even when the objects are unfamiliar and when the static view of the objects does not contain information about structure at all [Johansson, 1973; Ullman, 1979], a phenomenon that is referred to as the "kinetic depth effect" [Wallach and O'Connell, 1953].

The basic concept of using image displacements as a principal source of information about scene structure and image motion is generally credited to Gibson [1950]. Researchers then started to provide a mathematical framework for many of Gibson's ideas [Koenderink and van Doorn, 1975; Koenderink and van Doorn, 1976; Prazdny, 1980]. Nowadays most studies in computer vision devoted to visual motion analysis are dominated by the computational approach of Marr [1982]. The goal is to recover from dynamic imagery the 3D motion parameters and the structure of the objects in view. The suggested strategy attempts to solve the problem in two steps [Ullman, 1979]. First, accurate image displacements between consecutive frames have to be computed, either in the form of correspondences of features or as dense motion fields (optical flow fields). In the case where the motion between image frames is relatively "large", image features are isolated and tracked through a sequence of frames. This is the so-called correspondence problem. Otherwise, in the case of "small" motion, the dynamic imagery is regarded as a three-dimensional function of two spatial and one temporal variables, and from the

spatio-temporal derivatives and some additional information derived by making assumptions about this function, the velocity in the image plane, or optical flow, is computed. In a second step, the 3D motion and the structure of the scene are computed from the equations relating the 2D image velocity to the 3D parameters. This step is called the "structure from motion" problem.

Ullman, who first applied this strategy in a computational approach, describes many psychophysical experiments to support the viewpoint of two independent processes being involved. He argues that in human motion perception the correspondence of features is established completely independently of the 3D scene interpretation and even independently of the global forms of objects' outlines; it is only established between small components (or features) of objects. This viewpoint is not universally accepted among psychologists. Others set up experiments that defeated the two-step theory [Jenkin and Kolers, 1986]. Nevertheless, most of the motion research in computer vision has arisen from this computational theory, and therefore the majority of existing studies are devoted either to the problem of correspondence or optical flow or to the computation of structure from motion.

The classical motion estimation problem is concerned with the inference of 3D information from a sequence of images derived by a monocular observer. Although hybrid techniques have also been developed, where motion information is combined with shape information obtained from a stereo system, the discussion here is restricted to pure monocular methods. In general, the reconstruction of the 3D motion and the structure of scene in view cannot be unique, because in principle any arbitrary motion is possible. Some motion model therefore has to be employed. Since many of the structures in the visual world are rigid, rigidity or at least piecewise rigidity is usually assumed as a basis for structure from motion computation.

The plethora of mathematical models and computational techniques that have been employed in the past might be classified in various ways. In the past, methods were distinguished according to whether they used as input correspondence or optical flow. Other criteria in use are the choice of image projection model, the types of features which are corresponded, whether the proposed solutions involve iterative numerical methods or are in closed form, and whether a long sequence of images is used as input or correspondences between only two or three images are established. Here, research on the motion estimation problem is first described from a historical point of view.

Following the two-step approach, the two problems of computing correspondence or optical flow and the estimation of structure from motion have been studied in parallel.

In the evolution of the study of the structure from motion problem three phases can be distinguished. First, work dealt with the question of the existence of a solution, i.e. can we extract any information from a sequence of images about the structure and 3D motion of the scene that cannot be found from a single image? Several theoretical results have appeared that deal with questions such as what can be recovered from a certain number of feature points in a given number of frames ([Ullman, 1979], [Aloimonos and Brown, 1989]) under either orthographic or perspective projection. Then, the uniqueness aspects of the problem were studied. Non-linear algorithms for the recovery of structure and motion from point or line correspondences and optic flow appeared increasingly in the literature. Algorithms dealing with correspondence were based on iterative approximation techniques, so that they lacked guaranteed convergence. Later, "linear" algorithms were developed and closed form solutions introduced [Tsai and Huang, 1984; Spetsakis and Aloimonos, 1990; Adiv, 1985] that allowed proofs of uniqueness.

Although research along these lines has been accompanied by many experiments, none of the existing techniques can be used as a basis for an integrated system, working robustly in a general environment. Motion estimation, as it has been traditionally addressed, involves steps that are mathematically ill-defined, i.e. the computation of optical flow and the correspondence of features are underconstrained. Therefore additional assumptions have to be made, and the algorithms used in the computation of structure from motion are very sensitive to noise. As a result, motion research has shifted its focus to the robustness issue.

In order to overcome sensitivity, researchers started using redundant information. Algorithms have been developed that employ long sequences of image frames as input (multi-frame approaches). These techniques, however, still require the correspondence of features. Furthermore, in most studies many assumptions about the scene and the continuity of the motion were employed.

Since the computation of optical flow or correspondence is provably ill-posed, some researchers started an effort to circumvent this computational step. The only image representation that is well defined is the image motion component perpendicular to gray-level edges. A small number of 3D motion estimation studies do not rely on exact image motion measurements, but only consider the flow measurements parallel to the image gradient. These are, in the continuous case, the so-called direct methods which only employ the spatio-temporal derivatives, and in the discrete case, the methods using lines as the input features.

2.2 Computation of exact image measurements

The relative motion between the 3D scene and the observer (camera) gives rise to apparent motion in a sequence of images. The instantaneous 3D motion of any object point results in a velocity vector being assigned to that point's pixel in the image plane. The computation of these exact image velocities is the first computational step employed in most motion estimation techniques. To solve this problem, either features in different image frames are corresponded or, if the motion between different frames is very small, a dense vector field is computed.

2.2.1 The correspondence problem

The problem of correspondence is to establish a match between parts of one image frame and their counterparts in a subsequent frame that represent the same three-dimensional object. Correspondence is usually addressed in two steps. First, primitives in the image are chosen and a similarity measure between primitives in different frames is defined. Correspondence is then computed by minimizing some function over all possible matches. Ullman, who was the first to study the correspondence problem for 3D motion interpretation, argues that the elements to be corresponded should not be raw image pixels, but image features (or tokens) like line fragments, bars, and blobs. He also suggests that the matching of features should be local and matches between different pairs of features have to be independent. The independence assumption reduces the problem of minimizing the mean of the similarity measures in the least squares sense to a linear problem. However, independence is not a valid assumption in general. Most of the algorithms found in the literature fall within the framework of token matching presented by Ullman. They only differ in the complexity and dimensionality of the monocular primitives, the similarity measure chosen, and the way the minimization problem is solved.

The selected tokens are features that differ from their surroundings, such as corners [Nagel, 1983] or points of high interest [Moravec, 1977] evaluated by various operators. Different approaches have been developed to address the minimization problem. Some approaches use the structure from motion problem to help solve the correspondence problem. By restricting the possible shape of the scene or the motions of the objects in view, only certain combinations of correspondences are possible. Smoothness assumptions about the motion in space and time are made [Cheng and Aggarwal, 1990; Weng *et al.*, 1987a]; these assumptions basically relate to a 3D translation parallel to the image plane. Explicitly, the case of pure translation has been studied by Lawton [1983] and Sethi and

Jain [1987], who exploit the fact that under translational motion points are all moving radially away from one point. In order to make the minimization computationally feasible heuristics have been employed. For example, the assumption is made that features move in a way similar to features in their neighborhoods. The matching process is then solved locally by means of iterative relaxation algorithms ([Barnard and Thompson, 1980; Fang and Huang, 1984; Ranade and Rosenfeld, 1980].

The correspondence problem is ill-posed by its nature. One of the assumptions underlying any correspondence technique is that features in the image plane correspond to moving features in the scene. In the general case, however no operator can be constructed that solves this problem. Furthermore, any approach proposed for the solution of the minimization of a functional relating features before and after the motion is based on various assumptions about the structure of the scene in view or the motion involved. These assumptions are incorporated into the form of the functional to be minimized. If they do not hold, the minimization will produce erroneous results.

2.2.2 Optical flow

The relative motion of the observer with respect to the scene gives rise to motion of the brightness patterns in the image plane. The instantaneous changes of the brightness pattern in the image plane are analyzed to derive the optical flow field, a two-dimensional vector field reflecting the image displacements.

The optical flow value of each pixel is computed locally—that is, only information from a small spatio-temporal neighborhood is used to estimate it. In general, it is not possible to compute an image point's true velocity by observing only a small neighborhood. Imagine that you are watching a feature (line, bar, piece of contour) through an aperture that is small compared to the feature at two instants of time (see Fig. 2.1). Watching through this small aperture, it is impossible to determine where each point of the feature has moved to. The only information directly available from local measurements is the component of the velocity which is perpendicular to the feature, the "normal flow". We cannot determine the component of the optical flow parallel to the feature. This ambiguity, which is referred to as the "aperture problem", exists independently of the technique employed for local estimation of flow. In cases where the aperture is located around an endpoint of a feature, the true velocity can be computed, because the exact location of the endpoint at two instants of time is known. Thus, the aperture problem exists in regions that have strongly oriented intensity gradients (e.g. edges), and may not exist at locations of higher-order intensity variations, such as corners.

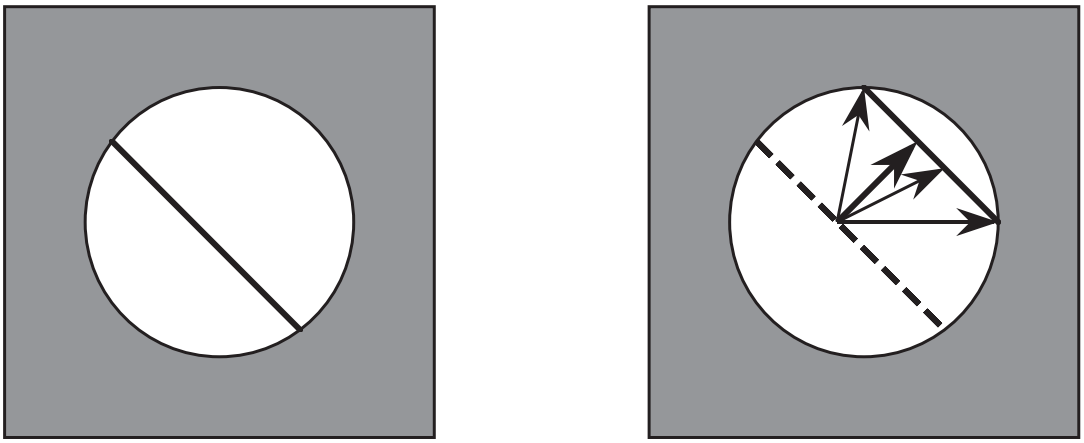


Figure 2.1: (a): Line feature observed through a small aperture at time t. (b): At time $t + \delta t$ the feature has moved to a new position. It is not possible to determine exactly where each point has moved to. From local measurements only the flow component perpendicular to the line feature can be computed.

Therefore, any optical flow procedure has to involve two computational mechanisms. In the first step, assuming the conservation of some form of information about the image, locally available velocity information is computed. According to the kind of information that is assumed not to change, three different kinds of approaches can be distinguished [Singh, 1990]: gradient based approaches, which assume that the image intensity does not change; correlation based techniques, which are based on the assumption of conservation of the local intensity distribution; and spatio-temporal energy-based approaches, which are analogous to gradient-based approaches in spatio-temporal frequency space. In a second step, in order to compute the other component of the optical flow vectors, additional assumptions have to be made. Either some kind of smoothness is assumed or the shape of the scene is geometrically modeled to obtain constraints on the optical flow values.

The gradient-based approach introduced by Horn and Schunck [1981] is based on the assumption that for a given scene point the intensity E at the corresponding image point remains constant over time. If scene point P projects onto image point (x, y) at time t and onto image point $(x + \delta x, y + \delta y)$ at time $(t + \delta t)$, one can write

$$E(x, y, t) = E(x + \delta x, y + \delta y, t + \delta t). \tag{2.1}$$

If we develop the right hand side of equation (2.1) in a first-order Taylor's series expansion and denote by $u(x, y)$, $v(x, y)$ the velocity $(\frac{dx}{dt}, \frac{dy}{dt})$ of image point (x, y) (the

components of the optical flow vector), we obtain the following equation that relates the optical flow to the partial derivatives E_x, E_y, E_t of E:

$$E_x u + E_y v + E_t = 0. \tag{2.2}$$

From this constraint, which has been called the optical flow constraint equation [Horn and Schunck, 1981], the aperture problem can be easily derived. The linear equation defines a line in velocity space ((u, v)-space). Thus only the vector component in the direction of the gradient (E_x, E_y) can be computed. If we use the motion constraint equation, everywhere in the image the normal flow can be computed.

In order to derive a second constraint, Horn and Schunck suggested the assumptions that the optical flow varies smoothly in most parts of the image. As a measure of departure from smoothness they used S, the sum of the squares of the flow's first order derivatives, i.e. $S = u_x^2 + u_y^2 + v_x^2 + v_y^2$. If the optical flow constraint equation is written as $L = E_x u + E_y v + E_t = 0$, then the following expression has to be minimized: $\lambda \int\int L^2 dxdy + \int\int S dxdy$, which denotes the sum of the error in the image motion and the departure from smoothness over all data points. λ denotes a weighting function that controls the relative importance of smoothness and consistency with the image data. The above minimization problem can be solved using methods of the calculus of variations by deriving the corresponding Euler-Lagrange equations, which turn out to be a pair of elliptic second-order partial differential equations. The method works well for image areas where the optical flow is a smooth function, but near discontinuities, for example at motion boundaries, the solution is much smoother than it should be.

Employing additional smoothness assumptions to recover an unknown function when the available constraints are not sufficient is a general mathematical technique known as regularization. Although it has been widely used before, Poggio *et al.* [1984] have officially introduced it to computer vision as a tool for applications in low-level vision. Various other regularization techniques, which use different smoothness constraints, have been developed to reconstruct optical flow. Hildreth [1984] proposed requiring smoothness only along image contours which are detected as zero-crossings of the Laplacian of Gaussian filtered image. With this technique, optical flow is computed only near discontinuities. The edge detector cannot differentiate between texture and motion boundary discontinuities; therefore smoothing across non-motion boundaries is prevented, where actually it should be propagated. Nagel [1987] and Nagel and Enkelmann [1986] introduced a more general technique, requiring smoothness along the orientation of contours by employing a weighting function in the smoothness term. Uras *et al.* [1988] require that the gradient of the image brightness does not change over time ($\frac{dI}{dt} \bigtriangledown I = 0$), an

assumption that is valid only at planar brightness patches in the image plane.

The main problem with all these smoothing techniques is the existence of motion boundaries. This realization has led to various studies dealing explicitly with discontinuities, which follow one of two general approaches. Discontinuous regularization techniques [Shulman and Hervé, 1989; Shulman, 1990; Aloimonos and Shulman, 1989] reconstruct the optical flow by smoothing as little as possible near discontinuities. Probabilistic approaches such as Markov random field techniques [Geman and Geman, 1984] localize the discontinuities, or more exactly the places where the probability of having a discontinuity is highest, in order to ignore these values and reconstruct the optical flow at all other locations.

A gradient-based technique different from the one introduced in [Horn and Schunck, 1981] has been developed by Nagel [1983] and Enkelmann [1988]. These techniques also fall within the framework of regularization approaches. In order to compute the optical flow values at corner points they employ a Taylor series expansion up to the second degree in equation (2.1), and then minimize a functional expressing the deviation of the image motion over a neighborhood. Some other gradient-based techniques, in which not the image intensity function itself but functions of it are assumed to be preserved, have been introduced by [Buxton and Buxton, 1984; Waxman *et al.*, 1988]. A comparative study of minimization approaches using a smoothness term applied to gradient-based techniques can be found in [Snyder, 1989].

A second class of techniques to derive locally available image information is based on correlation. Assuming the conservation of the local intensity distribution, a small window in the first image is searched for a corresponding window in the next image, where the search criterion is maximal cross-correlation. The cross-correlation, which is a measure of similarity, is usually applied to low-pass filtered [Wong and Hall, 1978] or band-pass filtered images [Burt *et al.*, 1983; Anandan, 1989; Anandan and Weiss, 1985].

A third group of local image motion estimation techniques computes velocity in spatio-temporal frequency space [Watson and Ahumada, 1985; Adelson and Bergen, 1985; Fleet and Jepson, 1990; Heeger, 1988]. Similar to the optical flow constraint equation, in frequency space the spatial and temporal frequencies $(\omega_x, \omega_y, \omega_t)$ of any pattern are related to the velocity components (u, v) through a linear equation $(\omega_x u + \omega_y v + \omega_t = 0)$ which represents a plane in spatio-temporal frequency space. The motion of the visual stimulus is encoded in the orientation of the plane. Spatio-temporal energy filters such as Gabor filters which are tuned to certain patterns and motions are used to derive the normal component of the flow. Heeger goes further and also addresses the problem of

complete optical flow recovery. He assumes translation of a highly textured surface, so that in every pixel's neighborhood a number of spatial frequency components are available.

In some studies assumptions not about smoothness but about scene geometry have been employed to derive the additional constraint necessary to compute optical flow (e.g. [Subbarao, 1988; Murray and Buxton, 1984; Waxman and Wohn, 1985; Waxman *et al.*, 1987b]). For example, Waxman and Wohn [1985] assume locally planar surfaces. Under this assumption the flow components are second-order polynomials in the image coordinates. The constants in the polynomials are interpreted as Taylor coefficients, and a polynomial is fitted to the normal flow values in a small area in order to recover the optical flow.

Optical flow as previously defined is the apparent motion of image brightness patterns. As pointed out by Horn and Schunck [1981], in the general case optical flow is not equal to the "motion field", the projection of the 3D motion vector on the image plane. For example, a sphere at a fixed location illuminated by a moving light source will cause a change of shading in the image plane. Therefore, a non-zero optical flow field will be obtained although the motion field is zero everywhere. On the other hand, zero optical flow will be measured if a sphere with no texture on it is rotating under arbitrary, fixed illumination. Luckily, such extreme cases occur rather rarely. Verri and Poggio [1989] have analyzed the difference between the two vector fields assuming Lambertian reflectance. Only under very restrictive assumptions, namely when a Lambertian surface which is illuminated uniformly (in space and time) undergoes translation, are the normal component of the optical flow and the motion flow the same. Since in this thesis only normal flow is used, the issue of the difference between the normal flow and the normal component of the motion field will be further elaborated later.

2.3 Segmentation

As a basis for structure from motion computations the rigid motion model is usually employed. In the case where the scene consists of multiple moving objects, every object is assumed to be moving rigidly relative to the observer. Thus, the image first has to be segmented into regions that correspond to areas in the scene which undergo the same relative motion. For a moving observer to perform segmentation on the basis of time varying imagery amounts to detecting independently moving objects. Such objects give rise to discontinuities in the flow field at their boundaries. These, however, are not the only places where the flow field is not smooth; surfaces that are separated in depth also

cause discontinuities.

In the literature the detection of independently moving objects has always been treated by searching for places in the image where the motion field deviates from rigid motion. In all these studies, some knowledge about either the observer's motion or the structure of the scene is assumed. Existing studies can be classified into two categories: approaches which employ only the spatio-temporal intensity function as input and approaches which require the computation of optical flow.

Optical flow fields are used by Jain [1984] to discriminate non-stationary objects using polar transforms for observers undergoing only translation. Adiv [1985] performs segmentation by assuming planar surfaces undergoing rigid motion. In order to determine the motion parameters of the moving plane and to group the motion vectors he employs a Hough transform. Thompson and Pong [1990] describe various principles for detecting independently moving objects if the observer has knowledge about its own motion or about the structure of the scene. Knowing the motion also means knowing the direction of the flow vector at every point. The method detects the places where deviations from the expected direction occur. In the case where the shape is known they compare the values of the motion and stereo disparities between points on different sides of flow discontinuities to discriminate between motion and depth boundaries.

Techniques using as input the spatio-temporal derivatives of the image intensity function (or the so-called normal flow) have been developed in [Sharma and Aloimonos, 1991] and [Nelson, 1991]. For an observer moving with known translational motion Sharma and Aloimonos detect places in the image where the normal flow vector's direction is inconsistent with the range of possible directions corresponding to the actual translation. With such a strategy only a small number of discontinuities will be recognized. In order to increase this number they repeatedly change the direction of translation and perform new computations. Nelson [1991] developed two different techniques for independent motion detection. He describes the geometrical area in which a normal flow vector lies if the observer's motion is known and the depth of the scene is constrained by an upper bound. With the first technique normal flow vectors are detected as originating from independently moving objects if they do not fall in this area. The second technique is designed only to detect objects which rapidly change their motion. The change in flow will be smooth for static image scene points and large for rapidly accelerating objects.

Several authors have also proposed methods for embedding the detection of independently moving objects in general motion analysis systems. For example, Burt *et al.* [1989] model the shape of the scene by a plane and consider orthographic projection. The two

components of the optical flow vectors, which under these simplifying assumptions are described as linear functions of the image coordinates, are computed at different levels of resolution. Discontinuities are detected if local values do not conform with the values computed at lower resolution.

2.4 Non-rigid motion models

More complicated models have been used in the computation of shape from motion that handle some aspects of non-rigidity. Ullman [1983] presented an approach to non-rigid motion estimation based on computing the object structure and motion that is most consistent with the data and as rigid as possible. In [Koenderink, 1984] bending transformations are studied and global data is used to determine motion information. Motion and shape recovery for isometric motions have been studied and closed form solutions have been developed for the case where motion information is used in conjunction with photometric stereo [Chen and Penna, 1986] and with shape information about the object before the motion [Penna, 1992]. In [Goldgof *et al.*, 1988] Gaussian curvature is used in the study of piecewise rigid and homothetic motions. Shulman and Aloimonos [1988] describe non-rigid motion fields locally as a sum of vector fields, with the weighting factors being constant. Given the shape of the object, they apply regularization to find the smoothest motion consistent with the image data. Various deformable spatial models, like superquadrics and snakes have been used to model non-rigid motion [Terzopoulos *et al.*, 1988; Pentland *et al.*, 1991]; the purpose of these studies, however, was mainly visualization and not motion estimation.

2.5 What could be computed

Assuming that detailed and accurate image information is available in the form of correspondences or dense optical flow fields, the 3D motion and the structure of the scene are estimated from the equations relating 2D to 3D measurements. These equations are determined by the specific geometric model of image formation that is used. Different geometric projection models have been employed. Orthographic projection [Ullman, 1979] leads to linear equations, but in general is not realistic and should be considered only when lenses of very long focal length are used and the field of view is very small. A more adequate model is given by perspective projection. The image is projected either on a sphere or on a plane. The resulting equations relating 3D to image motion are nonlinear. From 2D imagery alone, not all motion components can be computed. The parameters

which are computable under perspective and orthographic projection are different, and also shape has a different meaning under these two projections.

For reasons of simplicity, let us assume a coordinate system $(OXYZ)$ which is fixed to the camera with O being the nodal point of the camera and the Z-axis pointing along the optical axis. In the discrete case the scene in the first view is regarded as a set of points $P_i = (X_i, Y_i, Z_i)$ in the 3D coordinate system. The image plane is assumed to be perpendicular to the optical axis at a distance from the origin equal to the focal length. A natural coordinate system (ox, oy) parallel to (OX, OY) is induced on the image plane, with its origin o at the intersection of the optical axis with the image plane. If the camera undergoes a rigid motion then at a second instant of time every point P_i has coordinates $P_i' = (X_i', Y_i', Z_i')$ in the new coordinate system. Because of rigidity, we have

$$P_i' = RP_i + T \qquad \text{for } i = 1 \ldots n, \tag{2.3}$$

where R is a 3 by 3 rotation matrix whose elements $((r_{11}, r_{12}, \ldots, r_{33}))$ depend on three independent parameters, corresponding to rotations around the X-, Y-, and Z-axes and a translation vector $T = (\Delta X, \Delta Y, \Delta Z)$. In Fig. 2.2 the imaging geometry using perspective projection is shown.

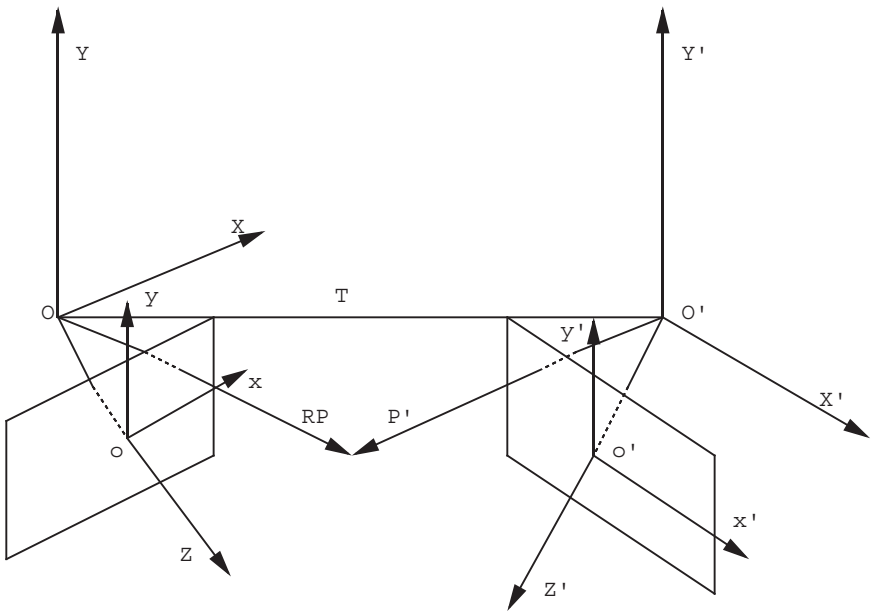


Figure 2.2: Imaging geometry under perspective: The discrete case.

Under orthography every point (X, Y, Z) projects onto a point $(x, y) = (X, Y)$ in the image plane. Thus, under this projection equation (2.3) becomes

$$\begin{aligned} x_i' &= r_{11}x_i + r_{12}y_i + r_{13}Z_i + \Delta X \\ y_i' &= r_{21}x_i + r_{22}y_i + r_{23}Z_i + \Delta Y \end{aligned} \tag{2.4}$$

Clearly, from these equations the translational component parallel to the Z-axis cannot be derived. Translation parallel to the image plane is displayed as a parallel shift of all image points. The 3D coordinates of the scene points depend only on the rotational motion. Since the translation along the Z-axis cannot be derived, structure under orthography is derivable only as the differences between depth values (i.e., from any two points P_i, P_j only $(Z_j - Z_i)$ can be computed).

Under perspective projection image points and their 3D counterparts are related as follows: $x = \frac{Xf}{Z}$ and $y = \frac{Yf}{Z}$. If we set f, the focal length, equal to 1, equation (2.3) becomes

$$\begin{aligned} x'_i &= \frac{(r_{11}x_i + r_{12}y_i + r_{13})Z_i + \Delta X}{(r_{31}x_i + r_{32}y_i + r_{33})Z_i + \Delta Z} \\ y'_i &= \frac{(r_{21}x + r_{22}y + r_{23})Z_i + \Delta Y}{(r_{31}x + r_{32}y + r_{33})Z_i + \Delta Z} \end{aligned} \tag{2.5}$$

Dividing the numerator and denominator by ΔZ, it becomes clear that the depth values Z and Z' as well as the translational components are computable up to a scale factor. For example, an object at distance Z moving with translation T will produce the same image measurements as a similar object that is twice the distance from the camera and is moving with twice the translational velocity. Thus, what is meant by structure under perspective projection is the ratio of the depth values between two image points (i.e. for any two points P_i, P_j, we can only compute $\frac{Z_i}{Z_j}$). Furthermore, under perspective projection the depth values depend on the translational values, as opposed to the rotational values, for the case of orthography. As regards the translation, only its direction can be computed $(\frac{\Delta X}{\Delta Z}, \frac{\Delta Y}{\Delta Z})$. These coordinates define the so-called focus of expansion—the point at which the image plane will be intersected by the trajectory of the camera's nodal point moving with translation $(-\Delta X, -\Delta Y, -\Delta Z)$.

2.6 Structure from motion under orthography

In the computational analysis of the structure from motion problem the simplified model of orthography preceded the use of perspective projection. Ullman [1979] showed that three views of four non-coplanar points are sufficient to determine shape and motion uniquely (up to a Necker reflection). That reconstruction of motion and shape from only two views is not possible has been proved for point correspondences [Aloimonos and Brown, 1989] as well as optical flow [Hoffman, 1982]. Optical flow allows one to compute surface orientation only as a straight line in gradient space, and in the case of correspondence the surface orientation lies on a hyperbola. By imposing additional smoothness

assumptions Aloimonos and Brown [1989] derived structure and motion from two views through the solution of a regularization problem. Due to the limited applicability of the orthographic projection model only a relatively small number of studies have been devoted to it. Recently, however, starting from the work on affine shape by Koenderink and van Doorn [1991], orthographic projection models have again been used in various applications [Ullman and Basri, 1991; Jacobs, 1992]. It turns out that two views are enough to reconstruct affine shape, that is, to reconstruct the affine coordinates of every object point with respect to three linearly independent vectors attached to the object. Actually, affine coordinates remain constant not only under rigid motion but under any linear transformation.

2.7 Discrete approaches

The problem of structure from motion becomes much harder when a perspective projection model is employed. As is clear from equation (2.5), the image motion is related to the 3D motion and structure in a non-linear way. It is this non-linearity that has generated a large amount of research in the area of structure from motion.

Initial attempts at addressing this problem started with the observation that five point correspondences [Roach and Aggarwal, 1980] are needed, since any point correspondence gives two equations and the number of unknowns is equal to nine (four scaled depth values, two translational and three rotational motion parameters). Such approaches suggested obtaining the structure and motion parameters through iterative methods [Martin and Aggarwal, 1978]. These techniques were not very successful, because it was later shown using tools from projective geometry [Kruppa, 1913], that five points in two views give rise to ten possible solutions [Faugeras and Maybank, 1990].

Longuet-Higgins [1981] and Tsai and Huang [1984] later presented a linearization technique that opened new avenues to the study of uniqueness proofs. They showed that the image measurements in the first and second frames are related through the equation

$$(x' \;\; y' \;\; 1)^T E (x \;\; y \;\; 1) = 0, \tag{2.6}$$

where $E = T_s R$ with $T_s = \begin{pmatrix} 0 & -\Delta Z & \Delta Y \\ \Delta Z & 0 & -\Delta X \\ -\Delta Y & \Delta X & 0 \end{pmatrix}$, a matrix of the translational components, and R the rotation matrix. From the nine parameters of the E-matrix, the so-called essential parameters, only eight can be determined because of the unrecoverable scale factor in the translation. Thus, given the correspondence of eight points in general

position the E-matrix can be computed uniquely by solving eight linear equations. Having found E, the motion parameters are derived uniquely by taking the singular value decomposition of this matrix. On the basis of this decomposition Tsai and Huang [1984] showed that seven points, unless they are traversed by two planes with one plane containing the origin or by a cone containing the origin, uniquely allow the computation of the five motion parameters from two views. However, this linearization technique is extremely sensitive to errors in the image measurements and thus it cannot give rise to a robust algorithm.

In order to increase the stability of the linearization technique, researchers have considered using more than eight point correspondences to obtain the E-matrix. Least squares minimization, which was originally used, is not justified, because it requires the essential parameters to be independent. Since these parameters are combinations of five motion parameters the statistical assumptions underlying the least squares technique are violated. Subsequently, constrained minimization techniques were developed [Spetsakis and Aloimonos, 1988; Horn, 1990; Aisbett, 1990; Philip, 1991]. These techniques address the minimization in two steps; in one step they solve for the translational and in the other for the rotational components. In [Spetsakis and Aloimonos, 1989; Daniilidis and Nagel, 1990] the linearization technique has been analyzed statistically. Spetsakis and Aloimonos [1989] proved that under the assumption of Gaussian noise the best result for the E-matrix is obtained by weighted least mean square minimization, where the weight factors are dependent on the motion parameters and the image coordinates. However, they also demonstrated that even the best possible technique is immensely sensitive to noise, and an error of 1 percent in the values of the image measurements (unit of measurement equal to the focal length) causes about 100 percent error in the values of the 3D motion parameters.

2.8 Multi-frame approaches

The desire for robustness initiated the use of long image sequences as input to algorithms for the computation of structure and motion. In so-called multi-frame approaches, motion estimation is addressed as a nonlinear minimization problem. A variety of multi-frame techniques have appeared. Most of of these techniques iteratively sharpen the values of the unknown parameters. For long image sequences the orthographic projection model has also been used [Tomasi and Kanade, 1991]. Techniques have been published in which the motion is assumed to be known and only the structure has to be estimated [Matthies *et al.*, 1989; Crowley and Stelmaszyk, 1990], as

well as techniques which rely on additional stereo data, so that the structure only needs to be computed more accurately [Young and Chellappa, 1990; Cui *et al.*, 1991; Weng *et al.*, 1987b]. Other techniques employ geometrical models of the scene in view [Chandrashekar and Chellappa, 1991]—for example, the algorithms developed for road following by Dickmanns and Graefe [1988b; 1988a].

In most techniques a model of the motion's time evolution is employed, and the parameters of the motion and scene are estimated recursively. This recursive estimation usually is accomplished through the use of Kalman filters. In this optimization technique, basically two algebraic stochastic models are employed: the plant model, to estimate the change in the state variables (functions of the motion and depth parameters) and the measurement model, relating the state variables to the image measurements. The stochastic disturbance terms in both models are assumed to be zero mean, unit variance Gaussian. The cycle of recursive estimation at every step (for every new image frame) consists of a prediction of the new state by employing the plant equation, followed by a correction update by means of the measurement model. The state variable after considering a number of frames is viewed as the value of an a posteriori probability density function, dependent on the measurements of the already evaluated frames. For linear measurement models [Legters Jr. and Young, 1982], linear Kalman filtering is the optimal least mean square estimator. In the general case, however, the estimation model is nonlinear. Iterated extended Kalman filters (IEKF) have therefore been employed which iteratively search for the peak of the a posteriori conditional probability; but such an iterative technique cannot guarantee convergence to the maximum value.

A technique that does not require a motion model has been developed by Spetsakis and Aloimonos [1991]. Between any two image frames they estimate for every point the normal distances of the rays which pass through the image center and the image point. Motion estimation is formulated as least mean squares minimization of these normal distances. The optimization is solved in two steps, one for the translation and one for the rotation.

2.9 Continuous approaches

When optical flow fields are used and the motion between successive image frames is very small, velocity is expressed as instantaneous change of position in order to derive the equations relating the 3D motion to the image measurements. The relative velocity $\dot{X}$ of every point X with respect to a camera that moves with translational velocity $T = (U, V, W)$ and rotational velocity $\omega = (\omega_x, \omega_y, \omega_z)$ when the XYZ-coordinate system

is attached to the focal point of the camera is given by

$$\dot{X} = -\omega \times R - T \tag{2.7}$$

Differentiating the expressions for perspective projection ($x = \frac{X}{Z}, y = \frac{Y}{Z}$) with respect to time and substituting for $\dot{X}$ results in the following equations for the optical flow:

$$\begin{aligned} u = \frac{dx}{dt} &= \frac{-U + xW}{Z} + \omega_x xy - \omega_y(x^2 + 1) + \omega_z y \\ v = \frac{dy}{dt} &= \frac{-V + yW}{Z} + \omega_x(y^2 + 1) - \omega_y xy - \omega_z x \end{aligned} \tag{2.8}$$

Assuming that the surface in view is smooth and thus the optical flow field varies smoothly, a series of algorithms have appeared that estimate the 3D motion from only local information. The surface patch in the scene around the optical axis is approximated by either a plane [Longuet-Higgins and Prazdny, 1980] or a quadratic [Waxman *et al.*, 1987b]. The resulting depth values are substituted in equation (2.8), a Taylor's expansion is performed, and the velocity vector at the center and its first and second order partial derivatives are related to the motion and surface parameters. The resulting equations are non-linear; as has been shown in [Longuet-Higgins and Prazdny, 1980], at least one equation of degree three has to be solved. For the computation of the derivatives Longuet Higgins and Prazdny suggested using the simple differential operators of divergence, curl, and shear [Koenderink and van Doorn, 1976]. Supported by biological findings, they argued that the human visual system may possess channels tuned to these four basic types of relative motion. In [Thompson *et al.*, 1984] these differential operators are employed to roughly classify object motions according to the prevailing type of motion, such as translation or rotation parallel or perpendicular to the optical axis.

Bruss and Horn [1983] developed a global technique for structure and motion estimation. This technique, however, relies on very unrealistic assumptions. They formulate the problem as least mean squares minimization of the error between the expected and the measured image motion, where the optimization is over the motion parameters and every depth value. Consequently a system of nonlinear equations has to be solved. Prazdny [1981] and later Burger and Bhanu [1990] suggested techniques which explicitly decompose the flow field into its translational and its rotational components. The structure of a translational flow field is such that all vectors lie on straight lines which pass through one point, the focus of expansion or focus of contraction. The rotational flow vectors under perspective are independent of depth. The way the decomposition is addressed is to search for the correct rotation to be subtracted in order to be left with a flow field having the properties of pure translation. To evaluate this property various error functions have

been employed, such as for example the variance of the intersections of one displacement vector with all the other vectors. A decomposition method using spherical projection was introduced by Nelson and Aloimonos [1988]. On the sphere a translational flow field has very nice properties: Its vectors are along geodesics, all emanating from one point and flowing into another point with the two points separated by 180 degrees. The rotational parameters are found from the flow measurements along the equators perpendicular to the coordinate axes. There the translational flow is clockwise on half of the equator and counterclockwise on the other half, while the rotational flow is constant. The motion is estimated by searching for the correct rotation to be subtracted from the vectors along the equators.

Theoretical studies investigating the number of possible solutions for the motion and the scene in view for a given optical flow field have been conducted in [Horn, 1987] and [Negahdaripour, 1989]. It has been shown that a noise-free retinal motion field uniquely determines the 3D motion, unless the surface in view belongs to a certain class of hyperboloids of one sheet.

Despite all these efforts, no optical flow technique has been developed that can be used for robust motion estimation. In fact, since the computation of optical flow is itself an ill-posed problem, only approximations to the true flow field can be derived. Any method that uses optical flow only locally will be very unstable because completely different observer motions produce locally similar motion fields. For example, in an area near the y-axis of the image plane, 3D rotation around the X-axis produces a flow field similar to the one produced by translation along the Y-axis. To address motion estimation globally in an analytic way would mean solving a non-linear optimization problem, a computationally infeasible task in general. A global optical flow technique that does not make unrealistic assumptions has thus not yet appeared.

2.10 Overcoming the aperture problem

Researchers trying to overcome the problems inherent in the discrete and continuous approaches began to use primitives other than points. Such primitives include planar curves and straight lines (infinitely long lines, not line segments). For the case of straight lines the analysis proceeded similarly to the study of point correspondences. First, iterative algorithms [Liu and Huang, 1988; Faugeras *et al.*, 1987] were derived, and later a linearization technique established the uniqueness properties of the approach [Spetsakis and Aloimonos, 1990]. Not much research has been devoted to the extraction of motion from planar contours. Bergholm [1988] studied uniqueness properties. In general it is

impossible to estimate motion from two monocular views unless information is available about the orientation of the plane on which the contours lie [Kanatani, 1990].

These approaches rely on the extraction of features and suffer from numerical instability. Very small errors in the extracted representation of the lines or contours cause large errors in the computed motion parameters. On the other hand, a very promising approach to overcoming the difficulties with retinal correspondences appears to be the use of normal flow. Since techniques along these lines do not rely on the intermediate computation of exact image measurements, but make use only of the changes of the patterns in the image plane, they are referred to as direct methods.

All the studies on the estimation of 3D motion from normal flow fields that have appeared up to now either deal with limited rigid motion models or assume shape information. In [Aloimonos and Brown, 1984] the case of purely rotational motion was studied. By minimizing in the least squares sense the difference between the predicted and observed normal flow values, linear equations relating the rotation parameters to the normal flow were derived. A similar result was reported by Horn and Weldon [1987], who presented several methods for the problem of motion and structure computation for the purely translational case, for the case of translation only, for known rotation, and for known structure. For the case of purely translational motion these authors make use of the the fact that the scaled depth $\frac{W}{Z}$ is positive for all points in the image plane if the observer is approaching the scene. Therefore, every normal flow measurement constrains the location of the FOE; it has to be in the half-plane (or hemisphere in case of spherical projection) which is on the opposite side of the gray level edge from the normal flow vector. From an algebraic point of view every normal flow measurement supplies an inequality. As algorithmic strategies for obtaining the FOE, methods like linear programming and perceptron learning were suggested. In [Negahdaripour and Horn, 1987] and [Negahdaripour, 1986] closed-form solutions are presented for the case in which scene in view is modeled as either a plane or a quadratic patch. In [White and Weldon, 1988] translation and rotation are estimated for an observer rotating around the direction of translation, and recently a hybrid technique has appeared [Taalebi-Nezhaad, 1990], using both optical flow and image gradients for addressing 3D motion in the general case (rotation and translation). In this thesis the general case is addressed. The contribution lies in the introduction of several novel geometric properties of a normal flow field due to rigid motion that give rise to simple pattern matching techniques for recovering 3D motion. Furthermore, it is shown how activities such as tracking and fixation facilitate motion estimation.

Chapter 3

Preliminaries

3.1 Overview

The synthetic approach imposes a hierarchy on navigational capabilities. This thesis presents solutions to the most basic processes of 3D motion estimation on the basis of a sequence of images acquired by a monocular observer. As discussed before, in the past all navigational tasks were regarded as applications of the general principle of performing complete scene recovery, usually using the model of rigid motion. However, while it is appropriate for the purpose of egomotion estimation to model the relative motion between the scene and the observer as rigid, for the case when the motion of objects is computed, this assumption in general can be used only locally. Therefore, here the processes of egomotion recovery and 3D object motion recovery are approached in two different ways. Below an outline is given of the active and qualitative techniques developed in the next three chapters and the results obtained.

It has been suggested that visual problems should be studied in the form of modules that are directly related to the visual task the observer is engaged in. Along these lines, it is argued that in many cases when an object is moving in an unrestricted manner (translation and rotation) in the 3D world, we are interested only in the motion's translational components. For a monocular observer, using only the normal flow, the problem of computing the direction of translation and the time to collision is solved in Chapter 4. The basic idea of the motion parameter estimation strategy lies in the employment of fixation and tracking. Fixation simplifies much of the computation by placing the object at the center of the visual field, and the main advantage of tracking is the accumulation of information over time. It is shown how tracking is accomplished using normal flow measurements; then tracking is used for two different tasks in the solution process. First,

it serves as a tool to compensate for the lack of existence of an optical flow field and thus to estimate the translation parallel to the image plane. Second, it is utilized to gather information about the motion component perpendicular to the image plane. While the object is moving and its distance to the image plane changes, the rotation of the camera tracking the object also has to be adjusted. From this change in the tracking motion, information about the change in depth can be derived. By combining the outputs of the two computations the direction of translation is then obtained.

Chapter 5 is devoted to the estimation of egomotion for an observer moving rigidly in a static environment. Usually the term "passive navigation" is used to describe the set of processes by which a system can estimate its motion with respect to the environment. Passive navigation is a prerequisite for any other navigational ability. A system can be guided only if there is a way for it to acquire information about its motion and to control its parameters.

New constraints of global nature relating 2D image measurements to 3D motion parameters are introduced. The approach is based on an analysis of the properties of the normal flow field. The fact that motion is rigid defines geometric relations between certain values of the normal flow field. Normal flow vectors are classified according to their direction; two different types of classification are introduced. Considering only the signs of the values in each class, patterns are found in the image plane. These patterns are regions of positive and negative signs, which are separated by conic sections and straight lines. The position of the patterns is related to the motion parameters; actually each pattern depends only on a subset of the parameters.

These theoretical findings are then utilized to develop techniques for the estimation of motion. For the most general case (a passive observer moving with three translational and three rotational parameters) it is shown how the patterns can be searched for in order to find the parameters describing the motion. The algorithmic procedure consists of three computational steps. The strategy lies in checking constraints imposed by the 3D motion parameters on the normal flow field in order to gradually reduce the space of possible solutions. In the first step candidate solutions for the axis of translation and the direction of rotation are computed by fitting a small number of patterns to the normal flow values. These patterns are of lower dimensionality than the others. In a second step the third rotational component is computed from the normal flow vectors that are only due to rotation. Finally, by looking at the complete data set, all solutions that cannot give rise to the given normal flow field are discarded from the solution space.

It should again be emphasized that the patterns are defined only by the sign of the

normal flow. It has been shown in the past that recovering 3D motion from noisy flow fields by analytic methods is a problem of extreme sensitivity, with researchers reporting very large errors in the motion parameter estimates under small perturbations in the input. The geometric constraints, on the other hand, allow us to estimate motion very robustly, since the patterns will not be affected by perturbations smaller than 100 percent in the normal flow values.

Chapter 5 concludes with a discussion of how the general results can be used in motion estimation algorithms for an observer with restricted motion capabilities. For the two most common restrictions, the cases of an observer that cannot rotate around the Z-axis and an observer with only translational abilities, algorithms are presented.

In Chapter 6 it is shown that if the observer is active and supplies additional information, the general constraints can be exploited to solve various navigational tasks with only a small computational effort. In particular, it is explained how the egomotion estimation problem becomes easier if tracking is employed. In this case the problem of finding the axis of rotation and the direction of translation reduces to a one-dimensional search problem. Furthermore, a qualitative strategy for bringing the focus of expansion to the center of the image is outlined. The idea is to demonstrate that an active observer can alter its motion parameters to predefined values without going through the intermediate stage of computing its current motion.

Since all the techniques described in the following chapters are based on the use of normal flow as an image representation, a description of this input is given in the following section. Then the basic equations employed are introduced and the choice of the coordinate system used is discussed. Since models should be directly related to the task being performed, I argue that the coordinate system should also be chosen differently for the tasks of egomotion and object motion estimation.

3.2 Input

Due to the aperture problem the only image motion measurement that can be uniquely defined from a sequence of images is the normal flow, the component of the flow perpendicular to edges (see Fig. 2.1). Normal flow can be computed in various ways. As discussed in Chapter 2, it is permissible to assume the conservation of local image information to derive a local description of image motion. The difficulty in the computation of normal flow is only due to the discrete aspect of digital images. Computing normal flow in images is as difficult as detecting edges.

Here, the image intensity is assumed to remain constant, and the normal flow is computed from the spatiotemporal derivatives of the image intensity function by employing the motion constraint equation. Usually it is assumed that the normal flow coincides with the "normal motion field", the normal component of the projection of the 3D motion on the image plane. This fact is expressed in the assumption $\frac{dI}{dt} = 0$, which says that the two fields are the same. However, the two fields are not equal in general. Their difference is investigated below.

Let $I(x, y, t)$ denote the image intensity, and consider the optical flow field $(u, v) = \vec{v}$ and the motion field $\vec{\bar{v}} = (\bar{u}, \bar{v})$ at a point (x, y), where the local (normalized) intensity gradient is $\vec{n} = (I_x, I_y)/\sqrt{I_x^2 + I_y^2}$. The normal motion field at point (x, y) is by definition

$$\begin{aligned} \bar{u}_n &= \vec{\bar{v}} \cdot \vec{n} && or \\ \bar{u}_n &= \left(\tfrac{dx}{dt}, \tfrac{dy}{dt}\right) \cdot \tfrac{(I_x, I_y)}{\sqrt{I_x^2 + I_y^2}} && or \\ \bar{u}_n &= \left(\tfrac{dx}{dt}, \tfrac{dy}{dt}\right) \cdot \tfrac{\nabla I}{||\nabla I||} && or \\ \bar{u}_n &= \tfrac{1}{||\nabla I||}\left(I_x \tfrac{dx}{dt} + I_y \tfrac{dy}{dt}\right) \end{aligned}$$

Similarly, the normal flow [Horn and Schunck, 1981] is

$$u_n = -\frac{1}{|| \bigtriangledown I||} I_t$$

Thus, when approximating the differential $\frac{dI}{dt}$ by its total derivative we get

$$\bar{u}_n - u_n = \frac{1}{|| \bigtriangledown I||} \frac{dI}{dt}$$

This shows that the two fields are close to equal when the local image intensity gradient $\bigtriangledown I$ is high. Thus, if we measure normal flow only in regions where the intensity gradients are of high magnitude, we guarantee that the normal flow measurements can be used for inferring 3D motion. It should be noted that this result can be inferred from the analysis of Verri and Poggio [1989] and Singh [1990], who proved that the two flow fields are exactly the same only in the case when a uniformly illuminated Lambertian surface undergoes pure translation.

Concerning implementation of the algorithms for finding normal flow, in the experiments conducted in this thesis the images were first convolved with either a box filter of kernel size 3 to 5 or a Gaussian of the same kernel size and standard deviation on the order of $\sigma \in [1.3, 1.7]$. The normal flow was computed by using 3×3 Sobel operators to estimate the spatial derivatives in the x and y directions and by subtracting the 3×3 box-filtered values of consecutive images to estimate the temporal derivatives. Because

the directions of normal flow vectors at corner points cannot be accurately estimated, a preprocessing step was utilized, where through a multi-resolution technique [Fermüller and Kropatsch, 1992] edge points of high curvature were detected, and the normal flow was not estimated at these points.

3.3 Image formation and the choice of coordinate system

Consider the monocular imaging situation where the observer and the scene are in motion relative to each other. In order to obtain the equations relating the 3D scene to the image measurements in a general form, two coordinate systems are employed. The reference coordinate system (X, Y, Z) is fixed to the observer with the center O being the nodal point of the camera. Another coordinate frame, which we will call the "scene frame", is fixed at a point $S = (X_s, Y_s, Z_s)$ on an object in the scene. At the time of observation the reference frame and scene frame axes are parallel. The rigid motion of any point $P = (X, Y, Z)$ in the scene can then be described through a translation $T_s = (U_s, V_s, W_s)$ of the scene frame with respect to the reference frame and a rotation $\omega = (\alpha, \beta, \gamma)$ with respect to the scene frame, which leads to the following equations [Bandopadhay and Ballard, 1991]:

$$\begin{aligned} \dot{X} &= -U_s - \beta(Z - Z_s) + \gamma(Y - Y_s) \\ \dot{Y} &= -V_s - \gamma(X - X_s) + \alpha(Z - Z_s) \\ \dot{Z} &= -W_s - \alpha(Y - Y_s) + \beta(X - X_s) \end{aligned} \tag{3.1}$$

As image formation model we use perspective projection on the plane. The image plane is parallel to the XY plane and the viewing direction is along the positive Z axis (see Figure 3.1). Under this projection the image position $p(x, y)$ of a 3D point $P(X, Y, Z)$ is defined by the relation

$$(x, y) = (\frac{fX}{Z}, \frac{fY}{Z}) \tag{3.2}$$

The constant f denotes the focal length of the imaging system. The equations relating the velocity (u, v) of an image point p to the 3D velocity can be derived by differentiating (3.1) and substituting from (3.2):

$$u = \frac{(-U_s f + xW_s)}{Z} + \alpha(\frac{xy}{f} - \frac{xX_s}{Z}) - \beta(\frac{x^2}{f} + f - \frac{xX_s}{Z} - \frac{Z_s f}{Z}) + \gamma(y - \frac{Y_s f}{Z})$$

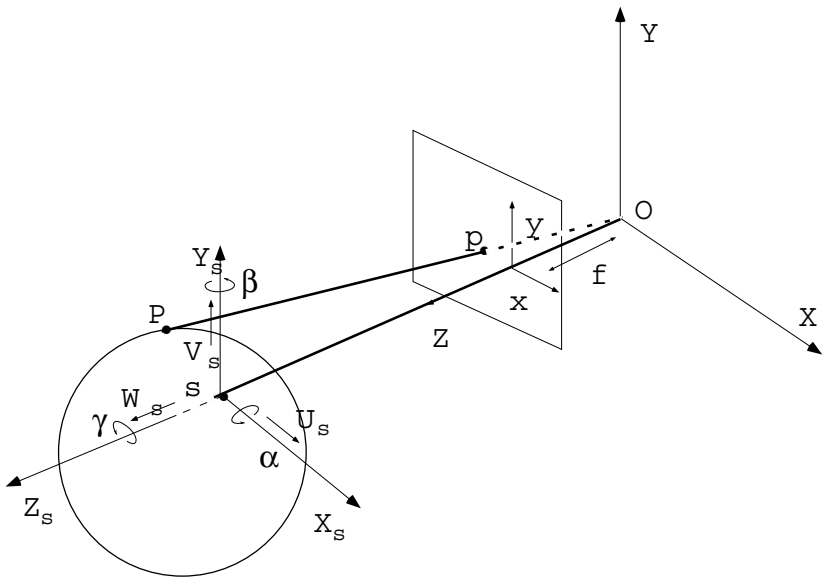


Figure 3.1: Imaging geometry and motion representation.

$$v = \frac{(-V_s f + yW_s)}{Z} + \alpha(\frac{y^2}{f} + f - \frac{yY_s}{Z} - \frac{Z_s f}{Z}) - \beta(\frac{xy}{f} - \frac{yX_s}{Z}) - \gamma(x - \frac{X_s f}{Z}) \tag{3.3}$$

The above equation is a more general form of the optical-flow based constraint equation first derived by Longuet-Higgins and Prazdny [Longuet-Higgins and Prazdny, 1980]. In fact, if the origin of the scene frame coincides with the origin of the reference frame, then $X_s = Y_s = Z_s = 0$ and $T = T_0 = (U, V, W)$ and (3.3) becomes

$$\begin{aligned} u &= \frac{(-Uf + xW)}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y \\ v &= \frac{(-Vf + yW)}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x \end{aligned} \tag{3.4}$$

Since the motion parameters are expressed relative to the scene coordinate system, prediction of the position of the moving entity (object or observer) at the next time instant is dependent on the choice of this coordinate system's position. In the case of egomotion it makes sense to attach the scene coordinate system to the observer, simply because the quantities recovered are directly related to the way the observer moves. On the other hand, when the observer needs to make inferences regarding another object's motion, the ideal place to put the origin of the scene coordinate system would be the mass center of the object (the natural system).

Since the mass center is not known, different choices have to be made. Most commonly the camera's nodal point is chosen as the center of the scene coordinate system (a "camera-centered" coordinate system). Rotation is described around the nodal point. In the case of object motion this leads to different values for the motion parameters for each new frame, which is an unwelcome effect in the task of finding translational motion.

We therefore decided to attach the center of rotation to the object's point of intersection with the optical axis (an "object-centered" coordinate system). The active observer is free in its choice of the center and will therefore decide on a point belonging to a neighborhood of non-uniform brightness with distinguishable features.

This approach can be justified by the following argument: When choosing as fixation point the mass center of the object's image or a point in its neighborhood, the resulting motion parameters are in many cases close to those of the natural system. In the natural coordinate system with center O_{natural} the velocity v at point P is due to the translational and the rotational components

$$v = t_{\text{natural}} + \omega \times \overrightarrow{O_{\text{natural}}P}$$

and in the object-centered coordinate system with center O_{object} the same velocity is expressed as

$$v = t_{\text{object}} + \omega \times \overrightarrow{O_{\text{object}}P}$$

Therefore the difference in translation between t_{natural} and t_{object} (see Figure 3.2) is given by

$$\begin{aligned} t_{\text{natural}} - t_{\text{object}} &= \omega \times (\overrightarrow{O\text{object}P} - \overrightarrow{O\text{natural}P}) \\ &= \omega \times \overrightarrow{O_{\text{object}}O_{\text{natural}}} \end{aligned}$$

This value becomes smaller as $\overrightarrow{O_{\text{object}}O_{\text{natural}}}$ decreases.

In order to stress the different analyses for different coordinate systems, throughout the thesis rotation in an object-centered coordinate system is denoted by $(\omega_1, \omega_2, \omega_3)$ and rotation in a camera-centered coordinate system by (α, β, γ).

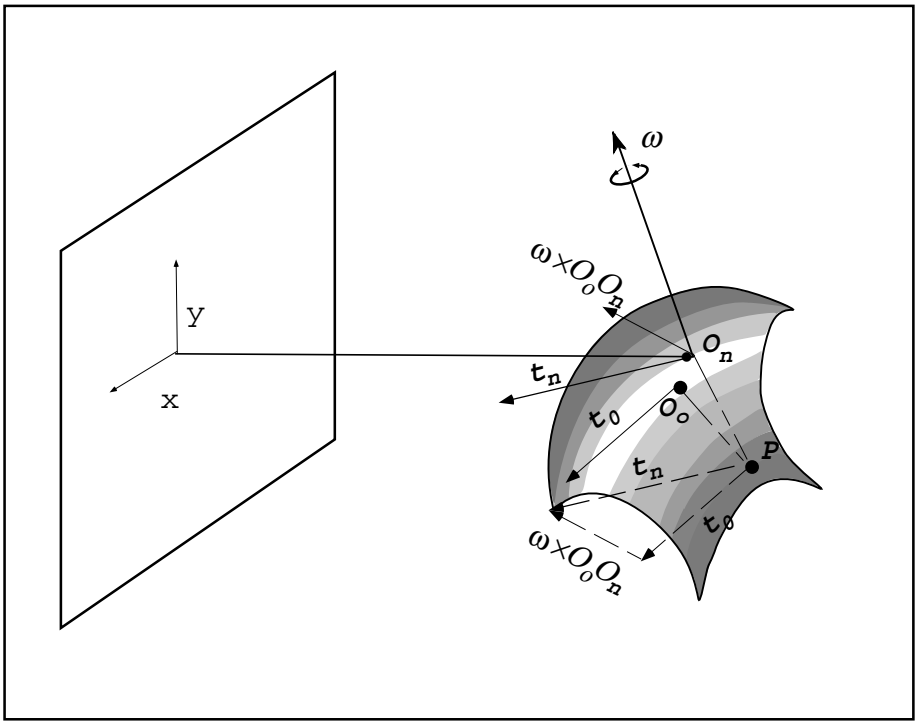


Figure 3.2: The difference in translation between t_n in the natural system with center O_n and t_o in the object centered system with center O_o is $\omega \times \overrightarrow{O_o O_n}$.

Chapter 4

Active 3D Motion Estimation

A method of estimating the direction of translation and the time to collision for a monocular observer that has the capability of tracking is presented. The observer derives its required tracking movement from the image sequence and uses these tracking parameters as input to the computation of the object's 3D motion.

To begin with, the observer detects independent motion [Sharma and Aloimonos, 1991] and fixates on the object, thus causing the optical axis to pass through the object. The translational direction of an object moving with translational parameters (U, V, W) and rotating with velocity $(\omega_1, \omega_2, \omega_3)$ is represented in the image plane by the point $(\frac{U}{W}, \frac{V}{W})$, the Focus of Expansion (FOE). To give a graphical explanation: If we put the object at a distance equal to the focal length f in front of the nodal point of the camera, the FOE represents the intersection point of the image plane and the motion trajectory which passes through the nodal point (see Figure 4.1).

It has been argued that tracking is used in biological vision for the sake of simplifying the estimation of motion. Since our goal is to study computer vision for an active observer, our first question should concern the nature of the activities themselves. Therefore we should ask why one should proceed in a roundabout way and derive the tracking movement as an intermediate step. What do we gain from tracking?

Through tracking we can accumulate information over time and therefore add the parameter of time as additional component to the input information. Another advantage of tracking is that since it is accomplished over a number of steps, the tracking parameters can be corrected sequentially (smoothed) and we need not rely on just one measurement.

The idea of using the tracking parameters for motion estimation was used previously by Bandopadhay and Ballard [1991]. They provide closed form solutions for the compu-

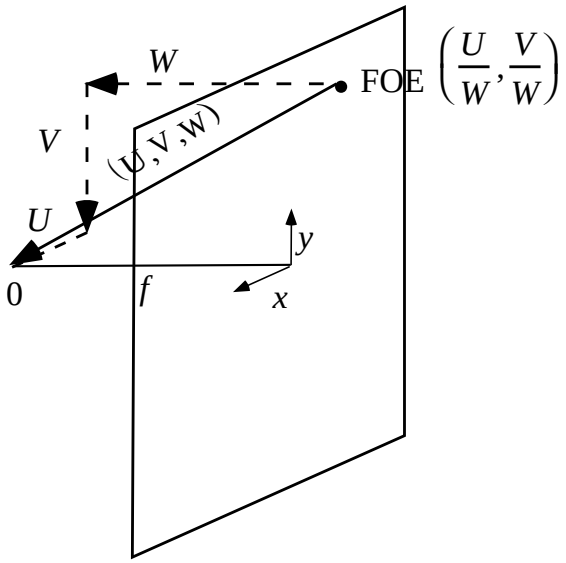


Figure 4.1: FOE in the image plane.

tation of the egomotion parameters for a binocular observer by employing the rotation angle and its first and second derivatives (angular velocity and acceleration). In their paper they did not show how tracking was actually done, whereas here a complete solution is proposed: First it is shown how to compute the tracking parameters using normal flow and then how to use them for 3D motion estimation.

The computation of the FOE and the time to collision is accomplished through three modules [Fermüller and Aloimonos, 1992] that involve the activities of fixation and tracking:

1. By fixating on an object point, which is considered to be the origin of the coordinate system, the image velocity at the center is obtained, which represents the projection of parallel translation. It is shown how tracking can be used to derive the projection of parallel translation from just the spatio-temporal derivatives.

2. In the next step, the output of the first module is used to acquire information about translation parallel to the optical axis. Again tracking is used, here as a tool for accumulating depth information over time.

3. In the third module it is shown how to estimate the time to collision (which is related to the FOE) from the spatio-temporal information at the fixated point.

4.1 Tracking gives parallel translation

The first activity used in this approach is fixation. This action provides us with linear relations between the 3D and the 2D velocity parameters. An object at distance Z in front of the camera moves in the 3D environment with translational velocity (U, V, W) and rotational velocity $(\omega_1, \omega_2, \omega_3)$. In an object-centered coordinate system with center $P(X_0, Y_0, Z_0)$ under perspective projection the optical flow (u, v) is related to these parameters through the following equations:

$$\begin{array}{rcl} \frac{dx}{dt} = u & = & \frac{Uf}{Z} - \frac{Wx}{Z} - \frac{xy\omega_1}{f} + \omega_2 \left(\frac{x^2}{f} + \frac{f(Z-Z_0)}{Z} \right) - \omega_3 y \\ \frac{dy}{dt} = v & = & \frac{Vf}{Z} - \frac{Wy}{Z} - \omega_1 \left(\frac{y^2}{f} + \frac{f(Z-Z_0)}{Z} \right) + \frac{\omega_2 xy}{f} + \omega_3 x \end{array}$$

In a small area around the center x, y and $\frac{(Z-Z_0)}{Z}$ are close to zero. The optical flow components due to rotation and due to translation parallel to the optical axis converge to zero; u becomes $\frac{Uf}{Z}$ and v becomes $\frac{Vf}{Z}$.

The flow at the center of the image gives the projection of parallel translation, but only normal flow is available. It is shown that tracking can be used for the evaluation of optical flow by an iterative technique and the convergence of the method to the exact solution is proven.

The problem of current optical flow algorithms is that additional constraints are employed. Constraints that impose a relationship on the values of the flow field are usually used, and this amounts to assumptions, such as smoothness, about the scene in view. This basic problem is overcome by providing the observer with activity. The computation is thus transferred to the active observer, which has the ability to iteratively adjust its motion to the given situation through its control mechanism.

In cases where the dominant motion of the object is translation towards the observer, the resulting optical flow vectors emanate from a point which lies inside the object's image. The coordinates of this point, the FOE, are consequently close to zero. Otherwise the optical flow pattern is due to vectors that are approximately parallel and have about the same magnitude. Typical normal flow patterns for both cases are shown in Figure 4.2.

For these cases, where the FOE lies inside the object, the normal flow vectors are mainly due to translation, because the rotational components near the object center are very small. Therefore a simple technique using only the direction of the normal flow measurements can be applied. Given the normal flow vector at a point, we know that the FOE lies in the half-plane which is on the opposite side of the gray-level edge from the normal flow vector. Considering every available normal flow measurement will narrow the

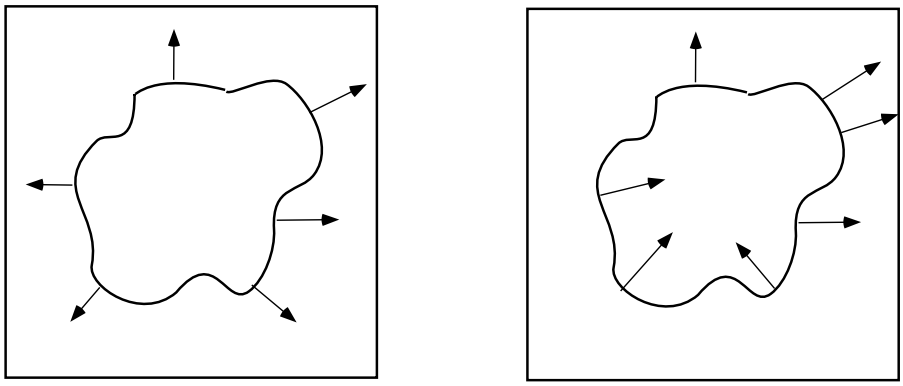


Figure 4.2: (a) Normal flow vectors emanating from a point inside the object. (b) Normal flow vectors when the translational component parallel to the image plane is not much larger than the component perpendicular to the plane.

possible location of the FOE to a small area (see Figure 4.3) (see also [Aloimonos, 1990; Horn and Weldon, 1987]). When dealing with such normal flow patterns, it would make no sense to use the method introduced in this chapter; we are concerned here with the more complicated case illustrated in Figure 4.2b.

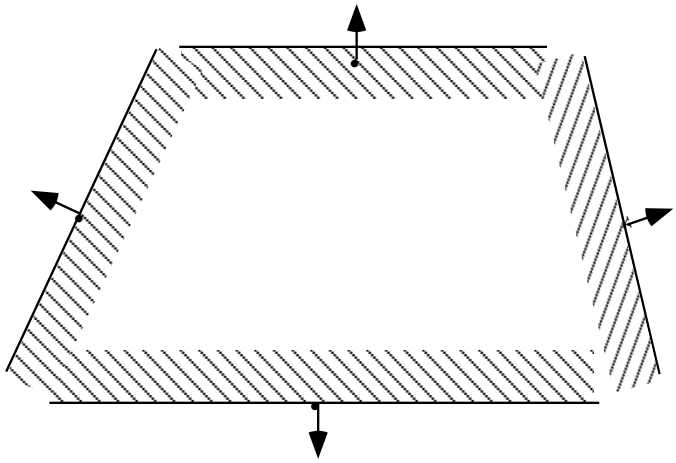


Figure 4.3: Each available normal flow measurement constrains the possible location of the FOE.

Let us compute the normal flow in a set of directions in a small area around the origin (fixation point). The normal flow is the projection of the optical flow on the gradient direction. The largest of the normal flow values in the different directions is therefore the one closest to the optical flow. Let us call this normal flow vector the "maximum normal flow" and denote it by (u^n, v^n) (see Figure 4.4a). We take it as an approximation to the correct optical flow and use it to track the fixated point. The purpose of tracking is to correct for the error in the approximation. In order to keep a point with optical flow (u, v)

in the center of the image the observer has to perform a movement that produces the same value of optical flow in the opposite direction. The way our observer accomplishes this task is by rotating the camera around the nodal point about the X- and Y-axes. While the observer is moving it takes the next image and again computes the normal flow vectors. If the maximum normal flow was equal to the optical flow, a new optical flow (due to object motion and egomotion) of zero will be achieved.

Usually, however, the maximum normal flow and the optical flow are not equal; they differ in magnitude or in direction, or both. An error in magnitude results in a flow vector in the direction of maximum normal flow, and an error in direction creates a flow vector perpendicular to it (see Figure 4.4b). The actual error is usually in both magnitude and direction. Thus the new flow vector is a vector sum of the two components. Again it can be approximated by the largest normal flow vector measurement. The new measured normal flow is used as a feedback value to correct the optical flow and the tracking parameters; the new normal flow vector is added to the maximum normal flow vector computed in the first step. Proceeding by applying the same technique to the successive estimated errors will result in an accurate estimate of the actual flow after a few iterations. The proof of convergence to the exact solution follows.

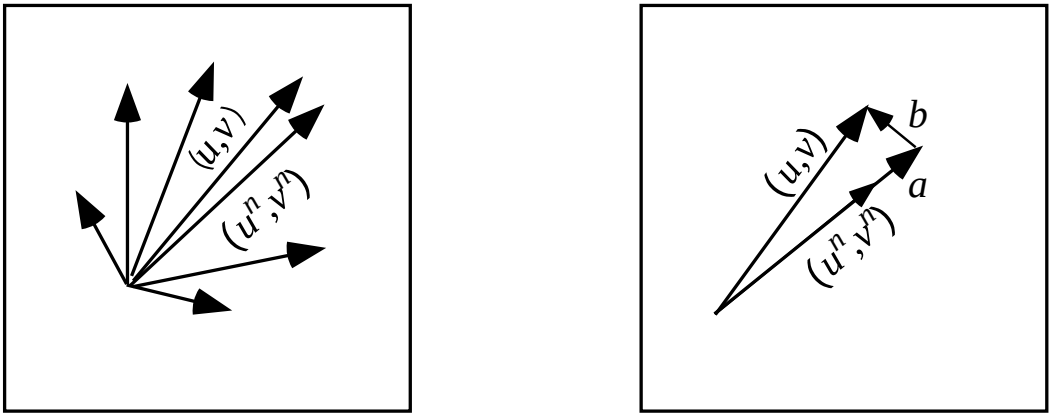


Figure 4.4: (a) Normal flow vectors measured in different directions. (b) The new flow vector (resulting from object motion and tracking) is due to 1) the error in magnitude, and 2) the error in direction.

We use here a simplified model to explain tracking. The change of the local coordinate system during tracking and the fact that the object is coming closer are not considered. Since for the purpose of optical flow estimation the number of tracking steps is small, the error originating from this model is not serious. In a specific application, the algorithm will stop when the computed error is smaller than a given threshold, which will cover model errors.

At each iteration step we are computing an approximation to the difference between the observer's egomotion and the object motion. Considering the possible sources of error we have to show that the approximation error will become zero.

Deviations of the chosen maximum normal flow from the optical flow value are due to the following causes:

- *Deviations covered through the model*:
  The fact that normal flow measurements are computed in a finite number of directions causes an error in direction of up to half the size of the angular separation between normal flow measurements. If measurements in n directions are performed, the maximum error p in direction is bounded by $p < \frac{\pi}{2n}$.

- *Deviations coming from simplifications and discrete computations*:
  In the evaluation of flow measurements the parts that are linear and quadratic in x, y, and $Z - Z_0$ are ignored. Furthermore, each measurement in one direction is computed as the average of the normal flow values in a range of directions. These factors may cause errors in magnitude as well as direction, and a vector different from the closest normal flow vector may be chosen.

- *General errors occuring in normal flow computation*:
  Sensor noise in normal flow measurements and the numerical computation of the derivatives of the image intensity function can influence the magnitude and the direction of the estimated value.

Let v be the magnitude of the actual optical flow. The error sources lead to specifying the error in magnitude, q, as a percentage of the actual value. q_i is the magnitude error in the maximum normal flow measurement at step i and p_i is the the angle between the maximum normal flow vector and the optical flow vector, where $q_i < q$ and $p_i < p$. Therefore the difference between the optical flow and the first measurement of maximum normal flow is given by $\mathit{diff}_1 = \begin{pmatrix} vq_1 \cos p_1 \\ v \sin p_1 \end{pmatrix}$, where the x-axis is aligned with the maximum normal flow vector (see Figure 4.5). The square of its magnitude is computed as

$$||\mathit{diff}_1||^2 = v^2 q_1^2 \cos^2 p_1 + v^2 \sin^2 p_1$$

The second normal flow vector, if measured from the direction of the maximum normal flow vector derived at the second step, is given by $\mathit{diff}_2 = \begin{pmatrix} ||\mathit{diff}_1|| q_2 \cos p_2 \\ ||\mathit{diff}_1|| \sin p_2 \end{pmatrix}$, and the

square of its magnitude is therefore

$$||diff_2||^2 = q_1^2 q_2^2 v^2 \cos^2 p_1 \cos^2 p_2 + q_1^2 v^2 \cos^2 p_1 \sin^2 p_2 + v^2 \sin^2 p_1 \sin^2 p_2 + q_2^2 v^2 \sin^2 p_1 \cos^2 p_2$$

In general, if we denote by $\{a,\ b\}$ the fact that either a or b has to be chosen, then $||diff_n||^2$ can be expressed as

$$||diff_n||^2 = v^2 \sum_{\text{all permutations}} \prod_{i=1}^{n} \{q_i^2 \cos p_i^2, \sin p_i^2\}$$

Since $q_i < 1$ and $\sin p_i < 1$ it follows that $\prod_i \{q_i^2 \cos p_i^2, \sin p_i^2\}$, and thus the whole term converges to zero. Therefore, we have shown the convergence of the approximation value to the actual optical flow value for the "simplified tracking model".

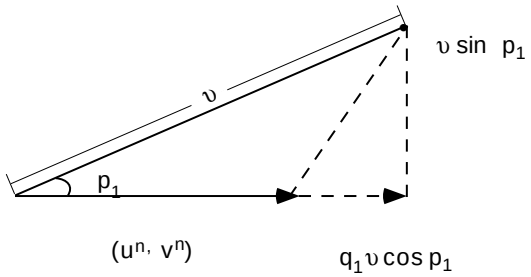


Figure 4.5: Difference between optical flow vector and maximum normal flow vector.

4.2 Estimating the FOE using tracking

When tracking continues over time, as an object comes closer and the value of Z becomes smaller, the optical flow value increases. In order to track correctly and adjust to the increasing magnitude of the optical flow value, the tracking parameters have to be changed. From the change of the tracking parameters the change in Z can be derived. If tracking is accomplished by rotation with a certain angular velocity, this means that the change in depth is derived from the angular acceleration. In the sequel we show the relation between image motion and tracking movement. The exact process of tracking is explained for a geometric setting consisting of a camera that is allowed to rotate around two fixed axes: X- and Y-. These axes coincide with the local coordinate system in the image plane at the beginning of the tracking process.

We describe rotation by an angle ϕ around an axis which is given by its directional cosines $n_1,\ n_2,\ n_3$, where $n_1^2 + n_2^2 + n_3^2 = 1$. The transformation of a point P with coordinates (X, Y, Z) before motion and (X', Y', Z') after motion is described, as shown

in equation (2.3), through the linear relation

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

where the transformation matrix R is of the following form:

$$\begin{pmatrix} n_1^2 + (1 - n_1^2)\cos\phi & n_1 n_2(1 - \cos\phi) - n_3 \sin\phi & n_1 n_3(1 - \cos\phi) + n_2 \sin\phi \\ n_1 n_2(1 - \cos\phi) + n_3 \sin\phi & n_2^2 + (1 - n_2^2)\cos\phi & n_2 n_3(1 - \cos\phi) - n_1 \sin\phi \\ n_1 n_3(1 - \cos\phi) - n_2 \sin\phi & n_2 n_3(1 - \cos\phi) + n_1 \sin\phi & n_3^2 + (1 - n_3^2)\cos\phi \end{pmatrix}$$

$$\equiv \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

Since the image coordinates (x, y) are related to the 3D coordinates through $x = Xf/Z$ and $y = Yf/Z$, we get the following equations, which can be derived from equation (2.5):

$$x' = \frac{(r_{11}x + r_{12}y + r_{13}f)f}{(r_{31}x + r_{32}y + r_{33}f)}$$

$$y' = \frac{(r_{21}x + r_{22}y + r_{23}f)f}{(r_{31}x + r_{32}y + r_{33}f)}$$

In order to compensate for the image motion (u, v) of the point P_o, which moves from (0, 0) to (u, v) in one time unit, the camera has to be rotated by ϕ, n_1, and n_2, where

$$\begin{aligned} u &= n_2 f \tan\phi \\ v &= -n_1 f \tan\phi \end{aligned}$$

Taking the flow measurements (u, v) at the center of the image at the beginning of the tracking process (time t_1), and assuming that the object doesn't change its distance Z_1 to the camera, we can conclude that during time interval Δt an image flow $(u\Delta t, v\Delta t)$ should be measured. The tracking motion necessary for compensation is given by

$$\frac{Uf}{Z_1} = n_2 \tan\phi.$$

But at time t_2 the object has moved to distance Z_2 and we measure a rotation

$$\frac{Uf}{Z_2} = n_2' \tan\phi'$$

Figure 4.6 shows the relationship between the 3D motion and the tracking parameters. Since $Z_2 - Z_1 = W\Delta t$, the change in the reciprocal of the rotation angle is proportional to $\frac{W}{U}$, because

$$\frac{1}{n_2 \tan\phi} - \frac{1}{n_2' \tan\phi'} = \frac{Z_2 - Z_1}{U\Delta t} = \frac{W\Delta t}{U\Delta t}$$

and the FOE $(\frac{U}{W}, \frac{V}{W})$ can be computed as

$$\frac{U}{W} = 1/(\frac{1}{n_2' \tan\phi'} - \frac{1}{n_2 \tan\phi}) = 1/(\frac{1}{n_2' \tan\phi'} - \frac{f}{u\Delta t})$$

and

$$\frac{V}{W} = 1/(\frac{1}{-n_1' \tan\phi'} - \frac{f}{v\Delta t}).$$

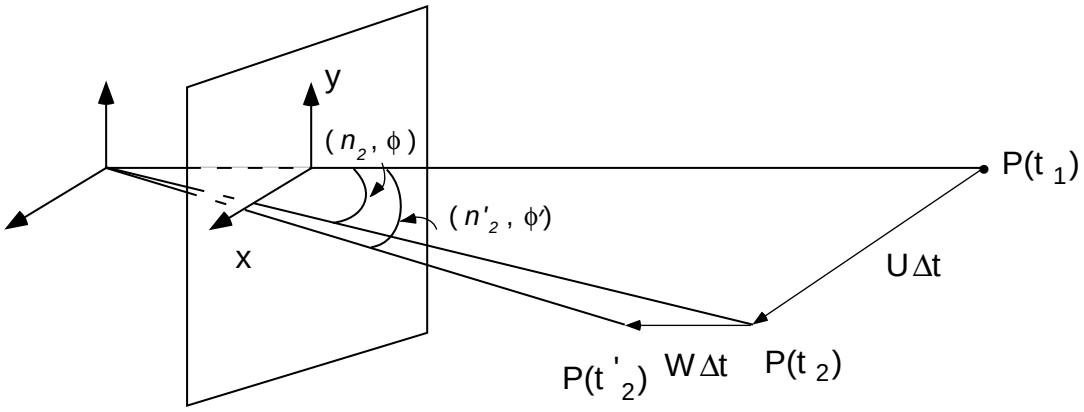


Figure 4.6: From the optical flow value, which is due only to translation parallel to the image plane, a translation of P from $P(t_1)$ to $P(t_2)$ is inferred, and therefore the tracking parameters (n_2, ϕ) are expected. But actually the point has moved to $P(t_2')$ and a rotation described by (n_2', ϕ') is measured.

It remains to be explained how tracking is actually performed, since we are facing the problem of a constantly changing local coordinate system. The next section is devoted to the computation of the tracking parameters.

4.3 Computation of tracking parameters

Unlike Section 4.1, where a "simplified model" was used, here in order to compute the tracking parameters the change of the local coordinate system is taken into account and the necessary parameter transformations between the coordinate systems are shown.

In the first module the projection of parallel translation at the beginning of the tracking process has been computed as described in Section 4.1. From these measurements

the rotational parameters ϕ_1, $n_{1,1}$ and $n_{2,1}$ necessary to track for one time interval were derived. When continuing with tracking, we have to consider the fact that through the rotation of the image plane the local coordinate system attached to it changes. At each tracking step, in the current local coordinate system an optical flow appears that is due to the change in the Z-distance. The rotation necessary to compensate for this value has to be computed and is added to the old rotation. The summation of rotational vectors is justified, since we are adding a very small vector.

The computation of the rotation vector from normal flow is done in the following way: In the new system the normal flow vectors is computed in different directions and the maximum value is taken. This vector spans from $(0,0)$ to (u_n, v_n). In order to compensate for this vector by rotation around the fixed X- and Y-axes, the point $(0,0)$ and the point (u_n, v_n) are transformed back to the old system through the equations

$$x_{\text{old}} = \frac{(r_{11}x_{\text{new}} + r_{12}y_{\text{new}} + r_{13}f)f}{(r_{31}x_{\text{new}} + r_{32}y_{\text{new}} + r_{33}f)} \quad \text{and} \quad y_{\text{old}} = \frac{(r_{21}x_{\text{new}} + r_{22}y_{\text{new}} + r_{23}f)f}{(r_{31}x_{\text{new}} + r_{32}y_{\text{new}} + r_{33}f)} \tag{4.1}$$

The same formula can be applied to compute from the coordinates the necessary rotation to transform one point into the other.

4.4 Estimating the time to collision

If the values of the motion parameters don't change over the tracking time, the value $\frac{Z}{W}$, the time to collision, expresses the time remaining until the object will hit the infinitely large image plane. A relationship between FOE and time to collision is inherent in the scalar product of the optical flow vector $\overrightarrow{(u,v)}$ with the vector in the gradient direction $\overrightarrow{(n_x, n_y)}$:

$$\begin{pmatrix} u \\ v \end{pmatrix} \begin{pmatrix} n_x \\ n_y \end{pmatrix} = ||v^n||$$

For the pixels near the center, for which we ignore the linear and quadratic parts in x, y and $\frac{Z-Z_0}{Z}$ in the relation between optical flow and the 3D parameters, we get the relationship

$$\frac{Uf}{Z}n_x + \frac{Vf}{Z}n_y = ||v^n||$$
$$\frac{Uf}{W}n_x + \frac{Vf}{W}n_y = ||v^n||\frac{Z}{W}$$

Since we know the FOE, we can compute the time to collision from this relationship by measuring the normal flow value in each of a set of directions and by solving an overdetermined system of linear equations by minimizing the squared error.

4.5 Experimental results

The method was tested on synthetic imagery by using the graphics package Swivel. In this way a simulation of object motion as well as camera rotation was made possible. In order to analyze the robustness of the method, the accuracy of the normal flow values at the centers of the images was evaluated. At every point v_{act}, the projection of the known optical flow value on the gradient direction computed there was determined. The error (err) in the normal flow values was defined as the standardized difference between v_{act} and the normal flow value, v_{meas} ($err = (v_{\text{act}} - v_{\text{meas}})/v_{\text{act}}\%$). In this way an average error of 76.14% and a standard deviation of 179.64% for the motion sequence at the beginning of the tracking process was computed. This constitutes a large error and is comparable to errors appearing in noisy real imagery.

The object displayed in Figure 4.7 moves in the direction $U/W = 4$ and $V/W = 2$, with an image motion at its center of $u = 0.004$ and $v = 0.002$ focal units, and the tracking process was accomplished over a sequence of 100 images.

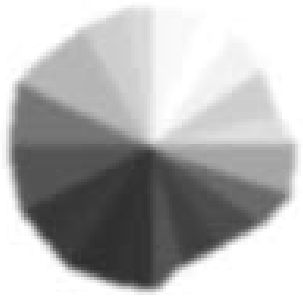


Figure 4.7: First image in the sequence used for tracking.

As regards the implementational details, the normal flow measurements were computed in ten directions in a 9×9 pixel area at the center of the image. When testing the first module, in which parallel translation is estimated, a threshold of 0.0002 focal units was used. The method converged very quickly, usually after two to three iterations. Rotation of increasing magnitude was added to the object motion, and it turned out that the algorithm converged in this situation even for relatively large rotations. (The object was 25 units away from the camera and moved with a translational velocity of $U = 0.1$, $V = 0.05$, $W = 0.025$ units per unit time, and the method converged for rotations of up to 0.3° per time unit around the x-, y- and z-axes.) Some graphical representations are

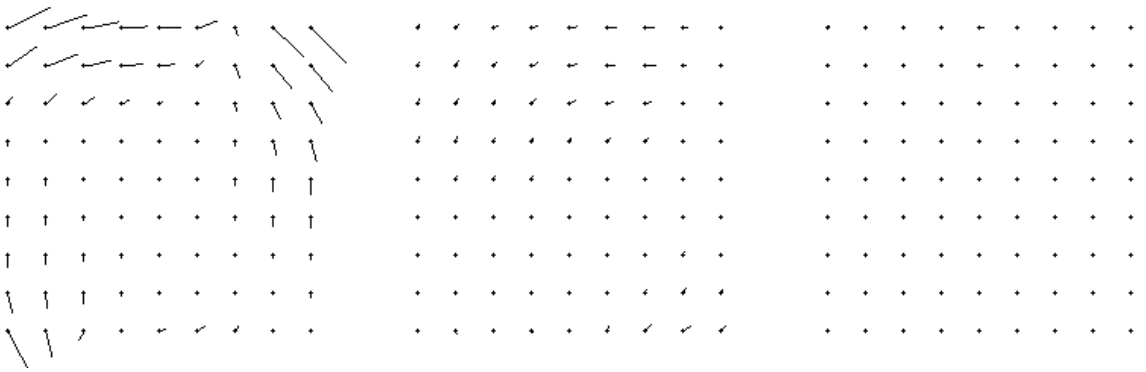


Figure 4.8: Normal flow fields for a tracking sequence.

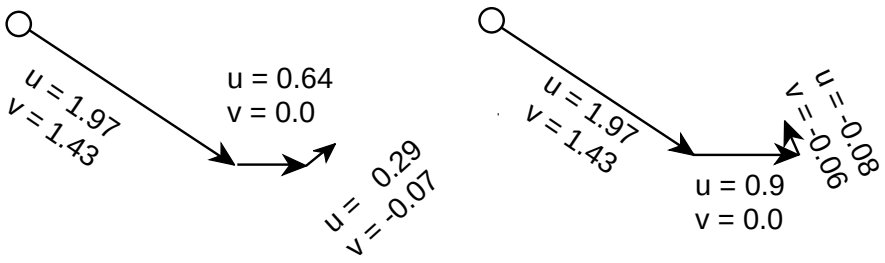


Figure 4.9: Maximum normal flow vectors for (a) no rotation and (b) rotation $\omega_x = 0.1°/\Delta t$, $\omega_z = 0.1°/\Delta t$.

given in the following figures. Figure 4.8 shows, for the case of no rotation, the three normal flow fields that were computed in the the 9×9 pixel area, before convergence was achieved. In Figure 4.9 two maximum normal flow vector sequences are displayed ((a) for no rotation, (b) for rotation $\omega_x = 0.1°$, $\omega_z = 0.1°$).

Using the estimates of parallel translation from this module and continuing the tracking over 100 steps resulted in FOE values having less than 15% error (e.g., for the case of no rotation an FOE of $U/W = 4.21$ and $V/W = 1.79$) was computed. These experiments demonstrate that the described technique for computing object motion can tolerate a large amount of noise in the input (normal flow). In particular, they show that tracking can be successfully accomplished using only normal flow under noisy conditions and that tracking acceleration can be employed for robust parameter estimation.

Chapter 5

Egomotion Estimation

For a monocular observer undergoing unrestricted rigid motion in the 3D world we compute the parameters describing this motion. Using a camera-centered coordinate system, the equations relating the velocity (u, v) of an image point to the 3D velocity and the depth Z of the corresponding scene point are (see also equation (3.4))

$$u = \frac{(-Uf + xW)}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y$$
$$v = \frac{(-Vf + yW)}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x$$

The number of motion parameters a monocular observer is able to compute under perspective projection is limited to five: the three rotational parameters and the direction of translation. We therefore introduce coordinates for the direction of translation, $(x_0, y_0) = (Uf/W, Vf/W)$, and rewrite the righthand sides of the above equations as sums of translational and rotational components:

$$\begin{aligned} u = u_{\text{trans}} + u_{\text{rot}} &= (-x_0 + x)\frac{W}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y \\ v = v_{\text{trans}} + v_{\text{rot}} &= (-y_0 + y)\frac{W}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x \end{aligned}$$

Since we can only compute normal flow, the projection of flow on the unit gradient direction (n_x, n_y), only one constraint can be derived at every point. The value u_n of the normal flow vector along the gradient direction is given by

$$\begin{aligned} u_n &= un_x + vn_y \\ u_n &= ((-x_0 + x)\frac{W}{Z} + \alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y)n_x \\ &\quad +((-y_0 + y)\frac{W}{Z} + \alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x)n_y \end{aligned} \tag{5.1}$$

This above equation should demonstrate the difficulties of motion computation using normal flow. A monocular observer not being able to measure depth is confronted with a motion field of five unknown motion parameters and one scaled depth component (W/Z) at every point. Since there is only one constraint for a single point and since no assumptions about depth should be made, there is no straightforward way to compute the motion parameters analytically.

5.1 Motion field interpretation

A motion field is composed of a translational and a rotational component. Only the first of these is dependent on distance from the observer. Therefore it seems reasonable to look for a way of determining the motion components by disregarding the depth components. The motion under consideration is rigid. Every point in 3D moves relative to the observer along a constrained trajectory. The rigidity constraint also imposes restrictions on the motion field in the image plane and these restrictions are reflected in the normal field as well. This is the motivation for investigating geometrical properties inherent in the normal flow field. The motion estimation problem then amounts to resolving the normal flow field into its rotational and translational components.

If the observer undergoes only translational motion, all points in the 3D scene move along parallel lines. Translational motion viewed under perspective results in a motion field in the image plane, in which every point moves along a line that passes through a vanishing point. This point is the intersection of the image plane with the translational trajectory passing through the nodal point. Its image coordinates are $x = Uf/W$ and $y = Vf/W$; the flow there has value zero. If the sensor is approaching the scene, all the flow vectors emanate from the vanishing point, which is then called the *focus of expansion* (FOE) (Figure 5.1). Otherwise the vectors point toward it, in which case we speak of the *focus of contraction* (FOC). The direction of every vector is determined by the location of the vanishing point; the lengths of the vectors depend on the 3D positions of the points in the scene. The vanishing point also constrains the direction of the normal flow vector at every point; it can only be in the half-plane containing the optical flow vector.

In the case of pure rotational motion, every point in 3D moves along a circle in a plane perpendicular to the axis of rotation. The perspective image of this circular path is the intersection of the image plane with the cone defined by the circle and the rotation axis (see Figure 5.2). Depending on the relation between the aperture angle of the cone for a given image point and the angle that the image plane forms with the rotation axis, different second order curves are obtained for the intersection: ellipses, hyperbolas,

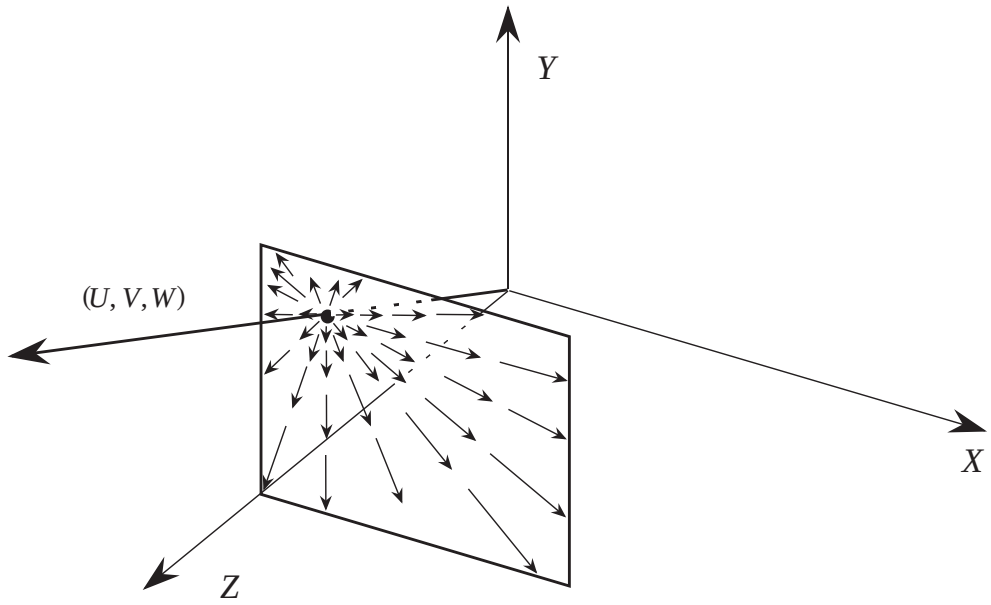


Figure 5.1: Translational motion viewed under perspective projection: the observer is approaching the scene.

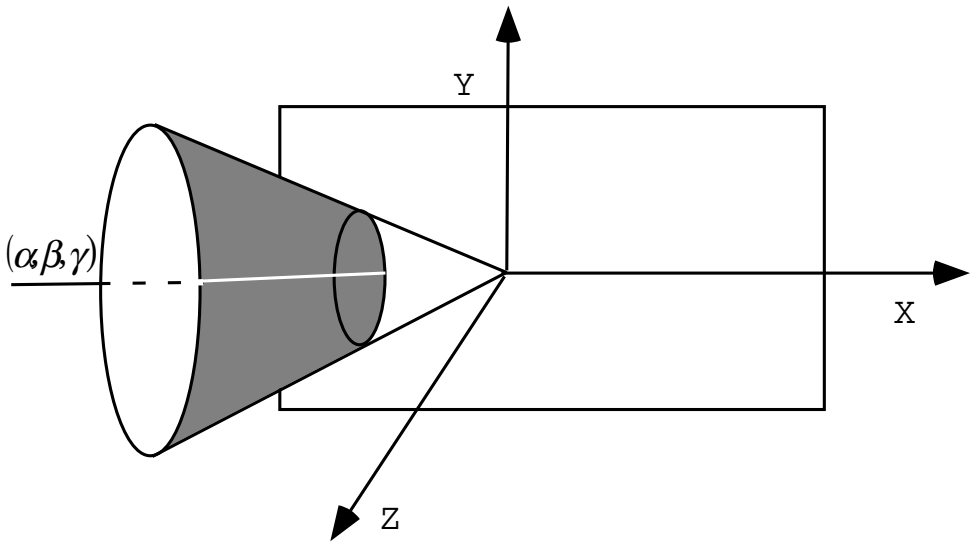


Figure 5.2: The intersection of the image plane with the cone (determined by the circular path in 3D and the rotation axis) defines the projection of rotational motion on the image plane.

parabolas, and even circles when the rotation axis and the optical axis coincide. The specific conic sections due to rotational motion are defined by the axis of rotation. The rotation axis given by the two parameters $(\frac{\alpha}{\gamma})$ and $(\frac{\beta}{\gamma})$ defines a family $M(\frac{\alpha}{\gamma}, \frac{\beta}{\gamma}; x, y)$ of

conic sections:

$$M(\frac{\alpha}{\gamma},\frac{\beta}{\gamma};x,y) = \\ (\frac{\alpha^2}{\gamma^2}x^2 + 2xy\frac{\alpha}{\gamma}\frac{\beta}{\gamma} + y^2\frac{\beta^2}{\gamma^2} + 2xf\frac{\alpha}{\gamma} + 2yf\frac{\beta}{\gamma} + f^2)/(x^2+y^2+f^2) = C \\ \text{with } C \text{ in } [0,\ldots,(1+\frac{\alpha^2}{\gamma^2}+\frac{\beta^2}{\gamma^2})] \tag{5.2}$$

5.2 Properties of selected vectors

A motion vector consists of a rotational component which can be parameterized by three unknowns and a translational vector which is everywhere directed away from (or towards) a point. However, the estimates we can compute at every point are only projections of the motion vector on the gradient direction. In this section geometrical relations of normal flow vectors in selected directions are investigated. To be more precise, we study the sign of the normal flow in certain directions and the locations of normal flow vectors of the same sign. Vectors which are perpendicular to rotational vector field lines and vectors perpendicular to lines emanating from a point are considered. For these vectors we find that the FOE and the axis of rotation separate the normal flow values in the image according to their sign by a second order curve and a straight line [Fermüller, 1993a].

The normal flow vector $\overrightarrow{u_n}$ is the projection of the optical flow vector $\overrightarrow{u}$ on the gradient direction and the value of the normal flow is therefore defined by the scalar product of the optical flow vector and the unit vector in the gradient direction. The flow vector can be decomposed into its translational and rotational components and the right hand side of equation (5.1) can be written as a sum of scalar products:

$$\begin{aligned} u_n &= \frac{W}{Z}((-x_0+x),(-y_0+y))(n_x,n_y) + \\ &\quad ((\alpha\frac{xy}{f} - \beta(\frac{x^2}{f}+f) + \gamma y),(\alpha(\frac{y^2}{f}+f) - \beta\frac{xy}{f} - \gamma x)(n_x,n_y) \end{aligned} \tag{5.3}$$

The goal is to achieve some kind of separation between translation and rotation. Therefore the normal flow vectors are classified according to their direction by defining two kinds of classes which are motivated by the concepts of rotation axis and FOE.

Any possible axis given by an orientation vector (A,B,C), where $A^2+B^2+C^2 = 1$, defines an infinite class of cones with axis (A,B,C) and apex at the origin. The intersection with the image plane gives rise to a set of conic sections, hereafter called vector field lines, or field lines of the axis (A,B,C), or just (A,B,C) field lines. It is

worth noting that the (A, B, C) field lines are the lines along which the image points would move if the observer rotated around axis (A, B, C). Normal flow vectors are combined into a single class if they are perpendicular to the vector field lines of the same axis (A, B, C). At a point (x, y) the orientation perpendicular to the (A, B, C) vector field lines is given by a vector $\overrightarrow{M} = (M_x, M_y)$:

$$\begin{aligned}(M_x, M_y) &= ((-A(y^2 + f^2) + Bxy + Cxf), \\ &\quad (Axy - B(x^2 + f^2) + Cyf))\end{aligned}$$

and its unit vector $\overrightarrow{m} = (m_x, m_y)$ is thus $\overrightarrow{m} = \frac{\overrightarrow{M}}{\|\overrightarrow{M}\|}$. We call the vectors of the class corresponding to the axis (A, B, C) the coaxis vectors (A, B, C). These are the normal flow vectors where the gradient (n_x, n_y) is equal to (m_x, m_y). In order to establish conventions about the vector's orientation, a vector will be said to be of positive orientation if it is pointing in direction (m_x, m_y). Otherwise, if it is pointing in direction $(-m_x, -m_y)$, its orientation will be said to be negative[1] (see Figure 5.3).

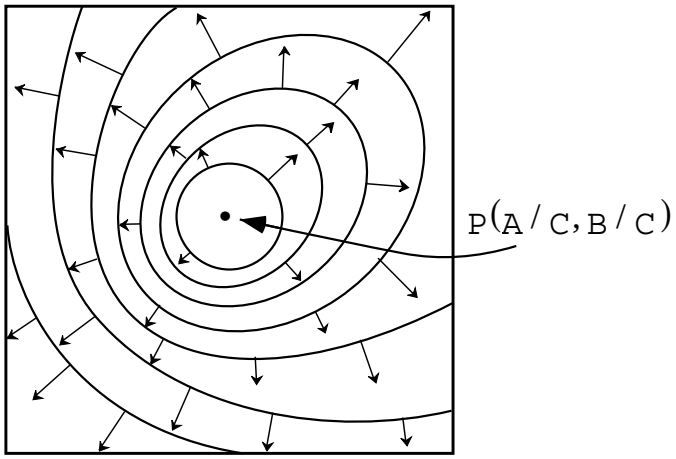


Figure 5.3: Field lines corresponding to an axis (A, B, C) and positive coaxis vectors (A, B, C).

Next we evaluate the translational components of the normal flow vectors in the chosen direction. The value t_n of any translational vector component at point (x, y) in direction (n_x, n_y) is given by

$$t_n = ((x - x_0, y - y_0)\frac{W}{Z})\overrightarrow{n}$$

[1]Obviously, the proposed classification is not based on an equivalence relation, since the intersection of the sets of normal flow vectors belonging to different axes is not empty. However, for our purpose this is not of importance.

Since $\frac{W}{Z}$ is positive in case the observer is approaching the scene, a classification into positive and negative values independent of the distance from the image plane is possible. The translational components of the coaxis vectors (A, B, C) are separated by a second order curve $h(A, B, C, x_0, y_0; x, y)$ given by

$$\begin{aligned} &h(A, B, C, x_0, y_0; x, y) = \\ &\quad x^2(Cf + By_0) + y^2(Cf + Ax_0) - xy(Ay_0 + Bx_0) \\ &\quad -xf(Af + Cx_0) - yf(Bf + Cy_0) + f^2(Ax_0 + By_0) = 0. \end{aligned} \tag{5.4}$$

Where $h(x, y) > 0$, the normal flow values are positive; where $h(x, y) < 0$, they are negative; and where $h(x, y) = 0$, the normal flow values have value zero. For any selected class of coaxis vectors there exists a curve h which is uniquely defined by the two coordinates x_0, y_0 of the FOE (see Figure 5.4a); furthermore it is linear in x_0 and y_0.

The rotational components of the flow vectors are defined only by the three rotational parameters α, β and γ. Along the positive direction of the coaxis vectors the value r_n of the rotational component is

$$r_n = ((\alpha\frac{xy}{f} - \beta(\frac{x^2}{f} + f) + \gamma y), (\alpha(\frac{y^2}{f} + f) - \beta\frac{xy}{f} - \gamma x)(n_x, n_y)$$

The coaxis vectors (A, B, C) and the rotational flow vectors form a right angle for all points on a straight line. Thus considering only the sign of the rotational component along the coaxis vectors (A, B, C) the image plane is separated by a straight line $g(A, B, C, \alpha, \beta, \gamma)$ into two halves containing values of opposite sign, where

$$g(A, B, C, \alpha, \beta, \gamma; x, y) = y(\alpha C - \gamma A) - x(\beta C - \gamma B) + \beta Af - \alpha Bf = 0 \tag{5.5}$$

Again the sign of $g(x, y)$ at a point (x, y) determines the sign of the coaxis vectors (A, B, C). The straight line is defined by only two parameters which characterize the axis of rotation, namely $\frac{\alpha}{\gamma}$ and $\frac{\beta}{\gamma}$ (see Figure 5.4b).

In order to investigate constraints for general motion the geometrical relations due to rotation and due to translation have to be combined. A second order curve separating the plane into positive and negative values and a line separating the plane into two half-planes of opposite sign intersect. This splits the plane into areas of only positive coaxis vectors, areas of only negative vectors, and areas in which the rotational and translational flow have opposite signs. In these last areas, no information is derivable without making depth assumptions (Figure 5.4c).

We thus obtain the following geometrical result for the case of general motion. Any class of coaxis vectors (A,B,C) is separated by a rigid motion into two groups. The

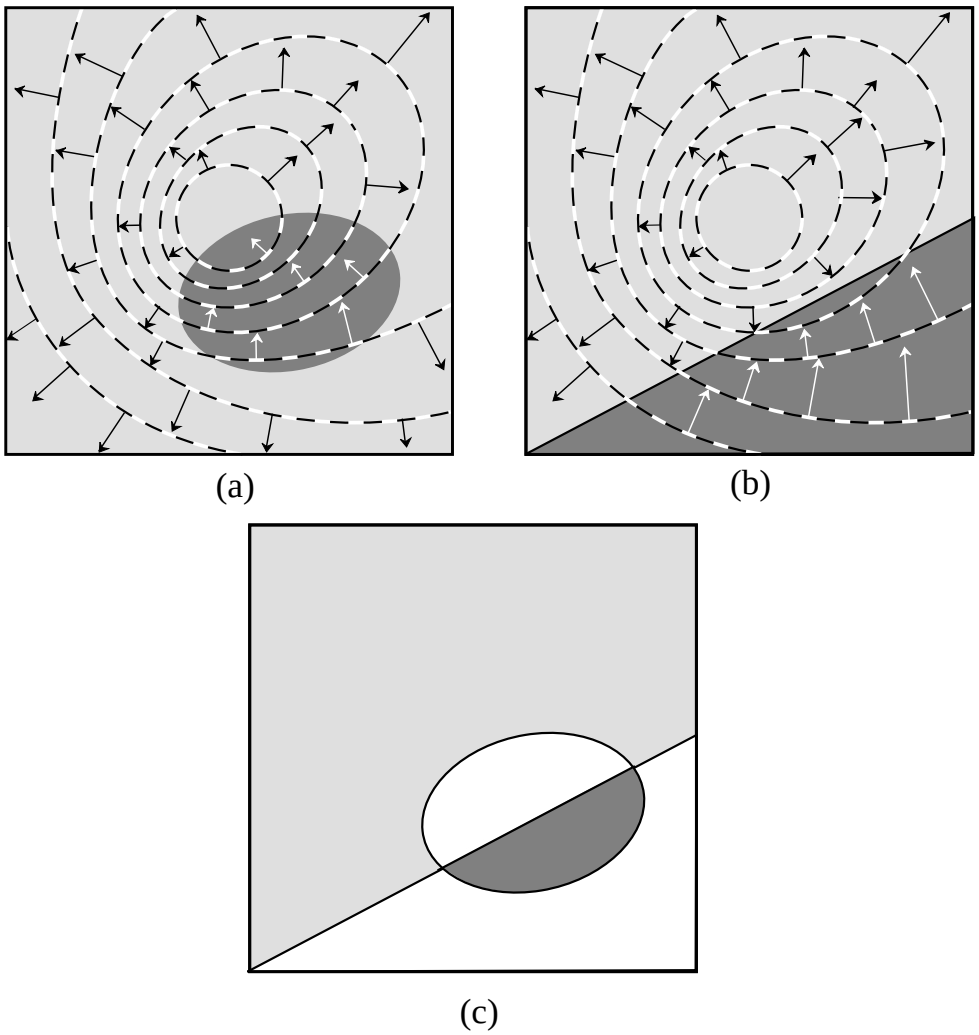


Figure 5.4: (a) The coaxis vectors (A, B, C) due to translation are negative if they lie within a second-order curve defined by the FOE, and are positive at all other locations. (b) The coaxis vectors due to rotation separate the image plane into a half-plane of positive values and a half-plane of negative values. (c) A general rigid motion defines an area of positive coaxis vectors and an area of negative coaxis vectors. The rest of the image plane is not considered.

FOE (x_0, y_0) and the rotation axis $(\frac{\alpha}{\gamma}, \frac{\beta}{\gamma})$ geometrically define two areas in the plane, one containing positive and one containing negative values. We call this structure on the coaxis vectors the coaxis pattern. It depends on the four parameters x_0, y_0, $\frac{\alpha}{\gamma}$ and $\frac{\beta}{\gamma}$.

For the second kind of classification of the normal flow vectors, namely the one defined as "perpendicular to the lines emanating from a defined point" (see Figure 5.5), similar patterns are obtained. In this case, the rotational components are separated by a second order curve into positive and negative values and the translational components are

separated by a straight line. We call the vectors perpendicular to straight lines passing through a point (r, s) the copoint vectors (r, s).[2]

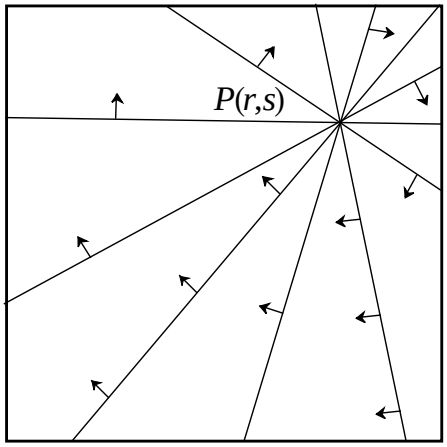


Figure 5.5: Positive copoint vectors (r, s).

At point (x, y) a copoint vector (r, s) of unit length in the positive direction is defined as

$$\frac{(-y+s, x-r)}{\sqrt{(x-r)^2+(y-s)^2}}.$$

The functions which define the curves are given as follows: The straight line $k(r, s, x_0, y_0; x, y)$ separating the translational components is

$$k(r, s, x_0, y_0; x, y) = y(x_0 - r) - x(y_0 - s) - x_0 s + y_0 r = 0 \tag{5.6}$$

and the second-order curve $l(r, s, \alpha, \beta, \gamma; x, y)$ separating the rotational components is (similar to $h(A, B, C, x_0, y_0; x, y)$) defined as

$$\begin{aligned} &l(r, s, \alpha, \beta, \gamma; x, y) = \\ &\quad -x^2(\beta s + \gamma f) - y^2(\alpha r + \gamma f) + xy(\alpha s + \beta r) + xf(\alpha f + \gamma r) \\ &\quad +yf(\beta f + \gamma s) - f^2(\alpha r - \beta s) = 0 \end{aligned}$$

The superposition of translational and rotational values again defines patterns in the plane which consist of a negative and a positive area. These patterns, called copoint patterns, are defined by the same four parameters which characterize the coaxis patterns.

[2]The copoint and coaxis vectors are dual to each other.

5.3 Search for motion patterns

Utilizing the geometrical constraints developed in the last section, motion estimation for a rigid moving observer will now be addressed through a search technique. The strategy involves checking constraints that a certain solution would impose on the normal flow field and in this way discarding impossible solutions. The search is performed in three steps, where from the first to the third step the constraints become more restrictive, hence the number of possible solutions computed at each step decreases. First a set S_1 of possible solutions for the FOE and axis of rotation is estimated by fitting a small number of patterns to the normal flow field. Two techniques, which use different patterns defined on certain coaxis vectors, are proposed for solving this task. Both fitting processes use the input in a qualitative way, since only the sign of the normal flow is employed. In the second step the third rotational parameter is computed, and the space of solutions is further narrowed to a set S_2. This can be performed by using normal flow vectors that do not contain translation (certain copoint vectors) and approximating the remaining rotational parameter from the given rotational vectors. An alternative approach is to have the active observer change its rotation and compute the third rotational component from the change in the perceived motion patterns. In the second approach loyalty to the exclusively qualitative use of normal flow is maintained. Finally, in the last step all impossible solutions are discarded by checking the validity of the motion parameters at every point.

5.3.1 First step: Pattern fitting

The direction of translation and the axis of rotation define patterns on subsets of the normal flow vectors. In the general case these patterns are described by four independent variables and searching for the solution would mean searching in a four-dimensional parameter space. By concentrating, in an initial search, only on a small number of normal flow vectors, we show how to tackle the problem. Clearly, such a restricted use of data will generally not result in a unique solution, but it allows us to either reduce the dimensionality of the problem ($\alpha\beta\gamma$-algorithm), or to employ motion vectors from all parts of the image plane (ARS-algorithm).

$\alpha\beta\gamma$-algorithm

One way to look at the optical flow vector is to imagine it as a sum of five vectors, each being due to only one of the motion parameters (either one of the two translational or one of the three rotational components). Consequently the value of the normal flow

vector at a point is computed as the sum of the five scalar products of these vectors and the unit vector in the gradient direction. The scalar product of two vectors is zero if the vectors are perpendicular to each other. Thus, by selecting normal flow vectors in particular directions, one or more of the motion components vanish.

The coaxis vectors which are dependent on only two of the three rotational parameters correspond to one of the three coordinate axes. These normal vectors and their patterns have special properties.

The coaxis vectors (A, B, C) when the orientation vector (A, B, C) is the Z-axis are perpendicular to circles whose center is the origin of the image plane, and we call them γ-vectors. Similarly, when (A, B, C) is the X- or Y-axis, the (A, B, C) coaxis vectors are called α-vectors and β-vectors and the corresponding field lines are hyperbolas whose major axes are the image plane's x- and y-axes, respectively. Figure 5.6 depicts these sets of vector field lines and the corresponding γ-, α-, and β-vectors in positive orientation.

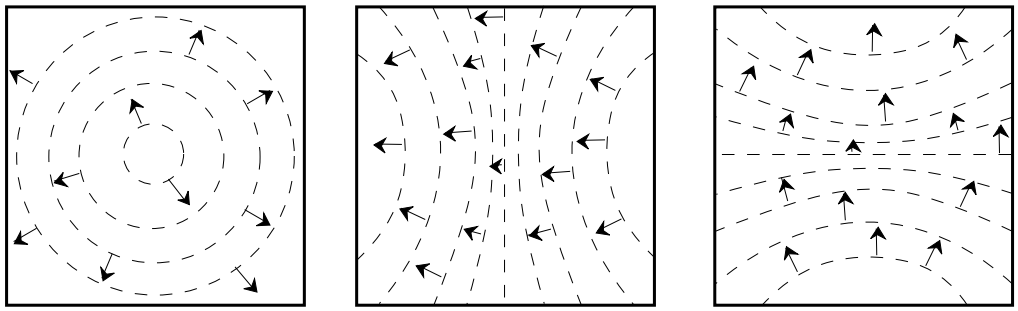


Figure 5.6: If the (A, B, C) axis is the Z-, X-, or Y-axis, the corresponding vector field lines are circles with center O (a), or hyperbolas whose axes coincide with the coordinate axes of the image plane (b and c). Normal flow vectors perpendicular to these field lines are called γ-, α-, and β-vectors.

The value of the α-, β- and γ-vectors due to rotation only can be described by a one-parameter function. Thus the dimensionality of the corresponding patterns is also reduced by one and the search for these patterns can be limited to a three-dimensional parameter space. This becomes clear by substituting into equation (5.5) for the triple (A, B, C) the orientation vectors of the coordinate axes $((1, 0, 0), (0, 1, 0)$ and $(0, 0, 1))$. The rotational components of the γ-vectors are separated by a line passing through the center, which has equation $y = \frac{\beta}{\alpha}x$. For the rotational components of the α-vectors the line is parallel to the x-axis and is defined by the equation $y = \frac{\beta}{\gamma}f$. The β-vectors are separated by a line parallel to the y-axis having equation $x = \frac{\alpha}{\gamma}f$.

The second-order curves separating the translational components of the α-, β- and γ-vectors are obtained from equation (5.4). For the γ-vectors the curve reduces to a circle, which has the FOE and the image center as two diametrically opposite points. Equation (5.4) reduces to

$$h(0,0,1,x_0,y_0,;x,y) = (x - \frac{x_0}{2})^2 + (y - \frac{y_0}{2})^2 - (\frac{x_0}{2})^2 + (\frac{y_0}{2})^2 = 0$$

The curves separating the α- and β-vectors become hyperbolas of the form

$$h(1,0,0,x_0,y_0,;x,y) = y^2 x_0 - xyy_0 - xf^2 + f^2 x_0 = 0$$

and

$$h(0,1,0,x_0,y_0,;x,y) = x^2 y_0 - xyx_0 - yf^2 + f^2 y_0 = 0$$

Figure 5.7 shows the α-, β- and γ-vectors for a general rigid motion.

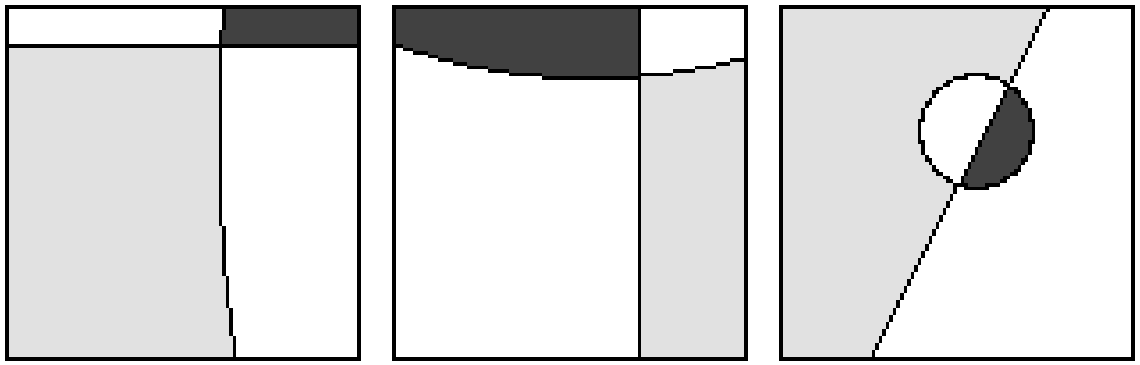


Figure 5.7: α-, β-, and γ-patterns for a general rigid motion.

The algorithm which computes the FOE and the axis of rotation from a given normal flow field by using only the α-, β- and γ-vectors works as follows. With each subset of normal flow vectors is associated a three-dimensional parameter space that spans the possible locations of the FOE and of a line defined by the quotient of two of the three rotational parameters. A search in the three-dimensional subspaces is accomplished by checking the patterns which the subspaces' parameter triples define on selected values of the normal flow field. The α-patterns are fitted to the α-vectors, which provides possible solutions for the coordinates of the FOE: x_0, y_0, and the quotient $\frac{\beta}{\alpha}$. Similarly, the fitting of the β- or γ-patterns results in solutions for x_0, y_0 and $\frac{\beta}{\gamma}$ or $\frac{\alpha}{\gamma}$. The objective is to find the four parameters defining the directions of the translational and rotational axes which give rise to three successfully fitted patterns. Therefore the three subspaces' patterns are combined and the parameter quadruples which define possible solution are determined.

Since only subsets of the normal flow values are considered in the fitting process, the fitting alone does not uniquely define the motion, but only constitutes a necessary condition. Usually there will be a number of parameter quadruples $\{x_0, y_0, \alpha/\gamma, \beta/\gamma\}$ that are selected as candidate solutions through pattern fitting.

The range of values for the coordinates of the FOE and for $\frac{\beta}{\gamma}$ and $\frac{\alpha}{\gamma}$ is $[-\infty, +\infty]$. If a wide-angle lens or a logarithmic retina [Tistarelli and Sandini, 1992] is employed, most of the directions representing the FOE lie in a bounded area of the image plane. Alternatively, in order to cover all possible cases, a coordinate transformation on the sphere can be performed, in which case the coordinates are expressed by two angles.

ARS-algorithm (Axis of rotation search algorithm)

For any rigid motion there exists one class of coaxis vectors which does not contain any rotational components. This set is defined by the actual rotation axis $\frac{A}{C} = \frac{\alpha}{\gamma}$ and $\frac{B}{C} = \frac{\beta}{\gamma}$. Coaxis vectors of this kind are due only to translation and the pattern of these vectors is solely defined by the two-parameter second-order curve $h(\alpha, \beta, \gamma, x_0, y_0; x, y)$. There is only one curve separating the positive from the negative values and thus the pattern is defined on the whole image plane. Since $h(\alpha, \beta, \gamma, x_0, y_0; x, y)$ is linear in x_0 and y_0, the problem of finding the FOE from the normal vectors due only to rotation reduces to estimating the linear discriminant function separating two classes of values (labeled positive and negative).

The pattern is due to only two parameters. In order to find the axis of rotation a search in the two-dimensional parameter space of $\frac{\alpha}{\gamma}$ and $\frac{\beta}{\gamma}$ is performed. For every possible rotation axis the data is checked for linear discrimination. If a second-order curve can be found that separates the positive from the negative values the quadruple (x_0, y_0, $\frac{\alpha}{\gamma}$, $\frac{\beta}{\gamma}$) will be added to the set of possible solutions.

Concerning the computational aspect of solving the discrimination problem, different algorithms from the pattern recognition literature can be applied. For example, the Ho-Kashyap algorithm decides whether a data set is linearly discriminable and will also find the best discrimination.

5.3.2 Second step: Computation of complete rotational motion

Detranslation

Proper selection of normal flow vectors also enables the elimination of the normal flow's translational components. By choosing as normal flow vectors the copoint vectors

defined by the locus of the FOE, this can be achieved. With the location of the FOE the directions of the translational motion components are defined. The optical flow vectors lie on lines passing through the FOE. The normal flow vectors perpendicular to these lines (the copoint vectors (r, s), where $r = x_0$ and $s = y_0$) do not contain translational, but only rotational components. This can be seen from equation (5.1). If the selected gradient direction at a point (x, y) is $((y_0 - y), (-x_0 + x))$ the scalar product of the translational motion component and a vector in the gradient direction is zero. This technique of eliminating the translational component, in the future referred to as "detranslation", is applied to compute the third rotational component and to further reduce the possible number of solutions.

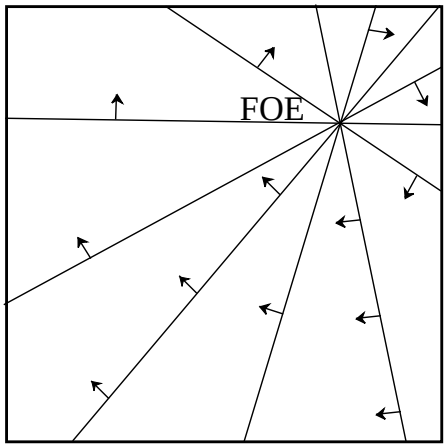


Figure 5.8: Normal flow vectors perpendicular to lines passing through the FOE are due only to rotation.

For each of the possible solutions computed in the second module the normal flow vectors perpendicular to the lines passing through the FOE have to be tested to see if they are due only to rotation (see Figure 5.8). This results in solving an overdetermined system of linear equations. Since two of the rotational parameters are already computed, there is only one unknown, the value γ. Every point supplies an equation of the form

$$\gamma = \frac{u_n}{(\frac{\alpha}{\gamma}(\frac{xy}{f}n_x + (\frac{y^2}{f} + f)n_y) - \frac{\beta}{\gamma}((\frac{x^2}{f} + f)n_x + \frac{xy}{f}n_y) + (yn_x - xn_y))} \tag{5.7}$$

Provided the chosen normal flow vectors are due only to rotation, the solution to the overdetermined system gives the γ value. In a practical application a threshold has to be chosen to discriminate between possible and impossible solutions. The value of the residual is used to confirm the presumption that the selected normal flow values are purely rotational. Usually "detranslation" will not result in only one solution, but will provide a set S_2 of possible parameter quintuples.

Alternatively to detranslation a different approach to computing the third rotational component may be taken. It requires the observer to be active in order to acquire, through a controlled motion, additional information about its rotation.

Pattern change through rotation

From one pattern search alone, we can only derive the ratios of the three rotational components. However, if the observer is active it has the capability of changing its rotational velocity. In the image plane such a change will result in the superposition of an additional flow field on the existing one. A pattern search on the new image measurements will provide the new rotation axis. From the change of the rotation axes the complete rotation can then be derived.

The first pattern fitting supplies the two parameters k_1 and l_1, where

$$k_1 = \frac{\alpha_1}{\gamma_1} \quad \text{and} \tag{5.8}$$

$$l_1 = \frac{\beta_1}{\gamma_1}. \tag{5.9}$$

A change of the rotational velocity by $(\alpha^*, \beta^*, \gamma^*)$ will alter the parameters of the rotation axis to k_2 and l_2:

$$k_2 = \frac{\alpha_2}{\gamma_2} = \frac{\alpha_1 + \alpha^*}{\gamma_1 + \gamma^*} \tag{5.10}$$

$$l_2 = \frac{\beta_2}{\gamma_2} = \frac{\beta_1 + \beta^*}{\gamma_1 + \gamma^*} \tag{5.11}$$

From (5.8) and (5.9) we obtain

$$\gamma_1 = \frac{k_2\gamma^* - \alpha^*}{k_2 - k_1}.$$

Similarly from (5.10) and (5.11) we obtain

$$\gamma_1 = \frac{l_2\gamma^* - \beta^*}{l_2 - l_1}$$

In this analysis the change of the local coordinate system between the two measurements is not considered. If the measurements are performed within a small interval of time, such a simplification is legitimate. Concerning the amount of search to be performed, knowledge about the first pattern allows us to restrict the search in the new normal flow field to a small fraction of the complete parameter space. If the additional rotation is performed for a small amount of time only, this will result in a very small

change in the location of the FOE. Thus only a tiny spatial neighborhood of the original FOE has to be considered. Furthermore, the parameters of the additional rotation supply information about the direction in which $g(A, B, C, \alpha, \beta, \gamma)$ (the straight line separating the positive from the negative rotational components) will change its slope and intercept. Additional reduction of the search space can be achieved by restricting the change of velocity to only one of the rotational motion parameters.

5.3.3 Third step: Derotation

The modules described so far considered only subsets of the normal flow vectors. Clearly, after having found possible solutions for the FOE and the axis of rotation, we can test every candidate solution for its correctness by employing any class of coaxis vectors. Since the quadruple $(x_0, y_0, \frac{\alpha}{\gamma}, \frac{\beta}{\gamma})$ defines a pattern on every class of coaxis vectors, we only have to test for the existence of this pattern. However, a pattern in the general case is defined only on parts of the image plane. Thus even by testing every possible class of coaxis vectors not every normal flow vector will be tested.

In order to eliminate all motion parameters which are in contradiction to the given normal flow field, every normal flow vector has to be checked. This check is performed in the "derotation" technique. With every parameter quintuple computed in the second step a possible FOE and a rotation are defined. The three rotational parameters are used to derotate the normal flow vectors by subtracting the rotational component $(u_{\text{rot}}, v_{\text{rot}})$. At every point the flow vector $(u_{\text{der}}, v_{\text{der}})$ is computed:

$$\begin{aligned} u_{\text{der}} &= u_n n_x - u_{\text{rot}} n_x \\ v_{\text{der}} &= v_n n_y - v_{\text{rot}} n_y \end{aligned} \tag{5.12}$$

If the parameter quintuple defines the correct solution, the remaining normal flow is purely translational. Thus it has to have the property of an emanating motion field. Since the direction of optical flow for a given FOE is known, the possible directions of the normal flow vectors can be determined. The normal flow vector at every point is confined to lie in a half-plane (see Figure 5.9). The technique checks all points for this property and eliminates solutions that cannot give rise to the given normal flow field.

5.3.4 The complete capability

In this section we give a summary of the complete technique in the form of a block diagram. The computation of an observer's egomotion is performed in three steps, where

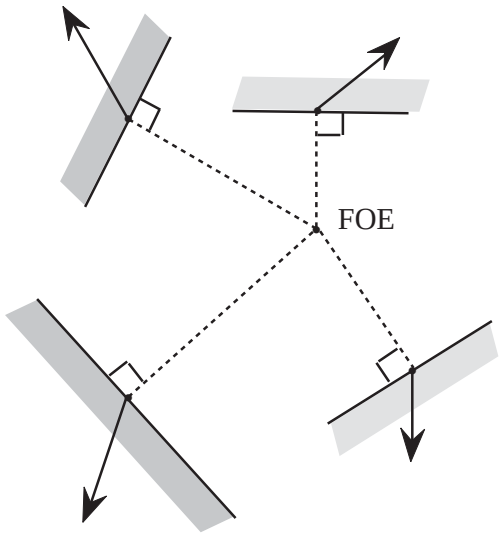


Figure 5.9: Normal flow vectors due to translation are constrained to lie in half-planes.

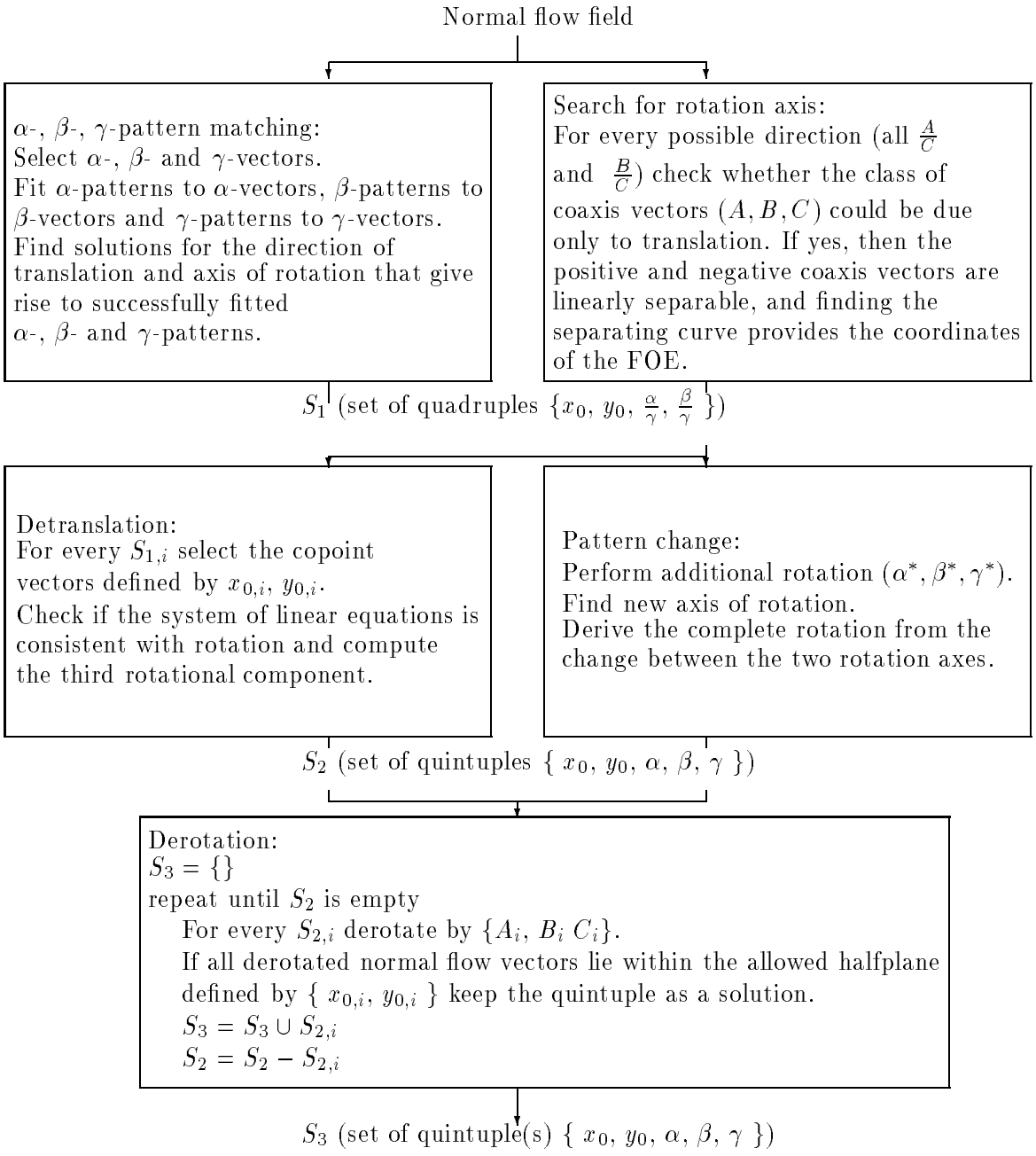
for each of the first two steps two alternative modules can be chosen. The sets of candidate solutions which are determined in the five modules are called S_1, S_2, S_3. To denote single solutions or single parameters, subscripts are used: $S_{1,i}$, $S_{2,i}$, $x_{0,i}$, $y_{0,i}$, and so on. The input to the algorithm is a normal flow field and outputs are all possible solutions for the direction of translation and the rotation which can give rise to this normal flow field.

It can easily be shown that from normal flow fields, in general, 3D motion cannot be uniquely computed. From two flow fields a common normal flow field can be constructed: take two different optical flow fields that originate from different scenes and different rigid motions. At every point in the image plane there exist two motion vectors. A normal flow vector, which is defined as the projection of a flow vector, is constrained to lie on a circle. Thus the intersection of the two circles defines the normal flow vector which is due to both motions (Figure 5.10).

The algorithm determines the complete set of solutions. If for a given normal flow field the algorithm finds more than one solution, then from the normal flow field alone the 3D motion cannot be determined uniquely. In this case one is obliged to use matching of prominent features to eliminate the incorrect motion parameters.

The computed 3D motion parameters and the normal flow values supply two linear equations in u and v at every point from which the optical flow field can be determined:

$$\begin{aligned} \frac{u - u_{\text{rot}}}{v - v_{\text{rot}}} &= \frac{u_{\text{trans}}}{v_{\text{trans}}} \\ u_n &= u n_x + v n_y \end{aligned} \tag{5.13}$$


Normal flow field
α-, β-, γ-pattern matching:
Select α-, β- and γ-vectors.
Fit α-patterns to α-vectors, β-patterns to β-vectors and γ-patterns to γ-vectors.
Find solutions for the direction of translation and axis of rotation that give rise to successfully fitted α-, β- and γ-patterns.
Search for rotation axis:
For every possible direction (all A/C and B/C) check whether the class of coaxis vectors (A, B, C) could be due only to translation. If yes, then the positive and negative coaxis vectors are linearly separable, and finding the separating curve provides the coordinates of the FOE.
S1 (set of quadruples {x0, y0, α/γ, β/γ})
Detranslation:
For every S1,i select the copoint vectors defined by x0,i, y0,i.
Check if the system of linear equations is consistent with rotation and compute the third rotational component.
Pattern change:
Perform additional rotation (α*, β*, γ*).
Find new axis of rotation.
Derive the complete rotation from the change between the two rotation axes.
S2 (set of quintuples { x0, y0, α, β, γ })
Derotation:
S3 = {}
repeat until S2 is empty
For every S2,i derotate by {Ai, Bi Ci}.
If all derotated normal flow vectors lie within the allowed halfplane defined by { x0,i, y0,i } keep the quintuple as a solution.
S3 = S3 ∪ S2,i
S2 = S2 − S2,i
S3 (set of quintuple(s) { x0, y0, α, β, γ })

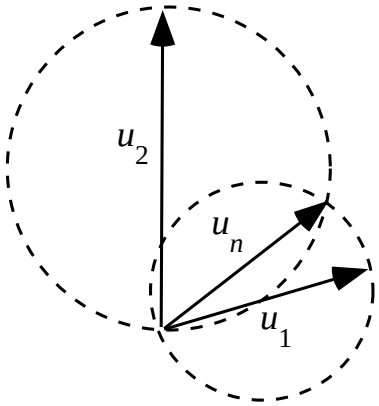


Figure 5.10: The intersection of two circles defines the possible location of the normal flow vector which corresponds to two different optical flow vectors.

The unique solution is then derived by checking prominent feature points of the first frame for their existence in the second frame at the locations computed by the optical flow values.

5.4 Restricted motion capabilities

In the previous sections we showed how to compute an observer's 3D motion in the most general case, i.e. if the motion conists of three translational and three rotational components. In many practical applications, however, the observer has only restricted motion capabilities. For our motion decoding strategy a limited degree of freedom in the 3D motion will result in a reduction of the parameter space in which the search is performed. Here we will discuss algorithms for two of the most common limitations: the case of an observer moving with only translational parameters and the case of an observer which cannot rotate around the Z-axis.

For the case of purely translational motion, techniques that employ only normal flow measurements have previously been proposed by Horn and Weldon [1987] and Negahdaripour [1986]. These authors make use of the fact that the scaled depth $\frac{W}{Z}$ is positive for all points in the image plane if the observer is approaching the scene. Therefore, every normal flow measurement constrains the location of the FOE; it has to be in the half-plane (or hemisphere in case of spherical projection) which is on the opposite side of the graylevel edge from the normal flow vector. Algebraically speaking, every normal flow measurement supplies an inequality; thus methods like linear programming or perceptron learning were suggested as algorithmic strategies. Here, however, we discuss an algorithm which is based on constraints different from those used by the above authors.

If the observer is moving only translationally, there will be no constraints due to rotation on the patterns. For both the coaxis and the copoint vectors, this means that only one curve separates the positive from the negative values. The coaxis vectors are separated by the second-order curve $g(A, B, C, \alpha, \beta, \gamma)$ and the copoint vectors by the straight line $l(r, s, \alpha, \beta, \gamma)$. Both curves are linear in the two unknowns x_0 and y_0. Thus finding the FOE is a problem of determining the linear discriminant function which separates the positive from the negative values. Linear discrimination can be solved using iterative methods, such as the perceptron algorithm [Rosenblatt, 1962; Duda and Hart, 1962], for which convergence to a correct solution is guaranteed.

For a purely translational flow field any class of coaxis vectors (A, B, C) and copoint vectors (r, s) is linearly separable. The larger the number of normal flow vectors in the class, the more accurately the discrimination function and thus the FOE can be determined. If, in a normal flow field, there does not exist any direction with a large enough number of vectors, we can compute the discriminant functions for several classes of coaxis vectors and take the average of the computed parameters as the final FOE.

The most common motion restriction we are dealing with in practical applications is the case of an observer with only two rotational degrees of freedom. One can consider the mammalian eye as an example.[3]

In such a case motion estimation through pattern fitting becomes much easier. The fact that the axis of rotation is defined by only one parameter $(\frac{\alpha}{\beta})$ affects our motion estimation technique in the two modules of the first step.

If the $\alpha\beta\gamma$-algorithm is employed, there does not exist a line g due to rotation for the α- and β-pattern because γ is zero. Depending on the sign of α or β all normal flow vectors have either positive or negative rotational components. The pattern which has to be located for these cases is an area covered by a hyperbola. This means that the search problem is three-dimensional for the γ-patterns, but only two-dimensional for the α- and β-patterns.

On the other hand, if pattern fitting is performed through a search for the rotation axis, only the one-dimensional space spanning the possible values for $\frac{\alpha}{\beta}$ needs to be considered. After having found candidate solutions for the axis of rotation and the FOE, the algorithm proceeds as described in the general case.

[3]Actually the human eye can be rotated around the Z-axis by a small amount (a motion referred to as cyclotorsion [Pahlavan and Eklundh, 1992]).

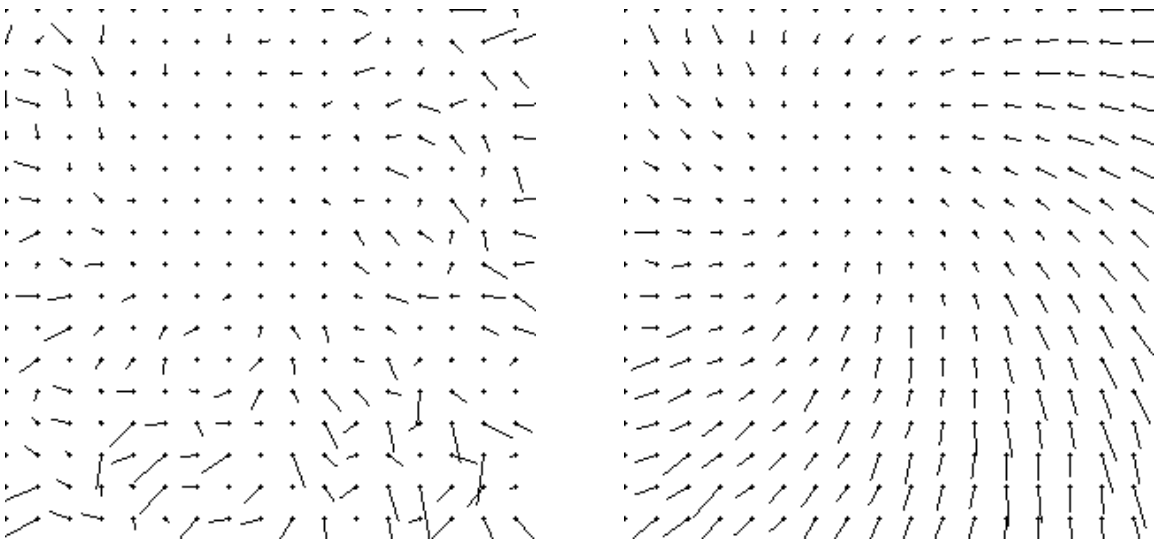
5.5 Experiments

In a series of experiments four modules of the general egomotion recovery strategy: α-, β-, γ-pattern matching, search for rotation axis, detranslation and derotation were tested. In the implementation of these modules the following approach was taken: The elimination of impossible parameters from the space of solutions involves discrimination on the basis of quantitative values. This was implemented in the following way: Normal flow values in certain directions are selected, if they are within a tolerance interval. In the pattern fitting and derotation modules counting is applied to discriminate between possible and impossible solutions. The quality of the fit, the "success rate", is measured by the number of values with correct signs normalized by the total number of selected values. The amount of rotation in the derotation module is computed through simple linear least squares minimization and the discrimination between accepted and rejected motion parameters is based on the value of the residual.

In the pattern fitting and derotation modules no quantitative use of values is made, since only the sign of the normal flow is considered. This limited use of data makes the module very robust, and the correct solutions for the axes of translation and rotation are usually found even in the presence of high amounts of noise. To give some quantitative justification of this we define the error in the normal flow at a point as a percentage of the correct vector's length. Since the sign of the vector is not affected as long as the error does not exceed the correct vector in value, the "pattern fitting" will find the correct solution in all cases of up to 100% error.

Several experiments have been performed on synthetic data. For different 3D motion parameters normal flow fields were generated; the depth value (in an interval) and the gradient direction were chosen randomly. In all experiments on noiseless data the correct solution was found as the best one. Figure 5.11 shows the optical flow field and the normal flow field for one of the generated data sets. The image size was 100×100, the focal length was 150, the FOE was at (–5, +30) and the ratio of the rotational components was $\alpha : \beta : \gamma = 10 : 11 : 150$.

In Figure 5.12 the fitting of the circle and the hyperbolas to the α- β- and γ-vectors and the coaxis pattern (α, β, γ) is displayed. The tolerance interval for the direction of normal flow vectors was taken to be 10°. Points with positive normal flow values are rendered in a light color and points with negative values are dark. Perturbation of the normal flow vectors' lengths by up to 50% did not prevent the method from finding the correct solution.



Normal flow field Optical flow field

Figure 5.11: Flow fields of synthetic data.

As a first example of a real scene, the NASA-Ames sequence[4] was chosen. The camera undergoes only translational motion; thus different amounts of rotation were added. For all points at which translational motion can be found the rotational normal flow is computed, and the new position of each pixel is evaluated. The "rotated" image is then generated by computing the new graylevels through bilinear interpolation. The images were convolved with a Gaussian of kernel size 5×5 and standard deviation $\sigma = 1.4$. The normal flow was computed by using 3×3 Sobel operators to estimate the spatial derivatives in the x- and y-directions and by subtracting the 3×3 box-filtered values of consecutive images to estimate the temporal derivatives.

When adding rotational normal flow on the order of a third to three times the amount of translational flow, the exact solution was always found among the best fitted parameter sets. The solution could not clearly be derived as a unique point in the five-dimensional parameter space; rather we obtained a number of solutions that form a "fuzzy blob" in the solution space (see Figure 5.15). All solutions with success rate higher than 99% were very close to the correct one with the FOE deviating by at most 6% of the focal length from the correct positions (x_0, y_0). In Figure 5.13 the computed normal flow vectors and the fitting of the α-, β- and γ-vectors for one of the "rotated" images is shown. Areas of negative normal flow vectors are marked by horizontal lines and areas of positive values by vertical lines. The ground truth for the FOE is $(-5\ ,-8)$, the focal length is 599

[4]This is a calibrated motion sequence made public at the Workshop on Visual Motion, 1991

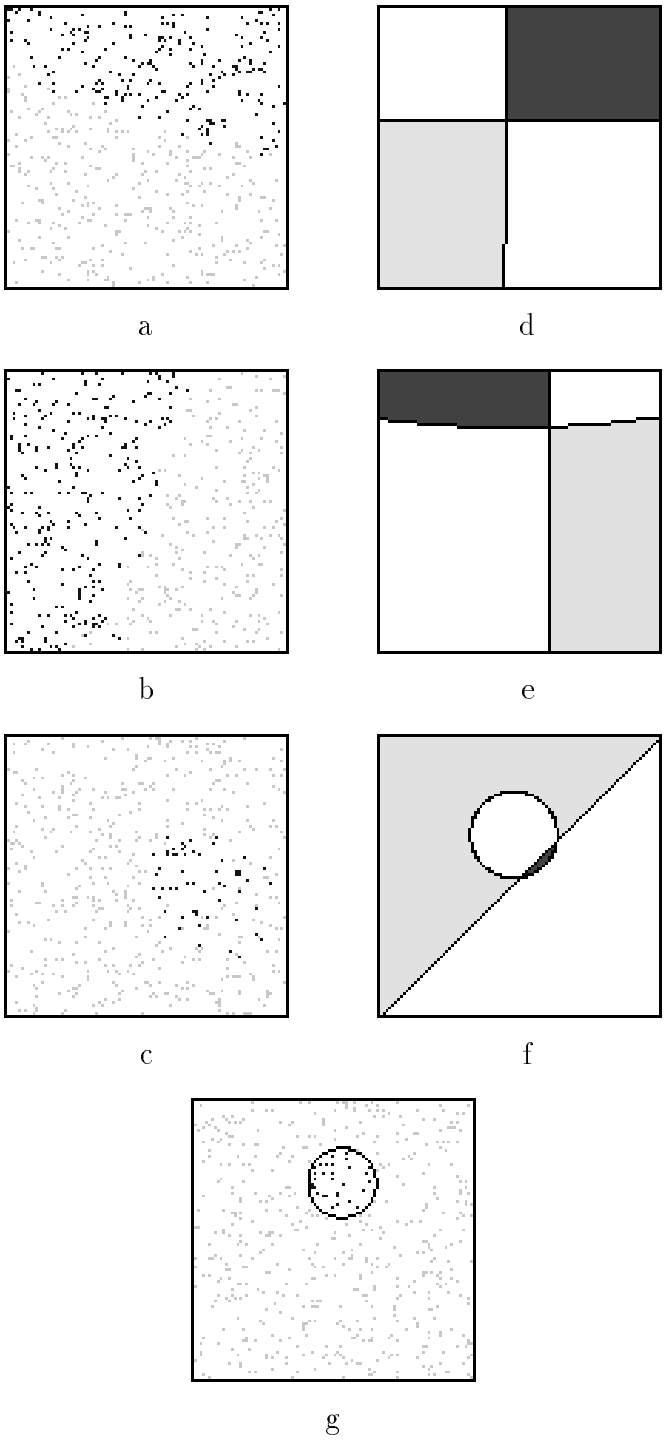
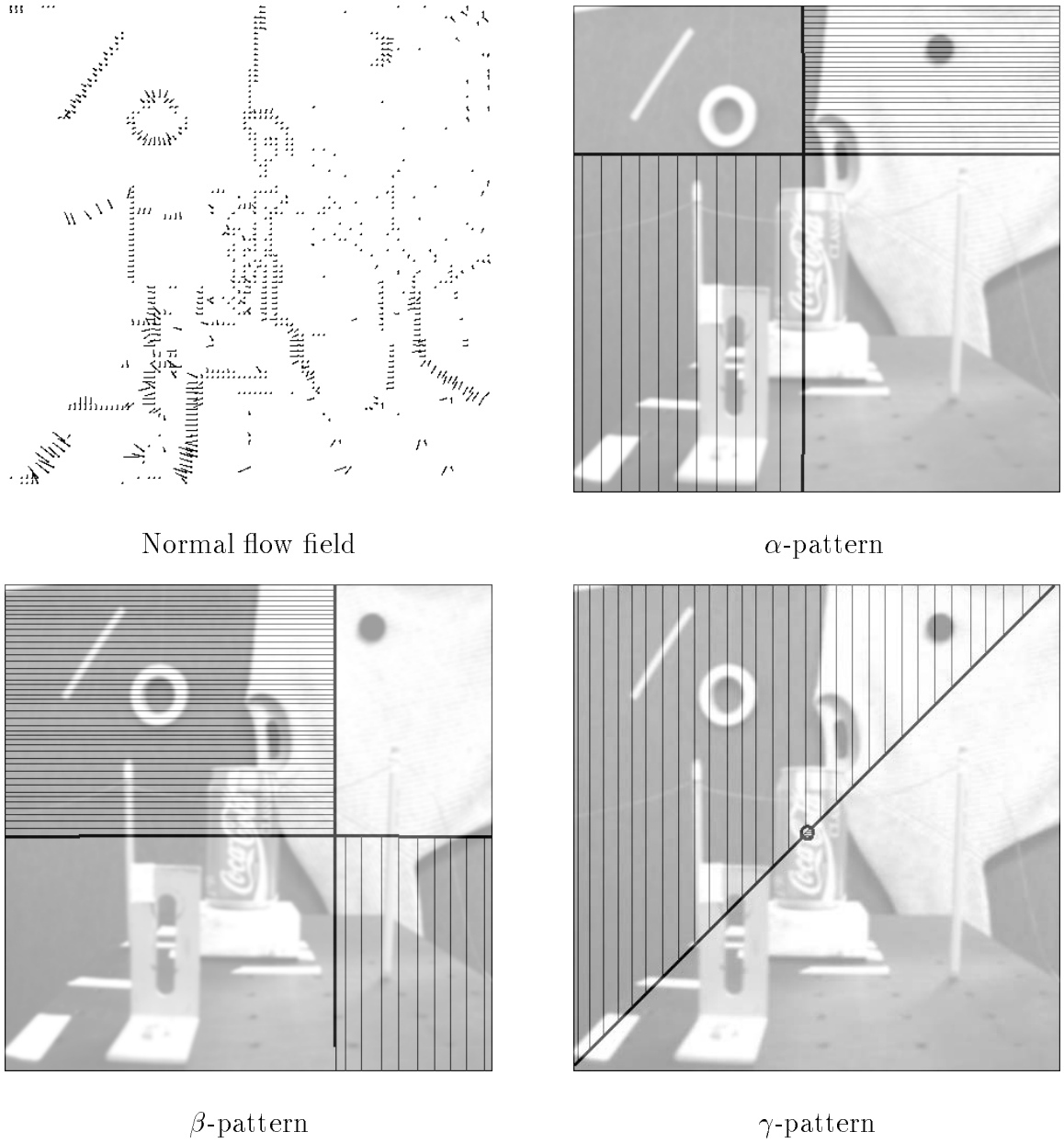


Figure 5.12: (a), (b), (c): Positive and negative α-, β-, and γ-vectors. (d), (e), (f) Fitting of α-, β-, and γ-patterns. (g): Separation of coaxis pattern (α, β, γ).

pixels, and the rotation between the two image frames is $\alpha = 0.0006$, $\beta = 0.0006$, and $\gamma = 0.004$. The algorithm found the correct solution. Figure 5.14 shows, overlaid on

the original image in black, the curves and lines separating positive and negative α-, β-, and γ-vectors. (Due to the large focal length the parts of the α- and β-hyperbolas which appear in the image plane are close to straight lines.) The curve separating the coaxis vectors (α, β, γ) is rendered in white. This is to visualize the fact that the intersection of the second order curves gives the FOE, and the intersection of the straight lines and the curve separating the coaxis vectors (α, β, γ) (white curve) gives the point where the axis of rotation pierces the image plane.



Normal flow field

α-pattern

β-pattern

γ-pattern

Figure 5.13: NASA scene: Normal flow field and fitting of α-, β-, and γ-vectors.

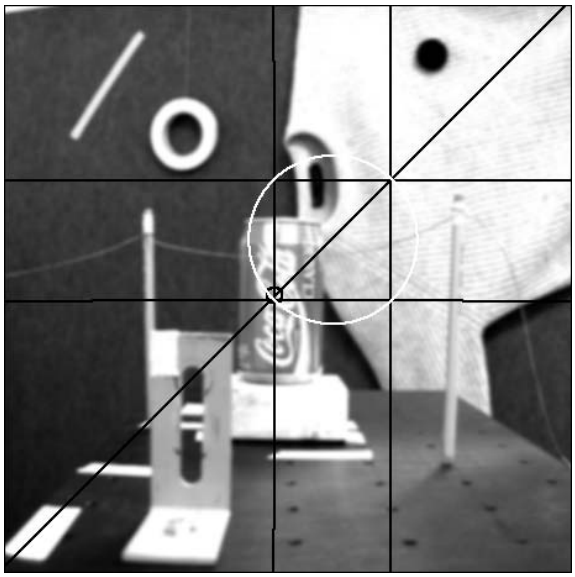


Figure 5.14: NASA scene with overlaid curves and lines separating positive and negative α-, β-, and γ-vectors and coaxis vectors (α, β, γ). At the intersection of the second order curves is the FOE (at the center of the coke can). The intersection of the straight lines denotes the point where the rotation axis pierces the image plane (over the pullover).

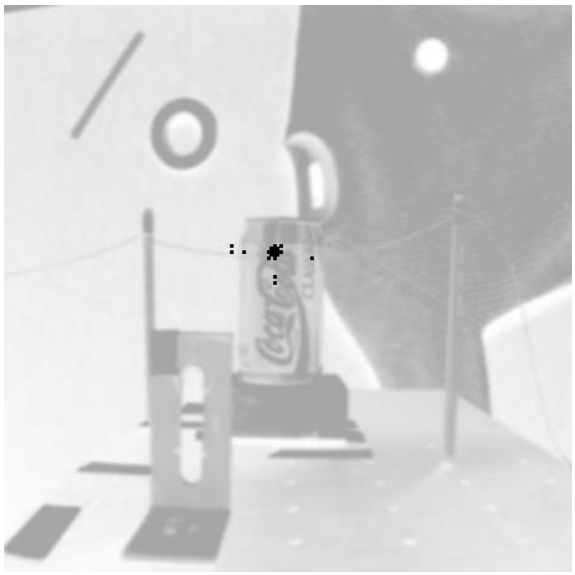


Figure 5.15: NASA scene: The FOE's of all solutions with a success rate of 99% and higher are marked by black squares. The actual solution is at the center of the blob.

A second series of experiments was performed on a series of images collected with the Royal Institute of Technology binocular head/ eye system [Pahlavan and Eklundh, 1992]. This is an active vision system consisting of two cameras, each of which has two rotational degrees of freedom (pan and tilt). Furthermore, each eye module has the capability of carrying a motorized zoom lens. The distance between the two eyes can be varied by changing the baseline. The head rests on a neck that can rotate around the X- and Y-axes (pan and tilt). The images were acquired by using only one camera, which was positioned in a fixed orientation relative to the baseline. The camera translated along the baseline, while at the same time rotating around the X- and Y-axes. The parameters of the camera were as follows: focal length in X-direction: 1163 pixels, focal length in Y-direction: 1316 pixels, image size: 574×652, center of the image: (332, 305) (measuring from the bottom left corner).

Figure 5.16: Image taken by KTH head.

The algorithm was run on three different sequences. In all cases it computed the axis of rotation and translation correctly, but due to the high value of the focal length a number of other solutions (in the range $\pm$ 0.08 times the focal length away from the true solution) had acceptable confidence. Furthermore, since the rotational components were very small with regard to the translational ones (the absolute values were five to ten times smaller) the amount of rotation was not computed correctly in all cases. Figure 5.16 shows the scene. For one of the settings the results are displayed: the ground truth is FOE = (–129, + 146) (measured from the image center); α = 0.000125663 rad; β = 0.000251327 rad, $\gamma = 0.0$ rad. Notice, because $\gamma = 0$, the lines separating the rotational components of the α- and β-vectors lie in infinity and thus do not appear in the image plane. Since the focal length is different in the X- and Y-directions the circle in the γ-

pattern is distorted to an ellipse. For this case the algorithm computed the correct FOE and the correct ratio $\frac{\beta}{\alpha}$; however, the estimated value for the rotation was equal to 90 percent of the actual one. In Figure 5.17 the positive and negative α-, β-, and γ-vectors and the corresponding patterns are displayed. For these experiments a tolerance interval of 6° has been allowed in the choice of normal flow vectors for the α-, β-, and γ-vectors. For clarity of the pictorial description, all the points corresponding to the chosen vectors were enlarged by a factor of four. Figure 5.18 shows the positive and negative coaxis vectors $(\alpha, \beta, 0)$. Since the rotation is very small in relation to the translation, a smaller tolerance interval, namely only 3°, was chosen. Figure 5.19 shows the computed normal flow field, and Figure 5.20 shows the conic sections separating the translation components of the α-, β-, γ- and coaxis vectors (α-, β-,0) overlaid on the image. At the intersection of these curves lies the FOE.

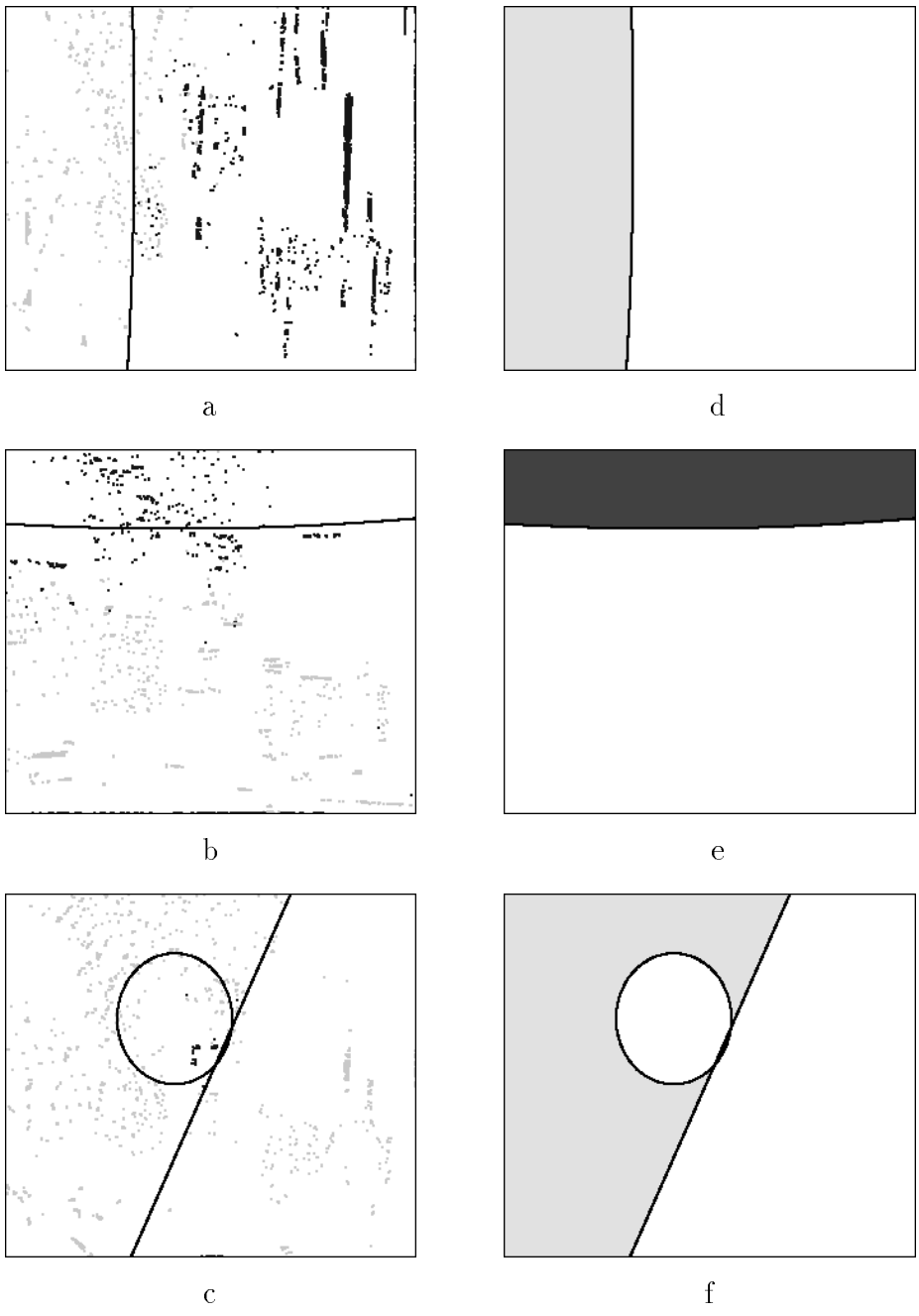


Figure 5.17: KTH scene: (a), (b), (c): Positive and negative α-, β-, and γ-vectors. (d), (e), (f) Fitting of α-, β-, and γ-patterns.

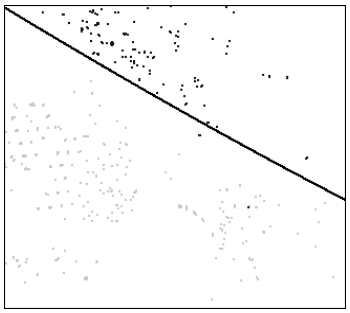


Figure 5.18: KTH scene: Positive and negative coaxis vectors $(\alpha, \beta, 0)$ and fitting of conic section.

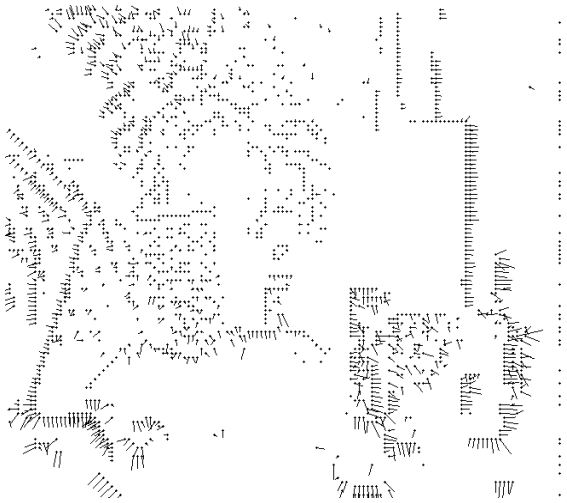


Figure 5.19: Normal flow field : Scene taken by KTH head.

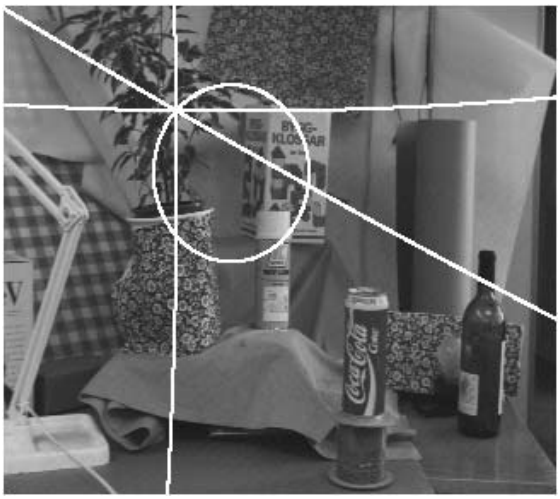


Figure 5.20: KTH scene: Curves separating the translational components of the α-, β-, γ-vectors and coaxis vectors $(\alpha, \beta, 0)$. At their intersection lies the FOE.

Chapter 6

Active Pattern Techniques

In this chapter two examples of the applicability of the global geometrical constraints to navigational tasks for an active observer are presented. First, it is shown how the ego-motion estimation problem becomes easier if additional information is obtained through tracking. Then a strategy for bringing the FOE to the center of the image plane is discussed. This is achieved by having the observer associate a pattern in the image plane with the motion it wants to obtain. The observer iteratively changes its motion parameters, where the change is derived from measurements of the current pattern. In this way the observer attains its goal without going through the intermediate computation of the parameters describing its motion.

6.1 The tracking constraint

Assume that an active observer in rigid motion is tracking, as before (Section 4.2), an environmental point whose image (x, y) lies at the center of the visual field ($(x, y) = (0, 0)$). Assume also that during a small time interval $[t_1, t_2]$ the motion of the observer remains constant and that during this time the camera, in order to correctly track, rotates around its x- and y-axes with rotational velocities $\omega_x(t), \omega_y(t)$ respectively, with $t \in [t_1, t_2]$. The tracking rotation adds to the existing flow field (u, v) a rotational flow field (u_{tr}, v_{tr}), where

$$u = \frac{-Uf + xW}{Z} + \frac{\alpha xy}{f} - \beta\left(\frac{x^2}{f} + f\right) + \gamma y$$

$$v = \frac{-Vf + yW}{Z} + \alpha\left(\frac{y^2}{f} + f\right) - \beta\frac{xy}{f} - \gamma x$$

$$u_{tr} = \omega_x\frac{xy}{f} - \omega_y\left(\frac{x^2}{f} + f\right)$$

$$v_{tr} = \omega_x \left(\frac{y^2}{f} + f \right) - \omega_y \frac{xy}{f}.$$

Here ω_x, ω_y are the tracking velocities at the time of the observation, and Z is the depth of the tracked point.

As before, if tracking rotation is represented by an angle ϕ around a rotation axis $(n_1, n_2, 0)$ with direction cosines n_1, n_2, then the introduced flow (u_{tr}, v_{tr}) is given by

$$\begin{aligned} u_{tr} &= n_2 f \tan\phi \\ v_{tr} &= -n_1 f \tan\phi \end{aligned}$$

Since the camera is continuously tracking the point at the origin, at any time $t \in [t_1, t_2]$ the introduced tracking motion compensates for the existing flow there, i.e.

$$\begin{aligned} n_{2_t} f \tan\phi_t &= \tfrac{Uf}{Z_t} + \beta f \\ n_{1_t} f \tan\phi_t &= -\tfrac{Vf}{Z_t} + \alpha f \end{aligned}$$

with the subscript t denoting the time of observation. Writing the above two constraints at times t_1 and t_2 we have

$$n_{2_{t_1}} f \tan\phi_{t_1} = \frac{Uf}{Z_{t_1}} + \beta f \tag{6.1}$$

$$n_{1_{t_1}} f \tan\phi_{t_1} = -\frac{Vf}{Z_{t_1}} + \alpha f \tag{6.2}$$

$$n_{2_{t_2}} f \tan\phi_{t_2} = \frac{Uf}{Z_{t_2}} + \beta f \tag{6.3}$$

$$n_{1_{t_2}} f \tan\phi_{t_2} = -\frac{Vf}{Z_{t_2}} + \alpha f \tag{6.4}$$

Subtracting (6.3) from (6.1) and (6.4) from (6.2), we obtain:

$$\begin{aligned} f(n_{2_{t_1}} \tan\phi_{t_1} - n_{2_{t_2}} \tan\phi_{t_2}) &= Uf \left[\tfrac{1}{Z_{t_1}} - \tfrac{1}{Z_{t_2}} \right] \\ f(n_{1_{t_1}} \tan\phi_{t_1} - n_{1_{t_2}} \tan\phi_{t_2}) &= -Vf \left[\tfrac{1}{Z_{t_1}} - \tfrac{1}{Z_{t_2}} \right] \end{aligned}$$

or by dividing,

$$\frac{V}{U} = \frac{n_{1_{t_2}} \tan\phi_{t_2} - n_{1_{t_1}} \tan\phi_{t_1}}{n_{2_{t_1}} \tan\phi_{t_1} - n_{2_{t_2}} \tan\phi_{t_2}}$$

In the sequel we denote the known quantity $\frac{n_{1_{t_2}} \tan\phi_{t_2} - n_{1_{t_1}} \tan\phi_{t_1}}{n_{2_{t_1}} \tan\phi_{t_1} - n_{2_{t_2}} \tan\phi_{t_2}}$, which is defined by the ratio of the tracking accelerations in the vertical and horizontal directions, by T. If $(x_0, y_0) = \left(\frac{Uf}{W}, \frac{Vf}{W}\right)$ is the FOE, the above equation becomes $\frac{y_0}{x_0} = \frac{V}{U} = T$, which is

a linear constraint on the FOE. It restricts the location of the FOE to a straight line passing through the origin of the image coordinate system with slope T.

Furthermore, we obtain a constraint on the rotational motion. Tracking adds a rotational flow field to the existing one. Let us call the composited rotational parameters A and B, where $A = \alpha + \omega_x$ and $B = \beta + \omega_y$. From equations (6.1) and (6.2) at the origin we have

$$\frac{n_{1_{t_1}} f \tan\phi_{t_1} - \alpha f}{n_{2_{t_1}} f \tan\phi_{t_1} - \beta f} = -\frac{V}{U}$$

Since at the center $n_{1_{t_1}} f \tan\phi_{t_1}$ is equal to $-\omega_x f$ and $n_{2_{t_1}} f \tan\phi_{t_1}$ is equal to $-\omega_y f$, we obtain

$$\frac{\omega_x + \alpha}{\omega_y + \beta} = \frac{A}{B} = -\frac{V}{U} = -\frac{y_0}{x_0} = -T$$

or $\frac{B}{A} = -T^{-1}$ and $\frac{y_o}{x_0} = T$.

In other words, tracking provides not only the line $\frac{y_0}{x_0} = T$ on which the FOE lies, but also defines the ratio of the two rotational parameters $(\frac{B}{A})$ (see Figure 6.1).

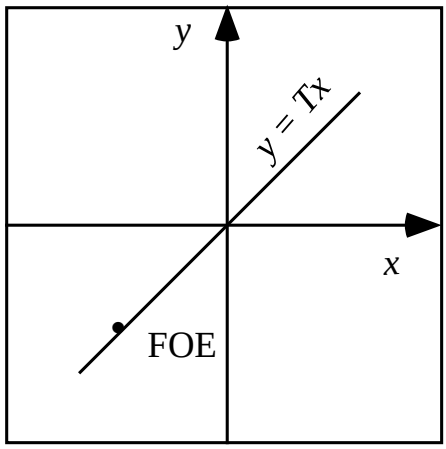


Figure 6.1: Fixation constrains the FOE to lie on the line $y = Tx$ and provides the value for the ratio $\frac{\beta+\omega_y}{\alpha+\omega_x} = -T^{-1}$.

When analyzing the motion field composed of the original motion and the tracking motion, these two constraints can be directly employed to reduce the dimensionality of the search for the γ-pattern. From the original three unknowns two are supplied by the tracking constraint. The line due to rotation $(y = \frac{B}{A})$ and the line through the center on which the FOE lies are known. Thus only the diameter of the circle which defines the FOE has to be found. Locating the pattern becomes a one-dimensional search problem.

After having found candidate solutions for the FOE and $\frac{A}{B}$, the two remaining rotational parameters can be found from the copoint vectors defined by the possible FOE's. As in detranslation, for every (x_{0_i}, y_{0_i}) the copoint vectors (x_{0_i}, y_{0_i}) have to be tested to see if they are due only to rotation. This results in solving an overdetermined system of linear equations, now in two unknowns. By considering all normal flow vectors $\vec{u}_{n_i} = u_{n_i}(n_{x_i}, n_{y_i})$, $i = 1, \ldots, k$, perpendicular to the lines passing through (x_{0_i}, y_{0_i}), we obtain

$$u_{n_i} = \left(A\frac{xy}{f} - B\left(\frac{x^2}{f} + f\right) + Cy\right) n_{x_i} + \left(A\left(\frac{y^2}{f} + f\right) - B\frac{xy}{f} - Cx\right) n_{y_i}$$

and since $\frac{A}{B} = -T$, we have

$$u_{n_i} = \left(-B\left(\frac{Txy}{f} + \frac{x^2}{f} + f\right) + Cy\right) n_{x_i} - \left(BT\left(\frac{y^2}{f} + f + \frac{xy}{f}\right) + Cx\right) n_{y_i} \quad i = 1, \ldots, k.$$

So, if the above k linear equations in the two unknowns B, C are consistent, we have found a possible FOE $((x_{0_i}, y_{0_i}))$ and we have computed its corresponding rotation. As a last step, in order to eliminate impossible solutions, derotation, as described in the general case, is performed.

6.2 Bringing the FOE to the center (Servoing the heading direction)

The observer wants to translate in the direction toward which it looks. Stated in terms of motion parameters, this means that translation is along the Z-axis, and the FOE is thus to be found at the center of the image coordinate system. One possibility for solving this problem is by computing the motion parameters and changing them in such a way that $\frac{U}{W}$ and $\frac{V}{W}$ become zero. Another way to accomplish this task is to iteratively change the motion parameters through rotation, where the change in rotation is obtained by checking necessary requirements on the pattern of the current image. We will explain here the idea behind this approach.

The task can be solved by employing only the α- and β-patterns. As can be seen from Section 5.3.1, the hyperbola separating the positive from the negative translational components of the α-vectors is given by

$$h(1, 0, 0, x_0, y_0; x, y) = y^2 x_0 - xyy_0 - xf^2 + f^2 x_0 = 0.$$

This curve passes through the point (x_0, y_0) and the point $(x_0, 0)$. The part of the curve for which the absolute value of the x-coordinate is smaller than $|x_0|$ has a y-

coordinate between zero and y_0 ($0 < y < y_0$); elsewhere the absolute value of x is larger than $|x_0|$. Figure 6.2 depicts two different α-hyperbolas and the corresponding FOEs.

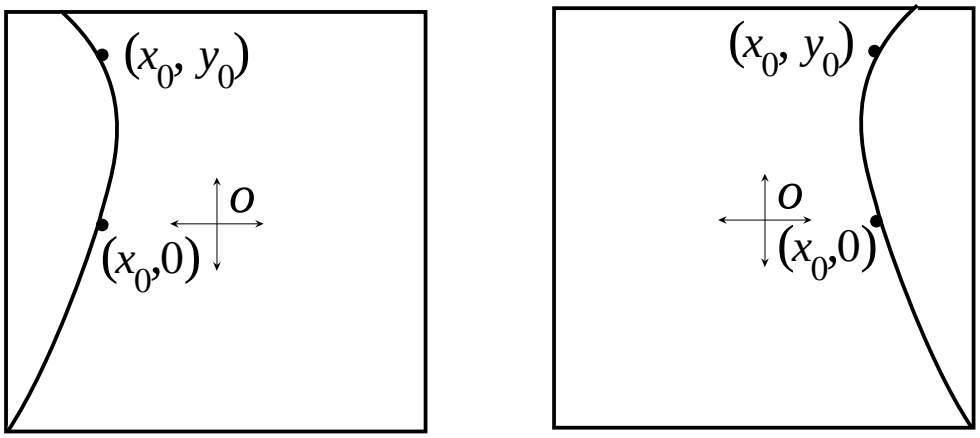


Figure 6.2: Possible α-hyperbolas.

If the observer is approaching the scene, i.e if there exists an FOE, then the area on the left side of a large enough image plane will contain positive values and the area on the right side will contain negative values. If the FOE coincides with the center of the coordinate system, the hyperbola degenerates to the y-axis.

Symmetrically, the β-vectors are separated by the hyperbola $h(0,1,0,x_0,y_0;x,y) = x^2y_0 - xyx_0 - yf^2 + f^2y_0 = 0$, which passes through the two points (x_0, y_0) and $(0, y_0)$. The absolute value of the y-coordinate is smaller than $|y_0|$ for all points with x between 0 and x_0 and larger otherwise. The hyperbola coincides with the x-axis if the FOE is the image center.

The lines separating the rotational components in the α- and β-patterns are defined by the equations $y = \frac{\beta f}{\gamma}$ and $x = \frac{\alpha f}{\gamma}$. If γ is small in relation to α and β or even zero, the line will be found outside the image plane and the rotational contributions to the α- and β-vectors will be either all positive or all negative.

For the α-pattern this means that if we add a large enough positive rotation around the Y-axis, all α-vectors will have positive rotational values. On the other hand, since during the rotation the local coordinate system changes, the x-coordinate of the FOE will become larger (move to the right in the image plane). The translational component of the pattern, which corresponds to the motion along the Z-axis, consists of a positive left half-plane and a negative right half-plane (see Figure 6.3a,b).

The translational components of the β-vectors, when the FOE is at the center, are positive in the lower half-plane and negative in the upper half-plane. A rotation with a

large enough negative α-component causes all rotational components of the β-vectors to be positive and a positive β causes a negative rotational contribution (see Figure 6.3c,d). In order to bring the FOE to the center, the observer pursues the following strategy. It

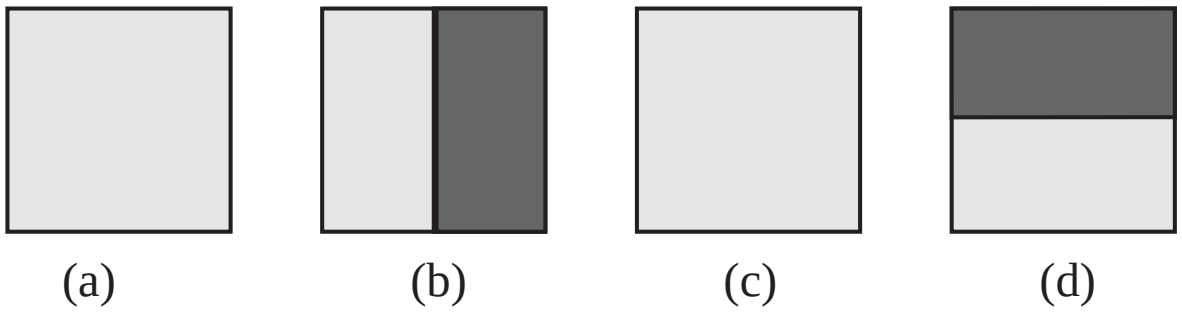


Figure 6.3: Rotational (a, c) and translational (b, d) patterns to be obtained if positive rotation around the Y-axis and negative rotation around the X-axis are added.

starts by changing its motion by a high positive rotation around the Y-axis and a negative rotation around the X-axis. In this way it adds a purely positive rotational field and at the same time changes the location of the FOE. This change occurs not because the actual direction of translation changes, but due to the fact that the FOE takes on a new value in a new local coordinate system. The observer continues rotating and continuously makes measurements. When the right half-plane of the α-pattern and the lower half-plane of the β-pattern become completely positive, it reverses its rotational components. It changes to a negative rotation around the Y-axis and a positive rotation around the X-axis, and thus makes the rotational contributions to all the investigated coaxis vectors negative. The rotation is performed until the left half-plane of the α-vectors and the upper half-plane of the β-vectors become negative. The final goal is to have the right and the lower half-plane positive at one step, and the left and the upper half-plane negative at the next step, where the camera should be looking at both measurements in the same direction. This means that compensation for the additional rotation has to be considered. If this is achieved, we have brought the FOE to the center of the coordinate system. Clearly the changes cannot be performed abruptly; a smart control strategy has to be developed to cope with the inertia of the system.

Chapter 7

Conclusions

The realization that vision is not a cognitive modality which exists in isolation, but is part of a larger system in interaction with its environment, opens new avenues for the study of problems involving visual processing.

To begin with, from a mathematical point of view, as has been shown throughout this thesis, activities of the observer introduce additional constraints which reduce the dimensionality of the problem space.

More important, the problems that are relevant to an observer/actor are different in nature [Aloimonos, 1990] from the problems defined in passive, reconstructive vision [Marr, 1982]. This viewpoint was elaborated for the problem of motion perception, which in the past was studied within the framework of the general structure-from-motion module. Consequently, egomotion perception and 3D-object motion perception were considered to be the same problem, whereas in this thesis these two problems are studied as perceptually different.

These ideas translate into a philosophy for building vision systems, the synthetic approach. This means that we should first develop basic capabilities and then equip the system with more and more memory and study capabilities of increasing complexity, where the complexity of a capability is determined by the complexity of the computational model in use. Two basic capabilities (egomotion estimation and the estimation of translational object motion) have been developed in a robust way. These two capabilities can form a basis of a system from which other capabilities can be developed. These capabilities include independent motion detection by a moving observer, landing or docking, target pursuit, and various manipulation tasks.

To summarize the technical results of the thesis, the problem of motion recovery has

usually been treated by using as input local image motion, with the published algorithms utilizing the geometric constraint relating 2D local image motion (optical flow, correspondence, derivatives of the image flow) to 3D motion and structure. Since it has proved very difficult to achieve accurate input (local image motion), a lot of effort has been devoted to the development of robust techniques. In this thesis a new approach to the problem of egomotion estimation is taken, which is based on constraints of a global nature. It has been shown that normal flow measurements form global patterns in the image plane. The positions of these patterns are related to the three dimensional motion parameters. By locating some of these patterns, which depend on only subsets of the motion parameters, using a simple search technique, the 3D motion parameters are found. The proposed algorithmic procedure is very robust, since it is not affected by small perturbations in the local image measurements (normal flow). As a matter of fact, since only the signs of the image measurements are employed, the direction of translation and the axis of rotation can be estimated in the presence of up to 100 percent error in the image measurements. If the observer is active and supplies additional information, the general constraints can be exploited to reduce the dimensionality of the parameter space in which the search is performed. Thus various navigational problems can be solved with only a small computational effort. If, for example, the observer possesses tracking and fixation capabilities, the egomotion estimation problem is reduced in dimensionality from 5 to 1.

In contrast to egomotion estimation where a camera-centered coordinate system is used, for object motion estimation an object-centered coordinate system is more appropriate. The main difference between the two techniques described in the thesis lies in the fact that egomotion estimation is based on global data while object motion is computed from local data. It has been argued by psychologists that biological organisms use tracking in the motion estimation process. In this thesis the advantages of tracking have been exploited to solve, for a monocular observer, the problems of computing a moving object's translational direction and time to collision. A complete solution to this problem was presented by showing how tracking can be performed when only normal flow measurements are used and how these parameters are of use in the 3D motion parameter decoding strategy.

The theoretical analysis and the experiments described demonstrate that the algorithms introduced here have the potential of being implemented in real hardware active vision systems, such as the ones described in [Ballard and Brown, 1992; Pahlavan and Eklundh, 1992].

Bibliography

[Adelson and Bergen, 1985] E.H. Adelson and J.R. Bergen. Spatiotemporal energy models for the perception of motion. *Journal of the Optical Society of America*, 2:284–299, 1985.

[Adiv, 1985] G. Adiv. Determining 3D motion and structure from optical flow generated by several moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:384–401, 1985.

[Aisbett, 1990] J. Aisbett. An iterated estimation of the motion parameters of a rigid body from noisy displacement vectors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:1092–1098, 1990.

[Aloimonos and Brown, 1984] J. Aloimonos and C.M. Brown. Direct processing of curvilinear sensor motion from a sequence of perspective images. In *Proc. Workshop on Computer Vision: Representation and Control*, pages 72–77, 1984.

[Aloimonos and Brown, 1989] J. Aloimonos and C.M. Brown. On the kinetic depth effect. *Biological Cybernetics*, 60:445–455, 1989.

[Aloimonos and Shulman, 1989] J. Aloimonos and D. Shulman. *Integration of Visual Modules: An Extension of the Marr Paradigm.* Academic Press, Boston, 1989.

[Aloimonos *et al.*, 1988] J. Aloimonos, I. Weiss, and A. Bandopadhay. Active vision. *International Journal of Computer Vision*, 2:333–356, 1988.

[Aloimonos, 1990] J. (Y.) Aloimonos. Purposive and qualitative active vision. In *Proc. DARPA Image Understanding Workshop*, pages 816–828, 1990.

[Anandan and Weiss, 1985] P. Anandan and R. Weiss. Introducing a smoothness constraint in a matching approach for the computation of optical flow fields. In *Proc. 3rd Workshop on Computer Vision: Representation and Control*, pages 186–194, 1985.

[Anandan, 1989] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1989.

[Bajcsy, 1988] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76:996–1005, 1988.

[Ballard and Brown, 1982] D.H. Ballard and C.M. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, New Jersey, 1982.

[Ballard and Brown, 1992] D.H. Ballard and C.M. Brown. Principles of animate vision. *CVGIP: Image Understanding: Special issue on Purposive, Qualitative, Active Vision*, Y. Aloimonos (Ed.), 56:3–21, 1992.

[Ballard, 1991] D.H. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[Bandopadhay and Ballard, 1991] A. Bandopadhay and D.H. Ballard. Egomotion perception using visual tracking. *Computational Intelligence*, 7:39–47, 1991.

[Barnard and Thompson, 1980] S.T. Barnard and W.B. Thompson. Disparity analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2:333–340, 1980.

[Barnes, 1983] J. Barnes. *The Presocratic Philosophers*, volume 2. Routledge, Chapman and Hall, New York, 1983.

[Bergholm, 1988] F. Bergholm. Motion from flow along contours: A note on robustness and ambiguous cases. *International Journal of Computer Vision*, 3:395–415, 1988.

[Brooks, 1986] R.A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2:14–23, 1986.

[Bruss and Horn, 1983] A. Bruss and B.K.P. Horn. Passive navigation. *Computer Vision, Graphics, and Image Processing*, 21:3–20, 1983.

[Burger and Bhanu, 1990] W. Burger and B. Bhanu. Estimating 3-D egomotion from perspective image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:1040–1058, 1990.

[Burt *et al.*, 1983] P.J. Burt, C. Yen, and X. Xu. Multi-resolution flow through motion analysis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 246–252, 1983.

[Burt *et al.*, 1989] P.J. Burt, J.R. Bergen, R. Hingorani, R. Kolczynski, W.A. Lee, A. Leung, J. Lubin, and H. Shvaytser. Object tracking with a moving camera. In *Proc. IEEE Workshop on Visual Motion*, pages 2–12, 1989.

[Buxton and Buxton, 1984] B.F. Buxton and H. Buxton. Computation of optic flow from the motion of edge features in image sequences. *Image and Vision Computing*, 2:59–75, 1984.

[Chandrashekar and Chellappa, 1991] S. Chandrashekar and R. Chellappa. Passive navigation in a partially known environment. In *Proc. IEEE Workshop on Visual Motion*, pages 2–7, 1991.

[Chen and Penna, 1986] S. Chen and A. Penna. Shape and motion of nonrigid bodies. *Computer Vision, Graphics, and Image Processing*, 36:175–207, 1986.

[Cheng and Aggarwal, 1990] C.-J. Cheng and J.K. Aggarwal. A two-stage hybrid approach to the correspondence problem via forward-searching and backward-correcting. In *Proc. International Conference on Pattern Recognition*, pages A 173–179, 1990.

[Crowley and Stelmaszyk, 1990] J. Crowley and P. Stelmaszyk. Measurement and integration of 3-D structures by tracking edge lines. In *Proc. First European Conference on Computer Vision*, pages 269–280, 1990.

[Cui *et al.*, 1991] N. Cui, J. Weng, and P. Cohen. Motion and structure from long stereo image sequences. In *Proc. IEEE Workshop on Visual Motion*, pages 75–80, 1991.

[Daniilidis and Nagel, 1990] K. Daniilidis and H.H. Nagel. Analytical results on error sensitivity of motion estimation from two views. *Image and Vision Computing*, 8:297–303, 1990.

[Descartes, 1978] R. Descartes. *Meditations and Passions*, volume 1 & 2 of *The Philosophical Works of Descartes*. E. Haldane and G. Ross (Eds.), Cambridge University Press, Cambridge, 1978.

[Dickmanns and Graefe, 1988a] E.D. Dickmanns and V. Graefe. Applications of dynamic monocular machine vision. *Machine Vision and Applications*, 1:241–261, 1988.

[Dickmanns and Graefe, 1988b] E.D. Dickmanns and V. Graefe. Dynamic monocular machine vision. *Machine Vision and Applications*, 1:223–240, 1988.

[Duda and Hart, 1962] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1962.

[Edelman, 1989] G. Edelman. *Bright Air, Brilliant Fire.* On the Matter of the Mind. Basic Books, 1989.

[Enkelmann, 1988] W. Enkelmann. Investigations of multigrid algorithms for estimation of optical flow fields in image sequences. *Computer Vision, Graphics, and Image Processing*, 43:150–177, 1988.

[Ernst and Newell, 1969] G.W. Ernst and A. Newell. *GPS: A Case Study in Generality and Problem Solving.* Academic Press, New York, 1969.

[Fang and Huang, 1984] J.Q. Fang and T.S. Huang. Some experiments on estimating the 3-D motion parameters of a rigid body from two consecutive image frames. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:547–554, 1984.

[Farah, 1990] M. Farah. *Visual Agnosia: Disorders of Object Recognition and What They Tell us about Normal Vision.* MIT Press, Cambridge, MA, 1990.

[Faugeras and Maybank, 1990] O.D. Faugeras and S. Maybank. Motion from point matches: Multiplicity of solutions. *International Journal of Computer Vision*, 4:225–246, 1990.

[Faugeras *et al.*, 1987] O.D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from motion from point and line matches. In *Proc. International Conference on Computer Vision*, pages 25–34, 1987.

[Fermüller and Aloimonos, 1992] C. Fermüller and Y. Aloimonos. Tracking facilitates 3-D motion estimation. *Biological Cybernetics*, 67:259–268, 1992.

[Fermüller and Aloimonos, 1993a] C. Fermüller and Y. Aloimonos. Recognizing 3-D motion. In *Proc. International Joint Conference on Artificial Intelligence*, 1993.

[Fermüller and Aloimonos, 1993b] C. Fermüller and Y. Aloimonos. The role of fixation in visual motion analysis. *International Journal of Computer Vision: Special issue on Active Vision*, M. Swain (Ed.), 1993.

[Fermüller and Kropatsch, 1992] C. Fermüller and W. Kropatsch. Multi-resolution shape description by corners. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 271–276, 1992.

[Fermüller, 1993a] C. Fermüller. Global 3-D motion estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, June 1993.

[Fermüller, 1993b] C. Fermüller. Motion constraint patterns. In *Proc. IEEE Workshop on Qualitative Vision*, June 1993.

[Fermüller, 1993c] C. Fermüller. Navigational preliminaries. In Y. Aloimonos, editor, *Active Vision*, Advances in Computer Vision. Lawrence Erlbaum, Hillsdale, NJ, 1993.

[Fermüller, 1993d] C. Fermüller. Qualitative interpretation of image derivatives. *International Journal of Computer Vision: Special issue on Qualitative Vision*, 1993.

[Fleet and Jepson, 1990] D.J. Fleet and A.D. Jepson. Computation of component velocity from local phase information. *International Journal of Computer Vision*, 5:77–104, 1990.

[Gelernter, 1959] H. Gelernter. Realization of a geometry theorem-proving machine. In *Information Processing: Proceedings of the International Conference on Information Processing*, UNESCO, 1959.

[Geman and Geman, 1984] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution and Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[Gibson, 1950] J.J. Gibson. *The Perception of the Visual World.* Houghton Mifflin, Boston, 1950.

[Goldgof *et al.*, 1988] D. Goldgof, H. Lee, and T.S. Huang. Motion analysis of nonrigid surfaces. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 375–380, 1988.

[Gregory, 1963] R.L. Gregory. Distortion of visual space as inappropriate constancy scaling. *Nature*, 119:678, 1963.

[Gregory, 1970] R.L. Gregory. *The Intelligent Eye.* McGraw-Hill, New York, 1970.

[Heeger, 1988] D. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1:279–302, 1988.

[Helmholtz, 1896] H. von Helmholtz. *Handbuch der Physiologischen Optik.* Leopold Voss, 1896.

[Hildreth, 1984] E. Hildreth. Computations underlying the measurement of visual motion. *Artificial Intelligence*, 23:309–354, 1984.

[Hoffman, 1982] D.D. Hoffman. Inferring local surface orientation from motion fields. *Journal of the Optical Society of America*, 72:880–892, 1982.

[Horn and Schunck, 1981] B.K.P. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[Horn and Weldon, 1987] B.K.P. Horn and E.J. Weldon. Computationally efficient methods for recovering translational motion. In *Proc. International Conference on Computer Vision*, pages 2–11, 1987.

[Horn, 1986] B.K.P. Horn. *Robot Vision*. McGraw Hill, New York, 1986.

[Horn, 1987] B.K.P. Horn. Motion fields are hardly ever ambiguous. *International Journal of Computer Vision*, 1:259–274, 1987.

[Horn, 1990] B.K.P. Horn. Relative orientation. *International Journal of Computer Vision*, 4:59–78, 1990.

[Horridge, 1987] G.A. Horridge. The evolution of visual processing and the construction of seeing systems. *Proceedings of the Royal Society, London B*, 230:279–292, 1987.

[Horridge, 1991] G.A. Horridge. Evolution of visual processing. In J.R. Cronly-Dillon and R.L. Gregory, editors, *Vision and Visual Dysfunction*. MacMillan, New York, 1991.

[Jacobs, 1992] D.W. Jacobs. Space efficient 3D model indexing. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 439–444, 1992.

[Jain, 1984] R. Jain. Segmentation of frame sequences obtained by a moving observer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:624–629, 1984.

[Jenkin and Kolers, 1986] M. Jenkin and P.A. Kolers. Some problems with correspondence. Technical Report RBCV-TR-86-10, University of Toronto, Toronto, Ontario, Canada, 1986.

[Johansson, 1973] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception and Psychophysics*, 14:201–211, 1973.

[Kanatani, 1990] K. Kanatani. *Group-Theoretical Methods in Image Understanding*. Springer-Verlag, Berlin, 1990.

[Kanizsa, 1979] G. Kanizsa. *Organization in Vision: Essays on Gestalt Perception*. Praeger, New York, 1979.

[Koenderink and van Doorn, 1975] J.J. Koenderink and A.J. van Doorn. Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta*, 22:773–791, 1975.

[Koenderink and van Doorn, 1976] J.J. Koenderink and A.J. van Doorn. Local structure of movement parallax of the plane. *Journal of the Optical Society of America*, 66:717–723, 1976.

[Koenderink and van Doorn, 1991] J.J. Koenderink and A.J. van Doorn. Affine structure from motion. *Journal of the Optical Society of America*, 8:377–385, 1991.

[Koenderink, 1984] J.J. Koenderink. Shape from motion and bending deformation. *Journal of the Optical Society of America*, A-1:1265–1266, 1984.

[Kruppa, 1913] E. Kruppa. Zur Ermittlung eines Objekts aus Zwei Perspektiven mit Innerer Orientierung. *Abhandlungen der Akademie der Wissenschaften, Wien*, 122:1939–1948, 1913.

[Lakoff, 1987] G. Lakoff. *Women, Fire and Dangerous Things: What Categories Reveal about the Mind.* University of Chicago Press, Chicago, IL, 1987.

[Lawton, 1983] T. Lawton. Processing translational motion sequences. *Computer Vision, Graphics, and Image Processing*, 22:116–144, 1983.

[Legters Jr. and Young, 1982] G.R. Legters Jr. and T.Y. Young. A mathematical model for computer image tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4:583–594, 1982.

[Liu and Huang, 1988] Y. Liu and T.S. Huang. Estimation of rigid body motion using straight line correspondences. *Computer Vision, Graphics, and Image Processing*, 43:37–52, 1988.

[Longuet-Higgins and Prazdny, 1980] H. C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proceedings of the Royal Society, London B*, 208:385–397, 1980.

[Longuet-Higgins, 1981] H.C. Longuet-Higgins. A computer algorithm for reconstruction of a scene from two projections. *Nature*, 293:133–135, 1981.

[Marr, 1982] D. Marr. *Vision.* W.H. Freeman, San Francisco, 1982.

[Martin and Aggarwal, 1978] W.N. Martin and J.K. Aggarwal. Dynamic scene analysis: A survey. *Computer Vision, Graphics, and Image Processing*, 7:356–374, 1978.

[Matthies *et al.*, 1989] L. Matthies, T. Kanade, and S.A. Shafer. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3:209–236, 1989.

[Moravec, 1977] H. Moravec. Towards automatic visual obstacle avoidance. In *Proc. International Joint Conference on Artificial Intelligence*, pages 584–585, 1977.

[Moses, 1976] J. Moses. The current capabilities of the MACSYMA system. In *Proc. ACM National Conference*, October 1976.

[Murray and Buxton, 1984] D.W. Murray and B.F. Buxton. Reconstructing the optic flow from edge motion: An examination of two different approaches. In *Proc. First Conference on AI Applications*, 1984.

[Nagel and Enkelmann, 1986] H.H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:565–593, 1986.

[Nagel, 1983] H.H. Nagel. Displacement vectors derived from second order intensity variations in image sequences. *Computer Vision, Graphics, and Image Processing*, 21:85–117, 1983.

[Nagel, 1987] H.H. Nagel. On the estimation of optical flow: Relations between different approaches and some new results. *Artificial Intelligence*, 33:459–483, 1987.

[Negahdaripour and Horn, 1987] S. Negahdaripour and B.K.P. Horn. Direct passive navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:163–176, 1987.

[Negahdaripour, 1986] S. Negahdaripour. *Direct Passive Navigation.* PhD thesis, Department of Mechanical Engineering, MIT, Cambridge, MA, 1986.

[Negahdaripour, 1989] S. Negahdaripour. Critical surface pairs and triplets. *International Journal of Computer Vision*, 3:293–312, 1989.

[Nelson and Aloimonos, 1988] R.C. Nelson and J. Aloimonos. Finding motion parameters from spherical flow fields (or the advantage of having eyes in the back of your head). *Biological Cybernetics*, 58:261–273, 1988.

[Nelson, 1991] R.C. Nelson. Qualitative detection of motion by a moving observer. *International Journal of Computer Vision*, 7:33–46, 1991.

[Pahlavan and Eklundh, 1992] K. Pahlavan and J.-O. Eklundh. A head-eye system—analysis and design. *CVGIP: Image Understanding: Special issue on Purposive, Qualitative, Active Vision*, Y. Aloimonos (Ed.), 56:41–56, 1992.

[Penna, 1992] M.A. Penna. Non-rigid motion analysis: Isometric motion. *CVGIP: Image Understanding*, 56:366–380, 1992.

[Pentland *et al.*, 1991] A. Pentland, B. Horowitz, and S. Sclaroff. Non-rigid motion and structure from contour. In *Proc. IEEE Workshop on Visual Motion*, pages 288–293, 1991.

[Pentland, 1986] A. Pentland, editor. *From Pixels to Predicates: Recent Advances in Computational and Robot Vision.* Ablex, Norwood, NJ, 1986.

[Philip, 1991] J. Philip. Estimation of three-dimensional motion of rigid objects from noisy observations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:61–66, 1991.

[Poggio *et al.*, 1984] T. Poggio, H. Voorhees, and A. Yuille. Regularizing edge detection. AI Memo 776, M.I.T. Artificial Intelligence Laboratory, Cambridge, MA, 1984.

[Prazdny, 1980] K. Prazdny. Egomotion and relative depth map from optical flow. *Biological Cybernetics*, 36:87–102, 1980.

[Prazdny, 1981] K. Prazdny. Determining instantaneous direction of motion from optical flow generated by a curvilinear moving observer. *Computer Vision, Graphics, and Image Processing*, 17:238–248, 1981.

[Ranade and Rosenfeld, 1980] S. Ranade and A. Rosenfeld. Point pattern matching by relaxation. *Pattern Recognition*, 12:269–275, 1980.

[Roach and Aggarwal, 1980] J.W. Roach and J.K. Aggarwal. Determining the movements of objects from a sequence of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2:554–562, 1980.

[Rosenblatt, 1962] F. Rosenblatt. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms.* Spartan, Washington, D.C., 1962.

[Sethi and Jain, 1987] I.K. Sethi and R. Jain. Finding trajectories of feature points in a monocular image sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:56–73, 1987.

[Sharma and Aloimonos, 1991] R. Sharma and Y. Aloimonos. Robust detection of independent motion: An active and purposive solution. Technical Report CAR-TR-534, Center for Automation Research, University of Maryland, College Park, MD, 1991.

[Shulman and Aloimonos, 1988] D. Shulman and J.Y. Aloimonos. (Non-)rigid motion interpretation: a regularized approach. *Proceedings of the Royal Society London B*, 233:217–234, 1988.

[Shulman and Hervé, 1989] D. Shulman and J-Y. Hervé. Regularization of discontinuous flow fields. In *Proc. IEEE Workshop on Visual Motion*, pages 81–86, 1989.

[Shulman, 1990] D. Shulman. *Regularization of Inverse Problems in Low-level Vision while Preserving Discontinuities.* PhD thesis, Center for Automation Research, University of Maryland, College Park, MD, 1990.

[Singh, 1990] A. Singh. *Optic Flow Computation: A Unified Perspective.* PhD thesis, Department of Computer Science, Columbia University, New York, NY, 1990.

[Sloman, 1989] A. Sloman. On designing a visual system. *Journal of Experimental and Theoretical Artificial Intelligence*, 1:289–337, 1989.

[Snyder, 1989] M. Snyder. On the mathematical foundations of smoothness constraints for the determination of optical flow and for surface reconstruction. In *Proc. IEEE Workshop on Visual Motion*, pages 107–115, 1989.

[Spetsakis and Aloimonos, 1988] M.E. Spetsakis and J. Aloimonos. Optimal computing of structure from motion using point correspondence. In *Proc. International Conference on Computer Vision*, pages 449–453, 1988.

[Spetsakis and Aloimonos, 1989] M.E. Spetsakis and J. Aloimonos. Optimal motion estimation. In *Proc. IEEE Workshop on Visual Motion*, pages 229–237, 1989.

[Spetsakis and Aloimonos, 1990] M. Spetsakis and J. Aloimonos. Structure from motion using line correspondences. *International Journal of Computer Vision*, 1:171–183, 1990.

[Spetsakis and Aloimonos, 1991] M. Spetsakis and J. Aloimonos. A multi-frame approach to visual motion perception. *International Journal of Computer Vision*, 6:245–255, 1991.

[Subbarao, 1988] M. Subbarao. *Interpretation of Visual Motion.* PhD thesis, Center for Automation Research, University of Maryland, College Park, MD, 1988.

[Taalebi-Nezhaad, 1990] M.A. Taalebi-Nezhaad. Direct recovery of motion and shape in the general case by fixation. In *Proc. DARPA Image Understanding Workshop*, pages 284–291, 1990.

[Terzopoulos *et al.*, 1988] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on deformable models: Recovering 3-D shape and nonrigid motion. *Artificial Intelligence*, 36:91–123, 1988.

[Thompson and Pong, 1990] W.B. Thompson and T.-C. Pong. Detecting moving objects. *International Journal of Computer Vision*, 4:39–57, 1990.

[Thompson *et al.*, 1984] W.B. Thompson, K.M. Mutch, and V.A. Berzins. Analyzing object motion based optical flow. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 791–794, 1984.

[Tistarelli and Sandini, 1992] M. Tistarelli and G. Sandini. Dynamic aspects in active vision. *CVGIP: Image Understanding: Special issue on Purposive, Qualitative, Active Vision*, Y. Aloimonos (Ed.), 56:108–129, 1992.

[Tomasi and Kanade, 1991] C. Tomasi and T. Kanade. Factoring image sequences into shape and motion. In *Proc. IEEE Workshop on Visual Motion*, pages 21–28, 1991.

[Tsai and Huang, 1984] R.Y. Tsai and T.S. Huang. Uniqueness and estimation of three–dimensional motion parameters of rigid objects with curved surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:13–27, 1984.

[Ullman and Basri, 1991] S. Ullman and R. Basri. Recognition by linear combination of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:992–1006, 1991.

[Ullman, 1979] S. Ullman. The interpretation of structure from motion. *Proceedings of the Royal Society, London*, B 203:405–426, 1979.

[Ullman, 1983] S. Ullman. Maximizing rigidity: The incremental recovery of 3-D structure from rigid and rubbery motion. AI Memo 721, M.I.T. Artificial Intelligence Laboratory, Cambridge, MA, 1983.

[Uras *et al.*, 1988] S. Uras, F. Girosi, and V. Torre. A computational approach to motion perception. *Biological Cybernetics*, 60:79–87, 1988.

[Verri and Poggio, 1989] A. Verri and T. Poggio. Motion field and optical flow: Qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:490–498, 1989.

[Wallach and O'Connell, 1953] H. Wallach and D.N. O'Connell. The kinetic depth effect. *Journal of Experimental Psychology*, 45:205–217, 1953.

[Watson and Ahumada, 1985] A.B. Watson and A.J. Ahumada. Model of human visual motion sensing. *Journal of the Optical Society of America*, 2:322–342, 1985.

[Waxman and Wohn, 1985] A.M. Waxman and K. Wohn. Contour evolution, neighborhood deformation and global image flow. *International Journal of Robotics Research*, 4:95–108, 1985.

[Waxman *et al.*, 1987a] A.M. Waxman, J. LeMoigne, L. Davis, E. Liang, and T. Siddalingaiah. A visual navigation system for autonomous land vehicles. *IEEE Journal of Robotics and Automation*, 3:124–141, 1987.

[Waxman *et al.*, 1987b] A.M. Waxman, B. Kamgar-Parsi, and M. Subbarao. Closed–form solutions to image flow equations for 3D structure and motion. *International Journal of Computer Vision*, 1:239–258, 1987.

[Waxman *et al.*, 1988] A.M. Waxman, J. Wu, and F. Bergholm. Convected activation profiles and measurement of visual motion. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 717–722, 1988.

[Weng *et al.*, 1987a] J. Weng, N. Ahuja, and T.S. Huang. Two-view matching. In *Proc. IEEE Computer Society Workshop on Computer Vision*, 1987.

[Weng *et al.*, 1987b] J. Weng, T.S. Huang, and N. Ahuja. A two step approach to optimal motion and structure estimation. In *Proc. IEEE Computer Society Workshop on Computer Vision*, 1987.

[White and Weldon, 1988] G. White and E.J. Weldon. Utilizing gradient vector distributions to recover motion parameters. In *Proc. International Conference on Computer Vision*, pages 64–73, 1988.

[Wittgenstein, 1953] L. Wittgenstein. *Philosophical Investigations.* MacMillan, New York, 1953.

[Wolf, 1983] P.R. Wolf. *Elements of Photogrammetry.* McGraw-Hill, New York, 1983.

[Wong and Hall, 1978] R.Y. Wong and E.L. Hall. Sequential hierarchical scene matching. *IEEE Transactions on Computers*, 27:359–366, 1978.

[Young and Chellappa, 1990] G-S.J. Young and R. Chellappa. 3-D motion estimation using a sequence of noisy stereo images: Models, estimation, and uniqueness results. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:735–759, 1990.