Dynamic Stereo Vision

Larry Matthies

October 1989

CMU-CS-89-195

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

This research was supported by the Defense Advanced Research Projects Agency, monitored by the Air Force Avionics Lab under contract F33615-87-C-1499, by the Office of Naval Research under contract N00014-81-K-0503, by the National Sciences and Engineering Research Council of Canada, and by the FMC Corporation. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the funding agencies.

Abstract

Sensing 3-D shape and motion is an important problem in autonomous navigation and manipulation. Stereo vision is an attractive approach to this problem in several domains. We address fundamental components of this problem by using stereo vision to estimate the 3-D structure or "depth" of objects visible to a robot, as well as to estimate the motion of the robot as it travels through an unknown environment.

We begin by using cameras on-board a robot vehicle to estimate the motion of the vehicle by tracking 3-D feature-points or "landmarks". We formulate this task as a statistical estimation problem, develop sequential methods for estimating the vehicle motion and updating the landmark model, and implement a system that successfully tracks landmarks through stereo image sequences. In laboratory experiments, this system has achieved an accuracy of 2% of distance over 5.5 meters and 55 stereo image pairs. These results establish the importance of statistical modelling in this problem and demonstrate the feasibility of visual motion estimation in unknown environments.

This work embodies a successful paradigm for feature-based depth and motion estimation, but the feature-based approach results in a very limited 3-D model of the environment. To extend this aspect of the system, we address the problem of estimating "depth maps" from stereo images. Depth maps specify scene depth for each pixel in the image. We propose a system architecture in which exploratory camera motion is used to acquire a narrow-baseline image pair by moving one camera of the stereo system. Depth estimates obtained from this image pair are used to "bootstrap" matching of a wide-baseline image pair acquired with both cameras of the stereo system. We formulate the bootstrap operation statistically by modelling depth maps as random fields and developing Bayesian matching algorithms in which depth information from the narrow-baseline image pair forms the prior density for matching the wide baseline image pair. This leads to efficient, area-based matching algorithms that are applied independently for each pixel or each scanline of the image. Experimental results with scale models of complex, outdoor scenes demonstrate the power of the approach.

Acknowledgements

The longer you work, the more people you have to thank. Hans Moravec played the most important role by starting me in this direction, influencing my experimental approach, bringing to my attention the statistical methods I've found so useful, and creating the laboratory used for the first half of this thesis. Steve Shafer and Takeo Kanade created the laboratory and computational facilities used in the balance of the thesis and made helpful observations at many stages of the work. Many people contributed to both the ideas and the software developed here. Chuck Thorpe contributed to the feature-tracking system described in chapter 2. Peter Highnam found Schonemann's "Procrustes" algorithm and provided code for the singular value decomposition. My views on representations, sensor fusion, and the use of probabilistic models were influenced by discussions and collaborations with Alberto Elfes, Radu Jasinschi, and Rick Szeliski. A variety of contributions to the ideas here, as well as to my general technical outlook, were made by Bruce Lucas, Bruce Krogh, Richard Stern, Wolfgang Forstner, Ernst Dickmanns, Carlo Tomasi, Ken Goldberg, and Jim Rehg. People instrumental in designing, building, and maintaining the robots, computers, and software systems I used include Greg Podnar, Kevin Dowling, Mike Blackwell, Leonard Hamey, Ralph Hyre, and Jim Moody. Finally, to the many people whose friendship I've enjoyed at CMU: thanks, and let's do it again — but not in grad school.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Autonomous robots are systems that can perform navigation and manipulation tasks without human intervention. Such systems are required for tasks that are too expensive, too hazardous, or too inaccessible for us to perform ourselves. Examples include repetitive manufacturing operations, handling hazardous materials, and exploring the oceans or other planets. To perform the full range of these tasks, robots must be able to sense their environments, build internal models of those environments, and construct and execute plans for achieving their goals. Moreover, these operations must be performed in a dynamic world in which both the robot and other objects move and change shape over time. We apply stereo vision to this sensing problem by developing algorithms for estimating the 3-D structure or *depth* of objects visible to a robot, as well as for estimating the *motion* of a robot as it travels through an unknown environment. These capabilities are the first milestones on the road to stereo systems that can estimate more general models of depth and motion; that is, toward systems for *dynamic* stereo vision.

We address two specific problems. In the first, a robot vehicle travels through an unknown environment and periodically uses on-board cameras to acquire a stereo image pair. The problem is to use this stereo image sequence to track 3-D point features, or *landmarks*, to estimate the motion of the vehicle. We model uncertainty in measured landmark coordinates with 3-D Gaussian distributions, develop statistical procedures to estimate the rotation and translation of the vehicle between successive stereo image pairs, and implement a system that tracks landmarks through the stereo image sequence. In laboratory experiments, this system has achieved an accuracy of 2% of distance over 5.5 meters and 55 stereo image pairs.

This work establishes a successful paradigm for feature-based depth and motion estimation. However, it embodies a very limited model of the 3-D structure of the enviroment. To extend this model, we develop methods for estimating *depth maps*, which specify depth at each pixel in the image. Our approach has two key elements. The first is the use of exploratory camera motions to reliably "bootstrap" matching between images. This involves estimating depth from a narrow-baseline image pair obtained by moving one camera, then using this depth information to constrain matching to a third, wide-baseline image acquired with the other camera. The second key element in our approach is its statistical formulation, which employs a random field model of the depth map and a Bayesian formulation of the matching problem. This formulation leads to

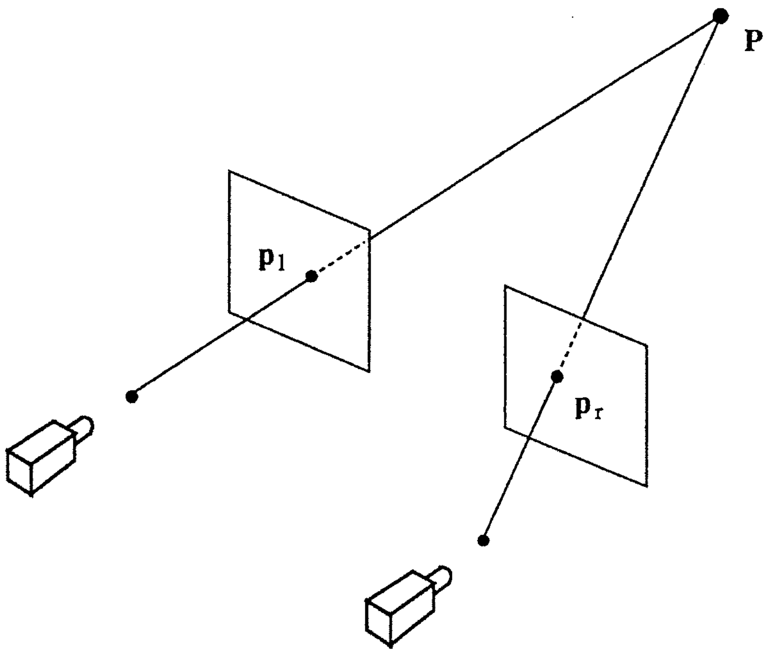


Figure 1.1: The basics of stereo triangulation: given *corresponding* image points, $\mathbf{p}_l$ and $\mathbf{p}_r$, and the positions and orientations of both cameras, we can compute the *depth* or 3-D location of the world point **P**.

simple, efficient matching algorithms based on correlation and dynamic programming. Results obtained with images of complex scenes demonstrate the power of the approach.

Before describing this work further, we will develop some background and motivation for the research.

1.1 Background and Motivation

Stereo vision is a triangulation-based range-finding technique in which two (or more) cameras are used to reconstruct the three-dimensional structure of a scene, as illustrated in figure 1.1. The fundamental computational problem in stereo is the *correspondence problem*, which requires finding *corresponding points* $\mathbf{p}_l$ and $\mathbf{p}_r$ in the two images. Given such points and the relative geometry of the cameras, it is a simple matter to compute the *depth*, or distance from the cameras, of the associated world point **P**. In principle, we can find the distance to every point in the scene by finding the corresponding point in the right image for every pixel in the left image. The resulting representation of scene depth at every pixel in the image, known as a *depth map*, is a starting point for computing a 3-D model of the scene. Relative motion between the cameras and the scene can be estimated by tracking the 3-D model through a time sequence of stereo images.

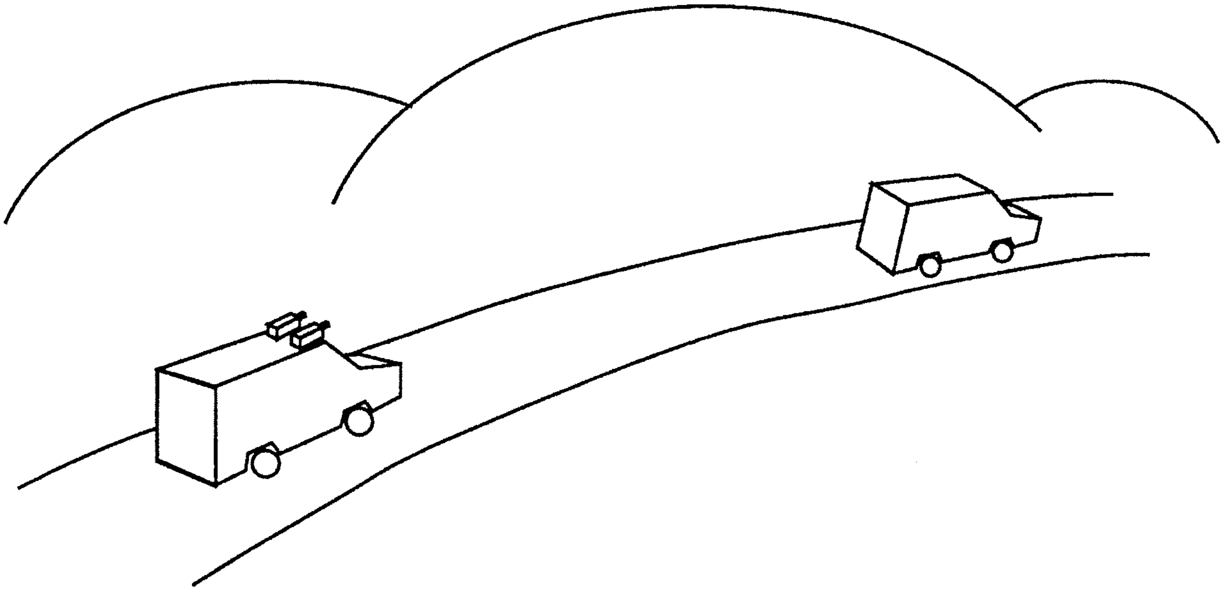


Figure 1.2: Application scenario: autonomous navigation

Interest in stereo arises from its roles in aerial surveying and in biological and robotic perception. We are concerned primarily with its role in robotics. In this case, we must ask why consider stereo at all, when there are devices like sonar, radar, and laser that measure depth more directly? There are a number of reasons for this, including the fact that stereo is:

- passive, because it doesn't project energy into the scene;
- non-scanning, because all pixels in a 2-D image can be acquired at the same point in time;
- non-mechanical, because it doesn't require mechanical scanning components, unlike laser range-finders for example;
- potentially low-power, compared to sensors that actively project energy.

These characteristics make stereo a candidate for robotic tasks requiring navigation, manipulation, or object recognition in civilian and military domains, including robotic operations in space.

As an example, in autonomous navigation a robot vehicle may use stereo cameras to survey terrain, detect obstacles, and estimate relative motions of other vehicles in the vicinity (figure 1.2). This introduces two, related series of problems, one concerning depth and one concerning motion (table 1.1). The depth problems require estimating:

- image-based depth models;
- 2-D and "2 1/2-D" world models;

Depth Estimation
Image-based depth models
2-D and “2 1/2-D” world models
3-D world models
Motion Estimation
Single rigid-body
Multiple rigid-body
Deformable body

Table 1.1: Problem hierarchies for stereo-based depth and motion estimation.

- 3-D world models.

Image-based depth models are representations of depth as a function of the image coordinates. These include the depth maps defined earlier, which specify depth at each pixel, and *feature-based* models that use sparse sets of points, line segments, or other geometric primitives defined in the image plane. 2-D world models are representations of the robot’s locality that are functions of two world coordinates, such as axes parallel to the ground plane. Examples include sets of polygons in one plane and 2-D volumetric models such as the spatial occupancy map [Elfes89]. These representations are useful for problems involving navigation in two dimensions, such as indoors. ”2 1/2-D” models are representations defined over two coordinates that express 3-D structure; terrain elevation maps [Hebert89] are a primary example. 3-D world models are defined over three spatial coordinates; they include boundary-based and volumetric object models in three dimensions [Requicha80]. In most work, an image-based model of some form is a necessary precursor to the various world models.

Motion problems add kinematics to the depth models. In order of increasing difficulty, these problems require estimating the motion(s) of:

- a single rigid body;
- multiple rigid bodies;
- multiple rigid or deformable bodies.

In the single rigid-body problem, the entire field of view is describable by a single rigid motion. This occurs when the robot vehicle travels through a static environment or when it observes moving objects against an imperceptible background. In the multiple rigid-body problem, two or more rigid motions are in view at once. This occurs when a single object moves against a perceivable background, when two or more objects move across the field of view, or when an

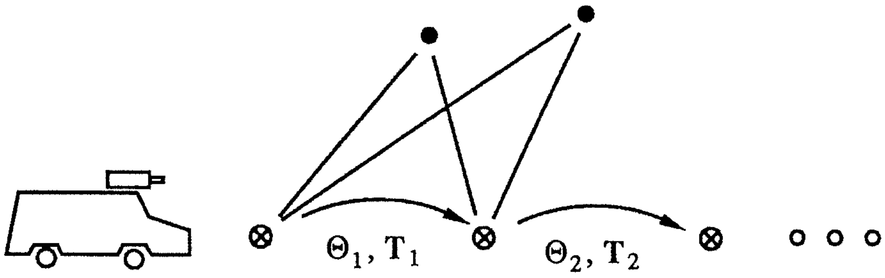


Figure 1.3: Estimating vehicle motion by tracking nearby landmarks

articulating object is in view. A robot vehicle operating in the presence of other vehicles is an example of this situation. This problem introduces the need to *segment* both the images and the world models into regions are that are consistent with distinct motions. In the final class of problems, the motions being observed may or may not be rigid. In addition to segmentation issues, this increases the complexity of modelling the motion and raises the issue of observability of the model.

We have illustrated these problems with the example of autonomous navigation, and the research we report here was largely motivated by this application. However, the problems themselves are common to many robotic tasks. Therefore, the two problem hierarchies also outline a general research agenda for robotic depth and motion estimation, using stereo or a variety of other sensors. To some degree, appropriate mathematical models for depth and motion already exist. However, the science of estimating such models from images still has many gaps. Techniques exist at all levels of the depth hierarchy, though not even the first level of depth map estimation has a complete mathematical and operational foundation leading to reliable performance in real systems. With motion, open issues remain for the single rigid-body case, while the problems of estimating multiple rigid and deformable motions are only beginning to be solved. Finally, the relatively recent idea of *controlling* motion to improve the estimation of depth, and consequently the motion itself, presents a host of new problems and opportunities [Aloimonos87,Geiger87].

1.2 Problems Addressed

In this research, we address two basic problems in visual depth and motion estimation. For each, we define the models to be estimated, develop a statistical formulation of the estimation problem, design sensing and estimation procedures that lead to reliable performance, and demonstrate this performance on real images of complex scenes.

The first problem is to estimate the rotation and translation of a robot vehicle as it travels through an unknown environment (figure 1.3). This is achieved by using on-board cameras to establish and track a 3-D world model consisting of point "landmarks" in the vicinity of the vehicle. In the language of the previous section, we use a feature-based depth model in estimating a model of single, rigid-body motion. The results of this work establish the importance of the statistical approach to this problem and give the first demonstration of accurate, reliable visual motion estimation in unknown environments. Extension to several more advanced motion problems appears to be fairly direct.

While this work is highly successful for motion estimation, the model of depth is very limited. A possible extension of the depth model is to use other features, such as edges or line segments. However, in complex environments, especially outdoors, these approaches appear unlikely to achieve robust performance or provide adequate representations of the environment. The same is true for related feature-based models, such as line junctions or curved segments. This prompts us to examine an alternate paradigm in which correlation-like operators are used to estimate depth "everywhere" in the image. In the discrete case, this implies using a "dense", pixel-based model of depth; that is, a depth map. The balance of our research considers how to formalize this paradigm and how to design a system to estimate depth maps reliably. This leads us to model depth maps as random fields and to develop Bayesian matching algorithms that use exploratory camera motions to "bootstrap" reliable stereo matching. Experimental results with images of complex scenes demonstrate that this approach is very successful. We conclude that both the dense depth paradigm and the use of exploratory camera motion are promising avenues for future research.

In the balance of this section, we review previous work on each of these problems and discuss our approaches and results in greater detail.

1.2.1 Motion Estimation

A great deal of effort has been expended in trying to estimate single rigid-body motion, especially from monocular image sequences. Successful estimation has been demonstrated with both monocular and stereo systems in contexts where the depth model is known in advance, in particular in mock-up demonstrations of satellite rendevous [Gennery86,Tietz82,Wunsche86]. When the depth model is not known in advance, monocular approaches are fundamentally limited because they cannot observe the absolute scale of the scene; moreover, the observability limitations become more severe in the context of more complex motion problems. Stereo does not suffer this limitation because it measures absolute depth. The first work to use stereo for motion estimation in unknown environments was the autonomous vehicle system developed by Moravec [Moravec80]. This system was designed to navigate to a pre-specified goal position. It used stereo first to create a world model consisting of 3-D point features (landmarks), then to track the landmarks to estimate the vehicle's position over time. The uncertainty models and the estimation algorithms used in this the system were relatively unsophisticated; nevertheless, promising performance was achieved.

The work here picks up where [Moravec80] leaves off. We develop a sequential, Bayesian formulation of the problem of estimating the rotation and translation of the vehicle between successive stereo image pairs. This formulation models uncertainty in the observed landmark coordinates with 3-D Gaussian distributions and sequentially updates estimates of the landmark coordinates to reflect the entire observation history. The image processing algorithms for creating and tracking the landmark model are refinements of algorithms developed in [Moravec80]. We simulate the performance of the estimation algorithms to demonstrate the importance of the statistical model. We also apply the entire system to sequences of images acquired by the vehicle to demonstrate the successful performance of the system on real images.

The principle contributions of this work are the development of the statistical model for this problem, the demonstration of the importance of this model in achieving reliable performance, and the demonstration of the feasibility of visual motion estimation in unknown environments. Extensions to more elaborate kinematic models and to other feature-based depth models are relatively direct; relevant work is described in [Ayache88,Dickmanns88,Gennery86,Young88]. The principle limitation lies in the feature-based depth model. Therefore, we address this issue next.

1.2.2 Depth Estimation

To develop systems that can operate effectively in complex environments, especially outdoors, we require a deeper understanding of how to estimate depth than that afforded by typical feature-based world models. Furthermore, it is not clear that we can define features that yield adequate depth information and that can be used robustly in a wide variety of domains. Therefore, we turn to depth map estimation to obtain a more general lowest level depth representation. As was the case for motion estimation, this problem has two components: (1) what is an appropriate formulation of the estimation problem, including its stochastic characteristics, and (2) how do we design a system to generate reliable estimates?

Statistical formulations of depth map estimation have taken two distinct approaches. The first approach, which is most common in photogrammetry, has focused on matching small image patches, modelled the noise in the images, and derived the error variance of the resulting disparity estimates [Forstner86,Forstner89,Gennery80]. In other words, such approaches use "area-based" or correlation-type matching operators and derive uncertainty in the depth estimates at each pixel; however, they do not explicitly model joint uncertainty in the depth estimates or in prior depth information. The second approach, which was introduced to computer vision in [Marroquin85,Marroquin87], deals with exactly the issue of joint uncertainty by modelling the depth map as a Markov random field (MRF). In [Marroquin85], the MRF model was used as the basis of a Bayesian approach to computing an "optimal" estimate of the disparity field. However, in this approach the prior density was used only to impose heuristic smoothness constraints on the estimated disparity field and no model was developed for the uncertainty in the resulting depth estimates[1]. Another stochastic approach to stereo is described in [Barnard89]; however, the stochastic aspect of this algorithm is a search method, rather than a statistical formulation of the matching problem *per se*.

The approach taken here extends the statistical models used in photogrammetry and the random field approach of [Marroquin85]. Because images are noisy, estimates of disparity must

[1] Extensions of this framework to depth estimation from image sequences are described in [Matthies89,Szeliski88].

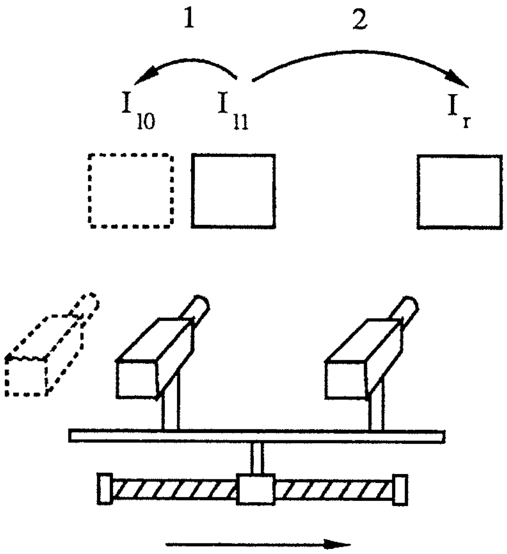


Figure 1.4: Camera motion and image acquisition for the bootstrap operation. Depth estimated by matching with images I_{l_0} and I_{l_1} is used to constrain matching between images I_{l_1} and I_r.

be noisy. Therefore, we model the disparity map as a random field with a joint Gaussian density. We define optimal disparity estimates with the *maximum a posterior probability* (MAP) criterion and develop area-based matching algorithms to estimate the posterior mean and variance of the disparity at each pixel in the image. Consideration of reliability leads us to propose a system design in which the stereo cameras are mounted on a precise translation stage, which in turn is mounted on the robot vehicle (see also [Geiger87]). Robust matching is achieved by using the translation stage to move the cameras a small fraction of the inter-camera baseline to acquire an image pair from one of the cameras. The relatively small baseline for this image pair assures reliable matching. The resulting depth map is used to constrain matching of a wide-baseline image pair obtained with both cameras (figure 1.4). We refer to this two-stage approach as a *bootstrap* operation. The matching algorithms are simple, efficient, and perform very well on images of complex scenes. Moreover, by using depth information from the narrow-baseline image pair to constrain matching in the wide-baseline image pair, the bootstrap operation employs knowledge about the scene itself to constrain matching, rather than the general smoothness heuristics employed by many wide-baseline stereo algorithms.

1.3 Thesis Overview

We address motion estimation in chapter 2. We begin by defining the landmark model, the coordinate frame conventions, and the rigid transformation equations that relate the coordinate frames of successive stereo pairs. We then formulate the problem of jointly estimating the 3-D coordinates of the landmarks and the rotations and translations of the vehicle between successive stereo pairs, given the image coordinates of the landmarks in each stereo pair. This problem is nonlinear and involves many unknowns, so we develop a solution in several stages. First, we compute 3-D observations of the landmark coordinates and model the uncertainty in these observations with 3-D, Gaussian probability densities. Then we derive least-squares, maximum-likelihood, and sequential Bayesian algorithms for estimating first the vehicle motion between frames, then both the vehicle motion and updated 3-D coordinates of the landmarks. These algorithms are embedded in a system that uses coarse-to-fine correlation to track landmarks through stereo image sequences. We use simulations to establish that motion estimates obtained with the full 3-D Gaussian uncertainty model are substantially superior to those obtained with a previous, simpler approach that used scalars to model landmark uncertainty. Finally, we do laboratory trials to establish the ability of the overall system to produce accurate motion estimates with stereo image sequences acquired by a real vehicle. We achieve an accuracy of 2% of distance with a sequence of 55 stereo pairs covering 5.5 meters of vehicle travel. These results demonstrate the importance of the uncertainty model and the practical feasibility of visual motion estimation in unknown environments. Two appendices give the camera models and camera calibration procedures used with the vehicle (appendix A), as well as detailed derivations of the estimators (appendix B).

We address depth estimation in chapters 3 to 5. In chapter 3, we introduce the conceptual framework that guides the rest of the work. This involves estimating the depth at each pixel, modelling the uncertainty of depth estimates at each pixel, and using an area-based approach to matching. We outline the steps we take in formalizing this as a statistical estimation problem. These steps are similar to those followed in chapter 2, except that here the variables to be estimated are the depth at each pixel, instead of the 3-D landmark coordinates and the vehicle motions of chapter 2. As part of the formalization, we model the depth map as a Gaussian random field and motivate the development of a Bayesian approach to the matching problem. We then consider what is necessary to solve the depth map estimation problem reliably. We conclude that redundant sensing is key and that one of the most attractive ways to achieve redundant sensing is by using camera motion to acquire more than two images. This leads us to propose an operational framework in which stereo cameras are mounted on a precise translation stage; in turn, the translation stage is mounted on the robot vehicle. Camera translation is used to acquire a narrow-baseline image pair from one of the cameras, plus a third image from the other camera. This is the basis of a *bootstrap* operation in which depth estimates obtained from the narrow-baseline image pair are used to constrain matching to the third, "wide-baseline" image. Chapter 3 closes by summarizing the issues involved in this bootstrap operation, as well as issues involved in extrapolating the bootstrap operation to depth estimation from stereo image sequences obtained as the vehicle travels through its environment.

Chapters 4 and 5 elaborate components of the bootstrap operation. In chapter 4, we study the model of depth at each pixel as a Gaussian random variable. To do so, we derive a basic, maximum-likelihood approach to estimating depth at a given pixel. This involves comparing intensities of two images in a window around the pixel and boils down to the familiar sum-squared-error matching operator. However, our goal is to examine the uncertainty in disparity estimates, so we derive a sub-pixel version of this operator and derive the variance of the estimation error. The balance of the chapter consists of experiments that examine the resulting error distribution for synthetic and real images. We find, not too surprisingly, that the Gaussian model is very good with synthetic images and reasonable, though not perfect, for real images. We conclude that the approach is worth pursuing.

Chapter 5 then develops single-scale matching algorithms for the bootstrap operation *per se*. We categorize possible algorithms into three classes:

- *fully independent algorithms*, which use windowed correlation methods to estimate depth independently for each pixel;
- *joint 1-D algorithms*, which use coupled estimators to jointly estimate the depth for all pixels in a single scanline, but estimate each scanline independently from other scanlines;
- *joint 2-D algorithms*, which couple the depth estimates within and across scanlines.

We judge the first two categories to be most practical and develop Bayesian matching algorithms for them. These algorithms are extensions of the basic maximum-likelihood operator derived in chapter 4. For the joint 1-D case, we develop two coupling models, one based on a heuristic, disparity-gradient constraint and one based on a correlated model of prior disparity information. Both lead to efficient, dynamic-programming algorithms for obtaining optimal disparity estimates for the entire scanline. In this chapter, we also examine several questions concerning the chosen direction and distance to translate the cameras to obtain the narrow and wide-baseline image pairs. Finally, we demonstrate that the new matching algorithms perform very well with images of complex scenes. We conclude that the overall framework we are pursuing is a successful and very promising approach to general depth estimation.

Chapter 6 summarizes the work of the thesis, reviews our main conclusions, and outlines directions for extension.

Chapter 2

Motion Estimation

In this chapter, we estimate the motion of a robot vehicle by using on-board cameras to track 3-D feature points, or landmarks, in the vicinity of the vehicle. This involves jointly estimating the motion of the robot and the positions of the landmarks from the image sequence acquired as the robot moves. We formulate the problem in a robot-centered coordinate frame; that is, we maintain the coordinates of the landmarks relative to the robot and estimate the rotation and translation of the robot between successive stereo image pairs. We formulate the problem as a statistical estimation problem, develop a system that implements the estimation and the image processing procedures necessary to accomplish the task, and demonstrate the performance of the system on real image sequences.

We begin by reviewing the background of this problem in mapping and navigation and by introducing relevant batch and recursive estimation paradigms. We then outline the structure of our approach, showing both the estimation-related and the image processing-related aspects of the processing cycle. Subsequent sections discuss the estimation side and then the image processing side of the cycle. Simulations and experimental results obtained with real image sequences are presented to show the performance of the system and to compare it to previous work that used a simpler statistical model [Moravec80]. The results show a marked improvement over the simpler model and demonstrate the feasibility of visual motion estimation in unknown environments.

The central issues in both the estimation and image processing aspects of this work are important to related problems in global mapping and visual trajectory estimation. In the final section of this chapter, we make these ties explicit by discussing related work and potential extensions in mapping and trajectory estimation. We also discuss the limitations of this work in terms of the depth information obtained. Following chapters will develop an approach to reducing these limitations.

2.1 Background and Methodology

Figure 2.1 illustrates the visual motion estimation problem as it is approached in this chapter. A robot vehicle, travelling through unknown terrain, uses sensors to detect highly localizable

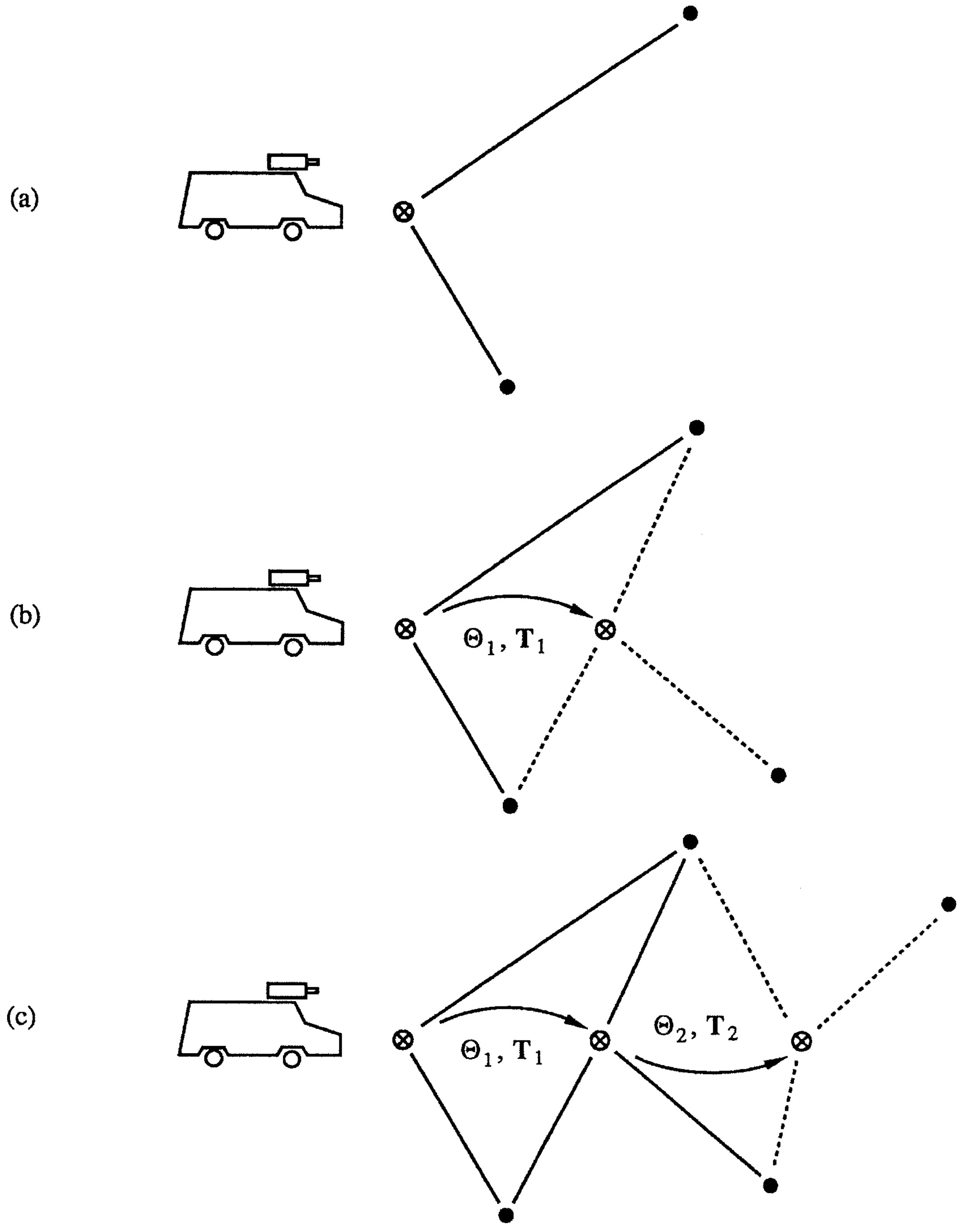


Figure 2.1: The robot navigation problem

(a) Robot in first position, establishes some landmarks; (b) robot in second position, picks more landmarks and relates its own position to its first position; (c) robot in third position, repeating the process. Θ is a vector of rotation angles and $\mathbf{T}$ is a translation vector.

features of its environment. It designates these features as *landmarks*, determines their 3-D coordinates, and stores the coordinates in a map. As the robot travels, it periodically observes the landmarks again and uses these observations to update estimates of its own position and the positions of the landmarks. Additional features of the environment may also be observed and added to the map. This process is repeated throughout the course of the journey. Because uncertainty will be present in all of the observations, estimates of the robot and landmark positions will also be uncertain. Our goal is to obtain optimal estimates of the robot and the landmark positions.

This problem may be approached in a *global* or a *local* manner. In the global approach, the goal is to produce optimal estimates of all of the landmarks observed in the course of the journey. In this case, a single map would contain all of the landmarks and note all of the robot positions indicated in figure 2.1. The landmark positions are modelled in a frame of reference that is fixed with respect to the ground. This is similar to problems in photogrammetry and geodesy that involve mapping of ground points from blocks of aerial photographs [Mikhail76,Slama80,Vanicek86]. The literatures in these areas contain well-developed paradigms for modelling and solving such problems. These paradigms involve establishing the functional relationships between observations and unknown parameters, modelling observational uncertainties, filtering out gross observational errors, and obtaining estimates of the unknowns via least squares estimation. In conventional aerial mapping applications, all of the observations are used to determine all of the unknowns in a single batch optimization procedure. However, both in photogrammetry and in robotics, there is interest in techniques that process observations on-line to incrementally update estimates of the unknowns. This is still a research issue; it has been discussed in a photogrammetric context in [Gruen84] and in robotic contexts in [DurrantWhyte88,Smith87].

While the global approach is appropriate for large-scale mapping applications, it is not suitable for local navigation or for estimating the incremental motion of a robot vehicle. The *local* approach is at the other end of the spectrum, in that it retains a model of only those objects in the vicinity of the robot. Depending on the application, it may also represent these objects in a robot-centered coordinate system. In this approach, successive observations are used both for estimating the current robot position and to update the parameters of the local 3-D model. This approach lends itself readily to incremental or recursive estimation procedures similar to the Kalman filter [Gelb74,Maybeck79]. The problems of feature tracking and recursive estimation that are encountered in this approach make it very similar to trajectory estimation problems in which the relative positions and velocities of two moving objects must be determined [Broida86,Gennery86,Tietz82,Wunsche86]. The primary difference is that in the case addressed here only position is estimated, so the kinematic model of robot motion is very simple.

Since the focus here is primarily on estimating robot position, we choose the local approach. This leads to a processing loop that repeatedly makes new observations, estimates the robot's current position, and updates a local landmark model. We will outline this loop in the next section, then explore the details and evaluate the performance of the resulting system in subsequent sections. At the end of the chapter, we will show how this work relates to both the global mapping and the trajectory estimation problems. We will also discuss the limitations of

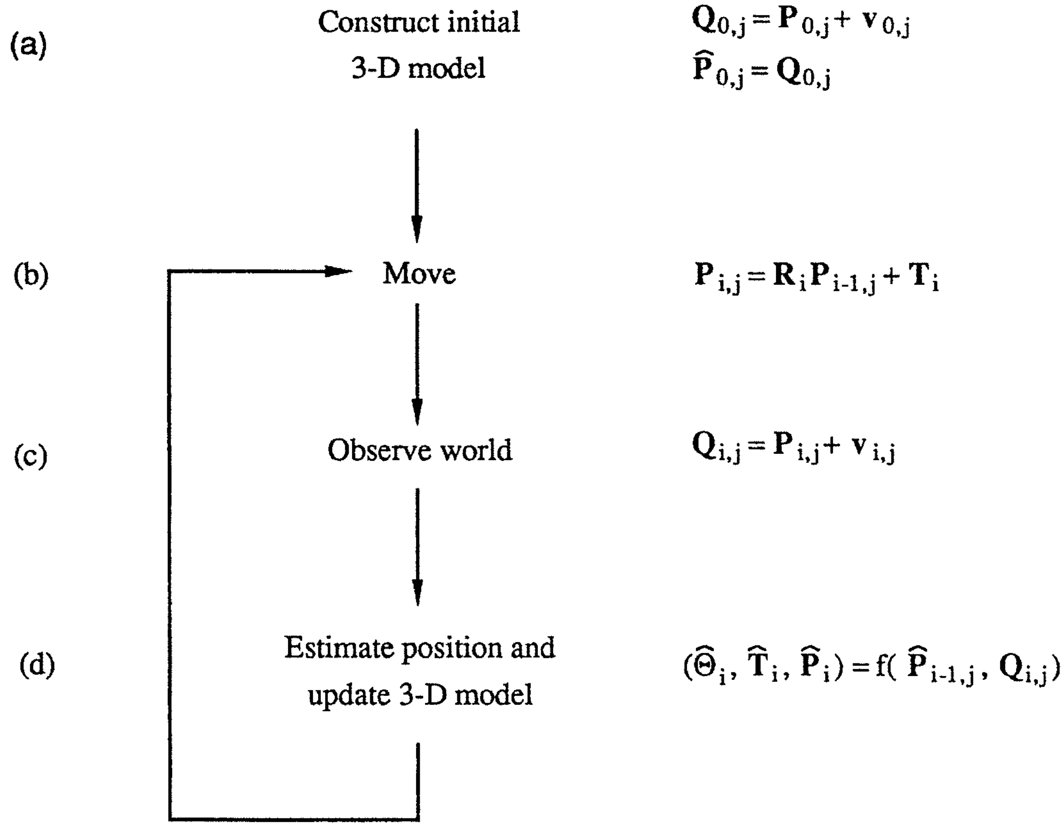

(a)
Construct initial 3-D model
$\mathbf{Q}_{0,j} = \mathbf{P}_{0,j} + \mathbf{v}_{0,j}$
$\hat{\mathbf{P}}_{0,j} = \mathbf{Q}_{0,j}$
(b)
Move
$\mathbf{P}_{i,j} = \mathbf{R}_i \mathbf{P}_{i-1,j} + \mathbf{T}_i$
(c)
Observe world
$\mathbf{Q}_{i,j} = \mathbf{P}_{i,j} + \mathbf{v}_{i,j}$
(d)
Estimate position and update 3-D model
$(\hat{\Theta}_i, \hat{\mathbf{T}}_i, \hat{\mathbf{P}}_i) = f(\hat{\mathbf{P}}_{i-1,j}, \mathbf{Q}_{i,j})$

Figure 2.2: Processing loop

the depth estimation and 3-D modelling procedures used in this chapter, in preparation for the more extensive depth estimation procedures developed in later chapters.

2.2 Structure of the Approach

The flowchart in figure 2.2 illustrates the structure of the approach. The system maintains a local world model consisting of the 3-D coordinates of the landmark points, represented in a robot-centered coordinate frame. This world model is initialized by finding correspondences in the first stereo image pair. Subsequently, the system operates in a loop that involves repeatedly moving the robot, locating the landmarks in the next image pair, estimating the motion of the robot between pairs, and updating estimates of the landmark positions. As landmarks become occluded or fall out of view, new landmarks are chosen from the new images and added to the world model.

For discussion, it is convenient to split the system into those aspects that are primarily concerned with estimation and those that are primarily concerned with image processing. The estimation aspects embrace the relevant equations of motion, models of the problem geometry, models of the observations and observation noise, and the procedures used for parameter estimation. These issues are addressed with methods drawn from the geodesy [Mikhail76,Vanicek86], optimal estimation [Gelb74,Maybeck79], and psychometric literatures [Schonemann66,Schonemann70]. The image processing aspects include algorithms for feature detection, stereo matching, and feature tracking. The image processing algorithms used here are refinements of those developed by Moravec [Moravec80].

We will discuss the estimation and the image-processing sub-systems in the following two sections, respectively.

2.3 Estimation Loop

We will formulate the processing cycle of figure 2.2 as a sequential, Bayesian estimation problem. This will lead to an estimation procedure that, on each cycle, uses the existing landmark model and new observations of the landmark coordinates to compute the vehicle motion between frames and to compute new estimates of the landmark coordinates relative to the current coordinate frame. The steps necessary to formalize this problem are to define [Maybeck79]:

1. the variables to be estimated,
2. the measurements or observations available,
3. the mathematical model describing how the measurements are related to the variables of interest,
4. the mathematical model of the uncertainties present, and
5. the performance evaluation criterion to judge which estimation algorithms are "best".

These steps are instantiated in our problem as follows.

(1) First, at time t_i, corresponding to acquisition of the i^{th} stereo image pair, the variables to be estimated are the rotation and translation vectors Θ_i, $\mathbf{T}_i$ that describe the vehicle motion between the previous and the current stereo pair, plus the 3-D coordinates $\mathbf{P}_{i,j}$ of the landmarks in the current coordinate frame. In the notation $\mathbf{P}_{i,j}$, the first subscript indexes the time step and the second subscript indexes landmarks at each point in time. We define the coordinate transformation between frames by

$$\mathbf{P}_{i,j} = \mathbf{R}_i \mathbf{P}_{i-1,j} + \mathbf{T}_i ,$$

where $\mathbf{R}_i$ is the rotation matrix corresponding to the vector Θ_i. This describes the change in the true coordinates of the landmarks, relative to the vehicle position, from image to image and formalizes step (b) in figure 2.2.

(2), (3), and (4) From each stereo pair, we measure the image coordinates $\mathbf{q}_{l_{i,j}} = [x_{l_{i,j}}\ y_{l_{i,j}}]^T$, $\mathbf{q}_{r_{i,j}} = [x_{r_{i,j}}\ y_{r_{i,j}}]^T$ of each landmark in view (figure 2.3) and use these to compute an *observation*

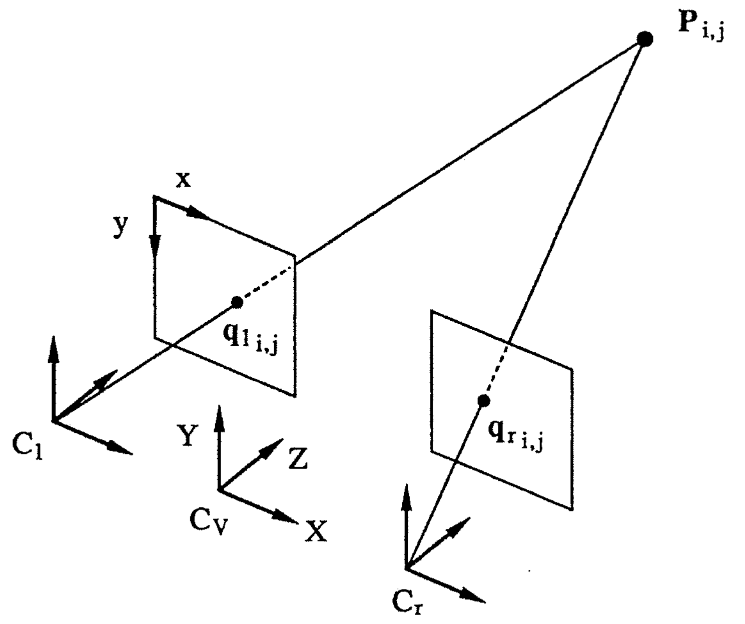


Figure 2.3: Stereo observation model

vector $Q_{i,j}$ of the 3-D landmark coordinates relative to the current vehicle position. Since the image coordinates contain some measurement error, the inferred 3-D coordinates will also contain measurement error. We model this by treating each $Q_{i,j}$ as the sum of the true landmark coordinates $\mathbf{P}_{i,j}$ and a noise vector $\mathbf{v}_{i,j}$:

$$Q_{i,j} = \mathbf{P}_{i,j} + \mathbf{v}_{i,j}.$$

We model the noise vectors $\mathbf{v}_{i,j}$ as zero-mean, Gaussian random vectors with covariance matrices $\Sigma_{\mathbf{v}_{i,j}}$. We assume that no prior information is available about the 3-D coordinates of landmarks; that is, the only information available about landmarks is obtained from the images. The initial world model, that is the landmark estimates $\hat{\mathbf{P}}_{0,j}$ at t_0, is equivalent to the observations made with the first stereo image pair (appendix A):

$$\hat{\mathbf{P}}_{0,j} \equiv Q_{0,j}.$$

Therefore, the error covariance of $\hat{\mathbf{P}}_{0,j}$ is $\Sigma_{\mathbf{P}_{0,j}} = \Sigma_{\mathbf{v}_{0,j}}$. The observations $Q_{0,j}$ and the resulting world model $\hat{\mathbf{P}}_{0,j}$ formalize step (a) of figure 2.2. Subsequent observations $Q_{i,j}$ are made in step (c). We assume that prior knowledge of the motion parameter vector $\mathbf{M} = [\Theta_i^T \ \mathbf{T}_i]^T$, if available, can be modelled by treating $\mathbf{M}$ as a Gaussian random vector with known mean and covariance.

(5) Following [Maybeck79], we will formulate the estimator in Bayesian terms. Therefore, at each iteration we will derive the probability distribution of the variables of interest and use this distribution to define our estimates. The distribution will be described by the conditional density

$$f(\mathbf{P}_{i,1}, \ldots, \mathbf{P}_{i,n}, \mathbf{M}_i | Q_{i,1}, \ldots, Q_{i,n})$$

of the current landmark and motion variables, given the most recent observations. We will use the maximum a posterior probability (MAP) criterion to define optimal estimates.

This defines the main components of the estimation loop. The details of the observation model and the estimation procedure are derived in the remainder of this section.

2.3.1 Observation Model

To simplify the notation, in describing the observation model we will dispense with the subscripts i,j. As shown in figure 2.3, with each stereo pair we measure the image coordinates $\mathbf{q}_l = [x_l\ y_l]^T$ and $\mathbf{q}_r = [x_r\ y_r]^T$ of the projections of each landmark $\mathbf{P} = [X\ Y\ Z]^T$ onto the left and right image, respectively. Our observation model includes a model of the uncertainty in $\mathbf{q}_l$ and $\mathbf{q}_r$, criteria for computing an estimate or observation $\mathbf{Q}$ of the 3-D landmark coordinates from $\mathbf{q}_l$ and $\mathbf{q}_r$, and a model of the resulting uncertainty in $\mathbf{Q}$. The observations $\mathbf{Q}$ are defined in a vehicle-centered coordinate frame C_V (figure 2.3). We use left-handed coordinate frames for the vehicle and the cameras. For each camera coordinate frame, the X axis extends parallel to the image plane from left to right, the Y axis extends upward parallel to the image plane, and the Z axis coincides with the optical axis of the camera. The origins of the camera coordinate frames coincide with the centers of projection of the lenses. Image coordinates are denoted x and y, with axes parallel to the camera coordinate frame.

Formally, the measured image coordinates are functions $\mathbf{h}_l$ and $\mathbf{h}_r$ of the landmark $\mathbf{P}$ with additive noise $\mathbf{v}_l$ and $\mathbf{v}_r$:

$$\begin{aligned}\mathbf{q}_l &= \begin{bmatrix} x_l \\ y_l \end{bmatrix} = \mathbf{h}_l(\mathbf{P}) + \mathbf{v}_l \\ \mathbf{q}_r &= \begin{bmatrix} x_r \\ y_r \end{bmatrix} = \mathbf{h}_r(\mathbf{P}) + \mathbf{v}_r\,.\end{aligned}$$

The measurement functions $\mathbf{h}_l$ and $\mathbf{h}_r$ define models of the coordinate transformations between the vehicle and the camera coordinate frames, as well as models of the perspective projection within each camera. In this section, we assume that the camera coordinate axes are parallel to the vehicle axes, with the origins placed symmetrically at distances $\pm b/2$ along the X axis of the vehicle frame. Therefore, in the camera frames the landmark coordinates are

$$\begin{aligned}\mathbf{P}_l = \begin{bmatrix} X_l \\ Y_l \\ Z_l \end{bmatrix} &= \mathbf{P} + \begin{bmatrix} b/2 \\ 0 \\ 0 \end{bmatrix} \\ \mathbf{P}_r = \begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} &= \mathbf{P} + \begin{bmatrix} -b/2 \\ 0 \\ 0 \end{bmatrix}.\end{aligned}$$

This model is idealized, because in general there will also be rotations and other translations between the vehicle and camera coordinate frames. Appendix A describes the extensions necessary to model the additional degrees of freedom and describes the calibration procedure that was used to determine the parameters of the camera models. The model above is very useful for basic simulations and was used in the simulations described in section 2.5. Experiments with real images used the extended model described in the appendix.

For both the simulations and the experiments with real images, the projection within each camera is modelled as an ideal perspective projection followed by scaling and translation of the image coordinates (see appendix A). The scaling and translation transform the ideal image coordinates (e.g. $X_l/Z_l, Y_l/Z_l$) into the coordinate system used for the actual images. The complete transformation from $\mathbf{P}$ to measured image coordinates is

$$\mathbf{q}_l = \begin{bmatrix} x_l \\ y_l \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} s_x & 0 & c_x \\ 0 & s_y & c_y \end{bmatrix} \begin{bmatrix} (X + b/2) \\ Y \\ Z \end{bmatrix} + \mathbf{v}_l \tag{2.1}$$

$$\mathbf{q}_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} s_x & 0 & c_x \\ 0 & s_y & c_y \end{bmatrix} \begin{bmatrix} (X - b/2) \\ Y \\ Z \end{bmatrix} + \mathbf{v}_r\,. \tag{2.2}$$

Here s_x, s_y, c_x, c_y denote the image plane scale and translation parameters. For the simulations, these are the same for both cameras; for the real images, separate parameters are determined for each camera.

We model the noise terms $\mathbf{v}_l$, $\mathbf{v}_r$ as zero mean, Gaussian random vectors with covariance matrices Σ_l and Σ_r, respectively. Chapter 4 develops a sub-pixel matching procedure that can be used to compute $\mathbf{q}_l$ and $\mathbf{q}_r$ to sub-pixel resolution and to estimate the covariance matrices.

Given the nonlinear measurement model of equations (2.1) and (2.2), our task is to compute an estimate $\mathbf{Q}$ of the 3-D coordinates of $\mathbf{P}$. For the idealized camera geometry, we compute $\mathbf{Q} = \left[\hat{X}\,\hat{Y}\,\hat{Z}\right]^T$ by inverting equations (2.1) and (2.2) to obtain

$$\begin{aligned} \hat{X} &= \frac{b(x_l + x_r - 2c_x)}{2(x_l - x_r)} \\ \hat{Y} &= \frac{bs_x(y_l + y_r - 2c_y)}{2s_y(x_l - x_r)} \\ \hat{Z} &= \frac{bs_x}{x_l - x_r}\,. \end{aligned} \tag{2.3}$$

Because the measured image coordinates are noisy, $\mathbf{Q}$ is noisy as well. The nonlinearity of (2.3) makes the uncertainty in $\mathbf{Q}$ non-Gaussian. Nevertheless, we use standard error propagation methods [Mikhail76] to approximate the uncertainty as Gaussian, with zero mean and with covariance

$$\Sigma_{\mathbf{v}} = \mathbf{J} \begin{bmatrix} \Sigma_l & 0 \\ 0 & \Sigma_r \end{bmatrix} \mathbf{J}^T, \tag{2.4}$$

where $\mathbf{J}$ is the matrix of first partial derivatives of (2.3) with respect to x_l, y_l, x_r, y_r, or the Jacobian. In section 2.5, we demonstrate that this approximation is adequate for position estimation in indoor navigation. The triangulation and uncertainty modelling procedures are summarized by writing $\mathbf{Q}$ as

$$\mathbf{Q} = \mathbf{P} + \mathbf{v}\,,$$

where $\mathbf{v}$ is a zero-mean, Gaussian random vector with covariance $\Sigma_{\mathbf{V}}$.

For non-ideal camera geometries the triangulation and error modelling procedures are similar, but somewhat more complex. The details are described in Appendix A.

Figure 2.4 interprets the observation model geometrically. Constant probability contours of the density of $\mathbf{v}$ describe ellipsoids that approximate the true error density. For nearby points the contours will be close to spherical; the farther the points the more eccentric the contours become (figure 2.4a). This illustrates the importance of modelling the uncertainty in $\mathbf{Q}$ by a full 3-D Gaussian density, rather than by a single scalar uncertainty factor s as done in earlier work [Moravec80]. Scalar error models are equivalent to diagonal covariance matrices $\Sigma = s\mathbf{I}$, where $\mathbf{I}$ is the 3×3 identity matrix. This model is appropriate when landmarks are very close to the camera, but it breaks down rapidly with increasing distance. Figure 2.4b shows a qualitative comparison of the Gaussian error model and the uncertainty regions that result from considering only quantization error in the image coordinates. The similarity of the models suggests that the Gaussian model will be useful even when quantization error is a significant component of the uncertainty in the measured image coordinates[1].

Where the Gaussian approximation breaks down is in failing to represent asymmetry in the true error density. The nonlinearity of the triangulation operation will cause the true error density to be skewed, not unlike the effects of quantization error shown in figure 2.4b. The skew is not significant when points are close, but becomes more pronounced the more distant the points. A possible consequence is biased estimation of point locations, which may lead to biased motion estimates. We have not examined this issue in detail; however, we will see some of its effect in simulations in section 2.5.

2.3.2 Estimation Procedure

Applying the foregoing model to each landmark gives a set of observations

$$\mathbf{Q}_{i,j} = \mathbf{P}_{i,j} + \mathbf{v}_{i,j} \tag{2.5}$$

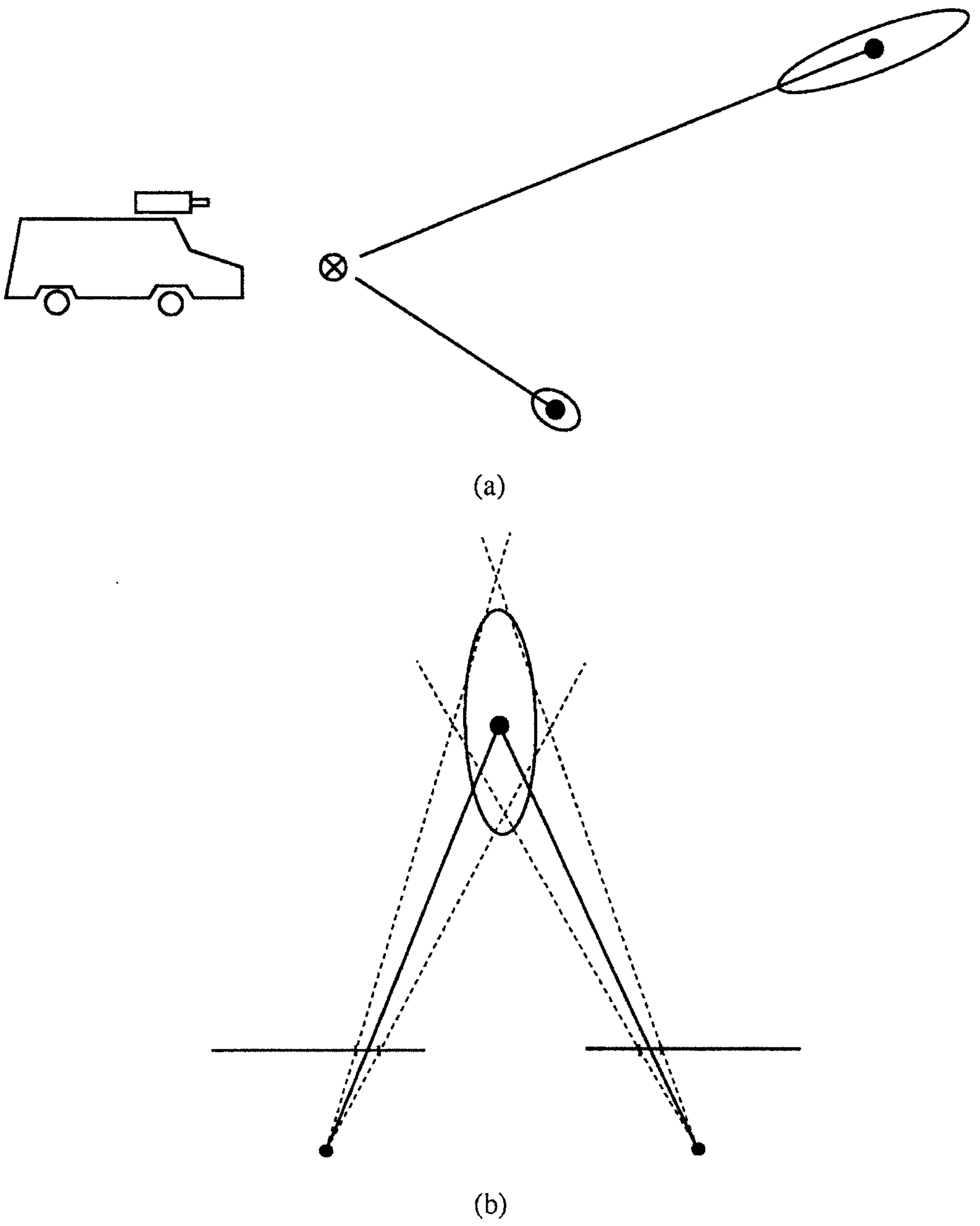
for each stereo pair. In this section, we derive a sequential Bayesian estimator that uses this sequence of observations together with the motion equation

$$\mathbf{P}_{i,j} = \mathbf{R}_i \mathbf{P}_{i-1,j} + \mathbf{T}_i \tag{2.6}$$

to compute estimates $\widehat{\Theta}_i$, $\widehat{\mathbf{T}}_i$ of the motion parameters and estimates $\widehat{\mathbf{P}}_{i,j}$ of the landmark coordinates. The landmark estimates are defined relative to the current coordinate frame and incorporate information from the entire observation history.

Two issues that complicate the derivation are the nonlinearity of the motion equation (2.6) and the large dimensionality involved in tracking many landmarks. We deal with the nonlinearity by using a simplified, least-squares formulation to compute an initial estimate of the motion parameters, then by linearizing the motion equation about the initial estimate and using iterative solution methods. We reduce the dimensionality of the problem by partitioning it to obtain

[1] A non-Gaussian distribution to model the effects of quantization error only is derived in [Blostein87].


(a)

(b)

Figure 2.4: Triangulation error: (a) error distribution is much more eccentric for distant points than for nearby points; (b) the Gaussian error model is a reasonable approximation even if the primary noise source is quantization of the image coordinates.

estimates of the motion parameters and each of the landmarks from separate systems of equations that are each no larger than 6×6.

For clarity, we present the derivation in several steps. First we discuss the least-squares initial estimate of the motion parameters. Next, we describe the linearization and the iterative solution method by deriving a maximum-likelihood estimate for the motion parameters alone. We then derive the complete estimator, which is a sequential Bayesian procedure that jointly estimates the motion and the landmark parameters. This procedure uses the maximum-likelihood motion estimate as part of the solution and employs matrix partitioning methods to reduce dimensionality. Finally, we give a concise summary of the the entire statistical model and the steps in computing the solution.

Because the following derivations refer entirely to the "previous" and "current" coordinate frames, we simplify the notation by dropping the subscript i from the motion parameters. We also condense the subscripts on the observations and the landmarks by writing $\mathbf{Q}_{pj}$, $\mathbf{Q}_{cj}$, $\mathbf{P}_{pj}$, and $\mathbf{P}_{cj}$ instead of $\mathbf{Q}_{i-1,j}$, $\mathbf{Q}_{i,j}$, $\mathbf{P}_{i-1,j}$, and $\mathbf{P}_{i,j}$.

Least-squares Estimation of Θ and T

From the observation (2.5) and motion (2.6) equations, the observations $\mathbf{Q}_{pj}$, $\mathbf{Q}_{cj}$ made from two successive stereo pairs are related to the unknown motion and landmark parameters by

$$\begin{aligned} \mathbf{Q}_{pj} &= \mathbf{P}_{pj} + \mathbf{v}_{pj} && (2.7) \\ \mathbf{Q}_{cj} &= \mathbf{R}\mathbf{P}_{pj} + \mathbf{T} + \mathbf{v}_{cj}\,. && (2.8) \end{aligned}$$

There is one such pair of equations for each landmark. To get initial estimates of the motion parameters, we reduce these equations to an ordinary least-squares problem for which a solution is known to exist. This is done by eliminating $\mathbf{P}_{pj}$ from (2.7), (2.8) and rewriting the one remaining equation in terms of a residual error vector $\mathbf{e}_j$:

$$\mathbf{e}_j = \mathbf{Q}_{cj} - \mathbf{R}\mathbf{Q}_{pj} - \mathbf{T}\,. \qquad (2.9)$$

Taking the squared length of each residual vector, applying scalar weighting factors w_j, and summing over all landmarks produces the cost expression

$$\sum_j w_j \mathbf{e}_j^T \mathbf{e}_j\,. \qquad (2.10)$$

The least-squares estimates are obtained by minimizing this expression over Θ and $\mathbf{T}$. The scalar weights w_j are defined to reflect the quality of the observations $\mathbf{Q}_{pj}$ and $\mathbf{Q}_{cj}$, for example by letting $w_j = (\det(\Sigma_{pj}) + \det(\Sigma_{cj}))^{-1}$. Moravec used this formulation (with different w_j) in an earlier approach to the motion estimation problem [Moravec80].

Following the standard solution procedure by differentiating (2.10) with respect to Θ and $\mathbf{T}$ does not lead to a linear optimization problem. However, a direct solution has been obtained in work on a related problem in the analysis of psychometric data[Schonemann66,Schonemann70]. This solution augments (2.10) with Lagrange multipliers that constrain $\mathbf{R}$ to be orthogonal. The resulting equations can be solved via the singular-value decomposition for the unique, orthogonal matrix **R** and vector **T** that minimize (2.10)[2]. Appendix B.1 gives a detailed derivation of this solution, expressed in terms appropriate to our application. We extract the corresponding rotation angles from **R** with well-known techniques ([Paul81, chapter 3]) and designate the resulting initial estimates as Θ_0 and $\mathbf{T}_0$.

Maximum Likelihood Estimation of Θ and T

As we will see shortly, the least-squares estimator of Θ and **T** is equivalent to a maximum-likelihood estimator derived from an observation model in which the error covariance matrices are scaled identity matrices, $\Sigma_\mathbf{V} = s\mathbf{I}$. The resulting motion estimates can be substantially inferior to those derived with the full error model. Unfortunately, using the full error model leads to a nonlinear optimization problem that does not appear to have a direct solution. To illustrate this problem and to show how it is solved via linearization, we now use the full error model to derive maximum likelihood estimates for Θ and **T**. These estimates will prove to be part of the larger solution for Θ, **T**, and $\mathbf{P}_{cj}$ that we derive subsequently.

Using (2.7) and (2.8) to eliminate $\mathbf{P}_{pj}$ as before, we obtain

$$\mathbf{Q}_{cj} = \mathbf{R}\mathbf{Q}_{pj} + \mathbf{T} + \mathbf{v}_j ,$$

where $\mathbf{v}_j$ subsumes the uncertainties in $\mathbf{Q}_{cj}$ and the product $\mathbf{R}\mathbf{Q}_{pj}$. For simplicity, suppose for the moment that $\mathbf{Q}_{pj}$ is noise-free, so that $\mathbf{v}_j = \mathbf{v}_{cj}$. Then the joint conditional density of the observations $\mathbf{Q}_{cj}$ given Θ and **T** is Gaussian,

$$f(\mathbf{Q}_{c1}, \ldots, \mathbf{Q}_{cn} | \Theta, \mathbf{T}) \propto \exp\left\{-\frac{1}{2}\sum_j \mathbf{e}_j^T \mathbf{W}_j \mathbf{e}_j\right\} ,$$

where $\mathbf{e}_j = \mathbf{Q}_{cj} - \mathbf{R}\mathbf{Q}_{pj} - \mathbf{T}$ and $\mathbf{W}_j$ is the inverse covariance matrix of $\mathbf{v}_j$. The maximum likelihood estimates of Θ and **T** are those that maximize this density. This is equivalent to finding Θ and **T** that minimize the summation in the exponent:

$$\sum_j \mathbf{e}_j^T \mathbf{W}_j \mathbf{e}_j . \tag{2.11}$$

Note that letting $\mathbf{W}_j = w_j\mathbf{I}$ reduces this expression to the objective function used in the least-squares solution (2.10). Unfortunately, the minimization problem is again nonlinear and the techniques that solve (2.10) do not to generalize to (2.11). Therefore, we resort to linearizing the problem and computing the estimates iteratively.

The linearization is obtained by taking a first-order expansion of (2.8) with respect to the rotation angles, evaluated at the initial estimate Θ_0:

$$\begin{aligned} \mathbf{Q}_{cj} &= \mathbf{R}\mathbf{P}_{pj} + \mathbf{T} + \mathbf{v}_{cj} \\ &\approx \mathbf{R}_0\mathbf{P}_{pj} + \left[\frac{d(\mathbf{R}\mathbf{P}_{pj})}{d\Theta}\right]_0 (\Theta - \Theta_0) + \mathbf{T} + \mathbf{v}_{cj} \\ &= \mathbf{R}_0\mathbf{P}_{pj} + \mathbf{J}_j(\Theta - \Theta_0) + \mathbf{T} + \mathbf{v}_{cj} . \end{aligned}$$

[2] Direct solutions that formulate the coordinate transformation in quaternion algebra are also known [Hebert83,Wertz78].

$\mathbf{R}_0$ denotes the rotation matrix for Θ_0 and $\mathbf{J}_j$ denotes the Jacobian[3] for landmark j, evaluated at $\Theta = \Theta_0$. Once again eliminating $\mathbf{P}_{pj}$, we obtain

$$\mathbf{Q}_{cj} = \mathbf{R}_0\mathbf{Q}_{pj} + \mathbf{J}_j(\Theta - \Theta_0) + \mathbf{T} + \mathbf{v}_j\,.$$

By error propagation, $\mathbf{v}_j$ is approximately a zero-mean, Gaussian noise vector with covariance $\Sigma_j = \Sigma_{cj} + \mathbf{R}_0\Sigma_{pj}\mathbf{R}_0^T$. Finally, we rewrite this equation as

$$\mathbf{Q}_{cj} - \mathbf{R}_0\mathbf{Q}_{pj} + \mathbf{J}_j\Theta_0 = \begin{bmatrix} \mathbf{J}_j & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix} + \mathbf{v}_j \tag{2.12}$$

and abbreviate it to

$$\mathbf{Q}_j = \mathbf{H}_j \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix} + \mathbf{v}_j\,,$$

where $\mathbf{Q}_j = \mathbf{Q}_{cj} - \mathbf{R}_0\mathbf{Q}_{pj} + \mathbf{J}_j\Theta_0$ and $\mathbf{H}_j = [\mathbf{J}_j\ \mathbf{I}]$. This equation models $\mathbf{Q}_j$ as a linear measurement of the unknowns Θ and $\mathbf{T}$, with additive Gaussian noise $\mathbf{v}_j$. Maximum likelihood estimates of Θ and $\mathbf{T}$ are obtained by minimizing the objective function (2.11), with $\mathbf{W}_j = (\Sigma_{cj} + \mathbf{R}_0\Sigma_{pj}\mathbf{R}_0^T)^{-1}$ and with the error vector $\mathbf{e}_j$ redefined as

$$\mathbf{e}_j = \mathbf{Q}_j - \mathbf{H}_j \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix}.$$

Differentiating the linearized objective function with respect to Θ and $\mathbf{T}$ and setting the derivatives to zero, we obtain the linear system

$$\left[\sum_j \mathbf{H}_j^T\mathbf{W}_j\mathbf{H}_j\right] \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix} = \left[\sum_j \mathbf{H}_j^T\mathbf{W}_j\mathbf{Q}_j\right]. \tag{2.13}$$

Inverting this, estimates of the motion parameters are given by

$$\widehat{\mathbf{M}} = \begin{bmatrix} \widehat{\Theta} \\ \widehat{\mathbf{T}} \end{bmatrix} = \left[\sum_j \mathbf{H}_j^T\mathbf{W}_j\mathbf{H}_j\right]^{-1} \left[\sum_j \mathbf{H}_j^T\mathbf{W}_j\mathbf{Q}_j\right]. \tag{2.14}$$

The motion equation is then re-linearized about the new estimate (i.e. the new Θ_0) and the solution is re-computed. The entire linearization and solution procedure is iterated until $\widehat{\Theta} \approx \Theta_0$. Appendix B.2 shows that the solution can be decomposed so that $\widehat{\Theta}$ is computed first, then $\widehat{\mathbf{T}}$ is obtained as a function of the optimal rotation. This allows the translation to be computed outside the iteration loop. The error covariance matrix of $\widehat{\mathbf{M}}$ is

$$\Sigma_{\mathbf{M}} = \left[\sum_j \mathbf{H}_j^T\mathbf{W}_j\mathbf{H}_j\right]^{-1}. \tag{2.15}$$

[3] A full derivation of $\mathbf{J}_j$ and an efficient method for computing it are given in appendix B.2.

This can be examined to evaluate the conditioning of the motion estimates. It will also be needed later to compute estimates of the landmark coordinates.

To recapitulate, we have used the full Gaussian error model to derive maximum likelihood estimates of Θ and $\mathbf{T}$. Since the resulting optimization problem is nonlinear, we developed a linearized formulation and iterated the solution starting from an initial estimate obtained by least-squares. The iterative procedure is considerably more expensive than using the least-squares estimate alone; however, it can produce motion estimates that are considerably superior to least-squares. An intuitive explanation for why and when this is the case can be obtained by recalling that the observational error density is relatively compact when landmarks are quite close, but becomes increasingly eccentric as landmarks become more distant (figure 2.4a). The least-squares objective function (2.10) provides an adequate model for the error density for the nearby case but not for the distant case, since it cannot reflect the eccentricity of the density. The objective function resulting from the maximum-likelihood approach (2.11) models both cases via the inverse covariance matrices $\mathbf{W}_j$. That is, $\mathbf{W}_j$ serves as a norm in measuring the length of residual vector $\mathbf{e}_j$. When a landmark is close, $\mathbf{W}_j$ is nearly equivalent to $w_j\mathbf{I}$. However, when the landmark is distant, $\mathbf{W}_j$ effectively gives less weight to triangulation errors along the line of sight that perpendicular to the line of sight. Given the nature of triangulation, this is appropriate. The relative quality of the motion estimates produced with the two methods is demonstrated by experiments in section 2.5.

Sequential Bayesian Estimation of Θ, T, and $\mathbf{P}_{cj}$

In the foregoing stages of the solution, we eliminated $\mathbf{P}_{pj}$ at the outset and solved only for Θ and $\mathbf{T}$. We will now formulate a Bayesian procedure for jointly estimating Θ, $\mathbf{T}$, and $\mathbf{P}_{cj}$. This formulation allows us to incorporate prior information about the motion parameters into the estimator; it also allows us to estimate the landmark coordinates in a sequential fashion, such that the estimates at each point in time incorporate information from the entire observation history. The estimator will encounter the difficulties with nonlinearity and large dimensionality we alluded to earlier. We solve these problems by linearizing the motion equation, using partitioned matrix methods to obtain the estimates from a series of low-dimensioned linear systems, and iterating the solution until the estimates converge. In the process, the least-squares and maximum-likelihood estimates of Θ and $\mathbf{T}$ appear as intermediate steps in the solution. For conciseness, we will denote the set of landmarks $\mathbf{P}_{cj}$ by the vector $\mathbf{P} = [\mathbf{P}_{c1}^T, \ldots, \mathbf{P}_{cn}^T]^T$ and the set of current observations $\mathbf{Q}_{cj}$ by the vector $\mathbf{Q} = [\mathbf{Q}_{c1}^T, \ldots, \mathbf{Q}_{cn}^T]^T$.

The estimator will be obtained from the conditional probability density $f(\mathbf{P}, \mathbf{M}|\mathbf{Q})$ of $\mathbf{P}$, $\mathbf{M}$ given the current observations $\mathbf{Q}$. From Bayes's theorem, this density is given by

$$f(\mathbf{P}, \mathbf{M}|\mathbf{Q}) = \frac{f(\mathbf{Q}|\mathbf{P}, \mathbf{M}) f(\mathbf{P}, \mathbf{M})}{f(\mathbf{Q})}.$$

$f(\mathbf{P}, \mathbf{M}|\mathbf{Q})$ is referred to as the *posterior* density of $\mathbf{P}$ and $\mathbf{M}$. $f(\mathbf{P}, \mathbf{M})$ is the *prior* density of these parameters and $f(\mathbf{Q}|\mathbf{P}, \mathbf{M})$ is the conditional density of the observations, given particular values of the parameters. For a given set of observations, $f(\mathbf{Q})$ is a constant scale factor that does not appear in our estimation procedure. We define the estimates by the MAP criterion; that is, the estimates $\widehat{\mathbf{M}}$, $\widehat{\mathbf{P}}$ are those values of **P**, **M** that maximize the posterior density. For Gaussians, the MAP estimate is equal to the mean of the posterior density [Maybeck79]. In this case, the error in the estimate,

$$\begin{bmatrix} \mathbf{P}-\widehat{\mathbf{P}} \\ \mathbf{M}-\widehat{\mathbf{M}} \end{bmatrix},$$

is a zero-mean, Gaussian random vector with covariance equal to the covariance of the posterior density. The full covariance matrix has dimension $(3n+6) \times (3n+6)$, which is too large to maintain. Therefore, the sequential estimation procedure maintains only the the 3×3 covariance matrices of the individual landmarks and the 6×6 covariance matrix of the motion parameters. These all lie on the main diagonal of the full posterior covariance matrix.

To obtain the estimator, we will now derive the prior density, the conditional density of **Q**, and the posterior means and covariances. We adopt the superscripts "–" and "+" from the Kalman filtering literature [Maybeck79] to distinguish between estimates of a quantity *before* incorporating new observations and updated estimates of the same quantity *after* incorporating new observations.

We will begin with the conditional density $f(\mathbf{Q}|\mathbf{M},\mathbf{P})$. From (2.5), observations made at the current time are modelled by

$$\mathbf{Q}_{cj} = \mathbf{P}_{cj} + \mathbf{v}_{cj}\,.$$

Because the noise terms $\mathbf{v}_{cj}$ are independent, zero-mean Gaussians with inverse covariance $\mathbf{W}_{\mathbf{v}_{cj}}$, the joint conditional density of **Q** given **P** is

$$f(\mathbf{Q}|\mathbf{P}) \propto \exp\left\{-\frac{1}{2}\sum_j \mathbf{e}_{\mathbf{v}_j}^T \mathbf{W}_{\mathbf{v}_{cj}} \mathbf{e}_{\mathbf{v}_j}\right\},$$

where $\mathbf{e}_{\mathbf{v}_j} = \mathbf{Q}_{cj} - \mathbf{P}_{cj}$. Since **Q** does not depend on **M**, $f(\mathbf{Q}|\mathbf{P})$ is equal to $f(\mathbf{Q}|\mathbf{P},\mathbf{M})$.

To derive the prior density, we start by considering the motion parameters alone. We assume that any prior information available about the motion parameters is statistically independent from step to step and that for each step the information can be modelled as a joint probability density for $\mathbf{M} = [\Theta^T\ \mathbf{T}^T]^T$. We model this density as Gaussian with mean $\widehat{\mathbf{M}}^-$ and covariance $\Sigma_{\mathbf{M}}^-$. Therefore, letting $\mathbf{e}_{\mathbf{M}} = \mathbf{M} - \widehat{\mathbf{M}}^-$, the prior density of **M** is

$$f(\mathbf{M}) \propto \exp\left\{-\frac{1}{2}\mathbf{e}_{\mathbf{M}}^T \mathbf{W}_{\mathbf{M}}^- \mathbf{e}_{\mathbf{M}}\right\},$$

where $\mathbf{W}_{\mathbf{M}}^- = (\Sigma_{\mathbf{M}}^-)^{-1}$.

Prior information about the landmarks $\mathbf{P}_{cj}$ is embodied in the estimates $\widehat{\mathbf{P}}_{pj}^+$ and error covariances $\Sigma_{\mathbf{P}_{pj}}^+$ obtained relative to the previous coordinate frame. To obtain a prior density for $\mathbf{P}_{cj}$, we relate the previous estimates to $\mathbf{P}_{cj}$ via the motion equation

$$\mathbf{P}_{cj} = \mathbf{R}\mathbf{P}_{pj} + \mathbf{T}\,.$$

Linearizing this about the initial motion estimate $\mathbf{M}_0 = [\Theta_0^T \ \mathbf{T}_0^T]^T$, we obtain

$$\begin{aligned}\mathbf{P}_{cj} &\approx \mathbf{R}_0\mathbf{P}_{pj} + \mathbf{J}_j(\Theta - \Theta_0) + \mathbf{T}_0 + (\mathbf{T} - \mathbf{T}_0) \\ &= \mathbf{R}_0\mathbf{P}_{pj} + \mathbf{T}_0 + \left[\begin{array}{cc} \mathbf{J}_j & \mathbf{I} \end{array} \right] \left[\begin{array}{c} \Theta - \Theta_0 \\ \mathbf{T} - \mathbf{T}_0 \end{array} \right] \\ &= \mathbf{R}_0\mathbf{P}_{pj} + \mathbf{T}_0 + \mathbf{H}_j(\mathbf{M} - \mathbf{M}_0) . \end{aligned} \tag{2.16}$$

Note that $\mathbf{P}_{pj}$ is a Gaussian random vector with mean $\widehat{\mathbf{P}}_{pj}^+$ and covariance $\Sigma_{\mathbf{P}_{pj}}^+$. Therefore, we see from (2.16) that the prior density of $\mathbf{P}_{cj}$, conditioned on $\mathbf{M}$ (i.e. $f(\mathbf{P}_{cj}|\mathbf{M})$), is Gaussian with mean

$$\mathbf{R}_0\widehat{\mathbf{P}}_{pj}^+ + \mathbf{T}_0 + \mathbf{H}_j(\mathbf{M} - \mathbf{M}_0)$$

and covariance $\Sigma_{\mathbf{P}_{cj}}^- = \mathbf{R}_0\Sigma_{\mathbf{P}_{pj}}^+\mathbf{R}_0^T$. Letting $\mathbf{W}_{\mathbf{P}_{cj}}^- = (\Sigma_{\mathbf{P}_{cj}}^-)^{-1}$, $\widehat{\mathbf{P}}_{cj}^- = \mathbf{R}_0\widehat{\mathbf{P}}_{pj}^+ + \mathbf{T}_0$, and

$$\mathbf{e}_{\mathbf{P}_j} = \mathbf{P}_{cj} - \widehat{\mathbf{P}}_{cj}^- - \mathbf{H}_j(\mathbf{M} - \mathbf{M}_0) ,$$

the joint conditional density of $\mathbf{P}$ given $\mathbf{M}$ is

$$f(\mathbf{P}|\mathbf{M}) \propto \exp\left\{ -\frac{1}{2} \sum_j \mathbf{e}_{\mathbf{P}_j}^T \mathbf{W}_{\mathbf{P}_{cj}}^- \mathbf{e}_{\mathbf{P}_j} \right\} .$$

The desired prior density of $\mathbf{M}$ and $\mathbf{P}$ is now given by

$$\begin{aligned} f(\mathbf{P}, \mathbf{M}) &= f(\mathbf{P}|\mathbf{M}) f(\mathbf{M}) \\ &\propto \exp\left\{ -\frac{1}{2} \left(\mathbf{e}_{\mathbf{M}}^T \mathbf{W}_{\mathbf{M}}^- \mathbf{e}_{\mathbf{M}} + \sum_j \mathbf{e}_{\mathbf{P}_j}^T \mathbf{W}_{\mathbf{P}_{cj}}^- \mathbf{e}_{\mathbf{P}_j} \right) \right\} . \end{aligned}$$

Next, from Bayes's theorem, the posterior density is

$$f(\mathbf{P}, \mathbf{M}|\mathbf{Q}) \propto \exp\left\{ -\frac{1}{2} \left(\mathbf{e}_{\mathbf{M}}^T \mathbf{W}_{\mathbf{M}}^- \mathbf{e}_{\mathbf{M}} + \sum_j \mathbf{e}_{\mathbf{P}_j}^T \mathbf{W}_{\mathbf{P}_{cj}}^- \mathbf{e}_{\mathbf{P}_j} + \sum_j \mathbf{e}_{\mathbf{v}_j}^T \mathbf{W}_{\mathbf{v}_{cj}} \mathbf{e}_{\mathbf{v}_j} \right) \right\} .$$

The MAP estimate is obtained by finding $\mathbf{M}$ and $\mathbf{P}$ that maximize this expression. This is equivalent to maximizing the log probability

$$\ln f = -\frac{1}{2} \left(\sum_j \mathbf{e}_{\mathbf{v}_j}^T \mathbf{W}_{\mathbf{v}_j} \mathbf{e}_{\mathbf{v}_j} + \sum_j \mathbf{e}_{\mathbf{P}_j}^T \mathbf{W}_{\mathbf{P}_{cj}}^- \mathbf{e}_{\mathbf{P}_j} + \mathbf{e}_{\mathbf{M}}^T \mathbf{W}_{\mathbf{M}}^- \mathbf{e}_{\mathbf{M}} \right) + K , \tag{2.17}$$

where K is a constant. Therefore, the optimal estimates minimize the sum of the quadratic forms.

The algebra leading to the estimates and error covariances is presented in appendix B.3. This involves partitioning the solution so that the motion estimates are generated first, then the landmark coordinates are estimated individually from low-order systems. The end result is that the linearized motion estimate is given by

$$\widehat{\mathbf{M}}^+ = \left[\begin{array}{c} \widehat{\Theta}^+ \\ \widehat{\mathbf{T}}^+ \end{array} \right] = \left[\mathbf{W}_{\mathbf{M}}^- + \sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{H}_j \right]^{-1} \left[\mathbf{W}_{\mathbf{M}}^- \widehat{\mathbf{M}}^- + \sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{Q}_j \right] \tag{2.18}$$

with error covariance

$$\Sigma_{\mathbf{M}}^{+} = \left[\mathbf{W}_{\mathbf{M}}^{-} + \sum_{j} \mathbf{H}_j^T \mathbf{W}_j \mathbf{H}_j \right]^{-1} . \qquad (2.19)$$

Here $\mathbf{Q}_j = \mathbf{Q}_{cj} - \mathbf{R}_0 \widehat{\mathbf{P}}_{pj}^{+} + \mathbf{J}_j \Theta_0$ and $\mathbf{W}_j = (\Sigma_{\mathbf{P}_{cj}}^{-} + \Sigma_{\mathbf{V}_{cj}})^{-1}$; these are essentially the same as in the maximum-likelihood solution, with the previous observations replaced by the previous landmark estimates.

To interpret these equations, note that the absence of prior motion information is modelled by letting $\mathbf{W}_{\mathbf{M}}^{-} = 0$; in this case, the Bayesian estimate reduces to the maximum likelihood estimate given in equations (2.14) and (2.15). When $\mathbf{W}_{\mathbf{M}}^{-} \neq 0$, the motion estimate is a weighted combination of the prior estimate and terms involving the new observations.

As in the maximum-likelihood case, the solution can be iterated by linearizing about the new estimate. Furthermore, the solution again can be partitioned to estimate $\widehat{\Theta}^{+}$ first, then to give $\widehat{\mathbf{T}}^{+}$ in terms of $\widehat{\Theta}^{+}$. For simplicity, it may be preferable to iterate the maximum-likelihood solution to convergence, then to do one iteration with (2.18) and (2.19) to incorporate the prior information.

Having computed the optimal motion estimate, the estimated landmark coordinates are (appendix B.3)

$$\widehat{\mathbf{P}}_{cj}^{+} = (\mathbf{W}_{\mathbf{V}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}}^{-})^{-1} (\mathbf{W}_{\mathbf{V}_{cj}} \mathbf{Q}_{cj} + \mathbf{W}_{\mathbf{P}_{cj}}^{-} \widehat{\mathbf{P}}_{cj}^{-}) , \qquad (2.20)$$

where $\widehat{\mathbf{P}}_{cj}^{-} = \widehat{\mathbf{R}}^{+} \widehat{\mathbf{P}}_{pj}^{+} + \widehat{\mathbf{T}}^{+}$ is computed using $\widehat{\mathbf{M}}^{+}$. As given earlier, $\mathbf{W}_{\mathbf{P}_{cj}}^{-} = (\mathbf{R}_0 \Sigma_{\mathbf{P}_{pj}}^{+} \mathbf{R}_0^T)^{-1}$. Finally, the error covariance for each landmark is

$$\Sigma_{\mathbf{P}_{cj}}^{+} = (\mathbf{W}_{\mathbf{V}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}}^{-})^{-1} + (\mathbf{W}_{\mathbf{V}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}}^{-})^{-1} \mathbf{W}_{\mathbf{P}_{cj}}^{-} \mathbf{H}_j \Sigma_{\mathbf{M}}^{+} \mathbf{H}_j^T \mathbf{W}_{\mathbf{P}_{cj}}^{-} (\mathbf{W}_{\mathbf{V}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}}^{-})^{-1} . \qquad (2.21)$$

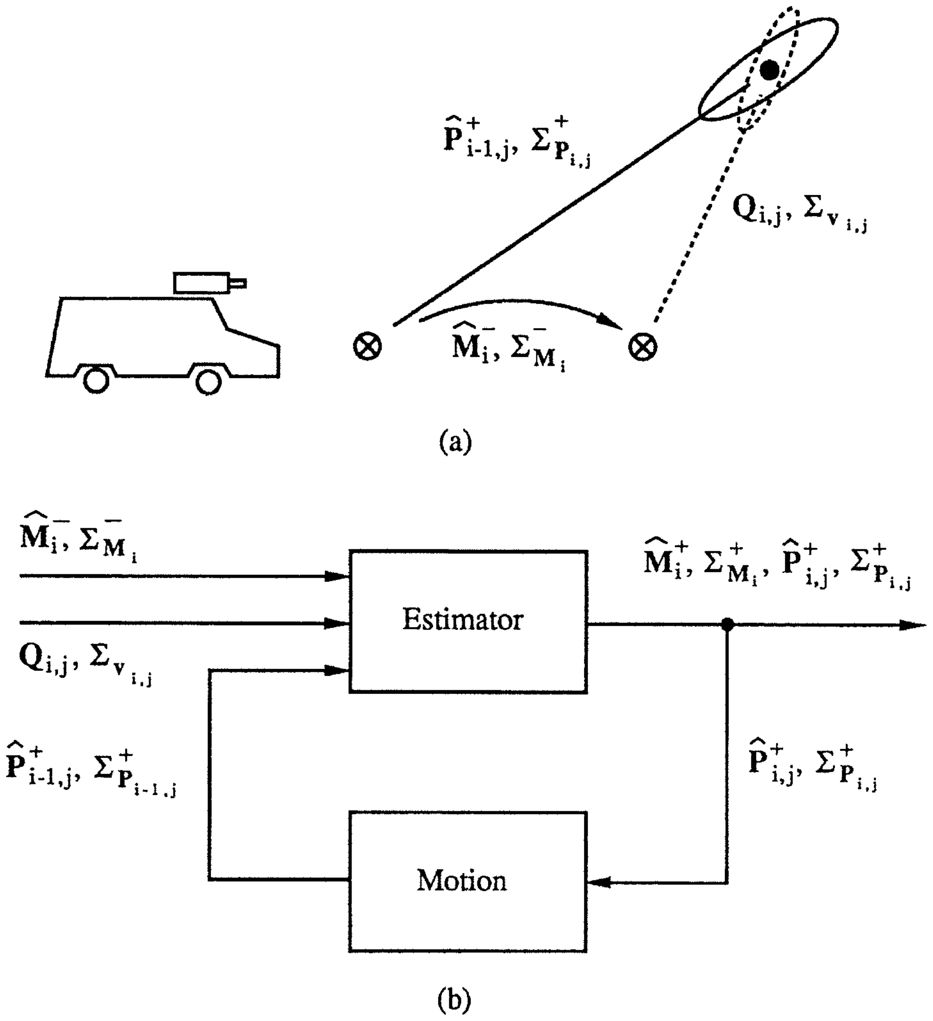
Equations (2.20) and (2.21) are applied independently to each landmark.

The landmark coordinate estimate is easy to interpret; we simply transform the previous estimate $\widehat{\mathbf{P}}_{pj}^{+}$ to the new coordinate frame, obtaining $\widehat{\mathbf{P}}_{cj}^{-}$, then combine it with the new observation $\mathbf{Q}_{cj}$, weighting each according to their respective covariances. To gain insight into the new error covariance, note that when the motion is known exactly (i.e. $\mathbf{W}_{\mathbf{M}}^{-} = \mathbf{0}$) the covariance reduces to

$$\begin{aligned} \Sigma_{\mathbf{P}_{cj}}^{+} &= (\mathbf{W}_{\mathbf{V}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}}^{-})^{-1} \\ &= (\mathbf{W}_{\mathbf{V}_{cj}} + \mathbf{R}_0 \mathbf{W}_{\mathbf{P}_{pj}}^{-} \mathbf{R}_0^T)^{-1} . \end{aligned}$$

That is, the covariance of the new landmark estimate is a function of the covariance of the new observation and the covariance of the previous estimate as transformed into the current frame. As a result, $\Sigma_{\mathbf{P}_{cj}}^{+}$ will be "smaller" than either its two component covariances. When the motion is not known exactly, but is estimated from the landmarks themselves, the estimate is not exact. Moreover, it is correlated with the landmarks. The additional term in (2.21) models these effects.

To summarize, we have derived a sequential Bayesian procedure for estimating the motion parameters and the landmark coordinates relative to the current coordinate frame. The inputs to the algorithm are the landmark estimates from the previous coordinate frame, the new landmark


$\hat{\mathbf{P}}^+_{i-1,j}, \Sigma^+_{\mathbf{P}_{i,j}}$
$\mathbf{Q}_{i,j}, \Sigma_{\mathbf{v}_{i,j}}$
$\hat{\mathbf{M}}^-_i, \Sigma^-_{\mathbf{M}_i}$
$\hat{\mathbf{M}}^-_i, \Sigma^-_{\mathbf{M}_i}$
$\mathbf{Q}_{i,j}, \Sigma_{\mathbf{v}_{i,j}}$
$\hat{\mathbf{P}}^+_{i-1,j}, \Sigma^+_{\mathbf{P}_{i-1,j}}$
Estimator
$\hat{\mathbf{M}}^+_i, \Sigma^+_{\mathbf{M}_i}, \hat{\mathbf{P}}^+_{i,j}, \Sigma^+_{\mathbf{P}_{i,j}}$
$\hat{\mathbf{P}}^+_{i,j}, \Sigma^+_{\mathbf{P}_{i,j}}$
Motion

(a)

(b)

Figure 2.5: Sequential estimation procedure

observations made in the current frame, and the prior density of the motion parameters (figure 2.5a). From these, the algorithm computes a posterior estimate of the motion parameters and estimates of the landmark coordinates relative to the current coordinate frame (2.5b). Because the landmark coordinates are effectively updated to reflect the entire observation history, the estimation error for each landmark should decrease over time and the accuracy of the motion estimates should correspondingly increase. This will be examined in simulation in section 2.5.

2.3.3 Summary

In this section we have presented a statistical model for the motion estimation problem and derived procedures for sequentially estimating the frame-to-frame motion of the vehicle and the landmark coordinates relative to each coordinate frame. We will now review the entire procedure to bring together the main results.

Table 2.1 summarizes the statistical model and the estimation procedures. At each point in time, we locate the landmarks $\mathbf{P}_{i,j}$ in the current stereo pair, measure their image coordinates $\mathbf{q}_{l_{i,j}}$ and $\mathbf{q}_{r_{i,j}}$, and estimate the covariance matrices $\Sigma_{l_{i,j}}$ and $\Sigma_{r_{i,j}}$ of the measurements. Procedures for this step are discussed in section 2.4 and in chapter 4. These measurements are used to compute the "observed" 3-D coordinates $\mathbf{Q}_{i,j}$ and associated covariances $\Sigma_{\mathbf{V}_{i,j}}$ (section 2.3.1 and appendix A). An initial estimate $\mathbf{M}_{i_0}$ of the motion between the previous and the current coordinate frame then is computed via least-squares and an adapted version of Schonemann's algorithm ([Schonemann66,Schonemann70], section 2.3.2, and appendix B.1). We described this algorithm as it would be applied to observations from the current and previous coordinate frames; however, in the sequential estimation context it would be applied to the current observations and the previous landmark estimates, as shown in Table 2.1. Given this initial estimate, the motion parameters are refined by linearizing the motion equation and iterating the procedure defined by equations (2.18) and (2.19). Finally, the converged motion estimate is used to compute the current estimates of the landmark coordinates via equations (2.20) and (2.21). The cycle then repeats with the movement of the vehicle and the acquisition of a new stereo pair.

The linearizations and Gaussian approximations used in both the observation model and the estimation procedures are potential weaknesses of this approach. As we discussed earlier, the linearized observation model is reasonably valid so long as landmarks are not extremely distant; we show later that performance with real images is quite good with this model. Also, the iterative estimation procedures converge rapidly unless all landmarks are extremely distant; for example, in the experiments reported later the final estimates were obtained after four to eight iterations.

Several other issues present possible limitations of or scope for extending the algorithms presented here. First, the spatial distribution of the landmarks affects the conditioning of the motion estimate. For example, if the landmarks happen to be collinear, one component of the vehicle rotation will be undetermined. Poor conditioning can be detected by examining the singular values [Golub83] of the covariance matrix of the computed motion parameters. If necessary, new landmarks can be searched for in regions of space that will improve the conditioning (see section 2.4.1). The second issue concerns calibrating the level of noise in the statistical model. As we show in chapter 4, the covariance matrices $\Sigma_{l_{i,j}}$ and $\Sigma_{r_{i,j}}$ of the measured

Variables	$\mathbf{P}_{i,j}$, Θ_i, $\mathbf{T}_i$
Motion equation	$\mathbf{P}_{i,j} = \mathbf{R}_i\mathbf{P}_{i-1,j} + \mathbf{T}_i$
Observed image coordinates	$\mathbf{q}_{l_{i,j}} = \mathbf{h}_l(\mathbf{P}_{i,j}) + \mathbf{v}_{l_{i,j}}\,,\quad \mathbf{v}_{l_{i,j}} \sim \mathrm{N}(\mathbf{0}, \Sigma_{l_{i,j}})$ $\mathbf{q}_{r_{i,j}} = \mathbf{h}_r(\mathbf{P}_{i,j}) + \mathbf{v}_{r_{i,j}}\,,\quad \mathbf{v}_{r_{i,j}} \sim \mathrm{N}(\mathbf{0}, \Sigma_{r_{i,j}})$
Observed 3-D coordinates	$\mathbf{Q}_{i,j} = \mathbf{P}_{i,j} + \mathbf{v}_{i,j}\,,\quad \mathbf{v}_{i,j} \sim \mathrm{N}(\mathbf{0}, \Sigma_{\mathbf{v}_{i,j}})$ Inverse covariance: $\mathbf{W}_{\mathbf{v}_{i,j}} = (\Sigma_{\mathbf{v}_{i,j}})^{-1}$
Landmark estimates at t_0	$\widehat{\mathbf{P}}^{+}_{0,j} = \mathbf{Q}_{0,j}\,,\quad \Sigma^{+}_{\mathbf{P}_{0,j}} = \Sigma_{\mathbf{v}_{0,j}}$ Inverse covariance: $\mathbf{W}^{+}_{\mathbf{P}_{0,j}} = (\Sigma^{+}_{\mathbf{P}_{0,j}})^{-1}$
Prior motion information	$\mathrm{f}(\mathbf{M}_i) = \mathrm{N}(\widehat{\mathbf{M}}^{-}_i, \Sigma^{-}_{\mathbf{M}_i})$ Inverse covariance: $\mathbf{W}^{-}_{\mathbf{M}_i} = (\Sigma^{-}_{\mathbf{M}_i})^{-1}$
Initial motion estimate	$\mathbf{e}_{i,j} = \mathbf{Q}_{i,j} - \mathbf{R}_i\widehat{\mathbf{P}}^{+}_{i-1,j} - \mathbf{T}_i$ $\min_{\mathbf{R}_i,\mathbf{T}_i} \sum_j w_{i,j}\,\mathbf{e}^T_{i,j}\,\mathbf{e}_{i,j}$ Solved to yield $\mathbf{M}_{i_0} = \left[\Theta^T_{i_0}\ \mathbf{T}^T_{i_0}\right]^T$
Linearization	$\mathbf{P}_{i,j} \approx \mathbf{R}_{i_0}\mathbf{P}_{i-1,j} + \mathbf{J}_{i,j}(\Theta_i - \Theta_{i_0}) + \mathbf{T}_i$ $= \mathbf{R}_{i_0}\mathbf{P}_{i-1,j} - \mathbf{J}_{i,j}\Theta_{i_0} + \mathbf{H}_{i,j}\mathbf{M}_i$ where $\mathbf{H}_{i,j} = [\mathbf{J}_{i,j}\ \mathbf{I}]$ and $\mathbf{M}_i = \left[\Theta^T_i\ \mathbf{T}^T_i\right]^T$
Refined motion estimate	Let $\mathbf{Q}'_{i,j} = \mathbf{Q}_{i,j} - \mathbf{R}_{i_0}\widehat{\mathbf{P}}^{+}_{i-1,j} + \mathbf{J}_{i,j}\Theta_{i_0}$, $\Sigma^{-}_{\mathbf{P}_{i,j}} = \mathbf{R}_{i_0}\Sigma^{+}_{\mathbf{P}_{i-1,j}}\mathbf{R}^T_{i_0}$, and $\mathbf{W}_{i,j} = (\Sigma_{\mathbf{v}_{i,j}} + \Sigma^{-}_{\mathbf{P}_{i,j}})^{-1}$. Then $\Sigma^{+}_{\mathbf{M}_i} = \left[\mathbf{W}^{-}_{\mathbf{M}_i} + \sum_j \mathbf{H}^T_{i,j}\mathbf{W}_{i,j}\mathbf{H}_{i,j}\right]^{-1}$ $\widehat{\mathbf{M}}^{+}_i = \Sigma^{+}_{\mathbf{M}_i}\left[\mathbf{W}^{-}_{\mathbf{M}_i}\widehat{\mathbf{M}}^{-}_i + \sum_j \mathbf{H}^T_{i,j}\mathbf{W}_{i,j}\mathbf{Q}'_{i,j}\right]$ Iterate, relinearizing about $\widehat{\mathbf{M}}^{+}_i$ each time
Updated landmark estimates	Let $\widehat{\mathbf{P}}^{-}_{i,j} = \widehat{\mathbf{R}}^{+}_i\widehat{\mathbf{P}}^{+}_{i-1,j} + \widehat{\mathbf{T}}^{+}_i$. Then $\widehat{\mathbf{P}}^{+}_{i,j} = (\mathbf{W}_{\mathbf{v}_{i,j}} + \mathbf{W}^{-}_{\mathbf{P}_{i,j}})^{-1}(\mathbf{W}_{\mathbf{v}_{i,j}}\mathbf{Q}_{i,j} + \mathbf{W}^{-}_{\mathbf{P}_{i,j}}\widehat{\mathbf{P}}^{-}_{i,j})$ $\Sigma^{+}_{\mathbf{P}_{i,j}} = (\mathbf{W}_{\mathbf{v}_{i,j}} + \mathbf{W}^{-}_{\mathbf{P}_{i,j}})^{-1} +$ $(\mathbf{W}_{\mathbf{v}_{i,j}} + \mathbf{W}^{-}_{\mathbf{P}_{i,j}})^{-1}\mathbf{W}^{-}_{\mathbf{P}_{i,j}}\mathbf{H}_{i,j}\Sigma^{+}_{\mathbf{M}_i}\mathbf{H}^T_{i,j}\mathbf{W}^{-}_{\mathbf{P}_{i,j}}(\mathbf{W}_{\mathbf{v}_{i,j}} + \mathbf{W}^{-}_{\mathbf{P}_{i,j}})^{-1}$

Table 2.1: Summary of statistical model and estimator equations.

image coordinates are directly proportional to the variance of the noise in the image. Thus, the image noise level functions as a single, global scale factor on the uncertainty in the entire model. In photogrammetry such a scale factor is referred to as the *reference variance* [Mikhail76]. Knowledge of the reference variance is not necessary for computing the parameter estimates, but it is needed to obtain properly scaled covariance matrices and to determine thresholds for error detection (appendix B.5). If necessary, a "posterior" estimate of the reference variance can be obtained at the end of the estimation cycle using techniques discussed in appendix B.4. Doing so allows the entire system to automatically adapt to the level of image noise. Finally, the procedures described in this section assume that no gross observation errors are present; in particular, they assume that landmarks are tracked correctly from frame to frame. Since this will not be true in practice, errors must be filtered out before or during the estimation process. Procedures for doing so are described in section 2.4.4 and appendix B.5.

2.4 Image Processing Loop

The previous section described the 3-D modelling and estimation aspects of the system loop shown in figure 2.2. These aspects dealt entirely with geometric abstractions of the locations of point landmarks, the projections of the landmarks in the image, the vehicle motion, and the appropriate statistical models and estimation procedures. In this section, we discuss the algorithms that provide input to the estimation loop by producing the measured image coordinates $\mathbf{q}_l$ and $\mathbf{q}_r$. Since these algorithms manipulate the images directly and correspond to components of the system loop of figure 2.2, we refer to these algorithms collectively as the *image processing loop*.

Image processing algorithms are associated with steps (a) and (c) of the system flowchart (figure 2.2); that is, with those steps that produce observations of the world. In step (a), the initial 3-D model is created by applying *feature selection* and *stereo matching* procedures to the initial stereo image pair. Landmarks are defined by 3-D points whose projections $\mathbf{p}_l$ and $\mathbf{p}_r$ can be precisely measured in successive stereo image pairs. Therefore, the feature selection process identifies points $\mathbf{q}_l$ in one image of a stereo pair that are likely to lead to trackable landmarks. The stereo matching procedure uses a correlation-based search to find the corresponding point $\mathbf{q}_r$ in the other image of the stereo pair. The resulting image coordinates are then used for triangulation in determining the initial 3-D model, $\hat{\mathbf{P}}_{0,j}$. Because landmarks continually drift out of view as the vehicle moves, the feature selection and stereo matching procedures are also applied at the end of the cycle (after step (d)) to replenish the landmark model.

Additional image processing operations are performed within the loop at step (c), when existing landmarks must be located in new stereo pairs. That is, image matching operations use the appearance of a landmark in the previous stereo pair to locate the landmark in the images of the new stereo pair, thereby producing new measurements $\mathbf{q}_l$ and $\mathbf{q}_r$. We refer to this as *feature tracking*. The tracking operation is implemented by correlation-based searches very similar to the stereo matching operation. The resulting image coordinates are converted into new 3-D observations $\mathbf{Q}_{i,j}$ via triangulation. Observations produced by feature tracking may include gross errors resulting from failure to locate landmarks. These errors are filtered out by thresholds applied to correlation coefficients, by 3-D *rigidity tests* applied after feature tracking, and by *outlier detection tests* applied as part of the estimation procedures that compute vehicle position.

The balance of this section describes the feature selection, stereo matching, feature tracking, and error detection algorithms in more detail. At the end of this section, we will review the system operation and present a more detailed flowchart that illustrates the combined estimation and image processing operations.

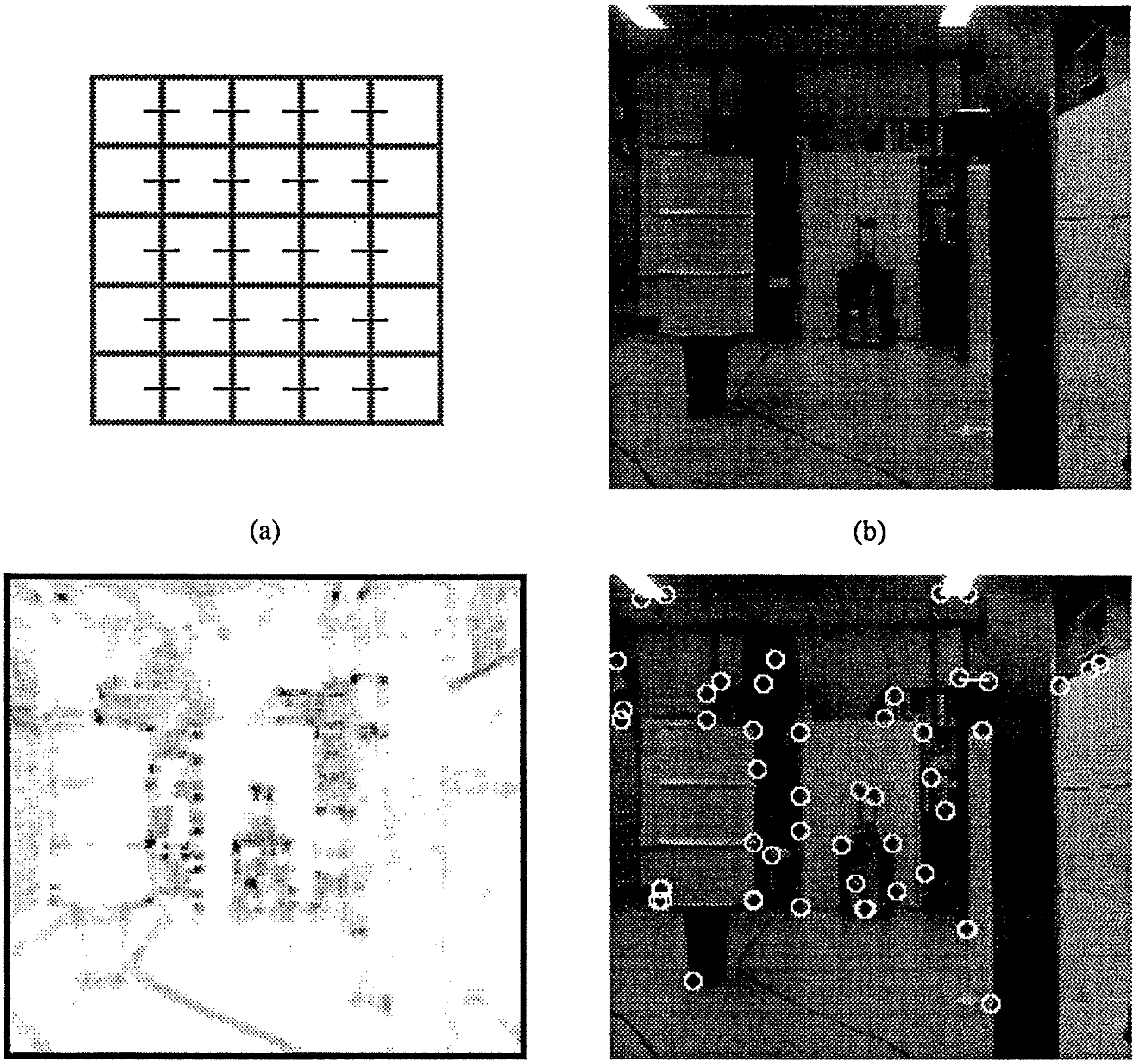
2.4.1 Feature Selection

Because stereo matching and feature tracking are implemented by correlation between images, the appearance of a landmark is modelled by a small patch of intensity around the landmark's projection in an image. For a landmark to be tracked reliably and accurately, the patch must exhibit intensity variation that allows the landmark to be localized in subsequent images. For example, pixels lying on extended edges cannot be localized in the direction parallel to the edge; such regions are not acceptable landmarks. Pixels on object corners or in certain kinds of texture are acceptable. Therefore, one issue to be addressed by the feature selection operator is how to find points that can be precisely localized in other images. Since motion estimates will only be well-conditioned if the landmarks are well-distributed in space, a second issue is how to select features so as to obtain good spatial distributions of landmarks.

For the experiments described in this chapter, the localizability issue was addressed in an informal manner by using the convolution-like *interest operator* developed by Moravec [Moravec80] to identify regions of the image having high intensity variation in multiple directions. This operator produces an *interest value* for each pixel in the image. The interest value is large for pixels that are localizable, such as those near vertices in the image, and low for pixels that are not localizable, such as pixels lying on extended edges or in areas of uniform intensity. For an $N \times N$ region Ω around a given pixel, the operator computes directional variances m_h, m_v, m_{d1}, and m_{d2} that respectively measure the intensity variation in the horizontal, vertical, and both diagonal directions over the area of the region. The measures are defined by

$$\begin{aligned}
m_h &= \sum_{x,y\in\Omega} [I(x,y) - I(x+1,y)]^2 \\
m_v &= \sum_{x,y\in\Omega} [I(x,y) - I(x,y+1)]^2 \\
m_{d1} &= \sum_{x,y\in\Omega} [I(x,y) - I(x+1,y+1)]^2 \\
m_{d2} &= \sum_{x,y\in\Omega} [I(x,y) - I(x-1,y+1)]^2
\end{aligned}$$

The horizontal measure m_h is illustrated in figure 2.6a. The interest value of the pixel at the center of the region is defined as the minimum of the directional variance measures. This operator accords high interest to regions with strong intensity variation in multiple directions,


(a)

(b)

(c)

(d)

Figure 2.6: Interest operator

(a) One of the directional variance terms: m_h is the sum of squares of differences of the pixels joined with dashes. (b) Example image. (c) "Interest" value. (d) Selected feature points.

less interest to regions with uni-directional intensity variation (such as regions along edges), and no interest to homogeneous regions of the image. In practice, the operator selects features like corners very well.

The notion of localizability can be defined formally in terms of the statistical uncertainty in the location of correlation peaks that are obtained when region Ω is matched in other images. This uncertainty is modelled by the covariance matrices Σ_l and Σ_r. Following this reasoning leads to a statistically-derived interest operator [Forstner88] that is very similar to the intuitively obtained Moravec operator. This is pursued in chapter 4.

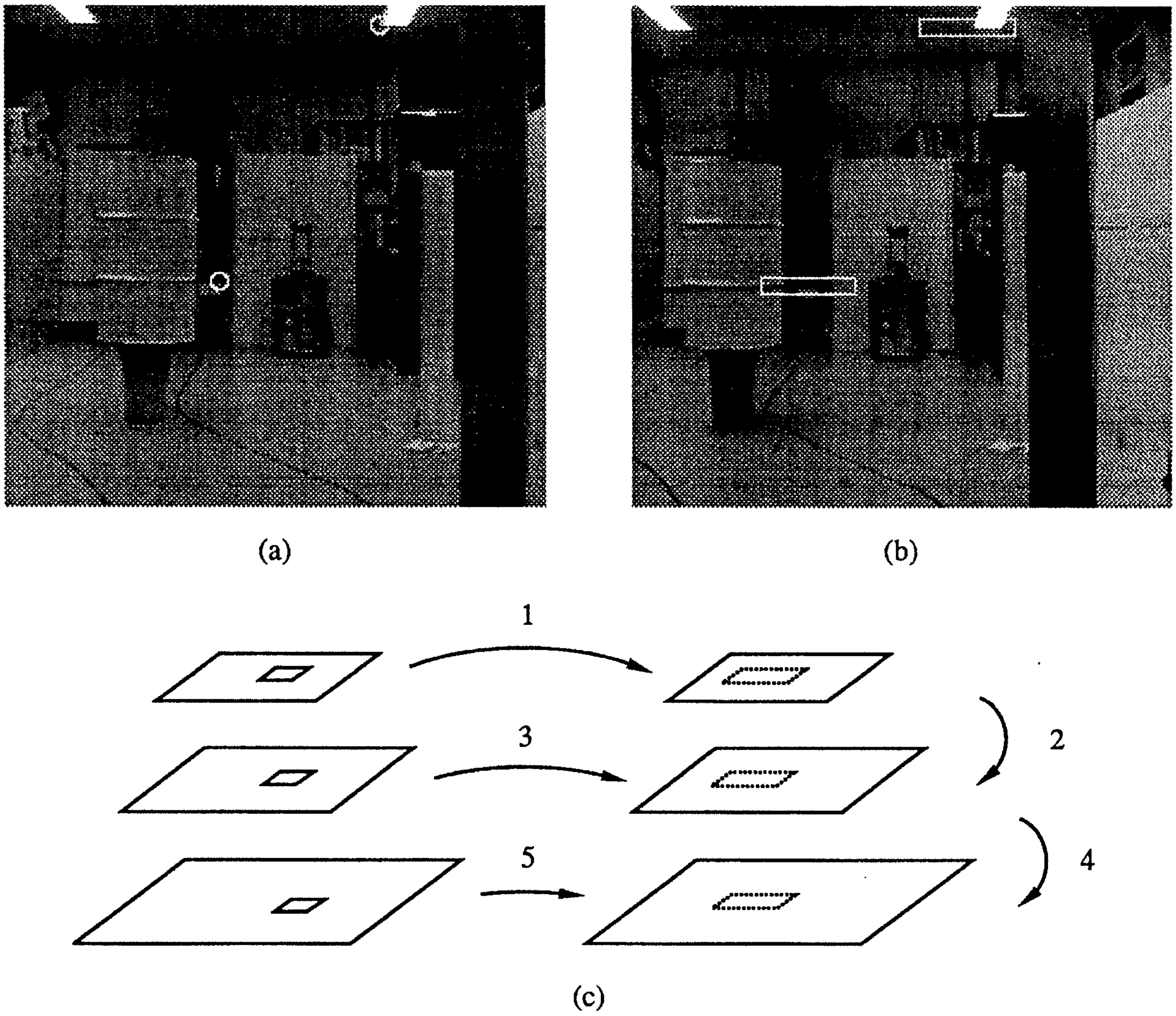
In addition to the localizability issue, it is also important to select pixels that will lead to landmarks that are well distributed in space. For this implementation, this was addressed with the simple expedient of partitioning each image into a 10 × 10 array of cells and identifying the most interesting pixel in each cell. This produced 100 *features* for each image. When creating the initial 3-D model, candidate landmarks were selected by sorting the features in descending order of interestingness and choosing the top N, where N was the number of landmarks being tracked ($\leq$ 50 in the work here). When replenishing the 3-D model, enough features are chosen to bring the number of landmarks back up to N; in this case, features are not chosen from grid cells already containing a landmark. A more complete approach to the spatial distribution issue is to search for features in areas of 3-D space that give good conditioning, rather than to settle for dispersion in the image. An approach of this nature has been developed for an object-tracking application described in [Wunsche86].

To illustrate the results obtained with this procedure, figure 2.6b shows one image from a test sequence and figure 2.6c shows the interest values computed for the same image. High interest pixels are black, low interest pixels are white. Note that strong intensity corners are very interesting, whereas areas along edges are less interesting and areas of uniform intensity are uninteresting. Figure 2.6d shows the selected features that result. Potential problems with this operator are discussed in [Thorpe84]. These include the possibility of choosing features that do not correspond to stationary points in 3-D; examples are the extremes of the barrel in figure 2.6 and chance alignments of foreground and background features at object boundaries.

2.4.2 Stereo Matching

Once features are extracted, corresponding points must be identified for each feature in the second image of the stereo pair. This is achieved by defining search windows for each feature within the second image and by using a coarse-to-fine, correlation-based search to find the best feature correspondences within the windows. Figures 2.7a and 2.7b illustrate two selected features and the corresponding search windows for a sample stereo pair. The coarse-to-fine search procedure is illustrated in figure 2.7.

The search windows are defined from knowledge of the relative camera geometry and from knowledge that the possible distance to a feature is confined to a pre-defined range $[Z_{min}, Z_{max}]$. From figure 2.3, the camera geometry is such that corresponding pixels are on or near corresponding scanlines in the two images. To allow for slight misalignment of the cameras, each search window extends above and below the nominal scanline by a small amount; in experiments,



(a)

(b)

(c)

Figure 2.7: Constrained image pyramid correlation for stereo matching

(a) Left image of a stereo pair, showing two particular feature points. (b) Right image of the same stereo pair, showing search windows for each feature. (c) Image pyramids and coarse-to-fine correlation procedure. Matching begins with the lowest resolution image and proceeds to the highest resolution.

this was 5% of the image height. The pre-defined distance range is translated into a disparity range that limits the horizontal extent of the search window. Z_{min} was determined by the fact that the cameras were set some distance back from the nose of the vehicle (yielding Z_{min} = 1.5m). For indoor operation, Z_{max} was determined by the size of the largest room in which the vehicle operated (yielding $Z_{max} \approx$ 10m); for outdoor operation, Z_{max} can be set to infinity. Given the baseline and lens focal length used in the experiments in section 2.5, the practical effect of this range was to limit the search window to about seven degrees of the visual field, or less than 20% of the total image width.

The search itself is conducted by coarse-to-fine, correlation-based search through an image pyramid [Moravec80] (figure 2.7c). Image pyramids are created for both images of the stereo pair. In this work, each level of the pyramid was created from the previous by averaging over 2×2 or 4×4 regions of the higher resolution image, then reducing the resolution by half. Resolutions from 480×512 down to 15×16 were computed. Then, starting with the lowest-resolution images, a small patch around the feature from the left image is correlated over the search window in the right image. For the experiments done here, the patch was 5×5 pixels. The correlation operation was Moravec's *pseudo-normalized* correlation [Moravec80]; this is similar to normalized correlation, but retains some sensitivity to bias and gain changes between the images. The position of best match is noted and scaled up to the next higher resolution image. The search window is reduced to a small region around the scaled up position of best match and the search is repeated at the new level. This process continues until the feature is matched in the highest resolution image. When features lie near the edges of the image, definitions of the search windows are modified to account for boundary effects. Finally, the correlation coefficients computed from the highest-resolution images are thresholded to eliminate features that match poorly.

The search procedure is applied independently for each selected feature and produces the left and right image coordinates, $\mathbf{q}_l$ and $\mathbf{q}_r$, for each feature. In the implementation, the image coordinates were computed only to pixel resolution and the covariance matrices were defaulted to $\Sigma_l = \Sigma_r = \mathbf{I}$. Methods for obtaining sub-pixel resolution and realistic covariance estimates are discussed in chapter 4. Finally, the image coordinates become input to the triangulation routines discussed in section 2.3.1.

2.4.3 Feature Tracking

At each time t_i, the feature tracking operation locates existing landmarks in the stereo pair for t_i and computes the respective image coordinates and covariance matrices $\mathbf{q}_{l_{i,j}}$, $\mathbf{q}_{r_{i,j}}$, $\Sigma_{l_{i,j}}$, $\Sigma_{r_{i,j}}$. This information becomes input to the triangulation algorithms that compute the 3-D observations $\mathbf{Q}_{i,j}$ and $\Sigma_{\mathbf{V}_{i,j}}$. Feature tracking is accomplished using correlation in much the same way as stereo matching, subject to search constraints derived by applying prior knowledge of the camera motion (e.g. $\widehat{\mathbf{M}}_i^-$) to the previous landmark model, $\hat{\mathbf{P}}_{i-1,j}^+$. However, this basic idea can be instantiated in many ways. We will describe the method implemented here, then discuss alternatives and possible extensions.

Generally, some prior motion estimate $\widehat{\mathbf{M}}_i^-$ will be available from odometry and the vehicle

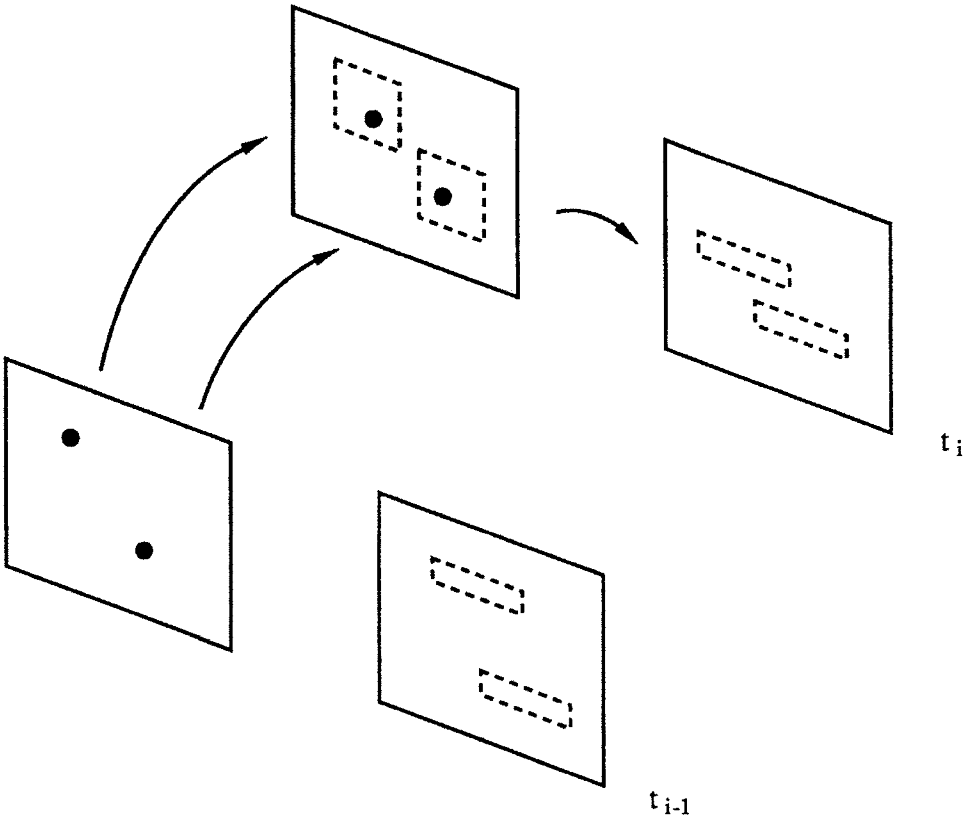


Figure 2.8: Feature tracking

Search windows are created in the left image of the new stereo pair using prior motion knowledge. The coarse-to-fine search procedure is then used to locate landmarks in the left image and then the right image of the new stereo pair.

controller. In the case of the Neptune vehicle used in section 2.5, the prior estimate was determined by a open-loop vehicle controller because no other motion sensor feedback was available from the vehicle. This information is used to transform the landmark estimates $\widehat{\mathbf{P}}_{i-1,j}^{+}$ forward in time to obtain the predicted coordinates relative to the next image pair, $\widehat{\mathbf{P}}_{i,j}^{-}$. The transformed 3-D coordinates are then projected onto the new image pair to obtain predicted image coordinates for each landmark. Search windows are established around each image prediction by reasoning about the uncertainty in the prediction. When a statistical model for $\widehat{\mathbf{M}}_i^-$ is available, the search window can be derived by propagating the random variable model into the predicted image coordinates and taking confidence intervals. In this implementation, a statistical model for $\widehat{\mathbf{M}}_i^-$ was not available, so search windows were established by propagating assumed worst-case error margins for $\widehat{\mathbf{M}}_i^-$ into the image coordinates. The prediction and window generation operation is illustrated in figure 2.8.

The prediction process generates search windows for both images of the new stereo pair. Because the camera geometry is known, an additional constraint is available in the form that the new image coordinates must lie on corresponding epipolar lines within the search windows. In principle, the coarse-to-fine correlation procedure could be applied in parallel to both new images to find new image coordinates satisfying both the search windows and the epipolar constraint[4]. A more straightforward approach was used in the implementation. First, the landmark was *reacquired* by using the correlator to find the landmark in just the left image of the new stereo pair. The search window in the right image was then contracted around the resulting epipolar line and the correlator was employed to match between the left and right images of the new stereo pair (figure 2.8). As in the stereo matcher, image coordinates were computed to pixel resolution and by default $\Sigma_{l_{i,j}} = \Sigma_{r_{i,j}} = \mathbf{I}$.The entire prediction, reacquisition, and matching operation is performed independently for each landmark in the 3-D model; after triangulation, this yields separate observations $\mathbf{Q}_{i,j}$, $\Sigma_{\mathbf{Q}_{i,j}}$.

We have already noted that if a statistical model is available for $\widehat{\mathbf{M}}_i^-$, this can be propagated to produce prior probability distributions for the predicted image coordinates. Search windows may be defined by confidence ellipses derived from these distributions. Furthermore, such distributions could be used in a Bayesian matching scheme that effectively applies a non-uniform weighting to the search window, making some pixels more likely matches than others *a priori*. Such a method was not implemented in this chapter; however, a similar insight plays a central role in later chapters.

Two other possible extensions are worth noting. First, in most situations the uncertainty in $\widehat{\mathbf{M}}_i^-$ will grow as a function of the absolute distance moved between frames; therefore, the size of the search windows will grow accordingly. Such growth will increase both the cost of feature tracking and the likelihood of error in feature tracking. It may be possible to analyze this growth to determine optimal image sampling rates. Finally, by tracking features independently, we fail to capitalize on correlations between predicted image coordinates. That is, to a certain extent errors between predicted and observed image coordinates must be consistent across all landmarks. This is not enforced by the current tracking procedure. Algorithms that track all landmarks in parallel, such as the global optimization approach developed in [Lucas84], in principle could enforce such consistency and thereby achieve more robust tracking. We leave this possibility for the future.

2.4.4 Error Detection

The feature matching and feature tracking procedures are not perfect: correspondence errors do occur. These errors must be detected, and the associated observation(s) must be rejected, before they corrupt estimates of the motion or the landmark coordinates. Therefore, several error detection mechanisms are incorporated within and following the stereo matching and feature tracking operations. We will summarize these mechanisms here; details of the derivations are given in appendix B.5.

First, the correlation coefficients computed by the coarse-to-fine search procedure are thresholded to reject matches with poor correlation. Poor correlation can result from several factors:

[4] For tracking, the 5×5 "source image" patch is taken from the previous image pair.

a landmark may have been occluded, may have fallen outside the field of view, or may have a markedly different appearance from the current viewpoint than it did from the previous viewpoint. In any event, the correlation threshold is an attempt to filter out such cases based on image appearance alone.

Two further tests use 3-D consistency considerations to filter errors. In the first, all features surviving the correlation threshold are subjected to a rigidity test. This is based on the constraint that landmarks must be stationary; therefore, a given set of landmarks should appear as a rigid cluster over time, with distances between landmarks remaining constant. The rigidity test enforces this constraint by verifying that distances between pairs of new landmark observations, $\mathbf{Q}_{i,j_1}$ and $\mathbf{Q}_{i,j_2}$, are approximately the same as the distances between the same pairs of estimated coordinates, $\hat{\mathbf{P}}^+_{i-1,j_1}$ and $\hat{\mathbf{P}}^+_{i-1,j_2}$, in the current landmark model. The test employs a loop that repeatedly rejects the landmark that appears to have shifted the most, until all changes are within a threshold (details are given in appendixB.5.1). Surviving landmarks are passed on to the motion estimation procedures. A major advantage of the rigidity test is that it can be performed before estimating the motion $\mathbf{M}_i$, since it does not require knowledge of the motion parameters.

The second 3-D consistency test uses outlier detection mechanisms within the motion estimation procedures. After computing a motion estimate $\widehat{\mathbf{M}}_i$, the outlier test computes *residual vectors*

$$\hat{\mathbf{v}}_{i,j} = \mathbf{Q}_{i,j} - \hat{\mathbf{R}}_i\hat{\mathbf{P}}^+_{i-1,j} - \hat{\mathbf{T}}_i$$

that reflect errors of fit between the existing landmark model, the new observations, and the inferred motion. The magnitude of these errors is used to reject the most outlying observation(s), after which the motion estimate is recomputed with the remaining landmarks. This process is repeated until all residuals are within a threshold. Appendix B.5.2 discusses this process in detail. For the experiments in section 2.5, only the rigidity test was employed.

2.4.5 Summary

We will conclude this section by summarizing the operation of the entire system, including the estimation and the image processing components. Figure 2.9 shows the combined processing loop. Before entering the loop, the system uses the feature selection and stereo matching procedures to obtain the image coordinates of a set of corresponding features from the first stereo pair. The triangulation and error modelling procedures use these coordinates to create the initial 3-D landmark model $\hat{\mathbf{P}}^+_{0j}$. Entering the loop, the robot vehicle then moves and acquires a second stereo pair. The known landmarks are located in the new images by using the *a priori* estimate $\widehat{\mathbf{M}}^-_1$ of the vehicle motion to create search windows in the new images and then applying the feature reacquisition and stereo matching routines to locate the features within the search windows. After triangulation, this produces the new 3-D observations $\mathbf{Q}_{1j}$. Since these observations may include gross correspondence errors, rigidity tests are applied as a filter. Next, the prior motion estimate, the previous landmark coordinates, and the new observations are used to compute a posterior estimate $\widehat{\mathbf{M}}^+_1$ of the vehicle motion together with updated estimates $\hat{\mathbf{P}}^+_{1j}$ of the landmark coordinates relative to the current coordinate frame. This estimation procedure may incorporate an additional error detection algorithm based on an outlier test. At this point, the number of

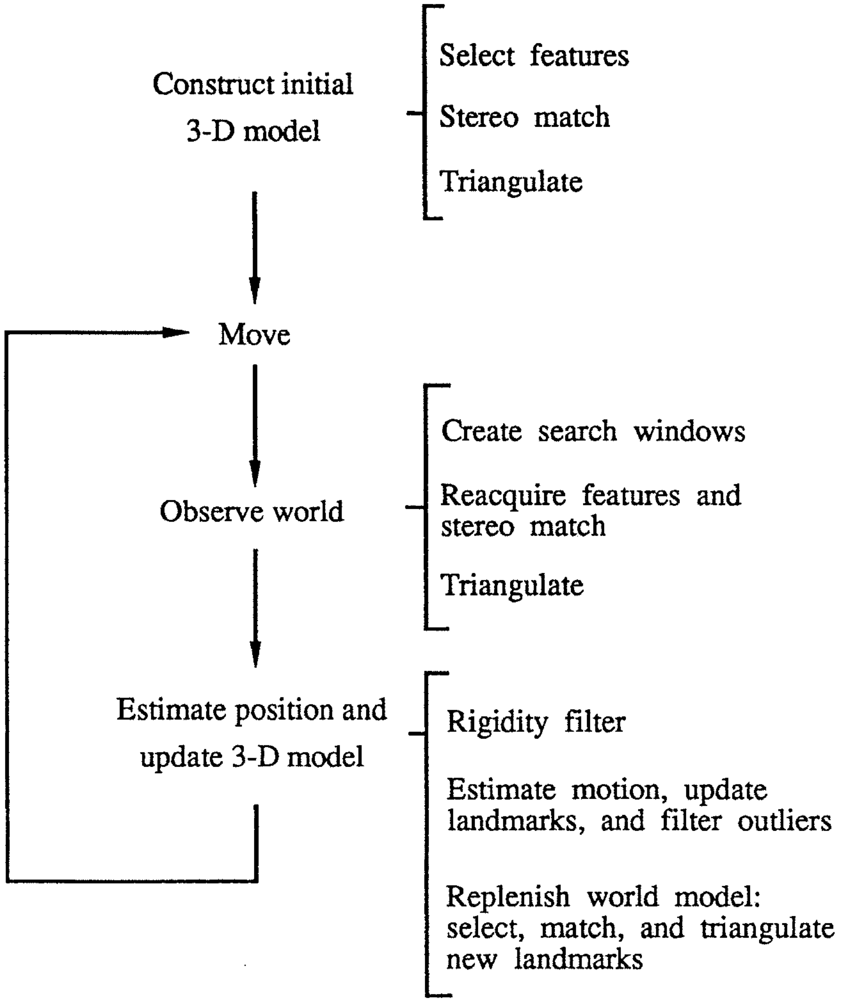

Construct initial 3-D model
Select features
Stereo match
Triangulate
Move
Observe world
Create search windows
Reacquire features and stereo match
Triangulate
Estimate position and update 3-D model
Rigidity filter
Estimate motion, update landmarks, and filter outliers
Replenish world model: select, match, and triangulate new landmarks

Figure 2.9: Expanded system loop flowchart

visible landmarks in the 3-D model will have been reduced due to matching errors and because some will now be outside the field of view. Therefore, the final step in the loop is to re-run the feature selection, stereo matching, and triangulation algorithms to replenish the world model.

In experiments, the system normally was configured to start with fifty landmarks in the world model. This number was chosen heuristically, based on experience with the reliability of the feature tracking operation. Field of view effects and error rejection tests typically reduced the number of landmarks remaining at the end of the loop to between fifteen and forty; the exact number was influenced by the amount of vehicle motion and the structure of the scene.

We examine the performance of the system in the following section.

2.5 Evaluation

The preceding sections have described the statistical formulation of our motion estimation problem, the numerical procedures used to solve it, and the image processing procedures used to detect and track landmarks over time. In this section, we evaluate the performance of the resulting system. The evaluation addresses three primary questions:

- How does performance with the full statistical model compare with the simpler model used in previous work?
- How does performance of each model vary as a function of distance to the landmarks?
- How do the motion estimates behave over time, given landmark tracking and sequential estimation?

We examine these questions via mathematical analysis, simulation, and laboratory experiments. We begin with a mathematical analysis of the correlation between successive motion estimates to see how the estimator should behave over time. We then use simulations to examine each of the above questions. Finally, we show results obtained with two image sequences obtained by driving a robot vehicle through a laboratory. These results confirm the conclusions drawn from the correlation analysis and demonstrate the robustness of the system.

2.5.1 Mathematical Analysis

Our primary concern here is to gain some understanding of how the estimates will behave over time. In particular, we wish to see what the effect is of continuing to track the same landmark over many frames. This effect, if any, will be reflected in the variance of global position and orientation estimates obtained by concatenating successive transformations $\widehat{\mathbf{M}}_i$. To examine this effect, we derive this variance for two successive transformations. For simplicity, we consider only translational motion.

Observations made at t_0 and t_1 can be written as

$$\begin{aligned}\mathbf{Q}_{0j} &= \mathbf{P}_{0j} + \mathbf{v}_{0j}\\ \mathbf{Q}_{1j} &= \mathbf{P}_{0j} + \mathbf{T}_1 + \mathbf{v}_{1j}\end{aligned}$$

Following the logic of section 2.3.2, we can eliminate $\mathbf{P}_{0j}$ to write

$$\mathbf{Q}_{10j} = \mathbf{Q}_{1j} - \mathbf{Q}_{0j} = \mathbf{T}_1 + \mathbf{v}_{10j},$$

where $\mathbf{v}_{10j}$ has covariance $\Sigma_{10j} = \Sigma_{\mathbf{v}_{1j}} + \Sigma_{\mathbf{v}_{0j}}$. We will denote the inverse of this covariance as $\mathbf{W}_{10j}$. With this notation, the maximum likelihood estimate of the translation is

$$\hat{\mathbf{T}}_1 = \left[\overline{\mathbf{W}_{10j}}\right]^{-1} \overline{\mathbf{W}_{10j}\mathbf{Q}_{10j}}. \tag{2.22}$$

Here the overline denotes summation over all landmarks. The covariance of $\hat{\mathbf{T}}_1$ is

$$\Sigma_{\mathbf{T}_1} = \left[\overline{\mathbf{W}_{10j}}\right]^{-1}.$$

We can derive the next motion estimate and its covariance in a similar fashion. The observations made at t_1 and t_2 can be written in terms of $\mathbf{P}_1$ as

$$\begin{aligned} \mathbf{Q}_{1j} &= \mathbf{P}_{1j} + \mathbf{v}_{1j} \\ \mathbf{Q}_{2j} &= \mathbf{P}_{1j} + \mathbf{T}_2 + \mathbf{v}_{2j}. \end{aligned}$$

From this, we obtain

$$\mathbf{Q}_{21j} = \mathbf{Q}_{2j} - \mathbf{Q}_{1j} = \mathbf{T}_2 + \mathbf{v}_{21j},$$

where $\mathbf{v}_{21j}$ has covariance $\Sigma_{21j} = \Sigma_{\mathbf{v}_{2j}} + \Sigma_{\mathbf{v}_{1j}}$ and $\mathbf{W}_{21j} = \Sigma_{21j}^{-1}$. The maximum likelihood estimate of the translation is

$$\hat{\mathbf{T}}_2 = \left[\overline{\mathbf{W}_{21j}}\right]^{-1} \overline{\mathbf{W}_{21j}\mathbf{Q}_{21j}}. \tag{2.23}$$

with covariance

$$\Sigma_{\mathbf{T}_2} = \left[\overline{\mathbf{W}_{21j}}\right]^{-1}.$$

Since (2.22) and (2.23) give the translation estimates as linear transformations of the Gaussian observations, standard error propagation methods [Mikhail76] can be used to derive the covariance matrix of $[\hat{\mathbf{T}}_1^T \; \hat{\mathbf{T}}_2^T]^T$. This is

$$\Sigma_{\mathbf{T}_1\mathbf{T}_2} = \begin{bmatrix} \Sigma_{\mathbf{T}_1} & -\Sigma_{\mathbf{T}_1}(\overline{\mathbf{W}_{10j}\Sigma_{\mathbf{v}_{1j}}\mathbf{W}_{21j}})\Sigma_{\mathbf{T}_2} \\ -\Sigma_{\mathbf{T}_2}(\overline{\mathbf{W}_{21j}\Sigma_{\mathbf{v}_{1j}}\mathbf{W}_{10j}})\Sigma_{\mathbf{T}_1} & \Sigma_{\mathbf{T}_2} \end{bmatrix}.$$

The diagonal elements are simply the separate covariance matrices of $\hat{\mathbf{T}}_1$ and $\hat{\mathbf{T}}_2$. The interesting observation is that the off-diagonal elements are negative; that is, $\hat{\mathbf{T}}_1$ and $\hat{\mathbf{T}}_2$ are negatively correlated. To see this more clearly, suppose that all of the observations have covariance $\sigma^2\mathbf{I}$; then $\Sigma_{\mathbf{T}_1\mathbf{T}_2}$ reduces to

$$\Sigma_{\mathbf{T}_1\mathbf{T}_2} = \frac{\sigma^2}{n}\begin{bmatrix} 2\mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & 2\mathbf{I} \end{bmatrix},$$

where a total of n landmarks are tracked. With this simplification, the covariance of $\hat{\mathbf{T}}_1 + \hat{\mathbf{T}}_2$ with tracking is $2\sigma^2/n\mathbf{I}$. In contrast, suppose that after estimating $\mathbf{T}_1$, we discard observations $\mathbf{Q}_{1j}$

and make new ones, $\mathbf{Q}'_{1j}$, to estimate $\mathbf{T}_2$. This will have the effect of decorrelating the motion estimates, so that the off-diagonal elements of $\Sigma_{\mathbf{T}_1\mathbf{T}_2}$ are zero. Therefore, without tracking, the covariance of $\hat{\mathbf{T}}_1 + \hat{\mathbf{T}}_2$ is $4\sigma^2/n\mathbf{I}$, or double what we have with tracking. We interpret this difference by observing that the negative correlation produced by tracking implies that errors in consecutive motion estimates tend to cancel each other.

Another useful way to view the same result is to consider the effect of an observation error at time t_1 for a single landmark. Using the same assumptions about the error model to simplify things, the translation estimates reduce to

$$\begin{aligned}\hat{\mathbf{T}}_1 &= \frac{1}{n}\overline{(\mathbf{Q}_{1j} - \mathbf{Q}_{0j})} \\ \hat{\mathbf{T}}_2 &= \frac{1}{n}\overline{(\mathbf{Q}_{2j} - \mathbf{Q}_{1j})}.\end{aligned}$$

Introducing an error $\mathbf{E}_{1k}$ into observation $\mathbf{Q}_{1j}$ then yields

$$\begin{aligned}\hat{\mathbf{T}}'_1 &= \frac{1}{n}\overline{(\mathbf{Q}_{1j} - \mathbf{Q}_{0j})} + \frac{1}{n}\mathbf{E}_{1k} \\ \hat{\mathbf{T}}'_2 &= \frac{1}{n}\overline{(\mathbf{Q}_{2j} - \mathbf{Q}_{1j})} - \frac{1}{n}\mathbf{E}_{1k}.\end{aligned}$$

When we compute the total translation, these errors cancel: $\hat{\mathbf{T}}_1 + \hat{\mathbf{T}}_2 = \hat{\mathbf{T}}'_1 + \hat{\mathbf{T}}'_2$. In addition to illustrating the results above regarding correlated errors, this example suggests that tracking tends to make the estimator robust against the occurrence of single outliers. That is, a single outlier may introduce a large error in the motion estimate for that time step; however, this error will be compensated for at the next time step.

This analysis has not considered the effect of using sequential estimation to refine the estimated landmark coordinates over time. Intuitively, we expect this to lead to more precise motion estimation than tracking without sequential estimation. We will not examine this theoretically. However, later we will use simulations to compare the performance of motion estimation without tracking, motion estimation with tracking, and motion estimation with tracking and sequential estimation.

2.5.2 Simulations

The simulations compare the performance of the least-squares estimator, the maximum-likelihood estimator, and the sequential Bayesian estimator. Since the primary difference between the least-squares estimator and the maximum-likelihood estimator is in the use of spherical versus ellipsoidal error models, we also refer to these as the "spherical" and "ellipsoidal" cases.

We present three sets of simulation results. The first is a base case that compares the standard deviations of position estimates obtained with each error model for a single step of vehicle motion. That is, it considers motion between only two consecutive stereo pairs. This illustrates the difference in the variability of position estimates with each model and reveals the effects on the motion estimates of coupling between the translational and rotational degrees of freedom. The second set also considers only two consecutive stereo pairs and tests limiting performance by tracking progressively more distant points. The last set examines the long range performance over many images of several different versions of our estimator.

The simulations were generated as follows. The "scene" consisted of random points uniformly distributed in a 3-D volume in front of the simulated cameras. For the first set of simulations, this volume extended 3 meters to either side of the cameras, 2 meters above and 1 meter below the cameras, and from 1.5 to 10 meters in front of the cameras. The cameras themselves were simulated as having 480 × 512 pixels and a field of view of 36 degrees. The stereo baseline was 0.2 meters. This duplicates fairly closely the conditions of the laboratory experiments to be described later. Image coordinates were obtained by projecting the points onto the images and adding Gaussian noise to the floating point image coordinates. These coordinates were input to the triangulation and motion solving algorithms. For the ellipsoidal error model, covariance matrices were computed as described in section 2.3.1. In the spherical case, weights were derived by taking the Z variance from the covariance matrix. Weights obtained by several other methods were tried and found to give very similar results. These include the volume and length of the major axis of the standard error ellipsoid and Moravec's half-pixel shift rule [Moravec80]. Trials were also performed in which the noisy image coordinates were rounded to pixel resolution, to simulate the effect of quantization error in the image coordinates. Since the results of these trials were essentially the same as the results without quantization, we present only the results obtained without quantization. The artificial noise in the image coordinates was uncorrelated between x and y and had a standard deviation in each coordinate of 0.1 pixels.

Single step, variable number of landmarks

The first set of simulations determined the standard deviation of the estimated motion between two consecutive stereo pairs when the true motion was 0.1 meters of forward translation, with no motion in the parameters. The results are shown in figures 2.10 and 2.11 plotted against the number of points used to compute the motion estimate.

For any given number of points tracked, the standard deviations are taken over 5000 random trials with entirely new points generated for each trial. In both figures, the top three curves were obtained with spherical modelling and the bottom three with ellipsoidal. Tilt implies rotation of the camera up or down, pan is the rotation about the vertical axis, and roll the rotation about the camera axis. The most significant thing to note is that the standard deviations obtained with the ellipsoidal model are a factor of 5 to 10 less than those of the spherical model. The size of the difference will vary with the distance to the points; for example, when they are within 1 to 2 meters of the cameras the factor is 2 to 4, and when they are within 2 to 5 meters it is 3 to 6. The case shown in the figures (points from 1.5 to 10 meters away) approximates the conditions of the indoor run with real data described later. Another point to note is that with the spherical model the estimates of roll and forward translation show less variation than the remaining parameters. This is because lateral translations and panning rotations have coupled effects on the errors of fit, as do vertical translations and tilting rotations. This shows up in the covariance matrix of the computed motion parameters as larger correlations between these

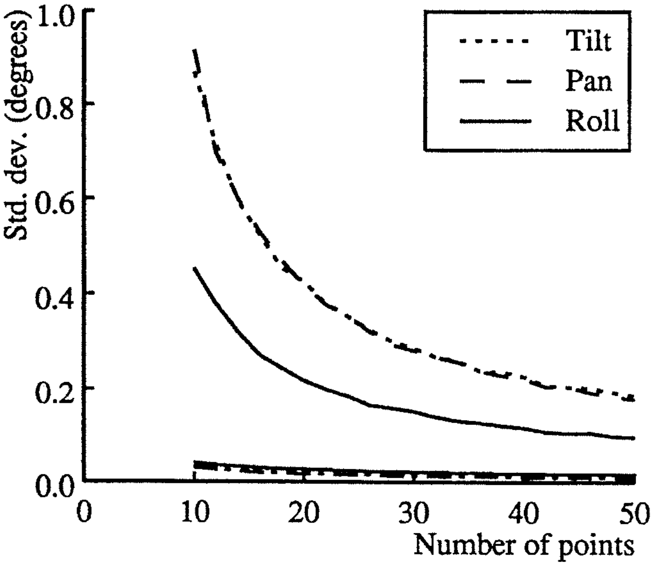


Figure 2.10: Standard deviation vs. number of points for rotations. Top three curves are for the spherical model, bottom three are for the ellipsoidal model (tilt and pan curves overlap).

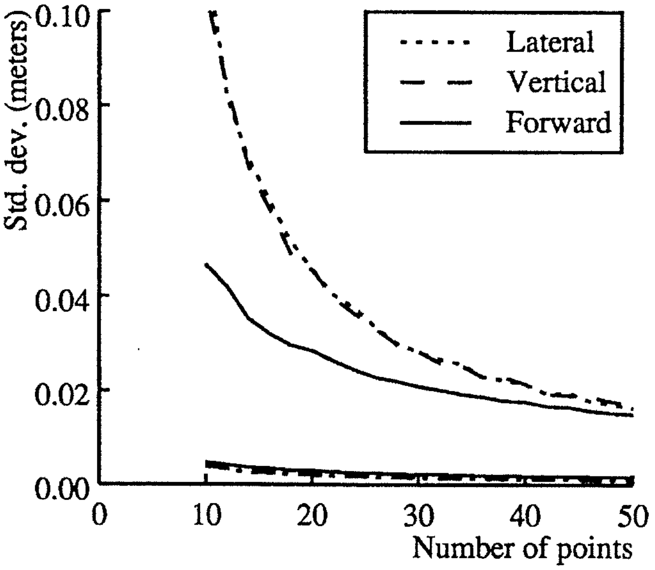


Figure 2.11: Standard deviation vs. number of points for translations. Top three curves are for the spherical model, bottom three are for the ellipsoidal model (lateral and vertical curves overlap).

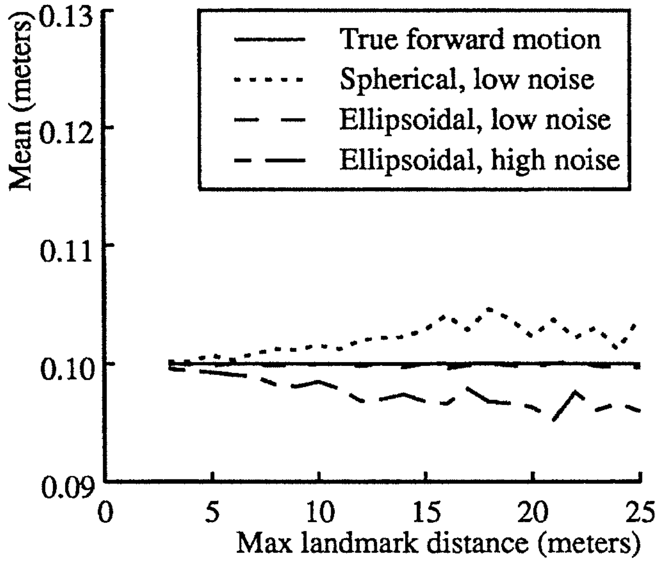


Figure 2.12: Mean estimated forward distance travelled vs. maximum distance to points

pairs of parameters than other pairs. These correlations are present with both error models, but the effects on the variance of the individual parameters is more apparent in the spherical case. Lastly, note that for a given level of performance fewer points are needed with the ellipsoidal model than the spherical, offseting the greater expense of the iterative motion solution needed in the ellipsoidal case. The exact relationship will depend on the camera configuration.

Single step, variable distance to landmarks

The second set of simulations illustrates the dependence of the standard deviation on the distance to the points in the scene. The initial volume for generating points was 1.5 to 3 meters away; this was expanded by moving the far limit back in stages until the final volume was 1.5 to 25 meters.

As with the previous experiment, for each volume 5000 random trials were performed with different landmarks generated for each trial. Ten landmarks were used for each trial. Figure 2.12 shows the mean of the forward translation estimates as a function of the maximum distance to the points. The true forward motion was 0.1 meter. Curves for a low-noise and a high-noise case are shown. In the low-noise case, the standard deviation of noise in the image coordinates was 0.1 pixels, as in the previous simulation; in the high-noise case, it was 4.0 pixels. Figure 2.13 shows the standard deviations of the motion estimates for just the low-noise case.

The standard deviation tells most of the story. With the ellipsoidal model, the standard deviation ranges from 1.5 percent to 12 percent of the true motion. On the other hand, with the spherical model the standard deviation is initially about seven percent of the actual motion and grows rapidly to 150 percent The other motion parameters, though not shown, behave similarly.

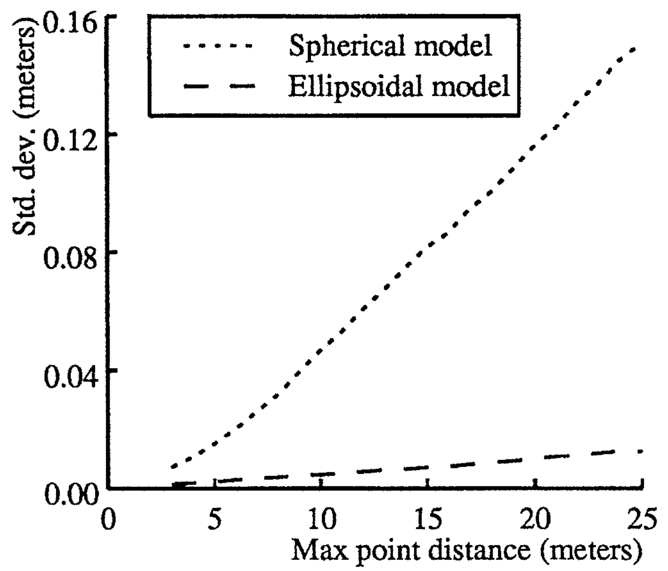


Figure 2.13: Standard deviation of estimated forward distance travelled vs. maximum distance to points

Looking at the means, with the ellipsoidal model the mean in the low-noise case is within 0.5 percent of the true motion for all distances. For the low-noise case with the spherical model, there is some bias toward over-estimation of the distance; however, the rapid growth of the standard deviation makes further interpretation of little value. In the high-noise case, the ellipsoidal model shows a gradually increasing bias toward under-estimation with increasing distance to the points. This was anticipated in section 2.3.1 as a possible result of the non-linearity in the triangulation operation. The existence of bias can be verified analytically, though we will not do so here. For the spherical model, estimates obtained in the high-noise case are completely unusable and the results are not shown. Thus, this experiment illustrates the strong contrast between the algorithms that develops with increasing distance to points.

Multiple steps

The last simulation looked at motion over a long sequence of images in order to compare the error growth experienced with different versions of the estimator. For this experiment, a single set of 10 randomly-generated landmarks was used for all trials with all versions of the estimator; thus, the only differences from trial to trial were the noise in the measured image coordinates and the estimator that was applied. This allows direct comparison of different estimators. Four variations of the estimator were examined:

- The maximum-likelihood estimator (ellipsoidal error model, equation(2.14)), with image coordinates in both the current and the previous image pair remeasured to estimate motion at each step. This is the uncorrelated case analyzed in section 2.5.1.

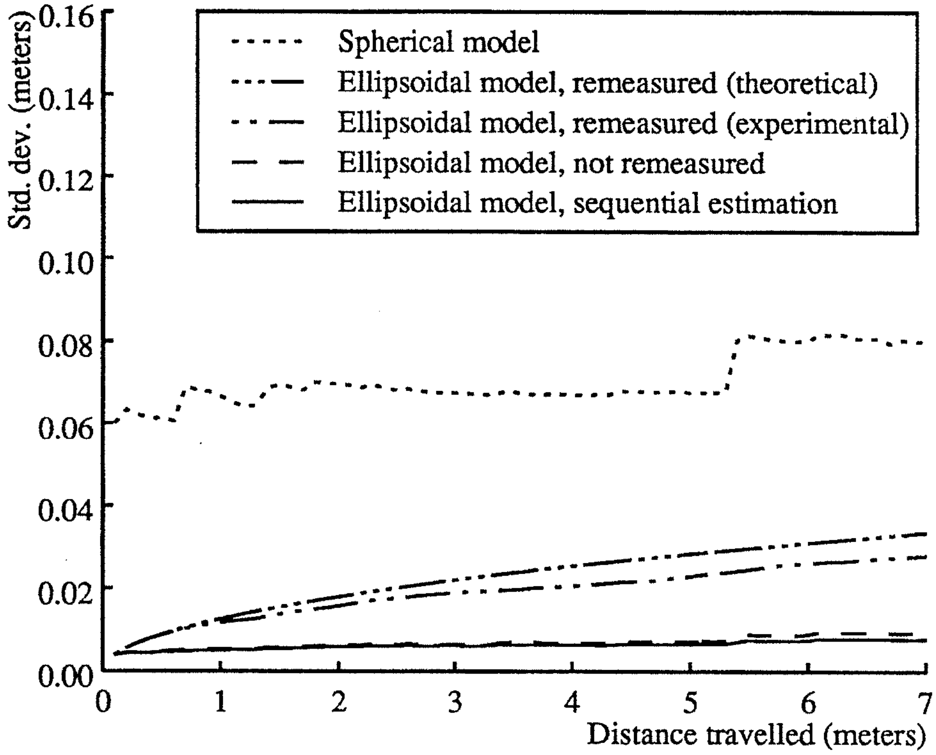


Figure 2.14: Standard deviation of estimated forward distance travelled vs. true distance

- The maximum-likelihood estimator without remeasuring image coordinates. This is the correlated case analyzed in section 2.5.1.
- The sequential Bayesian estimator (equations (2.18) through (2.20)).
- The least-squares estimator (spherical error model, from equation (2.10)), without remeasuring and without updating the landmark coordinates.

For simplicity, only translation was included in the estimators. The landmarks in this simulation were generated in a volume ranging from 1.5 to 10 meters in front of the cameras, with new landmarks added when existing ones passed out of view. The simulation covered a total of 70 steps of 0.1 meters forward per step. The noise standard deviation was 0.1 pixels. This imitates as closely as possible the first laboratory experiment described in the following section.

Figure 2.14 shows the standard deviations of the estimated total forward distance travelled, as a function of the true distance travelled. The top, dotted curve gives the results with the spherical error model; the remaining four curves give theoretical and experimental results with the ellipsoidal error model. As we expect, the least-squares results are distinctly worse than the other results. The fact that the curve does not rise steeply reflects the strong correlation between successive motion estimates. The abrupt increases in the curve, particularly at 5.4 meters, result from changes in the landmark configuration that occur when landmarks falling out of view are replaced by new landmarks. After each abrupt increase, the variance gradually decreases for a period of time. This is probably a reflection of the strong, negative correlation between successive motion estimates.

The second and third curves, that is the triple-dot-dash and double-dot-dash curves, show the theoretical and experimental results using the ellipsoidal error model with remeasurement, respectively. Based on the analysis in section 2.5.1, the theoretical curve is the function $\sqrt{n}\sigma_1^2$, where n is the step number and σ_1^2 is the variance for the first step from the experimental curve. The theoretical and experimental curves agree quite well. The fact that the experimental curve is slightly below the theoretical curve is explained by noting that in steady-state operation, landmarks will be somewhat closer on average than they are for the first few steps. Since the theoretical curve is obtained by extrapolating the results for the first step, we expect it to be somewhat higher than the steady-state results from the simulation.

The dashed and solid curves curves show results for the maximum-likelihood and sequential Bayesian estimators, respectively, without remeasuring landmarks as above. After two steps, the difference between the remeasured and non-remeasured maximum-likelihood estimates is in almost perfect agreement with the analysis of section 2.5.1. Moreover, the ratio of errors between the two approaches grows over time, so the importance of tracking is even greater than indicated by our initial analysis. The sequential estimator performs very marginally better than the maximum-likelihood estimator. We have not performed a thorough sensitivity analysis to determine what difference is to be expected. However, repeating the analysis in section 2.5.1 for simple examples of one-dimensional motions and one-dimensional landmark measurements suggests that little to no difference is to be expected. Therefore, it appears that, although sequential estimation does improve the estimated landmark coordinates, it does not significantly improve the estimates of vehicle motion.

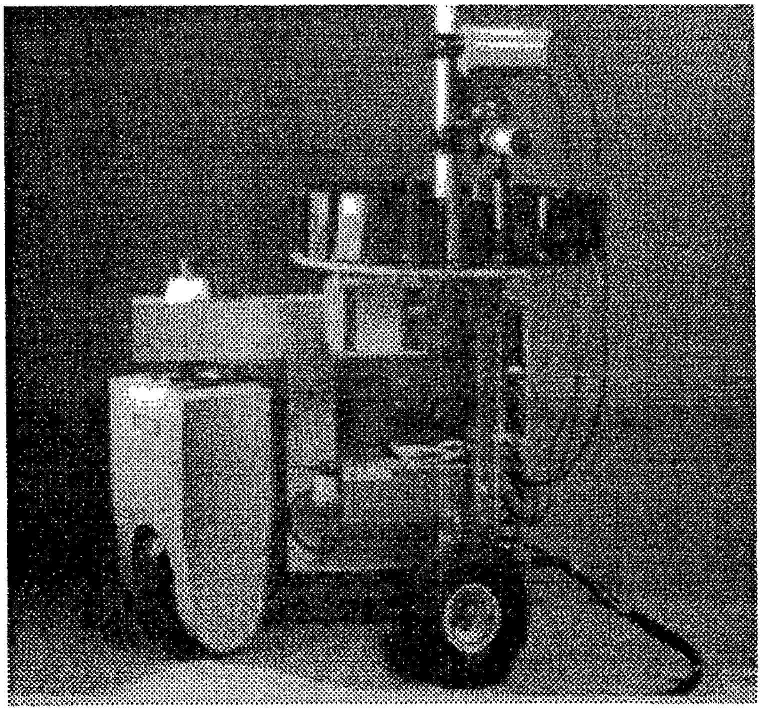


Figure 2.15: The robot vehicle "Neptune" used in the laboratory experiments

2.5.3 Laboratory Experiments

Experiments with the entire system were run with the "Neptune" robot vehicle [Podnar84] (figure 2.15) in the Mobile Robot Lab at Carnegie Mellon University. Neptune is a tricyle-type vehicle that provided a simple, mobile sensor platform for this and other research in autonomous navigation. It was steered and driven via the front wheel, while the rear wheels trailed passively. No on-board odometry was available. Power was supplied from off-board via a tether cable. An on-board 68000 controlled the motors and the sensors; all vision processing was done off-board on DEC Vax computers. Images were acquired with two, Sony CCD cameras set on a 20 centimeter baseline; 12.5 mm lenses were used, giving a field of view of roughly 36 degrees. These are the same specifications as used in the foregoing simulations.

For the experiments described here, the vehicle was driven manually through the room to acquire sequences of stereo image pairs. Two runs were made. For the first run, the vehicle was driven in a straight line in steps of approximately 10 centimeters between images, producing a sequence of 55 stereo pairs. The second run covered a curving trajectory, with each step not exceeding 7.5 centimeters in distance and five degrees in rotation; the resulting image sequence contained 94 stereo pairs. Figure 2.16 floor-plans of the laboratory, with dotted paths showing the actual position of the vehicle when each stereo pair was acquired. Images from these sequences were shown earlier in figures 2.6 and 2.7. The images were processed with the algorithms

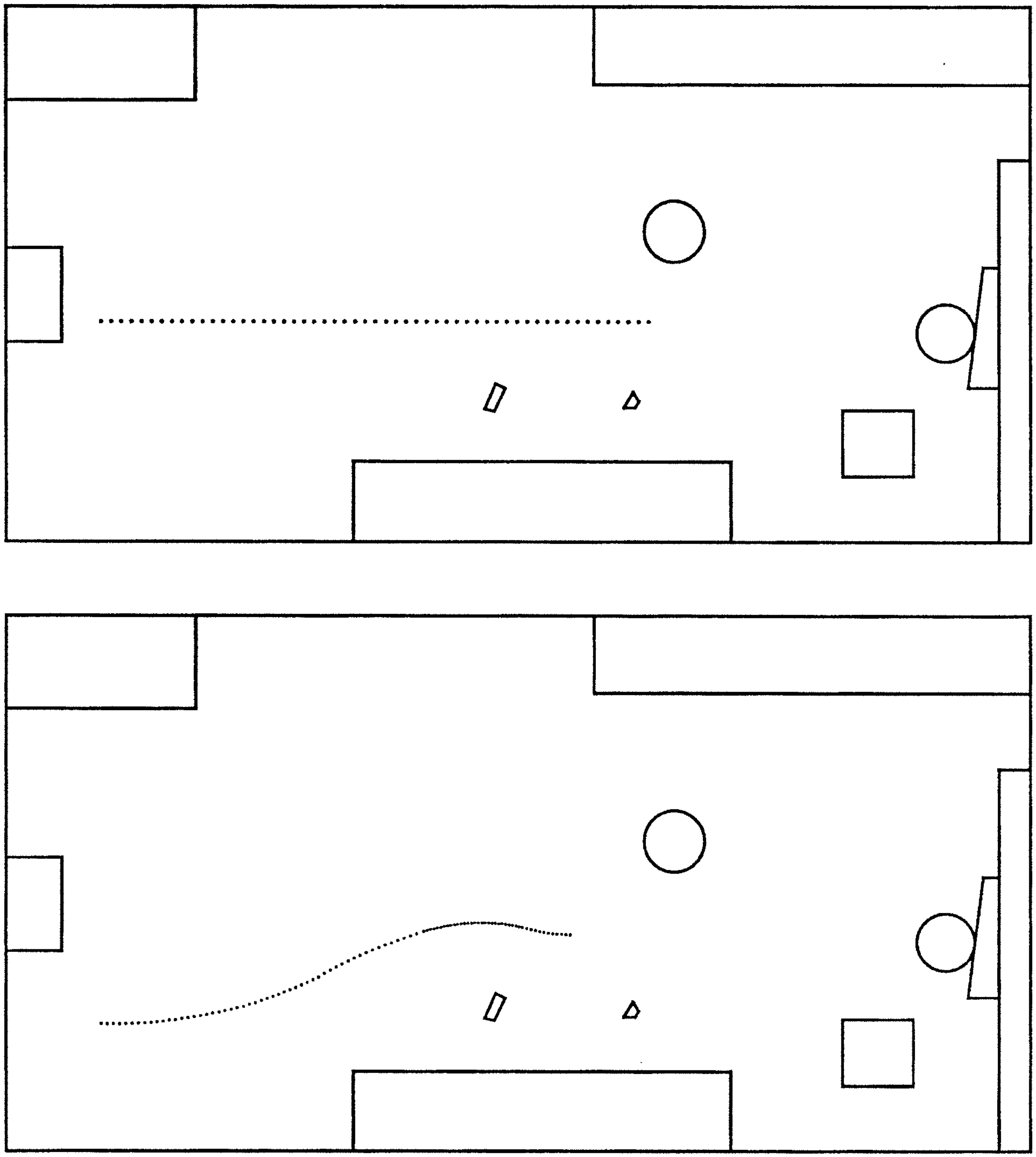


Figure 2.16: Floorplans of the Mobile Robot Lab showing true vehicle trajectories for the experiments with straight and curved motion. Dots mark the vehicle positions where images were acquired.

described earlier, with the exception that the outlier detection detection algorithm described in appendix B.5.2 was not implemented.

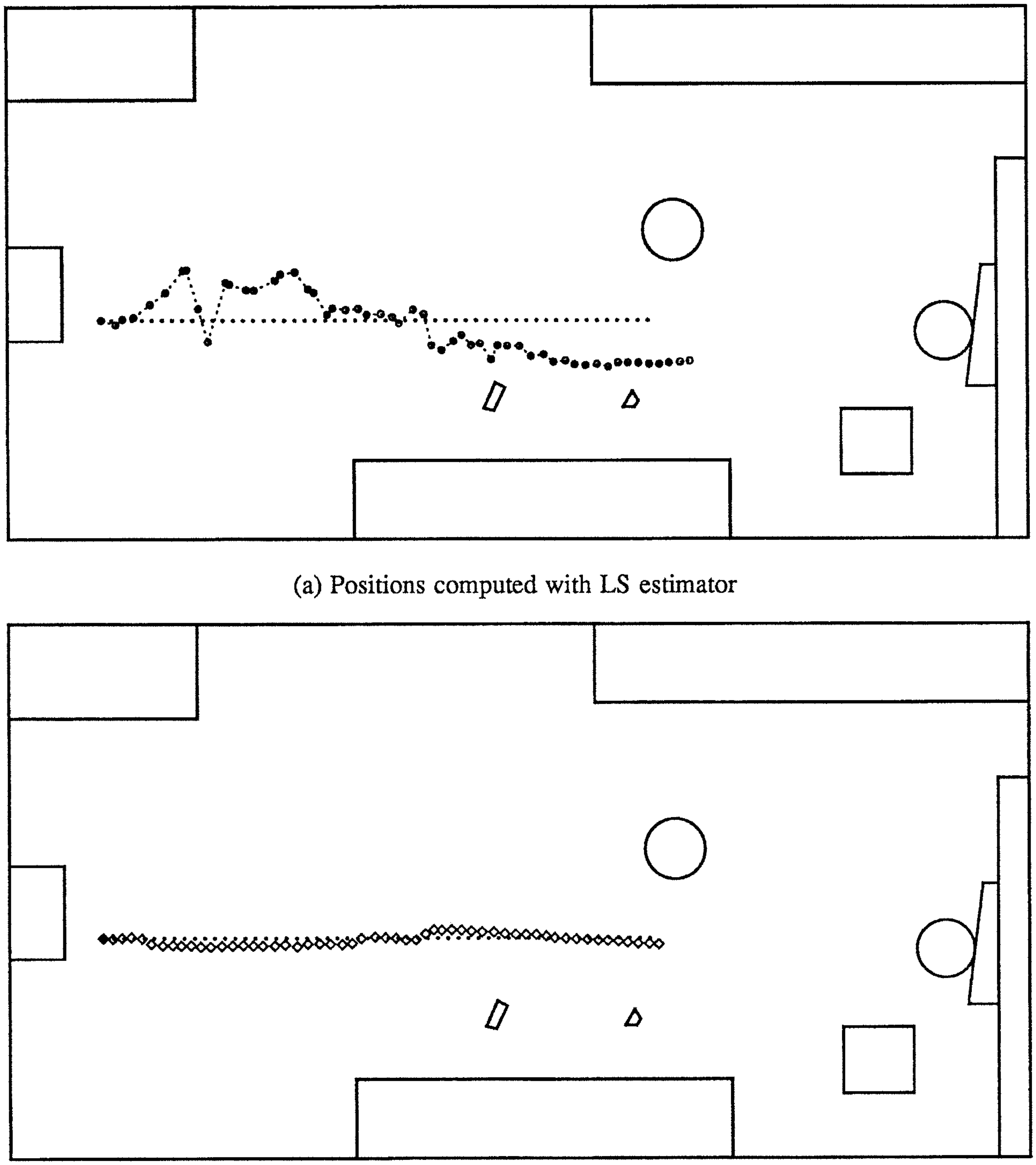
Figure 2.17 shows the vehicle positions estimated by the least-squares (LS) estimator and the maximum-likelihood (ML) estimator for the straight-line trajectory. Both estimators were constrained to estimate only motion in the floor plane; that is, two degrees of translation and one degree of rotation. The ML estimates are fairly stable throughout the trajectory. On the other hand, the LS estimates are fairly erratic early in the trajectory and only become relatively stable in the latter half of the trajectory. Note that the experimental conditions resulted in the majority of the landmarks being selected from the bookshelf against the far wall of the laboratory. Therefore, the motion estimates are consistent with the simulation results: spherical error model leads to poor performance when the landmarks are distance, better performance as the landmarks get nearer, and the ellipsoidal error model leads to good performance throughout. The final position obtained with the ellipsoidal error model was correct to within 2 percent of the distance and 1 deg of orientation. With the spherical model, the corresponding figures were 8 percent and 7 deg.

Repeating the experiment with six degree-of-freedom estimators led to similar results. It was notable that with the spherical model the error in roll was less than a degree, while in the other rotations it was between 5 deg and 12 deg. This is consistent with the observation made from the first simulation about correlations between estimates of rotation and translation.

The conditions of this experiment were very similar to those of the multi-step simulation. The results of the experiment and the simulation are in good agreement. In both cases, the LS estimator shows some erratic behavior, whereas the ML estimator is much more stable and precise. Regarding the precision of the ML estimator, the 2 percent error in forward distance achieved experimentally compares with a standard deviation of 0.13 percent obtained in the simulation. This difference of a factor of 15 is partly explained by the fact that in the simulation, features were localized to roughly one tenth of a pixel, whereas the implemented system computed correlation peaks only to pixel resolution. This suggests that a well-calibrated system with sub-pixel localization of features may achieve a precision better than 1 percent of distance. This compares favorably with odometry systems, which have been found to achieve about 1 percent [Marce86].

Figure 2.18 shows the estimated vehicle positions for the curved trajectory. In general, the results are similar to those for the straight trajectory. Looking at the results with the spherical error model for both the straight and curved trajectories, there is a tendency for large errors to from step to step to compensate. This is consistent with the correlation analysis we conducted earlier. For the ellipsoidal model, the estimated trajectories track the true trajectory fairly well in both cases, with the exception of two large errors made near the end of the curved trajectory. These are due to failures to reliably track a sufficiently large number of features to obtain accurate motion estimates. We expect that this can be remedied by implementing the outlier detection mechanism, plus additional mechanisms for monitoring conditioning and selecting landmarks according.

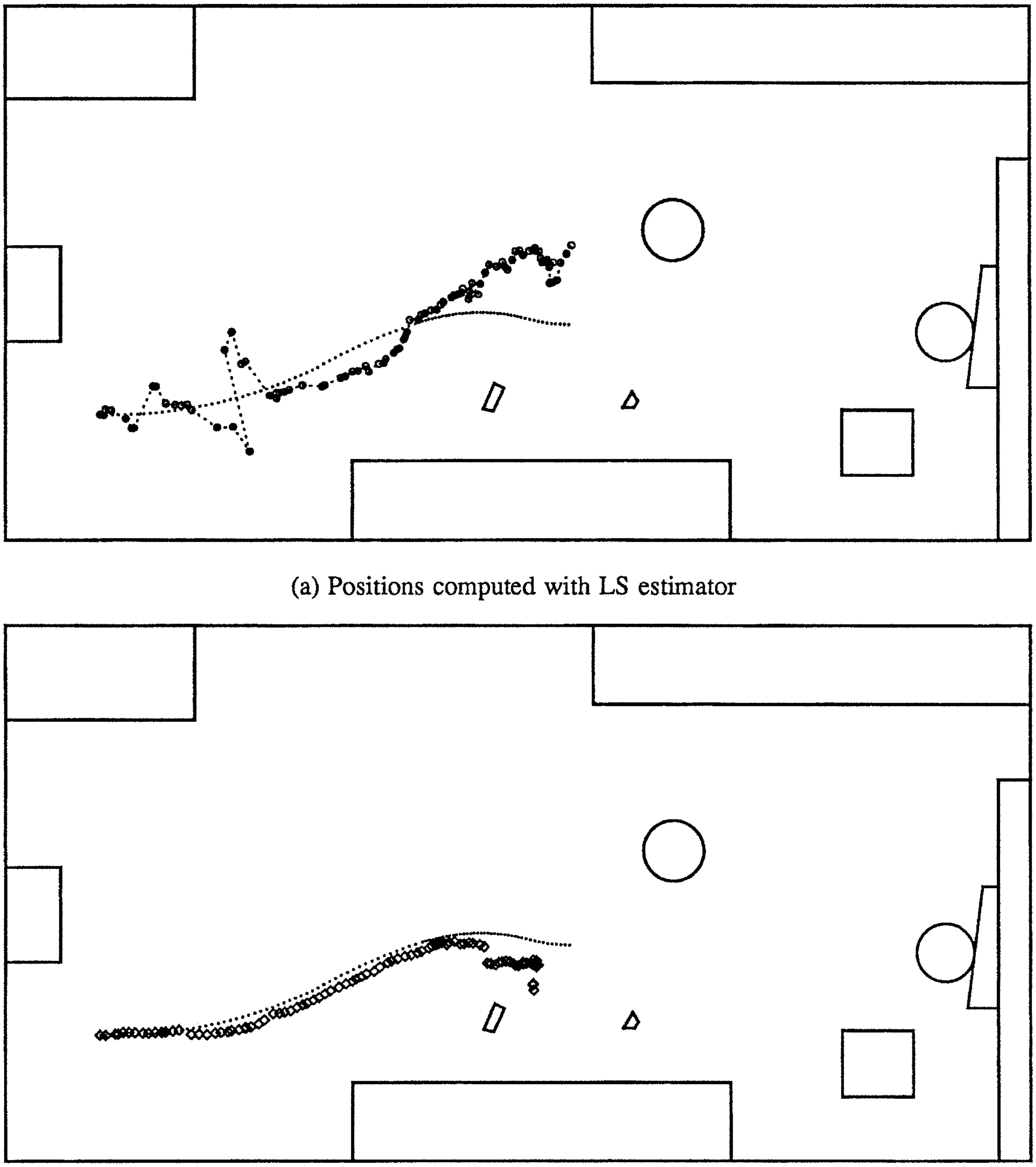
In conclusion, the results of experiments with real images support the conclusions drawn from the correlation analysis and the simulations. Moreover, the performance of the entire system is, on



(a) Positions computed with LS estimator

(b) Positions computed with ML estimator

Figure 2.17: Results for straight line motion



(a) Positions computed with LS estimator

(b) Positions computed with ML estimator

Figure 2.18: Results for curved motion

the whole, accurate and reliable. As noted above, even better performance should be achievable by tuning the implementation, as well as by using more precise calibration equipment than was available for this work. Thus, the results show that visual motion estimation may be competitive with other motion estimation systems on a precision basis; moreover, the whole system provides a good foundation for extension to other motion estimation problems.

2.6 Extensions and Related Work

Extensions of this work can go in many directions. Starting with unfinished business within the scope of this chapter, there are questions of performance and robustness with real image data that are unresolved. These include implementing and evaluating the adequacy of the outlier detection algorithm, estimating the noise level in the presence of outliers, and choosing landmarks so as to maintain adequate conditioning.

A first level of extension beyond the present scope involves related motion estimation problems for a single rigid body. One such problem is to estimate the parameters of a more extensive kinematic model. Appropriate kinematic models are well-developed in the engineering literature (e.g. [Wertz78]). Within computer vision, several successful efforts have been made to estimate single rigid-body motion when the object model is known in advance. This work has been done primarily in the context of aerospace applications; for example, monocular and binocular systems designed for docking or grasping satellites are described in [Gennery86,Tietz82,Wunsche86]. The general principles of recursive estimation of dynamic scene models are presented nicely in [Dickmanns88], together with applications to satellite docking, autonomous road following, and simulated aircraft landing. The next challenge is to extend such work to situations in which the object model is not known in advance, as was the case in this chapter. The main difficulty in doing so is to initialize a reliable scene description. The use of multiple image, together with the rigidity and outlier testing methods described, here should provide a good starting point.

Other extensions include the use of additional geometric primitives in the world model, global mapping of a static environment, and estimation in the presence of multiple rigid-body or deformable motions. Uncertainty modelling for line segments and planar patches is discussed in [Ayache88]. Recursive estimation of probabilistic world models for mobile robots is also discussed in [Kriegman87]. The global mapping issue is reminiscent of mapping from aerial photography, so we expect that closely related methods will be applicable. Relevant material is described in [Mikhail76,Vanicek86]. Applications of batch and recursive methods to similar problems in involving multiple coordinate frames in robotic workplaces are developed in [DurrantWhyte88,Smith87]. Estimation in the presence of multiple rigid-body motions introduces a new level of complexity, since it requires segmentation. A beginning is made on this problem in [Mulligan89,Zhang88]. Finally, we anticipate that thorough treatments of multiple rigid-body and deformable motion will require more complete models of disparity field estimation.

2.7 Summary

In this chapter, we have used stereo vision to estimate the frame-to-frame rotation and translation of a robot vehicle travelling through a static, unknown environment. This task is important because of its direct applications, because of its relevance to other pose-estimation problems, and because it is a precursor to more advanced problems in motion estimation.

Our approach was to track 3-D point landmarks and to use the apparent motion of the landmarks to estimate the actual motion of the vehicle. We introduced a statistical formulation of the estimation problem, using an estimation framework and Bayesian methods described in [Maybeck79]. The presence of large rotations make this problem non-linear. To solve it, we obtained initial estimates of the coordinate transformations by using an existing direct solution for a simpler uncertainty model [Schonemann70]; then we linearized our formulation and used iterative methods to refine the solution. We implemented a system for tracking point landmarks through real image sequences by extending methods previously developed in [Moravec80]. Finally, through simulations and laboratory experiments, we demonstrated (1) that our statistical formulation leads to a radical improvement in performance over previous work that did not employ an explicit statistical model, and (2) that the whole system performs accurately and reliably on long sequences of real images.

We identify three principal contributions of this work:

- it introduced statistical modelling of uncertainty to the problem of visual motion estimation in unknown environments and demonstrated the importance of such modelling,
- it integrated the estimation methods, image processing methods, and error detection methods necessary to make a system work in practice, and
- it gave the first demonstration of the feasibility of visual motion estimation in unknown environments.

A number of possible extensions to this work have just been described. Insofar as these concern estimation of rigid-body motion by tracking primitive geometric features, much relevant material already exists. However, problems of shape and motion estimation in unstructured environments require depth map estimation methods that go well beyond the selection and matching of primitive geometric features. A satisfactory approach to estimating such depth maps does not yet exist; therefore, we turn to this problem next.

Chapter 3

Depth Estimation: Overview

In the previous chapter, we solved a version of the single, rigid-body motion estimation problem by developing a system to estimate the position of a robot vehicle. We did this with a depth model consisting of 3-D point landmarks that we tracked through stereo image sequences. In the balance of our work, we consider the complementary problem of estimating depth maps from stereo image pairs. Our purpose in doing so is to build a more satisfactory foundation for addressing depth and motion estimation problems in complex, unstructured environments, as outlined in chapter 1.

To address this problem effectively, we must begin by developing perspective on what the problem is and what will be necessary to solve it reliably. That is, we must define what quantities we want to compute, we must formulate the problem of computing those quantities as a mathematical estimation problem, and we must consider what characteristics are necessary in the system design or operation to achieve a reliable solution. These are essentially the same steps that were executed in developing the system described in the previous chapter.

In the following sections, we define the problem of estimating depth maps and we motivate both the mathematical formulation and the system design we will employ. Components of the formulation and its operationalization are developed in detail in subsequent chapters.

3.1 Defining the Problem

We will start by reconsidering what it means to estimate depth from a stereo image sequence. The insight this provides will guide our approach to depth estimation with individual stereo image pairs. After establishing this context, the balance of our work will develop methods to estimate depth reliably for the first stereo pair in a sequence.

A stereo image sequence constitutes a pair of three-dimensional intensity functions $I_l(x,y,t)$ and $I_r(x,y,t)$, where t is the time dimension (figure 3.1). Relative to one of the cameras, say the left one, we denote scene depth by a function $Z(x,y,t)$ that gives the distance to the nearest object in the scene for all pixels of image sequence I_l. We assume that the stereo cameras are aligned so that objects appearing on a given scanline for the left camera appear on the same

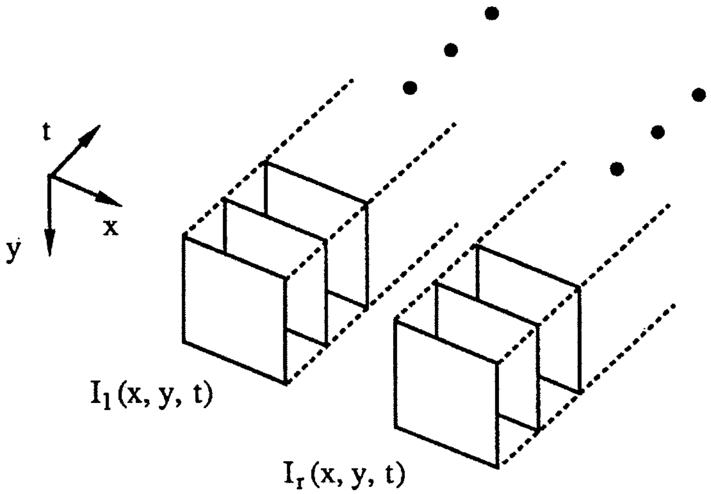


Figure 3.1: Problem: use sampled versions of I_l and I_r to estimate a sampled version of d.

scanline for the right camera[1]. In this case, an object seen at $I_l(x, y, t)$ in the left sequence will appear at $I_r(x - d(x, y, t), y, t)$ in the right sequence, where d, the *stereo disparity*, is inversely proportional to the depth Z. With this, a simple model of the relationship between the intensities of the left and right image sequences is

$$\begin{aligned} I_l(x,y,t) &= I(x,y,t) + n_l(x,y,t) && (3.1) \\ I_r(x,y,t) &= I(x + d(x,y,t), y, t) + n_r(x,y,t)\,, && (3.2) \end{aligned}$$

where I is an ideal, noise-free intensity signal, n_l and n_r represent noise in the images, and I_l and I_r are the intensities actually measured. Finding corresponding points boils down to finding d; likewise, maximizing the depth information extracted from the images amounts to estimating the disparity function $d(x, y, t)$ from the intensity functions $I_l(x, y, t)$ and $I_r(x, y, t)$. Estimating $d(x, y, t)$ is the long-term goal our research addresses. To proceed successfully, we must consider how to formulate this as an estimation problem *and* how achieve a reliable solution. These issues are the subjects of the following two sections.

3.2 Formulating the Estimator

Because the images contain noise, disparity estimates inevitably will be noisy. This suggests that we approach the above problem by formulating it as a statistical estimation problem, much as we did for motion estimation in chapter 2. Recall that the steps we took there were to define:

1. the variables to be estimated,

[1] This corresponds to the idealized camera model described in section 2.3.1.

2. the measurements available,
3. the mathematical model relating the measurements to the variables of interest,
4. the mathematical model of the uncertainties present, and
5. the performance criterion use to determine the "best" estimates.

We will proceed through the same steps here. In so doing, we are elaborating and adapting a statistical framework originally proposed in [Marroquin85] for an abstract image model. To keep things manageable, we will formulate the problem at a single scale of resolution.

(1) Whereas in chapter 2 the variables of interest were vehicle motion parameters $\mathbf{M}_i$ and 3-D landmark coordinates $\mathbf{P}_{i,j}$, in focusing on depth maps the variables we wish to estimate are the sampled disparities $d(x,y,t)$ for every pixel in the image sequence. The set $d(x,y)$ for each point in time is equivalent to the depth map we referred to in chapter 1 and is also known as a *disparity field.* In general, disparity can be a 2-D displacement vector; however, we will assume idealized camera geometry, in which case disparity is just a horizontal displacement for each pixel. We denote all pixels of the disparity field for each time t by the vector $\mathbf{d}(t)$.

(2) and (3) The measurements available are some form of comparison between the two image sequences. The comparison function can take many forms. We will consider just the simplest comparison obtained by differencing the two images obtained at each time t. For example (dropping the time index), to estimate the disparity at pixel (x_i, y_j), we measure the intensity differences between the two images for candidate values of disparity:

$$e(x_i, y_j; d(x_i, y_j)) = I_r(x_i - d(x_i, y_j), y_j) - I_l(x_i, y_j). \tag{3.3}$$

In practice, we actually examine intensity differences in a window around (x_i, y_j), with the assumption that disparity is constant over the window. We elaborate this model in chapter 4, where we also develop a linearized measurement model that approximates e as a linear function of d.

(4) We model the image noise functions n_l and n_r of equations (3.1) and (3.2) as stationary, Gaussian white fields. This makes the differences e random variables with distributions that are conditioned on the given value of disparity d. Therefore, for a given disparity field $\mathbf{d}$, the vector $\mathbf{e}$ of all measurements will have a conditional probability density $f(\mathbf{e}|\mathbf{d})$. Because we model the image noise as Gaussian, our model of $f(\mathbf{e}|\mathbf{d})$ will also be Gaussian. Details of the model are derived in chapters 4 and 5.

Prior information about $\mathbf{d}$ may be available from a number of sources, including terrain maps, other range sensors, and from previously processed images. This information generates uncertain predictions about the disparity at each pixel. We assume that this information can be modelled as a prior probability density for $\mathbf{d}$. The model we choose is to treat $\mathbf{d}$ as a Gaussian random vector with mean $\hat{\mathbf{d}}^-$ and inverse covariance matrix $\mathbf{W}_{\mathbf{d}}^-$. Therefore, for an image with M rows and N columns, the prior density of $\mathbf{d}$ is

$$f(\mathbf{d}) = (2\pi)^{-MN/2} |\mathbf{W}_{\mathbf{d}}^-|^{1/2} \exp\left\{-\frac{1}{2}\left[\mathbf{d} - \hat{\mathbf{d}}^-\right]^T \mathbf{W}_{\mathbf{d}}^- \left[\mathbf{d} - \hat{\mathbf{d}}^-\right]\right\}. \tag{3.4}$$

If $\mathbf{W}_\mathbf{d}^-$ is diagonal, then this density models the prior information as independent for each pixel, with prior means $\hat{d}^-(x,y)$ and variances $s^-(x,y)$. For example, such a model may be obtained by projecting information from another sensor, such as a laser scanner, onto the image plane of the cameras. The mean and variance at each pixel then characterize the predicted depth at each pixel and the level of uncertainty in the prediction. A non-diagonal $\mathbf{W}_\mathbf{d}^-$ implies correlation between pixels of the disparity field; consequences of this will be discussed later. Note that this probabilistic model makes the depth map a *random field* [Marroquin85,Marroquin87,Vanmarcke83].

(5) As in chapter 2, we use Bayes' theorem to derive a posterior density

$$f(\mathbf{d}|\mathbf{e}) = \frac{f(\mathbf{e}|\mathbf{d})f(\mathbf{d})}{f(\mathbf{e})} \tag{3.5}$$

from the prior and conditional densities and we employ the MAP criterion to define the "best" estimate $\hat{\mathbf{d}}^+$ of the disparity field. In general, the posterior inverse covariance matrix $\mathbf{W}_\mathbf{d}^+$ expresses the uncertainty in the estimated disparity field. We approximate only the diagonal elements of this matrix by deriving estimates of the posterior variance at each pixel in order to model the uncertainty in depth at each pixel. This is valuable, because this information may be very useful when disparity estimates are used by other parts of the robot system. For example, if disparity fields are used to build terrain maps for navigation, modelling the uncertainty in disparity allows us to model uncertainty in the terrain map, hence to take terrain uncertainty into account in motion planning.

The foregoing discussion gives a preview of how we will set up the estimation problem; now we will give an indication of how we eventually solve it. In general terms, applying the MAP criterion to (3.4) leads to an objective function defined over $\mathbf{d}$; minimizing this function defines our optimal estimate. To illustrate the nature of the objective function, as well as the issues involved in finding the global minimum, we will present a small example based on stereo algorithms in the literature. The objective function in the example is similar to one we obtain in chapter 5.

It has been popular [Barnard89,Horn86,Poggio85,Witkin87] to formulate the matching problem for each image pair as a variational problem. In abstract terms, this approach seeks to minimize an integral

$$q(d) = \int\int F(x,y,d,d_x,d_y,\ldots)\,\mathrm{dx}\,\mathrm{dy} \tag{3.6}$$

over possible disparity functions d, where F is a cost functional that measures the dissimilarity of I_l and I_r for candidate functions d. The functional may also depend on various derivatives of d. Typically, F measures the intensity error between the two images for any given d, as well as the departure of d from a pre-defined notion of how smooth it should be. A simple example is

$$q(d) = \int\int \left\{[I_r(x-d(x,y),y) - I_l(x,y)]^2 + \lambda\|\nabla d\|^2\right\}\mathrm{dx}\,\mathrm{dy}. \tag{3.7}$$

Here F measures the squared intensity error (e^2 in the notation of equation (3.3)) and the squared magnitude of the disparity gradient ($\|\nabla d\|^2$), with λ serving as a blending constant. The gradient term is a penalty function that biases the estimated disparity field to have low gradient. This penalty is a heuristic intended to capture the intuitive notion that surfaces are generally "smooth".

To make this example concrete, we discretize (3.7) by using forward differences to approximate ∇d. This replaces (3.7) by

$$\begin{aligned} q(\mathbf{d}) \quad = \quad & \sum_{x=1}^{N} \sum_{y=1}^{M} \left\{ [I_r(x - d(x,y), y) - I_l(x,y)]^2 \right\} + \\ & \lambda \sum_{i=1}^{N-1} \sum_{y=1}^{M-1} \left\{ [d(x+1,y) - d(x,y)]^2 + [d(x,y+1) - d(x,y)]^2 \right\}. \end{aligned} \tag{3.8}$$

We would minimize this objective function by searching over possible disparity fields **d**. This is closely related to the statistical framework discussed above. It has been shown that objective functions of this form can be equated to Bayesian estimation, where the intensity error term derives from the conditional density of the measurements and the disparity gradient term derives from the prior density of the disparity field [Poggio85,Szeliski88]. We elaborate on this connection in chapter 5, where we develop the statistical formulation in detail.

In practice, a number of problems make it difficult to find global minima of objective functions like (3.8). In particular, *false targets*, or matching ambiguity caused by such things as repetitive intensity patterns in the image, cause there to be multiple local minima. Discontinuities in the depth function that occur at object boundaries also make the search space discontinuous. If an initial estimate of the disparity field is available that is close to the true field, then it may be possible to use gradient descent to achieve the global minimum and find the correct estimate. The algorithm described in [Witkin87] takes this approach, using gradient descent in scale-space with a more elaborate objective function. However, if a good initial estimate is not available, gradient descent may not produce the correct result. In this case, a combinatorial search over possible disparity fields is required.

To summarize all of the discussion so far, in the previous section we motivated the problem of estimating disparity fields for a stereo image sequence. In this section, we chose to approach this as a statistical estimation problem and, restricting the discussion to a single stereo pair, we introduced the main steps we will take in formulating our approach. These steps will lead to objective functions that are minimized to estimate the disparity field. Gradient descent may be appropriate as the minimization algorithm if prior disparity estimates are available that are close to the true disparity; if such information is not available, the minimization will involve combinatorial search. To relate this back to stereo image sequences, the matching problem for the first pair of images (time t_0) is combinatorial. If images are acquired rapidly enough compared with the rate of variation of $d(x,y,t)$, then for $t > t_0$ the matching problem may be solvable with either gradient descent or combinatorial search. If images are acquired less rapidly, then the problem will involve combinatorial search for each pair of images.

The conclusion to draw from this discussion is that stereo matching reduces to an optimization problem that is generally difficult to solve. This has always been the crux of stereo research; the history of stereo research is largely one of trying to solve a combinatorial search problem efficiently and reliably. Experience has shown that this is actually a problem of both algorithm and system design. Therefore, our next step is to consider search algorithms and system design together, in order to identify a promising combination of the two.

3.3 Designing a Reliable System

Approaches to solving the stereo matching problem come in two basic types: those that augment the search algorithm and those that augment the sensor system. The first type seeks to use powerful search algorithms or knowledge about the scene to help find the best disparity field estimate for a given pair of images. Search algorithms include dynamic programming [Baker82,Ohta85], simulated annealing [Barnard89], methods based on accumulation of local support [Drumheller86,Marr76,Prazdny85,Stewart88,Szeliski85], and gradient descent or related methods that use multiple resolutions [Quam84,Witkin87]. Knowledge about the scene generally takes the form of heuristic assumptions about surface structure and is embodied in a variety of search constraints or penalty functions. These include the ordering constraints implicit in dynamic programming [Baker82,Ohta85], the "forbidden zones" discussed in [Drumheller86], penalty functions derived from surface smoothness assumptions [Barnard89,Boult88,Poggio85,Witkin87], and the various local support methods already mentioned. The search and knowledge-based algorithms have varying degrees of complexity and achieve varying levels of success. To date, there has not been a quantitative characterization of when or how well a given algorithm will work. Similarity, specific advantages are associated with specific techniques, but there is no one generally accepted set of search algorithms and knowledge sources that constitutes a "solution" to the problem.

The second type of approach uses more sensing, such as more images or combinations of images with other sensors, to constrain search or to resolve ambiguous image interpretations. Naturally, such methods can be used in place of or in addition to the methods above. Examples include trinocular stereo [Hansen88,Milenkovic85,Stewart88], combining stereo with focus [Krotkov88] or camera motion [Geiger87], and methods that process image sequences [Baker88,Bolles87,Matthies89,Xu85]. Such approaches are fundamentally more powerful than those that use only one stereo image pair, because disparity estimates can be verified by additional data instead of by agreement with assumed scene characteristics.

It is clear that to achieve reliability in complex, unstructured domains, methods from the second group must be employed. The problem is to find a combination of sensor configuration, sensing strategy, and search algorithm that can eventually perform well for stereo image sequences. Strong arguments can be made for using each of the redundant sensing methods listed above. Here, we observe that the use of fine motion to initialize stereo fusion is particularly attractive, because it can be made to work for cases in which even trinocular stereo will have difficulty and because it does not require controllable focus. It can also augment these other methods. Therefore, we choose to pursue the stereo/motion combination here.

Figure 3.2a illustrates one way to use fine motion to initialize stereo fusion. One or both cameras are mounted on a translation stage that can move the camera(s) parallel to the stereo baseline. Motion of one of the cameras is used to acquire a narrow-baseline image pair. The narrow baseline ensures that matching for this image pair will be comparatively easy; in the limit, success can be almost assured by shrinking the baseline. Depth information from this image pair is then used to constrain matching in a wide-baseline image pair acquired with both cameras. We refer to this whole procedure as a *bootstrap* operation. Other strategies combining

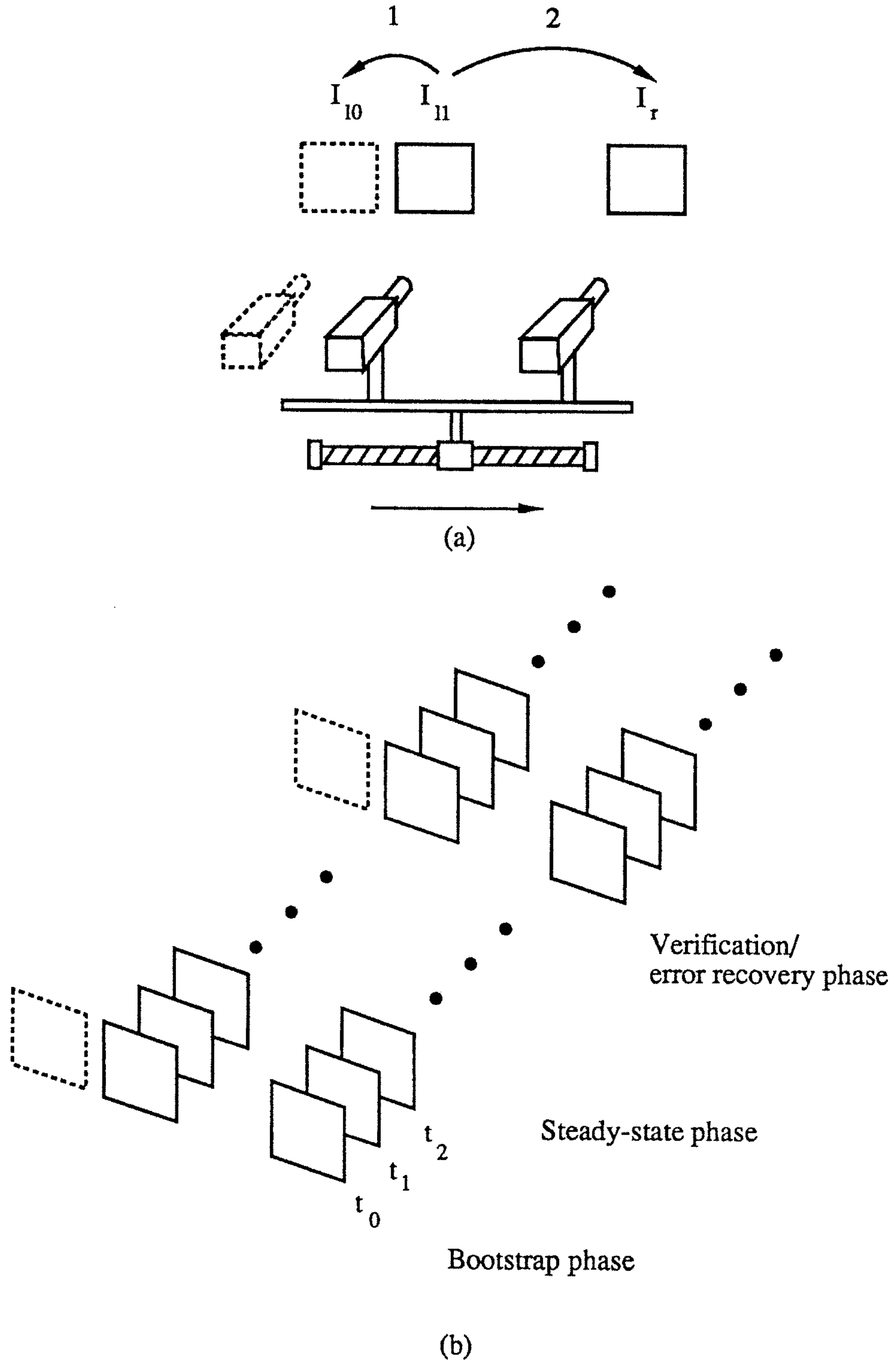


Figure 3.2: Operational framework: (a) bootstrap stage alone, (b) a larger scenario with bootstrap, steady-state, and verification/error recovery modes of operation

camera motion and stereo can be proposed [Geiger87]; however, this one is the simplest, is effective, and provides a good starting point for formalizing the approach.

Availability of the translational degree of freedom in the camera system is useful in a larger scenario involving stereo depth estimation over time. In this scenario, the bootstrap operation is used to obtain stereo fusion initially (figure 3.2b); thereafter, as the robot system executes task-oriented motions, depth maps for each new stereo pair are estimated using the depth map for the previous image pair to constrain search. We refer to this as a *steady-state* mode of operation. The correctness of such depth maps may degrade over time; if this can be detected, then the system can stop and use camera motion to re-initialize the depth in a *verification* or *error-recovery* operation.

This scenario raises many questions. In the remainder of this work, we confine ourselves to applying the single-scale, statistical formulation outlined in the previous section to the bootstrap operation. In chapter 4, we describe the measurement model in detail. This leads to a classical maximum-likelihood (least-squared-error) estimator for matching single pair of images. We derive the error variance of the estimator and experimentally examine the distribution of disparity estimation errors at individual pixels. This provides support for the Gaussian random field model of disparity. Chapter 4 also notes relationships between this estimator and the interest operator of chapter 2. Chapter 5 extends the maximum-likelihood estimator to Bayesian formulations for estimating the disparity field from the narrow and wide-baseline image pairs. This leads to efficient, area-based matching algorithms that estimate depth, either independently for each pixel or jointly for all pixels of each scanline. The performance of these algorithms is demonstrated with scale models of complex, outdoor scenes.

Chapter 4

Depth Estimation: Basic Disparity and Error Estimation

In the previous chapter, we outlined a statistical formulation of the depth map estimation problem. This involved probabilistic models of uncertainties in measured image intensities, prior depth information, and posterior depth estimates. Each of these uncertainties was modelled by a Gaussian random variable or, when the entire image is taken into account, by a Gaussian random field.

In this chapter, we begin to develop the details of the formulation and to validate the uncertainty models experimentally. We use measurements of intensity differences between images to derive a basic, maximum likelihood estimate of disparity and to derive the variance of the estimation error. We also review several general properties of maximum likelihood estimators that are important in our problem. These properties imply that the estimation error will tend to be Gaussian distributed. We then examine the estimation error experimentally, using real and synthetic images of constant-disparity scenes, to verify that these properties are observed in practice. In summary, we find that the statistical model is extremely good with synthetic images and satisfactory with real images. We also find that we can estimate the variance of these distributions reasonably well from the images themselves. We conclude that the Gaussian random field is a promising model of uncertainty for depth map estimation with real images.

We close this chapter by outlining some extensions to the basic estimator and by relating it to the interest operator and the landmark observation model of chapter 2. In the following chapter, we generalize the results of this chapter to Bayesian methods that incorporate prior disparity information, examine formulations for jointly estimating larger units of the disparity field, and apply these estimators to the bootstrap operation proposed in chapter 3.

4.1 Maximum-likelihood Disparity Estimation

Most area-based matching operators used in computer vision have their roots in statistical considerations similar to those we will employ below. However, the goals of these operators generally stopped at getting the best disparity estimate for pixels in an image. For the most part, the computer vision research community has only recently become concerned with the uncertainty of the disparity estimate, although this has been a concern in photogrammetry for a long time [Forstner86,Mikhail76,Ryan80,Vanicek86]. Exceptions to this statement are the area-based matchers described in [Gennery80] and [Anandan84], the theoretical treatments of the variance of extracted edge positions given in [Canny86,Nalwa86], and recent efforts to characterize the uncertainty in optical flow [Heeger88,Rives86]. Since the uncertainty of the disparity estimate is a central concern of ours, we will derive the disparity estimate and its error properties in detail. Our treatment is drawn primarily from similar, previous derivations in the photogrammetry literature [Forstner86,Ryan80] and from the thorough text by Van Trees [VanTrees68].

The disparity estimation problem requires finding the unknown shift between two noise-corrupted images $I_l(x,y)$ and $I_r(x,y)$. When the disparity is treated as a deterministic, unknown parameter, as it is in this chapter, maximum likelihood methods are appropriate for developing an estimator [VanTrees68]. Doing so involves three steps:

1. Defining a set of *observations* as functions of the unknown disparity.
2. Formulating the probability density of the observations conditioned on the disparity.
3. Determining the disparity estimate that maximizes the probability of the observations.

In general, the "disparity" may be a two-dimensional displacement vector. For clarity, we will start by considering only 1-D displacements in 1-D images. The extension to 2-D displacements in 2-D images is developed subsequently.

Formulation for 1-D Displacements

Before defining the observations, we will review our model of the stereo images themselves. We model the images as displaced versions of the same deterministic signal, with noise added separately to each image. Thus,

$$\begin{aligned} I_l(x) &= I(x) + n_l(x) \\ I_r(x) &= I(x + d(x)) + n_r(x) \end{aligned} \tag{4.1}$$

where $I(x)$ is the underlying deterministic signal, d is the displacement between images I_l and I_r, and n_l and n_r are the noise functions. For simplicity, we assume that n_l and n_r are stationary, Gaussian white noise processes with variance σ_l^2 and σ_r^2, respectively. Noise in real images is more complex; however, we verify in section 4.3.3 that the estimator performs well on real images despite of the simplicity of the noise model. The effect of perspective distortion between the two images is not expressed by equations (4.1) and is currently beyond the scope of our work.

We will define the observations as differences of a suitable representation of the local intensity variation in the neighborhoods of potentially matching pixels. In this thesis, the representation we use is the image itself and the observations are intensity differences in windows around the pixels being matched. Other representations, such as expansions in terms of localized basis functions [Adelson87,Kass84,Mallat87], may offer advantages in dealing with the issue of scale.

To find the disparity at pixel $I_l(x_i)$, we observe the intensity differences in a window around this pixel for each candidate disparity. Assuming that disparity is constant over the window, this gives a set of intensity errors

$$e(x_i + \Delta x_j; d) = I_r(x_i + \Delta x_j - d) - I_l(x_i + \Delta x_j) \tag{4.2}$$

where Δx_j indexes pixels in the window. We express the observations $e(x_i + \Delta x_j; d)$ together as the vector

$$\mathbf{e}(x_i; d) = [e(x_i + \Delta x_1; d), \ldots, e(x_i + \Delta x_n; d)],$$

where n is the size of the window. Under the noise model above, the conditional joint p.d.f. of $\mathbf{e}$ given d is

$$f(\mathbf{e}|d) = \frac{1}{(2\pi)^{n/2}\sigma} \exp\left(-\frac{1}{2\sigma^2}\mathbf{e}^T\mathbf{e}\right), \tag{4.3}$$

where $\sigma^2 = \sigma_1^2 + \sigma_2^2$ is the sum of the noise variances in both images. $f(\mathbf{e}|d)$ is called a *likelihood function* and a choice of d that maximizes it is called a *maximum likelihood estimate* (MLE). Maximizing (4.3) is equivalent to maximizing the *log-likelihood*,

$$\begin{aligned} \ell(d; \mathbf{e}) &= \ln f(\mathbf{e}|d) \\ &= -\frac{1}{2\sigma^2}\mathbf{e}^T\mathbf{e} + \text{constant terms}, \end{aligned} \tag{4.4}$$

which in turn is equivalent to minimizing the quadratic form. This is the familiar "squared intensity difference" matching criterion. This can be generalized by defining the observations as differences of linear transformations of the image (ie. convolutions). For one such approach, see [Kass86].

For digital images, minimizing (4.4) is accomplished in two steps. First, (4.4) is evaluated for every discrete d in a predefined search range to find the minimum to pixel resolution. This yields an initial estimate d_0 of d at pixel resolution. Then, an estimate of d at sub-pixel resolution can be obtained by taking a first-order expansion of $\mathbf{e}$ about $d = d_0$. This yields

$$\begin{aligned} e(x_i + \Delta x_j; d_0) &= I_r(x_i + \Delta x_j - d_0) - I_l(x_i + \Delta x_j) \\ &= I(x_i + \Delta x_j + d - d_0) - I(x_i + \Delta x_j) + n_r(x_i + \Delta x_j - d_0) - n_l(x_i + \Delta x_j) \\ &\approx \left[I(x_i + \Delta x_j) + (d - d_0) \left.\frac{\partial I(x_i + \Delta x_j + d - d_0)}{\partial d}\right|_{d=d_0} \right] - I(x_i + \Delta x_j) \\ &\quad + n_r(x_i + \Delta x_j - d_0) - n_l(x_i + \Delta x_j) \\ &= I'(x_i + \Delta x_j)(d - d_0) + n_r(x_i + \Delta x_j - d_0) - n_l(x_i + \Delta x_j). \end{aligned}$$

Since we are modelling the noise terms as white, we can abbreviate $n_r(x_i+\Delta x_j-d_0) - n_l(x_i+\Delta x_j)$ by $n(x_i + \Delta x_j)$ henceforth, where the variance of $n(x_i + \Delta x_j)$ is σ^2. Collecting all $e(x_i + \Delta x_j; d)$, $I'(x_i + \Delta x_j)$, and $n(x_i + \Delta x_j)$ into the vectors $\mathbf{e}$, $\mathbf{J}$, and $\mathbf{n}$, we obtain

$$\mathbf{e} \approx \mathbf{J}(d - d_0) + \mathbf{n}. \tag{4.5}$$

For implementation, the derivatives I' are estimated from I_l. Since I_l is noisy, the derivative estimates will also be noisy; this can be moderated by smoothing the image before differentiation.

With the linearized model of **e** in (4.5), the conditional density of **e** is

$$f(\mathbf{e}|d) = \frac{1}{(2\pi)^{n/2}\sigma} \exp\left(-\frac{1}{2\sigma^2}[\mathbf{e} - \mathbf{J}(d-d_0)]^T[\mathbf{e} - \mathbf{J}(d-d_0)]\right). \tag{4.6}$$

Taking the log of this and setting the derivative with respect to d to zero, we obtain the following, revised estimate of d:

$$\widehat{d} = d_0 + \frac{\mathbf{J}^T\mathbf{e}}{\mathbf{J}^T\mathbf{J}}.$$

This can be iterated to refine the disparity estimate. In practice, iterating will require estimating the intensity errors **e** at positions between pixels. This can be done by fitting curves to the discrete intensity image.

The uncertainty in the disparity estimate is expressed by the variance of the estimation error, $E[\tilde{d}^2] = E[(d - \widehat{d})^2]$. Assuming $\widehat{d}$ is unbiased ($E[\widehat{d}] = d$), standard error propagation techniques [Maybeck79] lead to the following estimate of the error variance:

$$E[\tilde{d}] = \frac{\sigma^2}{\mathbf{J}^T\mathbf{J}} \equiv \sigma_d^2. \tag{4.7}$$

As we discuss below, this expression is actually a lower bound on the error variance.

The variance estimate σ_d^2 relates the precision of the disparity estimate to the noise level σ^2 and the "edginess" of the images, as expressed by the squared intensity derivatives $\mathbf{J}^T\mathbf{J}$ [Forstner86]. Since these derivatives can be computed from I_l before attempting to match, the variance estimate can be used as an interest operator to decide where matching should be attempted [Forstner88]. In fact, the directional variance terms of the Moravec interest operator [Moravec80] used in chapter 2 are essentially the same as σ_d^2. Thus, (4.7) offers a specific definition of interest operators in terms of the achievable precision of disparity estimates at that point in the image. Equivalent expressions that relate the error variance to the image power spectrum are given in [Forstner89] and [Ryan80].

If σ^2 is not known in advance, it can be estimated from the residual intensity errors after matching is done. The estimate is given by

$$\widehat{\sigma}_d^2 = \frac{1}{n-1}\sum_{i=1}^{n}[I_r(x - \widehat{d}) - I_l(x)]^2,$$

where n is the number of pixels in the window. This is a special case of the *posterior estimate of the reference variance* [Mikhail76,Vanicek86] discussed in chapter 2 and appendix B.4. The denominator in this expression is the number of degrees of freedom in the data, which in this case is $n-1$ because it has been used to estimate one unknown. Note that $\widehat{\sigma}_d^2$ is a random variable itself, so consideration must be given to the variance of $\widehat{\sigma}_d^2$ before it is used. Since $(n-1)\widehat{\sigma}_d^2/\sigma^2$ has a χ^2 distribution with $n-1$ degrees of freedom [Mikhail76] and variance $2(n-1)$, the law for variance propagation through linear transformations implies that the variance of $\widehat{\sigma}_d^2$ will be

$2\sigma^4/(n-1)$. To see what this means in practice, in our experience a typical value for σ^2 in 8-bit images is about 1.3. For the 5×5 matching window used in our experiments, this leads to a standard deviation of $\hat{\sigma}_d^2$ of almost 0.5, which is very high relative to the true value. Better precision can be obtained by averaging over many non-overlapping windows.

Finally, we observe that the overlap of matching windows for nearby pixels will cause disparity estimates to be correlated for pixels separated by distances $\tau < w$, where w is the width of the matching window[1]. The existence of this correlation will be of interest later when we consider Bayesian formulations of the matching problem, so we will derive a model of the correlation here. From preceding results, for pixels x_i and $x_j = x_i + \tau$ we obtain disparity estimates as follows:

$$\begin{aligned}\hat{d}(x_i) &= d_0(x_i) + \frac{\mathbf{J}(x_i)\mathbf{e}(x_i; d(x_i))}{\mathbf{J}(x_i)^T\mathbf{J}(x_i)} \\ \hat{d}(x_j) &= d_0(x_j) + \frac{\mathbf{J}(x_j)\mathbf{e}(x_j; d(x_j))}{\mathbf{J}(x_j)^T\mathbf{J}(x_j)}\end{aligned}$$

We will abbreviate the quantities in these expressions by replacing the parameters x_i and x_j by the subscripts i and j. By definition, the covariance of $\hat{d}_i$ and $\hat{d}_j$ is $\sigma_{ij} = E[(\hat{d}_i - d_i)(\hat{d}_j - d_j)]$. Assuming that the disparity estimates are unbiased, this can be expanded to:

$$\begin{aligned}\sigma_{ij} = E[(\hat{d}_i - d_i)(\hat{d}_j - d_j)] &= E[\hat{d}_i\hat{d}_j] - d_i d_j \\ &= E\left[\left(d_{0i} + \frac{\mathbf{J}_i\mathbf{e}_i}{\mathbf{J}_i^T\mathbf{J}_i}\right)\left(d_{0j} + \frac{\mathbf{J}_j\mathbf{e}_j}{\mathbf{J}_j^T\mathbf{J}_j}\right)\right] - d_i d_j \\ &= \frac{\mathbf{J}_i^T E[\mathbf{e}_i\mathbf{e}_j^T]\mathbf{J}_j}{(\mathbf{J}_i^T\mathbf{J}_i)(\mathbf{J}_j^T\mathbf{J}_j)}.\end{aligned}$$

When $\tau = 0$, the above expression reduces to $\sigma^2/\mathbf{J}_i^T\mathbf{J}_i$, as we expect. For $\tau \neq 0$, $E[\mathbf{e}_i\mathbf{e}_j^T]$ is a matrix with all elements zero except for a diagonal of 1's at a distance τ from the main diagonal. Making the approximation that all elements of $\mathbf{J}_i$ and $\mathbf{J}_j$ are equal to the same constant c, the expression reduces to

$$\sigma_{ij} \approx \frac{(w - |\tau|)\sigma^2}{w^2c^2},$$

where w is the width of the window. Thus, the covariance function is approximately triangular for $|\tau| < w$ and zero for τ outside this region. Given the symmetric, tapering nature of the function, it may be reasonable to make the further approximation of modelling the covariance function $K_d(i,j)$ as exponential,

$$K_d(i,j) \approx \sigma_i\sigma_j\rho^{|i-j|},$$

for some $\rho \in (0,1)$, where σ_i and σ_j denote the standard deviations obtained at x_i and x_j, respectively. We will use this approximation in chapter 5 to design a joint Bayesian formulation of the stereo matching problem.

[1] The presence of correlated noise in the images would also induce correlation in the disparity estimates.

Formulation for 2-D Displacements

Estimating 2-D image displacements is important when epipolar lines do not correspond to scanlines, when tracking feature points through image sequences as in chapter 2, and when estimating 2-D optical flow [Anandan84,Heeger88]. In the 2-D case, we model the displacement estimate as a 2-D, Gaussian random vector and characterize the estimation error by the 2×2 covariance matrix of the probability density. Extending the 1-D formulation to 2-D is straightforward and has been done before. For completeness, we include the following derivation, which is based on that in [Forstner86]. Related derivations, both statistical and heuristic, have been given in several papers dealing with optical flow estimation [Anandan84,Heeger88,Nagel86].

First, let $\mathbf{x}_i = [x_p, y_q]^T$ denote a 2-D image coordinate vector, let $\Delta\mathbf{x}_j = [\Delta x_r, \Delta y_s]^T$ index pixels in a window around $\mathbf{x}_i$, and let $\mathbf{d} = [d_x, d_y]^T$ denote a 2-D image displacement vector. Then the observation equation (4.2) can be rewritten for 2-D as

$$e(\mathbf{x}; \mathbf{d}) = I_r(\mathbf{x}_i + \Delta\mathbf{x}_j - \mathbf{d}) - I_l(\mathbf{x}_i + \Delta\mathbf{x}_j).$$

Taking the first order expansion with respect to the vector $\mathbf{d}$ about an initial estimate $\mathbf{d}_0$ and proceeding as in the 1-D case, the updated estimate of the displacement vector is

$$\hat{\mathbf{d}} = \mathbf{d}_0 + (\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{e},$$

where $\mathbf{e}$ is the vector of intensity errors $[e_1, \ldots, e_n]^T$ and the Jacobian matrix

$$\mathbf{J} = \begin{bmatrix} \frac{\partial e_1}{\partial d_x} & \frac{\partial e_1}{\partial d_y} \\ \vdots & \vdots \\ \frac{\partial e_n}{\partial d_x} & \frac{\partial e_n}{\partial d_y} \end{bmatrix},$$

is evaluated at the initial estimate $\mathbf{d}_0$. As before, this estimate can be iterated until the correction is near zero. The 2×2 covariance matrix of the displacement estimate is

$$\Sigma_{\mathbf{d}} = \sigma^2(\mathbf{J}^T\mathbf{J})^{-1}. \tag{4.8}$$

The eigenvalues of this matrix determine the size of ellipses of constant probability; therefore, they determine how well localized is the displacement estimate. The ratio of the eigenvalues determines the eccentricity of the ellipses; the degree of eccentricity characterizes whether the displacement estimate is well localized in zero, one, or both dimensions. Being well localized in both dimensions is important in tracking feature points; thus, the interest operator of chapter 2 is closely related to (4.8). The degree of eccentricity also has significance in optical flow estimation; this is discussed in [Anandan84,Heeger88,Nagel86].

4.2 Properties of the Estimator

Several properties of maximum likelihood estimators are relevant to the performance of the operator above and to the experimental evaluation performed in section 4.3. These are whether the operator is unbiased, whether it achieves minimum variance, and whether the estimation errors are in fact Gaussian distributed.

An unbiased estimator is one for which the expected value of the estimation error, $E[\tilde{d}]$, is zero. For linear estimation problems with zero-mean, Gaussian noise, the MLE will always be unbiased. Therefore, disparity estimates will be unbiased when the images consist of a linear intensity ramp with added noise, $I(x) = ax+b+n(x)$. For arbitrary, nonlinear intensity variations, unbiasedness is not guaranteed. However, the MLE converges in probability to the correct value of the unknown parameter as the number of observations tends to infinity (ie. asymptotically). In image matching, this implies that larger windows are less likely to suffer from bias than smaller windows.

Regarding variance, it can be shown [VanTrees68] that for observations Y of an unknown parameter vector X,

$$Y = f(X) + n,$$

where n is noise, a lower bound for the variance of any unbiased estimator is given by the expression

$$E\left[\left(\frac{\partial \ln P(Y|X)}{\partial X}\right)^2\right]^{-1}.$$

This is known as the *Cramer-Rao bound* (CRB) and any estimator that achieves this bound is called *efficient*. The bound is achieved for linear problems with Gaussian noise. In other cases, the variance of a given estimator may be higher, but it will approach the lower bound asymptotically and will be close to the lower bound whenever the magnitudes of the estimation errors are small relative to the degree of nonlinearity [VanTrees68]. For our model of the disparity estimation problem, the CRB is the same as the variance estimate in equation (4.7). Therefore, for a linear ramp image we expect the error variance to be given accurately by (4.7), whereas for nonlinear intensity variations the actual variance of the estimator may be higher. The asymptotic property implies that the variance will approach the lower bound as the window size increases.

Finally, since our disparity field representation models depth estimates as Gaussian, we would like to know how well this model matches the actual sample distribution. Asymptotically, the MLE is Gaussian with mean equal to the true value of the unknown variable and variance equal to the CRB.

In summary, for linear ramp images with Gaussian white noise, the MLE is unbiased, minimum-variance, and Gaussian; for nonlinear images with possibly non-Gaussian, non-white noise, the MLE has these properties asymptotically. In the following section, we see to what extent these properties are observed experimentally with a specific size of match window.

4.3 Evaluation

The goal of our experimental evaluation was to verify that the estimator behaves approximately in accord with the foregoing mathematical model. This was done by checking for bias in the disparity estimates, by comparing the sample variance of the estimates against the theoretical lower bound, and by testing the normality of the sampling distribution. Three sets of experiments were performed. The first used synthetic, linear ramp images corrupted with synthetic, Gaussian white noise to verify the mathematical model under the conditions to which it applies exactly. The second used real images with synthetic noise to test the behavior when the intensity signal is realistic but the noise is ideal. Finally, the last experiment examines the behavior of the estimator with real images and real noise.

Implementing the operator requires computing intensity differences at arbitrary points between pixels, which in turn requires interpolating the image. Since we did not wish to make a study of the interpolation issue, we simply fitted the image with a cubic interpolating spline[2]. Intensity differences were computed by evaluating the spline at the given disparity value. Derivatives were estimated by central differences.

4.3.1 Linear Image with Synthetic Noise

With a linear ramp image, we expect estimates for small windows to be unbiased, to achieve the CRB, and to be Gaussian distributed. For this set of experiments, we generated reference images $I_r(x) = ax$ with slopes a of 2.0, 4.0, and 8.0. We then generated target images $I_t(x) = a(x + d)$ with offsets d of $-$ 0.2, 0.0, and + 0.2 pixels and added noise with variance $\sigma^2 = 1$ to each target image. For each offset, we performed 5000 trials of matching the reference to the target images with 5×5 windows around each pixel. At each pixel, we computed the sample mean and sample variance of the 5000 disparity estimates. These sample statistics were then averaged over several hundred pixels to get an impression of the general behavior of the estimator.

Table 4.1 summarizes the results. There is no appreciable bias. The sample variances are in close agreement with the lower bound. In fact, they average slightly below the lower bound for image gradients of 4.0 and 8.0. We have not explained this, but expect that it is due to minor numerical problems or imperfections in the random number generator. Figure 4.1 illustrates these results graphically by plotting histograms of the sample means for approximately 600 pixels. The histograms confirm the impression given by the averaged results in the table.

To visualize the adequacy of the Gaussian model for the estimation errors, Figure 4.2 shows a histogram of the disparity estimates for a single pixel plotted together with a Gaussian curve whose mean and variance equal the sample mean and sample variance for that pixel. The agreement between the histogram and the Gaussian curve is very close. A χ^2 goodness of fit test with this data accepts the Gaussian hypothesis at a 70% significance level. Therefore, for a 5×5 window under the ideal conditions of this simulation, the Gaussian model is very good indeed.

We conclude from these experiments that the behavior of the disparity estimator agrees very well with the mathematical model for linear ramp images with Gaussian white noise. Next, we check the behavior with more realistic image data.

[2] Parabolic blending [Rogers76]

Gradient	Sample mean (avg.)	CRB	Sample Variance (avg.)
2.0	0.00020	0.01	0.0101
4.0	0.00009	0.0025	0.00251
8.0	0.00004	0.000625	0.000627

(a) Results for linear ramp image, true disparity = 0.0 pixels

Gradient	Sample Mean (avg.)	CRB	Sample Variance (avg.)
2.0	-0.199	0.01	0.0100
4.0	-0.200	0.0025	0.00247
8.0	-0.200	0.000625	0.000616

(b) Results for linear ramp image, true disparity = - 0.2 pixels

Gradient	Sample Mean (avg.)	CRB	Sample Variance (avg.)
2.0	0.199	0.01	0.0100
4.0	0.200	0.0025	0.00247
8.0	0.200	0.000625	0.000616

(c) Results for linear ramp image, true disparity = 0.2 pixels

Table 4.1: Results for linear ramp image.

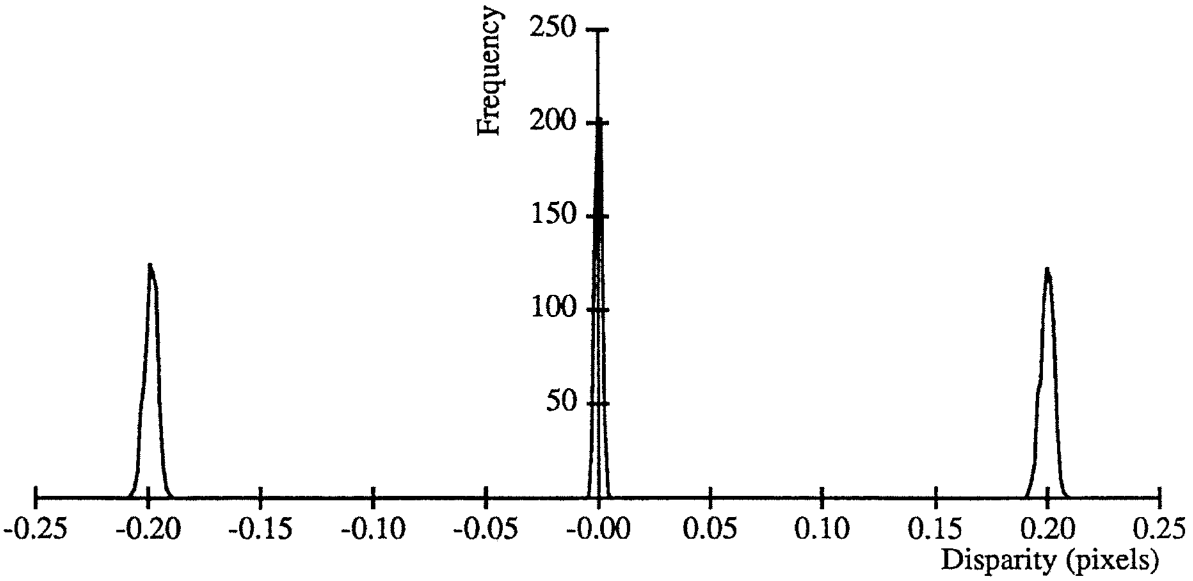


Figure 4.1: Bias plot, linear ramp image

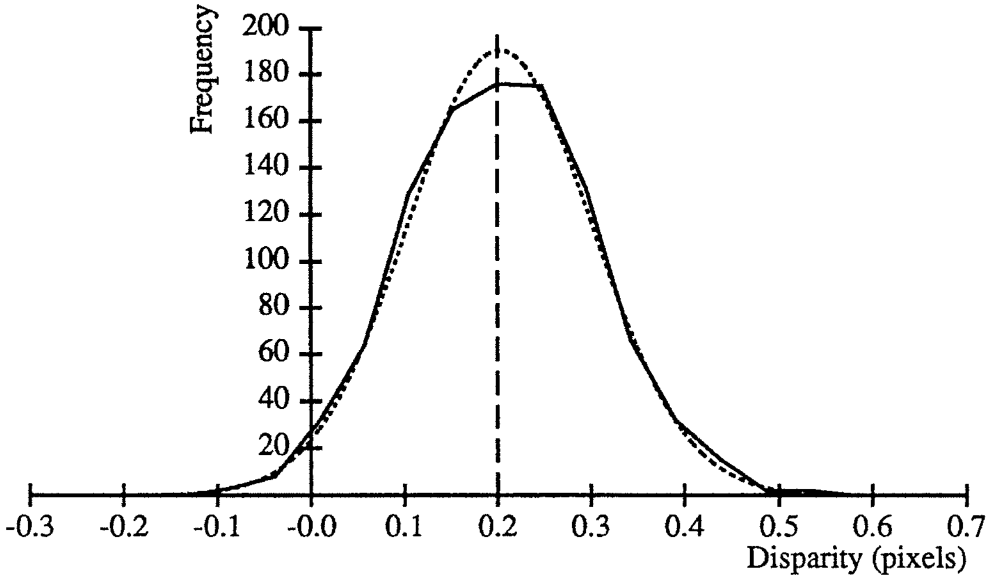


Figure 4.2: Disparity histogram, linear ramp image

4.3.2 Real Image with Synthetic Noise

The goals of the second experiment were to examine the bias, variance, and distribution characteristics with a real image when the noise was known to be Gaussian, white, and stationary. Therefore, the previous experiment was repeated with the center quarter of the image in Figure 4.3 used as the reference image. Target images were created by adding noise with $\sigma^2 = 1$ to the reference image. 5000 matching trials were performed for a true disparity of 0.0 pixels.

Figure 4.4 shows a histogram of the sample means. Over 90% of the means had an error of less than 0.001 pixels and the worst errors were less than 0.03 pixels. While cases leading to greater bias probably can be constructed, these results suggest that bias will not be significant with natural images.

Figure 4.5 plots the sample variance at each pixel against the variance estimate computed with equation (4.7) from the noise-free reference image. In other words, each point in the scatter plot compares the sample variance of the disparity estimates at one pixel with the theoretical lower bound. The points cluster very closely around the ideal, unit-slope line.

A χ^2 test for an arbitrary pixel accepted the Gaussian hypthesis at a significance level of 5%. The histogram of disparity errors is plotted with the Gaussian curve superimposed in Figure 4.6. The Gaussian model is satisfactory.

In summary, the simulations show that the estimator achieves near-optimum performance with a real image corrupted with synthetically generated Gaussian white noise.

Figure 4.3: Poster used for simulations with realistic data

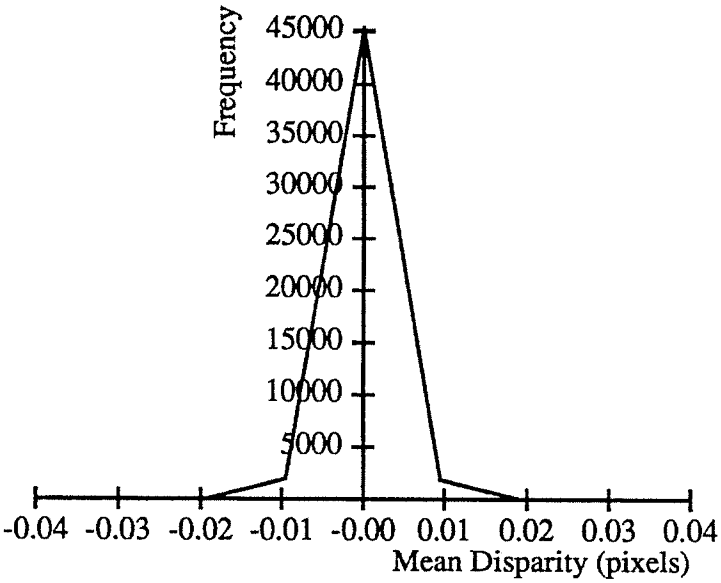


Figure 4.4: Bias plot, tiger poster

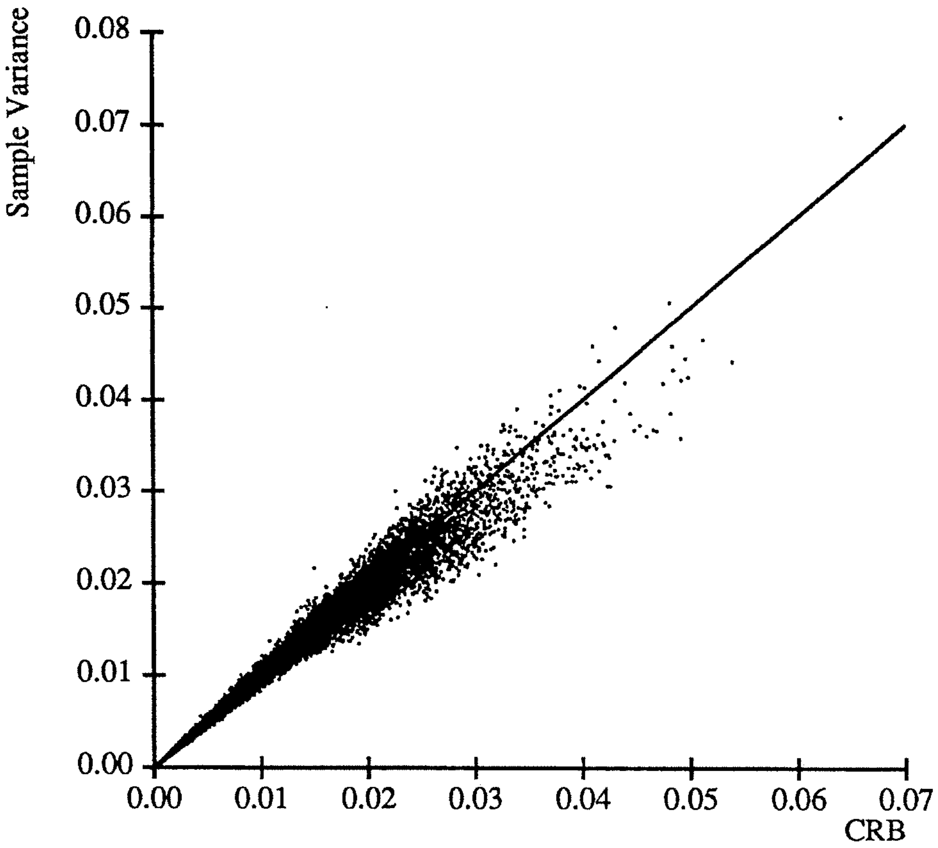


Figure 4.5: Theoretical variance lower bound (CRB) vs. sample variance. The line shows the theoretical variance, the dots show actual variance.

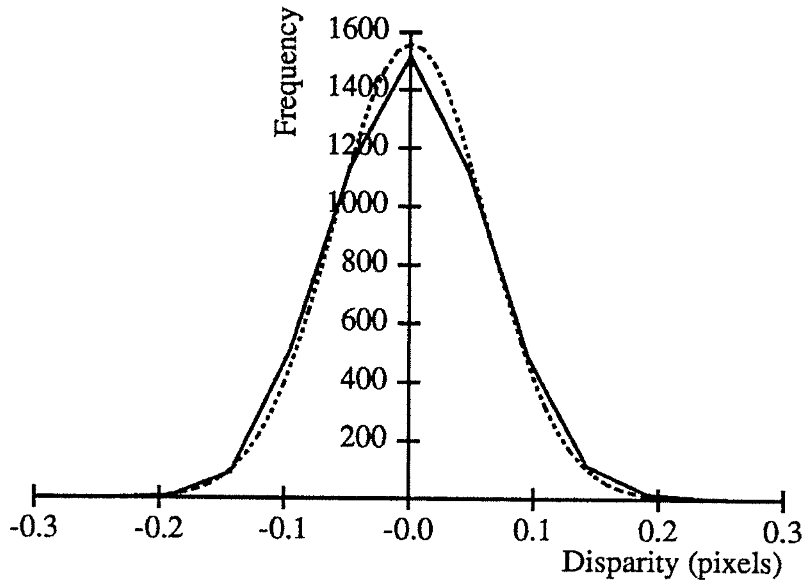


Figure 4.6: Histogram of disparity errors for a single pixel for the matching experiment with the tiger poster and synthetic noise. Superimposed is the Gaussian curve with mean and variance equal to the sample mean and variance.

4.3.3 Real Image with Real Noise

Noise in real cameras and digitizing systems departs from the idealized model used so far. Since modeling these characteristics more accurately is beyond our scope, the final set of experiments was done to verify that the estimator performs satisfactorily despite the inaccuracy of the noise model. In these experiments, two new images of the poster in Figure 4.3 were digitized for each matching trial. The true disparity was again zero. One form of noise compensation was performed. Electrical interference caused low-frequency, low-amplitude intensity fluctuations to roll down the images. To remove these, we subtracted out bias differences between corresponding scanlines of the image pair.

A histogram of the sample means computed for a 60 × 70 pixel segment of the image is shown in Figure 4.7. The results agree closely with those observed in simulation. The maximum deviation from the true disparity of zero is less than 0.02 pixels.

A histogram of 5000 disparity estimates at a single pixel is shown in Figure 4.8 with a Gaussian curve superimposed in the same manner as for the simulation result shown in Figure 4.6. The χ^2 test accepts the Gaussian hypothesis at a significance level of 50%. The agreement with the Gaussian model is excellent.

Finally, two factors complicate evaluation of the variance of the estimator. First, both images contained noise in this experiment, so the variance estimates at a single pixel fluctuate from image pair to image pair. Second, the image noise level is not constant, so efforts to compute ensemble statistics by averaging over time are complicated by the non-stationarity of the noise. Assuming

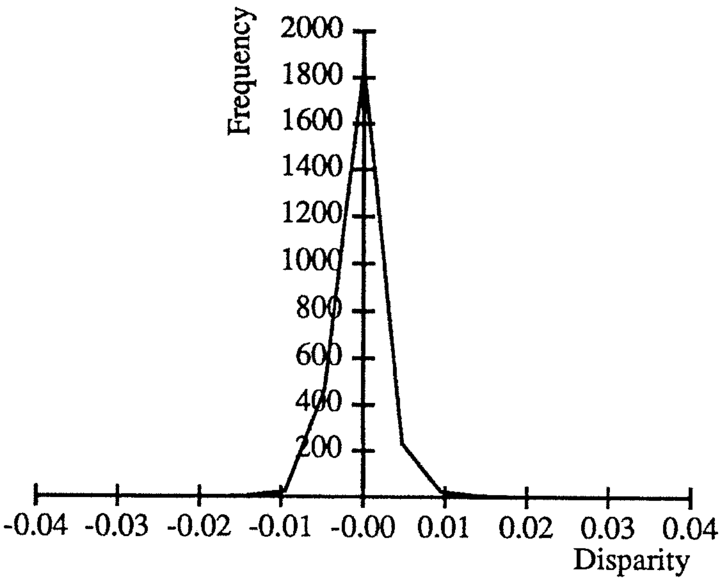


Figure 4.7: Bias histogram for real image with real noise

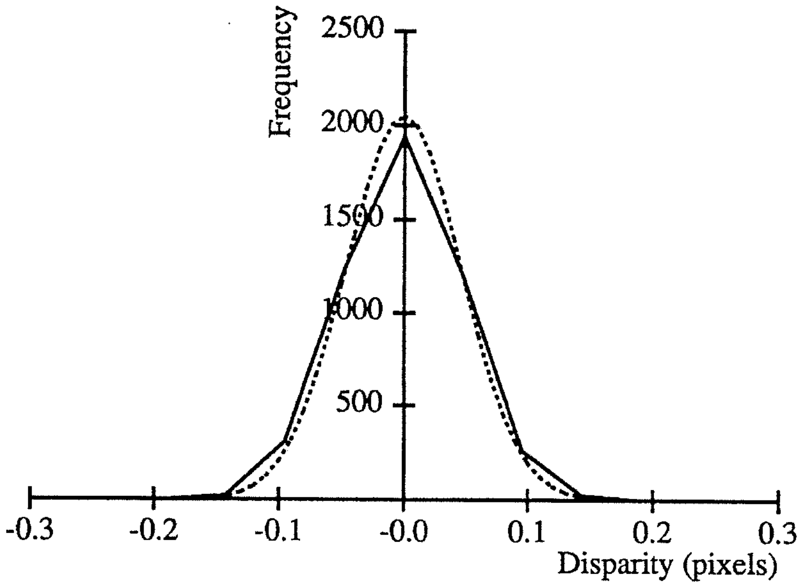


Figure 4.8: Disparity histogram with Gaussian curve for real image with real noise

that the noise is approximately stationary within each image, a simple way to overcome these problems is to compute sample variances by spatial averaging with a single image pair instead of time averaging over many image pairs[3]. This is done by taking the sample variance for all pixels with estimated variance within a small distance from a nominal value, then comparing the sample variance to this nominal value. By partitioning the range of the estimated variances and applying this technique to each partition, we obtain comparison plots analogous to Figure 4.5.

Results for disparity estimates from the entire tiger image are shown in Figures 4.9 and 4.10. For these results, σ^2 was estimated as described in section 4.1. Figure 4.9 shows the estimated standard deviation versus the sample standard deviation, while Figure 4.10 shows the number of pixels in each partition[4]. The agreement between the estimated and actual standard deviation is quite good, except for a constant offset of about 0.14 pixels. This offset may reflect timing jitter in the digitizer scanline clock. The larger variation at higher σ's may be due to the smaller number of pixels available for computing sample statistics, as indicated in Figure 4.10. We conclude that the results indicate that it is possible to obtain meaningful uncertainty estimates from the images themselves.

4.3.4 Conclusions

To recapitulate the results of the experiments, we found that the estimator did not produce significant bias in any of the experiments. The distribution of the estimation errors with a 5×5 match window was sufficiently close to Gaussian in all cases tried. The variance of the estimator was close to the theoretical lower bound for both the ideal, linear ramp image and for the realistic image when the noise was white and Gaussian. For the real image with real noise, the image noise level was determined from the images being matched by averaging the reference variance esimates over the whole image. Using this estimate for σ^2, we found that the variance of the estimator was again in moderately good agreement with the theoretical lower bound.

In conclusion, it was found that the Gaussian model of disparity uncertainty is a good description of the experimentally observed estimation errors. It was also found that the variance of the estimation errors could be estimated fairly well directly from the image pair using equation (4.7). Therefore, the variance estimate is a good description of the uncertainty in the disparity estimate and may be useful in subsequent reasoning about disparity uncertainty. Limitations of these conclusions lie in the fact that the experiments used a fronto-parallel, non-specular surface translated parallel to the image plane. The impact of surface slant, shininess, and other effects must be examined in the future.

[3] This technique was developed by Rick Szeliski

[4] We plot standard deviation instead of variance on the assumption that most people will find it easier to relate this to error as a fraction of the pixel size

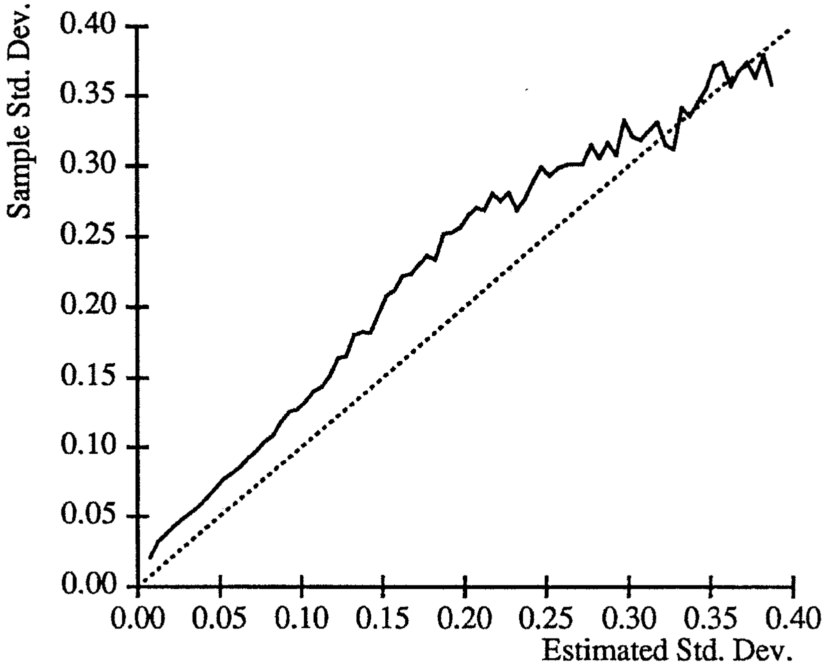


Figure 4.9: Variance scatter plot with spatial averaging

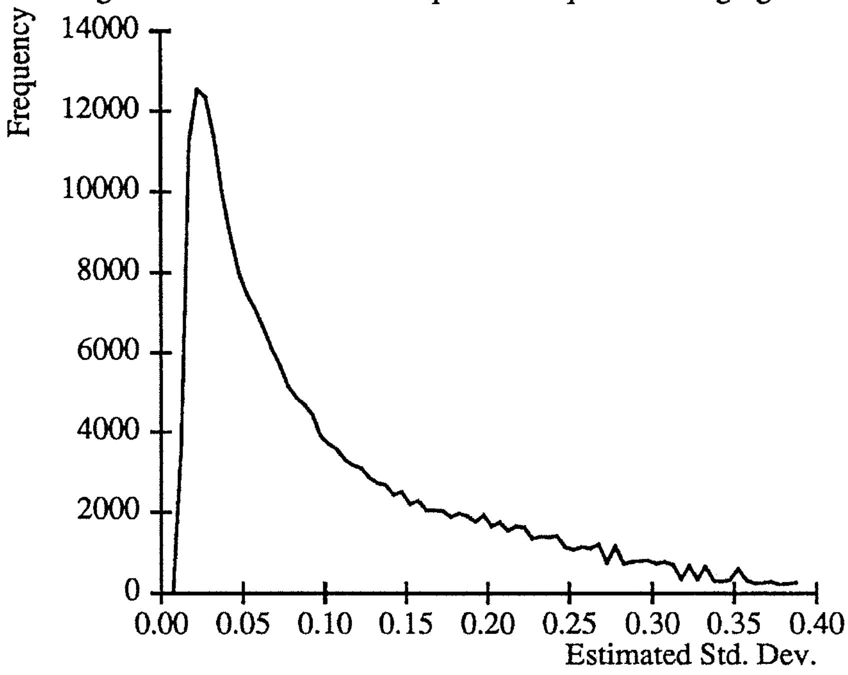


Figure 4.10: Number of pixels in each partition. Partition size is 0.005 sigma.

4.4 Extensions and Related Work

Many extensions to this work suggest themselves. Foremost is to improve the matching and/or noise models for real images. Modifying the matching model of equation (4.1) to include a linear transformation of the window from image I_l to image I_r [Forstner86,Gennery80], with the scale and offset terms of the transformation treated as unknowns, may improve the estimator. We have not considered the effect of correlated noise on the estimator. This may be important if the noise is spatially correlated or if image prefiltering is performed, for example to reduce resolution, since this may introduce correlations even if the noise was originally uncorrelated. The effects of perspective compression and non-Lambertian reflectance also need to be analyzed. Extensions at a higher level of abstraction include characterizing the probability of error in the disparity estimate and dealing with the issue of scale. Error probability has been discussed in [Gennery80]. A multi-scale approach to obtaining image difference measurements is given in [Kass86,Kass84].

4.5 Summary

The goals of this chapter were to elaborate a basic, probabilistic formulation of image matching and to examine the validity of the uncertainty models employed. To do so, we derived a maximum-likelihood, sub-pixel estimate of disparity, given observations of intensity differences between image pairs, and derived the corresponding error variance. This estimator is asymptotically unbiased, minimum variance, and Gaussian. We showed in simulation that these properties were observed for 5 × 5 matching windows for the ideal case of linear ramp images and for a real image, so long as the noise conformed to the Gaussian white model. For real image sequences with real noise, the estimator did not show significant bias and the estimation error was approximately Gaussian. A comparison of the estimated disparity variance and the sample variance showed that the sample variance was higher than the estimate by roughly a constant factor. Thus, the estimated variance appears to have been a good model of the actual uncertainty, up to an unmodelled, additive factor. We conclude that the Gaussian model of estimation error is valid under our experimental conditions, which consisted of a non-specular, fronto-parallel surface. We accept this is adequate justification for pursuing extensions of the ML formulation of this chapter to a Bayesian formulation of the bootstrap operation in the following chapter.

Chapter 5

Depth Estimation: Bootstrapping Stereo Fusion

In chapter 3, we outlined a statistical formulation for depth map estimation and described an approach to "bootstrapping" stereo fusion by using narrow-baseline and wide-baseline image pairs. Chapter 4 derived a classical, area-based, maximum-likelihood disparity estimator and presented experimental results to justify the use of a Gaussian random field model of depth maps. In this chapter, we extend the estimator of chapter 4 by developing Bayesian matching algorithms for the bootstrap operation.

We begin by identifying three classes of single-scale depth map estimation algorithms. These are algorithms that estimate depth independently for each pixel, jointly for each scanline, or jointly for the entire image. Phrased another way, these algorithms are either completely uncoupled, coupled in 1-D, or coupled in 2-D. We then develop algorithms for the independent and joint 1-D classes. These lead to simple, efficient algorithms that use depth estimates from the narrow-baseline image pair as prior densities for matching the wide-baseline image pair. For the joint 1-D case, we develop two algorithms that generalize current regularization-based approaches to matching. This is done by constraining the disparity field estimated from the wide-baseline image pair to have smoothness properties similar to those measured from the narrow-baseline image pair, rather than employing universal smoothness heuristics as existing algorithms do. We briefly examine the possibility of 2-D coupling. The main advantage of such approaches appears to be enforcing coherence between scanlines. Since area-based matching algorithms already achieve this to some degree by using 2-D image comparison operators, we conclude that 2-D coupling is probably unnecessary. After developing matching algorithms, we examine issues of sensitivity, computational complexity, and matching ambiguity that arise in determining both the direction and the distance to move the cameras in obtaining the narrow-baseline image pair. Finally, we show that the algorithms developed here perform very well on images of scale models of outdoor scenes.

The results of this and the previous chapter lead us to conclude that the three main components of the approach we have pursued — the random field model of depth, area-based matching, and the bootstrap operation — are very promising techniques for stereo depth estimation in complex,

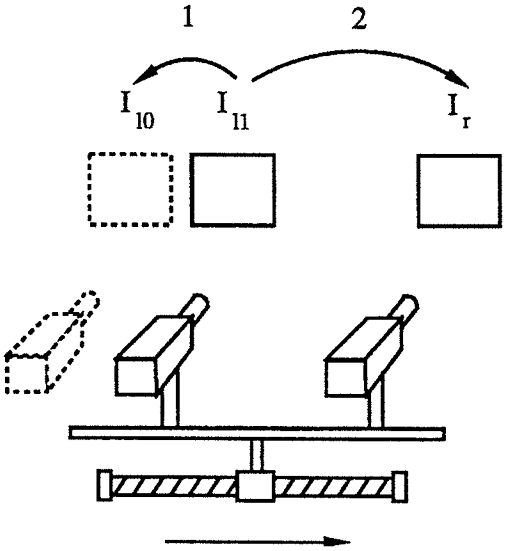


Figure 5.1: Images and matching steps in the bootstrap operation

unstructured environments. We close this chapter by discussing limitations of our work to date and outlining areas for extension.

5.1 Mathematical Models and Matching Algorithms

We begin by recalling the overall approach outlined in chapter 3. The goal is to estimate the disparity at every pixel in the image; that is, to estimate the entire disparity field, which we denote by the vector **d**. We model prior information about **d** as jointly Gaussian with mean $\hat{\mathbf{d}}^-$ and inverse covariance $\mathbf{W}_\mathbf{d}^-$. We model intensity differences between the two images as Gaussian conditioned on **d**. Bayes' theorem is used to obtain an estimate $\hat{\mathbf{d}}^+$ of **d**, with inverse covariance $\mathbf{W}_\mathbf{d}^+$. We operationalize this formalism in a bootstrap operation that uses fine camera motion to initialize stereo fusion.

There are many ways to use camera motion to assist stereo fusion. Figure 5.1 illustrates the specific scenario we explore in this chapter. We assume that two, narrow-baseline images I_{l_0} and I_{l_1} are taken with the left camera. Assuming that prior information guarantees that all disparities lie in the range $[d_{min}, d_{max}]$, we match I_{l_1} to I_{l_0} to sub-pixel precision using techniques similar to those of the previous chapter. The resulting disparity estimates determine a prior density for the disparity field that constrains matching for the wide-baseline image pair of I_{l_1} and I_r.

In the balance of this section, we develop mathematical models and matching algorithms for this scenario. The algorithms operate at a single scale of resolution. We distinguish three classes of models that differ in the model for $\mathbf{W}_\mathbf{d}^-$:

Fully independent model: This treats the prior density of **d** as completely uncorrelated, so that $\mathbf{W}_\mathbf{d}^-$ is diagonal. With this model, we estimate disparity for each pixel independent of all others <not quite right because of overlapping windows>. As we discussed in chapter 3, this provides for a very simple matching algorithm, but requires that prior information be sufficiently constraining to make the match unambiguous. The intent of the camera motion is to provide this information.

Joint 1-D model: This treats **d** as correlated within scanlines, but as uncorrelated across different scanlines. With this model, we estimate disparity for each scanline as a unit, but there is no interaction between scanlines. This can lead to matching algorithms that incorporate constraint neighboring pixels within each scanline, yet are efficient in space and time and allow separate scanlines to be processed in parallel. The key issues in developing such algorithms are to find reasonable correlation models and to find matching algorithms that can estimate jointly optimal disparities for the scanline under the assumed correlation model.

Joint 2-D model: This treats **d** as correlated across as well as within scanlines. With this model, the optimal estimate at each pixel depends on neighbors above and below, as well as to the left and right. This may lead to better estimates than the first two models, but the coupling between scanlines also increases the computational burden in determining optimal estimates.

We consider each of these models below. For the first two, we develop complete matching algorithms for the bootstrap operation. These algorithms are efficient and perform well on complex images. For the third model, the joint 2-D case, we examine issues involved in developing such an algorithm and contrast likely characteristics of such algorithms with the algorithms we develop for the joint 1-D case. We conclude that it does not appear attractive to purse the joint 2-D case, so we do not develop an algorithm for it.

5.1.1 Fully Independent Model

Figure 5.1 illustrates the images acquired and the sequence of matching operations performed in the basic bootstrap scenario. We assume that the left camera acquires image I_{l_0}, moves to acquire image I_{l_1}, and that the right camera acquires image I_r. Considering only 1-D images for simplicity, we model the images as shifted, noise-corrupted versions of the same deterministic signal:

$$\begin{aligned} I_{l_0}(x) &= I(x - d(x)) + n_{l_0}(x) \\ I_{l_1}(x) &= I(x) + n_{l_1}(x) \\ I_r(x) &= I(x + kd(x)) + n_r(x). \end{aligned}$$

Here $d(x)$ is the disparity function and the constant k is the ratio of disparity between I_{l_0} and I_{l_1} to the disparity between I_{l_1} and I_r. If the spacing between the images is equal in both cases, k is 1; in general we will want less spacing between I_{l_0} and I_{l_1} than between I_{l_1} and I_r, so in general k will be larger than 1.

To estimate disparity at pixel x_i in image I_{l_1}, we observe intensity differences between the images in a window around x_i in the same way as in chapter 4. Assuming that $d(x)$ is constant in a small region around x_i, the intensity errors between I_{l_0} and I_{l_1} and between I_{l_1} and I_r are, respectively,

$$\begin{aligned} e_{ll}(x_i + \Delta x_j; d) &= I_{l_0}(x_i + \Delta x_j + d) - I_{l_1}(x_i + \Delta x_j) \\ e_{lr}(x_i + \Delta x_j; d) &= I_r(x_i + \Delta x_j - kd) - I_{l_1}(x_i + \Delta x_j). \end{aligned}$$

We denote the intensity errors in a region around x_i for both image pairs by the vectors $\mathbf{e}_{ll}$ and $\mathbf{e}_{lr}$, respectively.

To derive the estimator, we will first extend the maximum-likelihood formulation of chapter 4 to a Bayesian formulation matching with the narrow-baseline image pair. We will then extend this to apply the result to the wide-baseline pair.

Formulation for I_{l_0} and I_{l_1}

In chapter 4, we used the conditional density $f(\mathbf{e}_{ll}|d)$ to obtain maximum-likelihood disparity estimates. Here, we use Bayes' theorem

$$f(d|\mathbf{e}_{ll}) = \frac{f(\mathbf{e}_{ll}, d)}{f(\mathbf{e}_{ll})} \tag{5.1}$$

to obtain expressions for the posterior density of d, given $\mathbf{e}_{ll}$, in terms of the joint density $f(\mathbf{e}_{ll}, d)$ and the marginal density $f(\mathbf{e}_{ll})$. As in chapter 2, we use the MAP criterion to define the optimal estimate. For a given set of observations, the marginal density in the denominator is a constant normalizing term that is not needed to arrive at our results. We assume that any prior information about d comes from external sources, such as a laser scanner or a map database, and is independent of the image noise.

We assume that the prior information can be modelled by a Gaussian density with mean $\hat{d}^-$ and variance s^-; that is,

$$f(d) \propto \exp\left\{ -\frac{1}{2}\frac{(d - \hat{d}^-)^2}{s^-} \right\}. \tag{5.2}$$

When d is independent of the image noise, the conditional density is the same as we gave in chapter 4:

$$f(\mathbf{e}_{ll}|d) \propto \exp\{-\frac{1}{2\sigma^2}\mathbf{e}_{ll}^T\mathbf{e}_{ll}\}.$$

With the MAP criterion, the optimal estimate of d maximizes $f(d|\mathbf{e}_{ll})$, which is equivalent to maximizing the log-likelihood

$$\ell(d) = -\frac{1}{2}\left\{ \frac{1}{\sigma^2}\mathbf{e}_{ll}^T\mathbf{e}_{ll} + \frac{(d - \hat{d}^-)^2}{s^-} \right\} + K, \tag{5.3}$$

where K is a constant. Therefore, we obtain disparity estimates to pixel resolution by maximizing (5.3) over d, or equivalently by minimizing the expression in parentheses,

$$\frac{1}{\sigma^2}\mathbf{e}_{ll}^T\mathbf{e}_{ll} + \frac{(d - \hat{d}^-)^2}{s^-}. \tag{5.4}$$

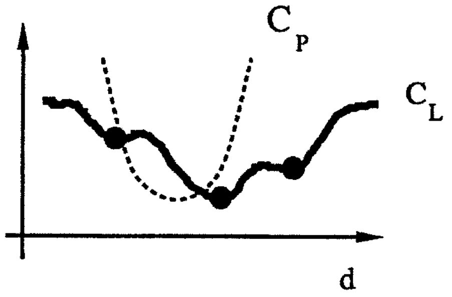


Figure 5.2: Bayesian matching for a single pixel. Curve C_P represents the quadratic cost term from the prior density; curve C_L illustrates the intensity error or "likelihood" term. Local minima of C_L become candidate disparities.

This is just a combination of the intensity error term of chapter 4, weighted by the inverse noise variance, with a quadratic penalty for deviation from the prior estimate, weighted by the variance of the prior estimate. Figure 5.2 illustrates this by plotting the quadratic term (curve C_P) and the intensity error term (C_L) as a function of disparity. The latter may have several local minima, as shown in the figure. Intuitively, we can view the local minima in C_L as defining candidate disparities and the prior term as influencing which candidate is considered optimal. Our implementation does exactly that by evaluating (5.4) only at local minima in C_L. The best local minimum according this criterion defines the disparity estimate to pixel resolution, which we denote d_0.

Sub-pixel disparity estimates are obtained by linearizing the observation equations about d_0, as done in chapter 4. Expanding the error observed at d_0 yields:

$$\begin{aligned} e_{ll}(x_i+\Delta x_j; d_0) &= I_{l_0}(x_i+\Delta x_j+d_0) - I_{l_1}(x_i+\Delta x_j) \\ &= I(x_i+\Delta x_j - d + d_0) - I(x_i+\Delta x_j) + n_{l_0}(x_i+\Delta x_j+d_0) - n_{l_1}(x_i+\Delta x_j) \\ &\approx \left[I(x_i+\Delta x_j) - (d-d_0)\left.\frac{\partial I(x_i+\Delta x_j - d + d_0)}{\partial d}\right|_{d=d_0} \right] - I(x_i+\Delta x_j) \\ &\quad + n_{l_0}(x_i+\Delta x_j) - n_{l_1}(x_i+\Delta x_j) \\ &= -I'(x_i+\Delta x_j)(d-d_0) + n_{l_0}(x_i+\Delta x_j) - n_{l_1}(x_i+\Delta x_j). \end{aligned} \tag{5.5}$$

Letting $\mathbf{J}$ be the vector of derivatives over a window around x_i and letting $\mathbf{n}_{l_0}$ and $\mathbf{n}_{l_1}$ be the corresponding noise vectors, the linearized measurement vector is

$$\mathbf{e}_{ll} \approx -\mathbf{J}(d-d_0) + \mathbf{n}_{l_0} - \mathbf{n}_{l_1}.$$

Substituting this approximation for $\mathbf{e}_{ll}$ into (5.3), the log-likelihood becomes

$$\ell(d) \approx -\frac{1}{2}\left\{ \frac{1}{\sigma^2}[\mathbf{e}_{ll} + \mathbf{J}(d-d_0)]^T[\mathbf{e}_{ll}+\mathbf{J}(d-d_0)] + \frac{(d-\hat{d}^-)^2}{s^-} \right\} + K. \tag{5.6}$$

The MAP estimate of d is obtained by taking the derivative $\mathrm{d}\ell/\mathrm{d}d$, setting it to zero, and solving for d. This produces

$$\begin{aligned}\widehat{d}_{ll}^{+} &= \left[\frac{\mathbf{J}^T\mathbf{J}}{\sigma^2}+\frac{1}{s^-}\right]^{-1}\left[\frac{\mathbf{J}^T\mathbf{J}}{\sigma^2}d_0+\frac{\mathbf{J}^T\mathbf{e}_{ll}}{\sigma^2}+\frac{\widehat{d}^-}{s^-}\right]\\ &= \left[\frac{\mathbf{J}^T\mathbf{J}}{\sigma^2}+\frac{1}{s^-}\right]^{-1}\left[\frac{\mathbf{J}^T\mathbf{J}}{\sigma^2}\left(d_0+\frac{\mathbf{J}^T\mathbf{e}_{ll}}{\mathbf{J}^T\mathbf{J}}\right)+\frac{\widehat{d}^-}{s^-}\right].\end{aligned}$$

Here, $(d_0+\mathbf{J}^T\mathbf{e}_{ll}/\mathbf{J}^T\mathbf{J})$ is the linearized maximum likelihood estimate derived in chapter 4 and $\mathbf{J}^T\mathbf{J}/\sigma^2$ is the corresponding error variance. Denoting these terms by $\widehat{d}_{ML}$ and σ^2_{ML} gives

$$\widehat{d}_{ll}^{+}=\left[\frac{1}{\sigma^2_{ML}}+\frac{1}{s^-}\right]^{-1}\left[\frac{\widehat{d}_{ML}}{\sigma^2_{ML}}+\frac{\widehat{d}^-}{s^-}\right]. \tag{5.7}$$

Thus, $\widehat{d}_{ll}^{+}$ is a weighted combination of the prior estimate $\widehat{d}^-$ and the maximum-likelihood estimate $\widehat{d}_{ML}$, where $\widehat{d}_{ML}$ is computed by linearizing about the best pixel-resolution disparity. As in chapter 4, this process may be iterated to further refine the disparity esimate. The form of (5.7) suggests the simplicifation of iterating $\widehat{d}_{ML}$ to convergence, then combining this with $\widehat{d}^-$ to compute $\widehat{d}_{ll}^{+}$.

By completing squares in the exponent of $f(\mathbf{e}_{ll}|d)f(d)$, it can be shown [DeGroot70] that $\widehat{d}_{ll}^{+}$ as above is the mean of the posterior density $f(d|\mathbf{e}_{ll})$ and that the posterior variance is

$$s_{ll}^{+}=\left[\frac{1}{\sigma^2_{ML}}+\frac{1}{s^-}\right]^{-1}. \tag{5.8}$$

Therefore, s_{ll}^{+} is the variance of the estimation error in $\widehat{d}_{ll}^{+}$.

To summarize what we have done so far, we assumed that a prior disparity estimate was available for each pixel and modelled this estimate as Gaussian, with mean $\widehat{d}^-$ and variance s^-. Using the conditional density $f(\mathbf{e}_{ll}|d)$ from the previous chapter, we derived the log-likelihood $\ell(d)$. The intensity error term of $\ell(d)$ is evaluated for all disparities in a search range $[d_{min}, d_{max}]^T$. Local minima of this term define a set of disparity candidates at pixel resolution; the candidate for which (5.4) is minimal becomes the initial disparity estimate d_0. We then used a first order expansion of $\mathbf{e}_{ll}$ about d_0 to derive the posterior mean $\widehat{d}_{ll}^{+}$ (5.7) and variance s_{ll}^{+} (5.8) of d, which define the "best" estimate of d and the variance of the estimation error. In practice, we decompose the calculation of $\widehat{d}_{ll}^{+}$ by iterating the linearized, maximum-likelihood estimate $\widehat{d}_{ML}$ to convergence, then combining $\widehat{d}_{ML}$ with the prior estimate $\widehat{d}^-$. If there is no prior information, s^- is infinite and the equations reduce to the maximum-likelihood estimator.

We repeat this procedure for each pixel in I_{l_1} to estimate the entire disparity field. In regions of the image with negligible intensity variation, this will not yield a meaningful disparity estimate. Such regions can be detected before attempting to match by thresholding σ^2_{ML}, which can be computed in advance. Thresholding σ^2_{ML} amounts to applying an interest operator; however, instead of choosing to match only at local maxima of the interest value, we match everywhere where interest falls within a threshold.

Formulation for I_{l_1} and I_r

The above operation is used to estimate a disparity field from the narrow-baseline image pair, I_{l_0} and I_{l_1}. This disparity field becomes the prior density for matching the wide-baseline image pair, I_{l_1} to I_r. Therefore, what were the posterior mean and variance, $\hat{d}^+_{ll}$ and s^+_{ll}, now become the prior mean and variance, $\hat{d}^-_{lr}$ and s^-_{lr}. The observed intensity errors for this image pair are

$$e_{lr}(x_i + \Delta x_j; d) = I_r(x_i + \Delta x_j - kd) - I_{l_1}(x_i + \Delta x_j).$$

The appropriate form of Bayes' theorem is

$$f(d|\mathbf{e}_{lr}, \mathbf{e}_{ll}) = \frac{f(\mathbf{e}_{lr}, \mathbf{e}_{ll}, d)}{f(\mathbf{e}_{lr}, \mathbf{e}_{ll})} = \frac{f(\mathbf{e}_{lr}|\mathbf{e}_{ll}, d)}{f(\mathbf{e}_{lr}|\mathbf{e}_{ll})} f(d|\mathbf{e}_{ll})$$

The conditional density $f(\mathbf{e}_{lr}|\mathbf{e}_{ll}, d)$ is somewhat more complex than the density $f(\mathbf{e}_{ll}|d)$ for the narrow-baseline case, because the sharing of I_{l_1} makes $\mathbf{e}_{ll}$ and $\mathbf{e}_{lr}$ correlated. For the moment, we will avoid the additional complexity by ignoring this correlation and using $f(\mathbf{e}_{lr}|d)$ instead of $f(\mathbf{e}_{lr}|\mathbf{e}_{ll}, d)$. This is equivalent to assuming that the narrow-baseline depth estimate is independent from I_{l_1} and I_r; for example, this would be the case if a new copy of image I_{l_1} was acquired for the wide-baseline match. Since we do not do this, the resulting estimator is sub-optimal.

With this simplification, the derivation of the estimator is very similar to the previous section. Collecting the observations over the area of the match window into the vector $\mathbf{e}_{lr}$ and following the MAP estimation method as before, we find that the optimal estimate of d minimizes

$$\frac{1}{\sigma^2}\mathbf{e}_{lr}^T\mathbf{e}_{lr} + \frac{(d - \hat{d}^-_{lr})^2}{s^-_{lr}}. \tag{5.9}$$

This expression is used to determine the best disparity estimate to pixel resolution in the same manner as before. If there is no prior information for the narrow-baseline case (i.e. $s^- = \infty$), then $s^-_{lr} = \sigma^2/\mathbf{J}^T\mathbf{J}$ and the above expression becomes

$$\frac{1}{\sigma^2}\mathbf{e}_{lr}^T\mathbf{e}_{lr} + \frac{(d - \hat{d}^-_{lr})^2}{\sigma^2/\mathbf{J}^T\mathbf{J}}.$$

This version is useful if the variance of the image noise is not well known, because then σ^2 factors out of both terms and does not affect the match decision. Minimizing either of these expressions produces the initial disparity estimate d_0.

Sub-pixel precision again is obtained by linearizing about d_0. Expanding e_{lr} in a similar fashion to (5.5), we obtain

$$\mathbf{e}_{lr} \approx k\mathbf{J}(d - d_0) + \mathbf{n}_r - \mathbf{n}_{l_1}.$$

Following through the MAP derivation of equations (5.6) through (5.8) leads to the following disparity estimate and error variance:

$$\hat{d}^+_{lr} = s^+_{lr}\left[\left(\frac{k^2\mathbf{J}^T\mathbf{J}}{\sigma^2}\right)\left(d_0 + \frac{\mathbf{J}^T\mathbf{e}_{lr}}{k\mathbf{J}^T\mathbf{J}}\right) + \frac{\hat{d}^-_{lr}}{s^-_{lr}}\right] \tag{5.10}$$

$$s^+_{lr} = \left[\frac{k^2\mathbf{J}^T\mathbf{J}}{\sigma^2} + \frac{1}{s^-_{lr}}\right]^{-1} \tag{5.11}$$

In (5.10), the term $(d_0 + \mathbf{J}^T\mathbf{e}_{lr}/\mathbf{J}^T\mathbf{J})$ is the ML disparity estimate for this image pair; the factor of $(1/k)$ scales the correction term so that the disparity estimate is in units of the narrow baseline. Likewise, the term $(k^2\mathbf{J}^T\mathbf{J}/\sigma^2)$ is the inverse of ML error variance, scaled into units of the narrow baseline. Therefore, we can rewrite (5.10) as

$$\hat{d}^+_{lr} = s^+_{lr}\left[k^2\frac{\hat{d}_{ML}}{\sigma^2_{ML}} + \frac{\hat{d}^-_{lr}}{s^-_{lr}}\right],$$

which shows that the disparity estimate is again a weighted combination the prior estimate and a new measurement obtained from images I_{l_1} and I_r. The weight of k^2 attached to the new measurement reflects the longer baseline used to obtain it.

If no prior information is available for matching the narrow-baseline image pair ($s^- = \infty$), (5.10) and (5.11) reduce to

$$\begin{aligned}\hat{d}^+_{lr} &= \frac{1}{k^2+1}\left[k^2\hat{d}_{ML} + \hat{d}^-_{lr}\right] \\ s^+_{lr} &= \frac{\sigma^2}{(k^2+1)\mathbf{J}^T\mathbf{J}}.\end{aligned}$$

That is, the new disparity estimate is a weighted combination of two measurements obtained with baselines in the ratio of $k : 1$, which results in a weight ratio of $k^2 : 1$. Note that if $k = 1$ (equal distances between both pairs of images), then the posterior disparity estimate is just the average of the two measurements and the posterior variance is half that of the measurements, as we would expect.

To summarize, by ignoring correlation between $\mathbf{e}_{lr}$ and the prior information about d, we were able to apply the same estimator to the wide-baseline image pair as the narrow-baseline image pair. An initial disparity estimate d_0 at pixel resolution is obtained by minimizing (5.9). From this, a sub-pixel estimate and the error variance are obtained from (5.10) and (5.11). This estimate can be iterated as described in the previous section. We also showed simpler forms of the equations that result when $s^- = \infty$. Finally, if the correlation is taken into account, it can be shown that different weights are obtained for the terms comprising $\hat{d}^+_{lr}$ and that the final variance is lower. We will not enter into the details here.

Overall Algorithm for the Bootstrap Operation

The entire procedure for estimating depth from the narrow and wide-baseline images consists of the following steps:

- Compute σ^2_d from image I_{l_1} and threshold it to determine which pixels to match.
- Match the narrow-baseline image pair for pixels within threshold. If prior information consists of disparity limits, use the ML operator; otherwise, use the Bayesian operator. Sub-pixel disparity estimates are computed by linearization and iteration.

- Match the wide-baseline image pair. In principal, search windows for this step could be established by deriving confidence limits from the prior estimate and centering the resulting range around the prior mean. In practice, we use the more conservative approach of assuming that the disparities from I_{l_0} and I_{l_1} are accurate to within a fixed fraction of the pixel width (generally 0.5 to 0.7), scaling this interval up according to the size of the wide baseline, and using the result as the search window half-width. Within this search range, we use the Bayesian operator. Again, sub-pixel disparity estimates are computed.

The results of this procedure are estimates of disparity, computed to sub-pixel resolution, and error variance for each pixel within threshold of the interest operator.

Discussion

This algorithm is simple and efficient, because it does not use global optimization or the expensive search methods sometimes used with global optimization. To achieve reliability, the algorithm requires appropriate choices of the narrow baseline and the ratio between the narrow and the wide baselines. This makes the choice of baseline, especially the automated choice of baseline, an important problem. We consider this problem in section 5.2.

Whereas the algorithm in this section estimates depth for each pixel independently, most binocular stereo algorithms attempt to gain reliability by using surface smoothness or "local support" heuristics that couple the depth estimate at each pixel to estimates at neighboring pixels. In the next section, we interpret these concepts in the context of the bootstrap operation. This leads to attractive re-statements of the existing heuristics into forms that have better physical and statistical justifications.

5.1.2 Joint 1-D Model

In terms of the probabilistic model of the entire disparity field, the fully independent algorithm above is modelled by prior and posterior densities for the disparity field in which the covariance matrix is diagonal; that is, there is no correlation in the field. In this section, we investigate formulations that are coupled with one dimension. The corresponding probabilistic models in which the disparity field is correlated within scanlines, but independent across scanlines. This leads to objective functions that require global optimization within scanline, but which can be minimized efficiently with dynamic programming.

As motivation, we begin with the gradient-based, surface-smoothness constraint discussed in chapter 3. In the bootstrap scenario, the heuristic of low disparity gradient can be replaced with the more justifiable constraint that gradients measured from the wide-baseline pair should be the same, up to noise, as those measured from the narrow-baseline pair. In one dimension, the this constraint leads to an objective function for which the global minimum can be found efficiently by dynamic programming. However, this objective function is still based on a somewhat *ad hoc* development and contains blending constants that must be defined heuristically. As a step toward a more rigorous probabilistic model, we recall from chapter 4 that disparity estimates obtained from the narrow-baseline image pair are in fact correlated. We use this observation to derive a joint, Bayesian estimator for disparity within scanlines that includes an exponential model of correlation in the prior density. The resulting objective function is closely related to the objective function based on the gradient constraint and can also be minimized by dynamic programming.

Gradient Constraint Formulation

As we noted in chapter 3, a common approach to stereo matching has been to augment image similarity measures with cost functions that penalize departures from smoothness in the estimated disparity field [Barnard89,Boult88,Horn86,Poggio85,Witkin87]. The predominant smoothness constraints have been based on first and second derivatives of the disparity field. In chapter 3, we looked an example based on first derivatives. In one dimension, this example chose the disparity function d to minimize the integral

$$q(d) = \int \left\{ [I_r(x - d(x)) - I_l(x)]^2 + \lambda (d'(x))^2 \right\} \mathrm{d}x, \tag{5.12}$$

where λ is a blending constant. The term $(d'(x))^2$ penalizes departures of the estimated disparity field from zero derivative; that is, it biases the algorithm to prefer surfaces that face the cameras directly. A suitable discrete version of this integral is obtained by using a forward difference approximation of $d'(x)$ to write

$$q(\mathbf{d}) = \left(\sum_{i=1}^{N} [I_r(x_i - d_i) - I_l(x_i)]^2 \right) + \lambda \left(\sum_{i=1}^{N-1} (d_{i+1} - d_i)^2 \right).$$

With this cost function, we seek the disparity vector $\mathbf{d} = [d_1, \ldots, d_n]^T$ that minimizes the total intensity error across the scanline plus the weighted, total "deviation from flatness" of $\mathbf{d}$. It is useful to note that the second summation is equivalent to the quadratic form $\mathbf{d}^T \mathbf{W}_g \mathbf{d}$, with

$$\mathbf{W}_g = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix}.$$

Thus, in 1-D the gradient constraint is equivalent to a quadratic form with a tri-diagonal coefficient matrix.

Unfortunately, constraining the estimated disparity field to have low gradients is purely heuristic. Although it has been shown that this tends to be true for surfaces with random orientations in space [Arnold80,Milenkovic85], this heuristic is by no means true everywhere in the image or for all scenes. On the other hand, the bootstrap scenario allows this heuristic to be replaced with another, more meaningful constraint. Because we are observing the *same* surface with both the narrow-baseline and the wide-baseline image pairs, the disparity gradients observed in both cases must be the same, up to the effects of noise. To relate this to (5.12),