# 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

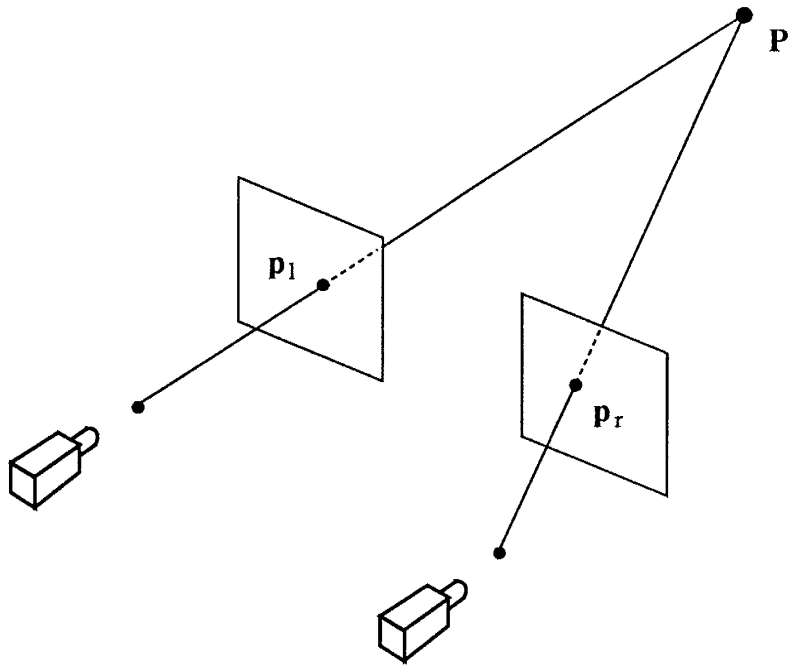



Figure 1.1: The basics of stereo triangulation: given *corresponding* image points, $\mathbf{p}_l$ and $\mathbf{p}_r$, and the positions and orientations of both cameras, we can compute the *depth* or 3-D location of the world point **P**.

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.

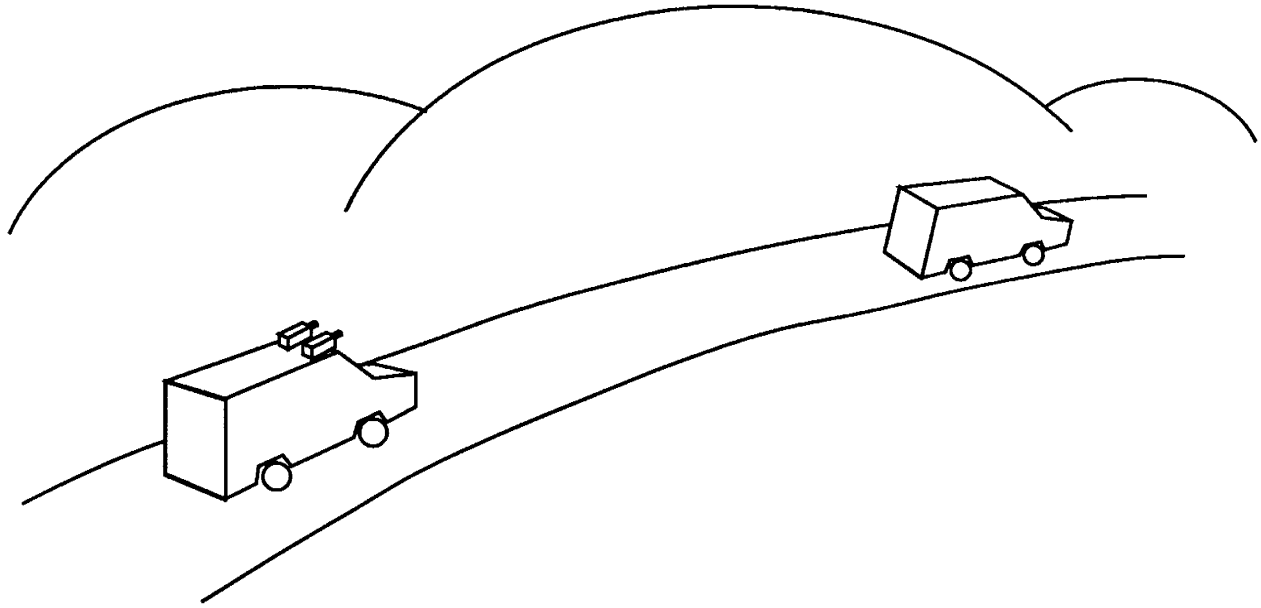


Figure 1.2: Application scenario: autonomous navigation

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

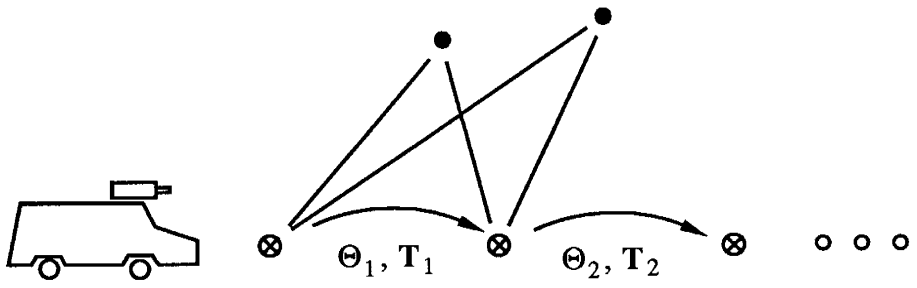



Figure 1.3: Estimating vehicle motion by tracking nearby landmarks

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].

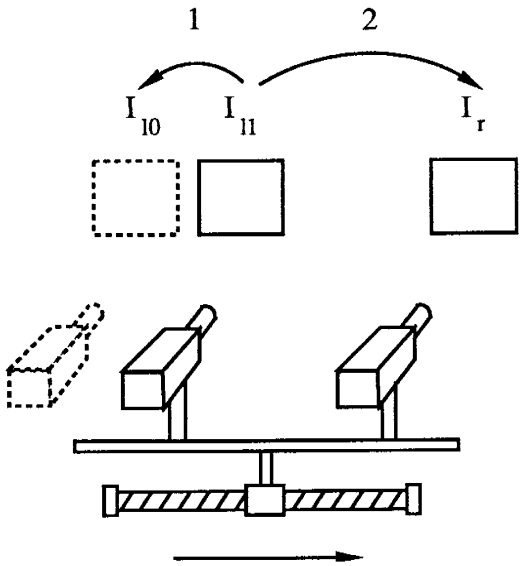



Figure 1.4: Camera motion and image acquisition for the bootstrap operation. Depth estimated by matching with images $I_{l_0}$ and $I_{l_1}$ is used to constrain matching between images $I_{l_1}$ and $I_r$.

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

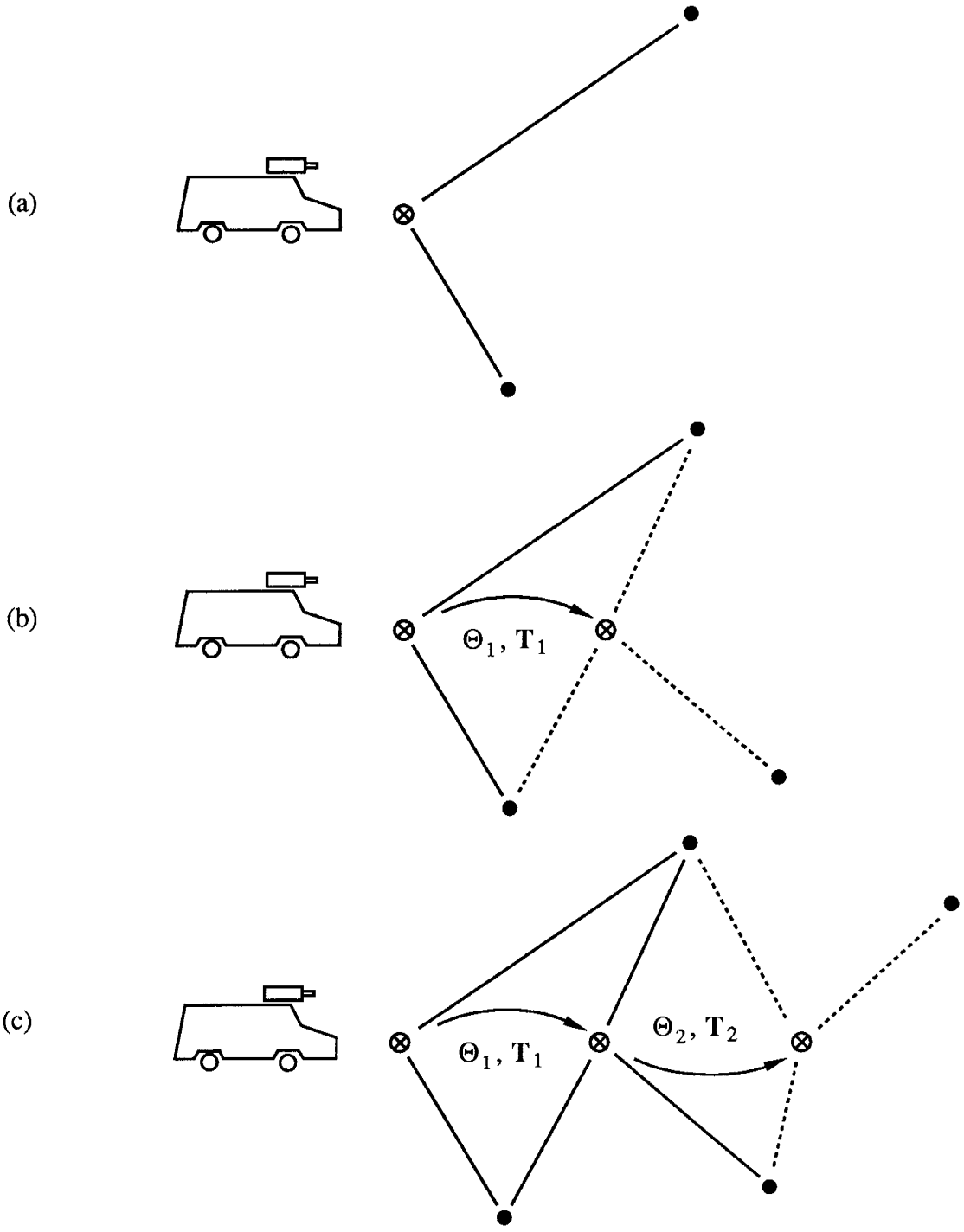



Figure 2.1: The robot navigation problem

(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. $\Theta$ 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

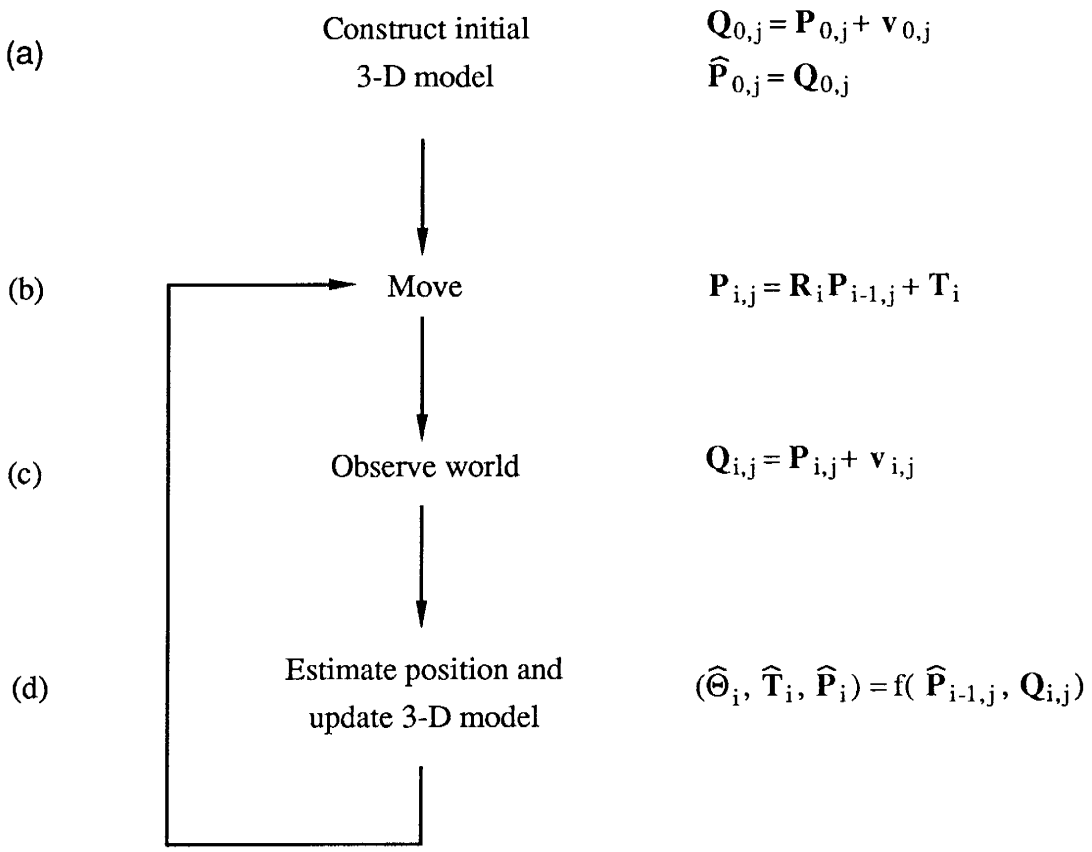



Figure 2.2: Processing loop

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 esti-

mation 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 $\Theta_i$, $\mathbf{T}_i$ 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 $\Theta_i$. 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*

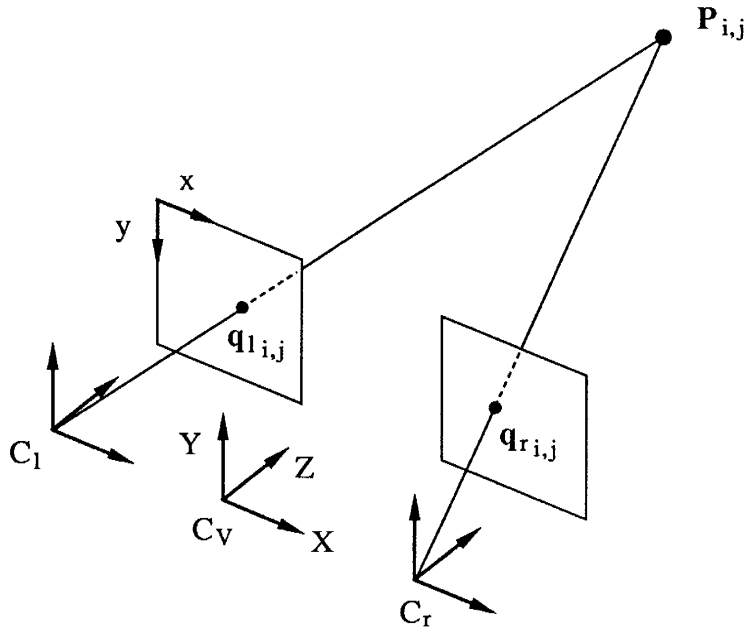



Figure 2.3: Stereo observation model

vector $\mathbf{Q}_{i,j}$ 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 $\mathbf{Q}_{i,j}$ as the sum of the true landmark coordinates $\mathbf{P}_{i,j}$ and a noise vector $\mathbf{v}_{i,j}$:

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

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):

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

Therefore, the error covariance of $\hat{\mathbf{P}}_{0,j}$ is $\Sigma_{\mathbf{P}_{0,j}} = \Sigma_{\mathbf{v}_{0,j}}$. The observations $\mathbf{Q}_{0,j}$ and the resulting world model $\hat{\mathbf{P}}_{0,j}$ formalize step (a) of figure 2.2. Subsequent observations $\mathbf{Q}_{i,j}$ 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

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

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 $\mathbf{v}_r$:

$$\mathbf{q}_l = \begin{bmatrix} x_l \\ y_l \end{bmatrix} = \mathbf{h}_l(\mathbf{P}) + \mathbf{v}_l$$

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

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

$$\mathbf{P}_l = \begin{bmatrix} X_l \\ Y_l \\ Z_l \end{bmatrix} = \mathbf{P} + \begin{bmatrix} b/2 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{P}_r = \begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} = \mathbf{P} + \begin{bmatrix} -b/2 \\ 0 \\ 0 \end{bmatrix}.$$

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 $\Sigma_l$ and $\Sigma_r$, 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 $3 \times 3$ 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

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

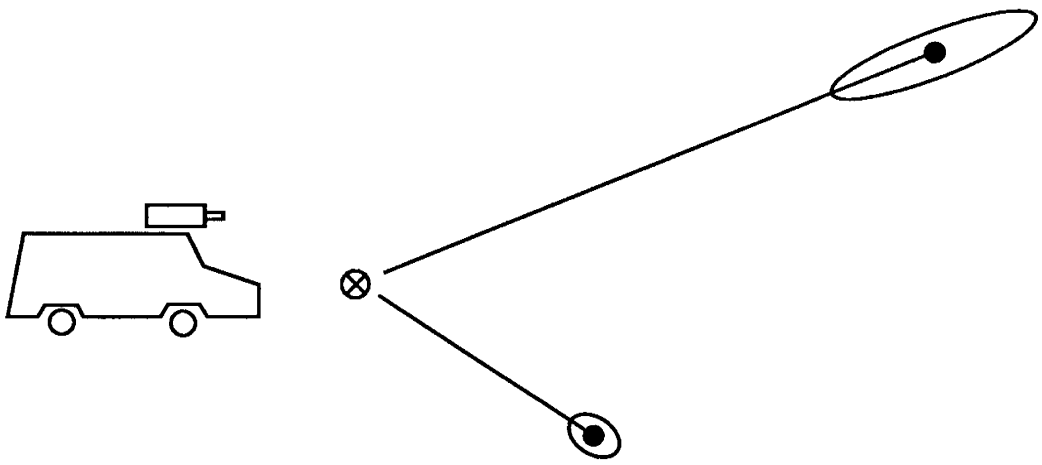
for each stereo pair. In this section, we derive a sequential Bayesian estimator that uses this sequence of observations together with the motion equation

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

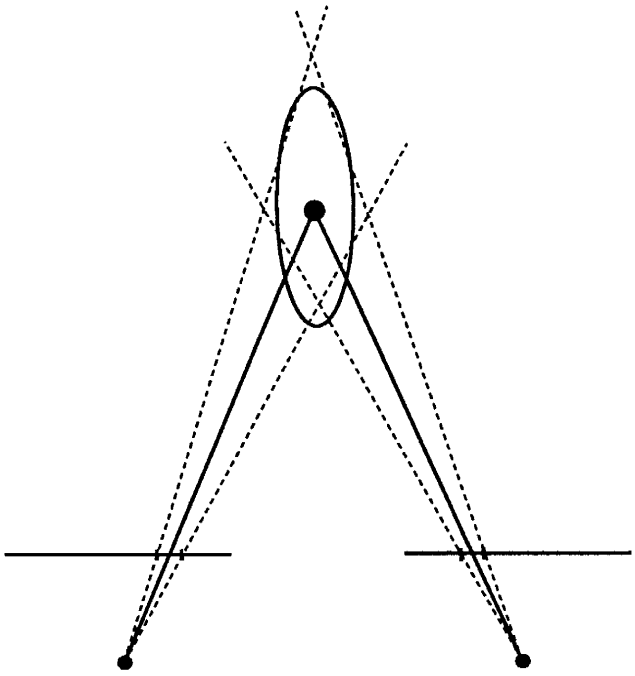
to compute estimates $\widehat{\Theta}_i$, $\widehat{\mathbf{T}}_i$ of the motion parameters and estimates $\widehat{\mathbf{P}}_{i,j}$ of the landmark coordinates. The landmark estimates are defined relative to the current coordinate frame and incorporate information from the entire observation history.

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 $6 \times 6$.

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 $\Theta$ 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

$$\mathbf{Q}_{pj} = \mathbf{P}_{pj} + \mathbf{v}_{pj} \tag{2.7}$$

$$\mathbf{Q}_{cj} = \mathbf{R}\mathbf{P}_{pj} + \mathbf{T} + \mathbf{v}_{cj}\,. \tag{2.8}$$

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 $\mathbf{e}_j$:

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

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

$$\sum_j w_j \mathbf{e}_j^T \mathbf{e}_j\,. \tag{2.10}$$

The least-squares estimates are obtained by minimizing this expression over $\Theta$ 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 $\Theta$ 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 $\mathbf{R}$ and vector $\mathbf{T}$ 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 $\mathbf{R}$ with well-known techniques ([Paul81, chapter 3]) and designate the resulting initial estimates as $\Theta_0$ and $\mathbf{T}_0$.

### Maximum Likelihood Estimation of $\Theta$ and T

As we will see shortly, the least-squares estimator of $\Theta$ and $\mathbf{T}$ 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 $\Theta$ and $\mathbf{T}$. These estimates will prove to be part of the larger solution for $\Theta$, $\mathbf{T}$, 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 $\Theta$ and $\mathbf{T}$ 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 $\Theta$ and $\mathbf{T}$ are those that maximize this density. This is equivalent to finding $\Theta$ and $\mathbf{T}$ 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 $\Theta_0$:

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

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

$\mathbf{R}_0$ denotes the rotation matrix for $\Theta_0$ 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 $\Theta$ and $\mathbf{T}$, with additive Gaussian noise $\mathbf{v}_j$. Maximum likelihood estimates of $\Theta$ 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 $\Theta$ 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

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

The motion equation is then re-linearized about the new estimate (i.e. the new $\Theta_0$) 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 $\hat{\mathbf{T}}$ 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 $\Theta$ 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 $\Theta$, T, and $\mathbf{P}_{cj}$

In the foregoing stages of the solution, we eliminated $\mathbf{P}_{pj}$ at the outset and solved only for $\Theta$ and $\mathbf{T}$. We will now formulate a Bayesian procedure for jointly estimating $\Theta$, $\mathbf{T}$, 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 $\Theta$ 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 $\mathbf{P}$, $\mathbf{M}$ 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 $3 \times 3$ covariance matrices of the individual landmarks and the $6 \times 6$ 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 $\mathbf{Q}$, 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 $\mathbf{Q}$ given $\mathbf{P}$ 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 $\mathbf{Q}$ does not depend on $\mathbf{M}$, $f(\mathbf{Q}|\mathbf{P})$ 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 $\mathbf{M}$ 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

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

Note that $\mathbf{P}_{pj}$ is a Gaussian random vector with mean $\hat{\mathbf{P}}_{pj}^+$ 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

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

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

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

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

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

with error covariance

$$\Sigma_{\mathbf{M}}^{+} = \left[ \mathbf{W}_{\mathbf{M}}^{-} + \sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{H}_j \right]^{-1} . \tag{2.19}$$

Here $\mathbf{Q}_j = \mathbf{Q}_{cj} - \mathbf{R}_0 \hat{\mathbf{P}}_{pj}^{+} + \mathbf{J}_j \Theta_0$ 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 $\hat{\mathbf{T}}^{+}$ 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)

$$\hat{\mathbf{P}}_{cj}^{+} = (\mathbf{W}_{\mathbf{v}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}^{-}})^{-1} (\mathbf{W}_{\mathbf{v}_{cj}} \mathbf{Q}_{cj} + \mathbf{W}_{\mathbf{P}_{cj}^{-}} \hat{\mathbf{P}}_{cj}^{-}) , \tag{2.20}$$

where $\hat{\mathbf{P}}_{cj}^{-} = \hat{\mathbf{R}}^{+} \hat{\mathbf{P}}_{pj}^{+} + \hat{\mathbf{T}}^{+}$ 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

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

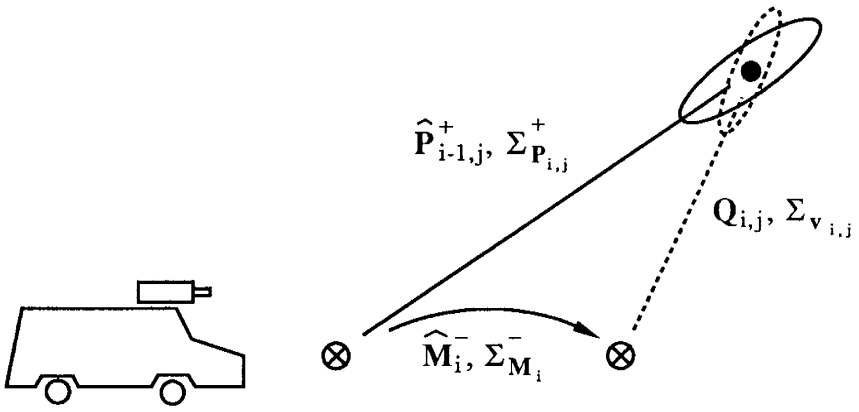
Equations (2.20) and (2.21) are applied independently to each landmark.

The landmark coordinate estimate is easy to interpret; we simply transform the previous estimate $\hat{\mathbf{P}}_{pj}^{+}$ to the new coordinate frame, obtaining $\hat{\mathbf{P}}_{cj}^{-}$, 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

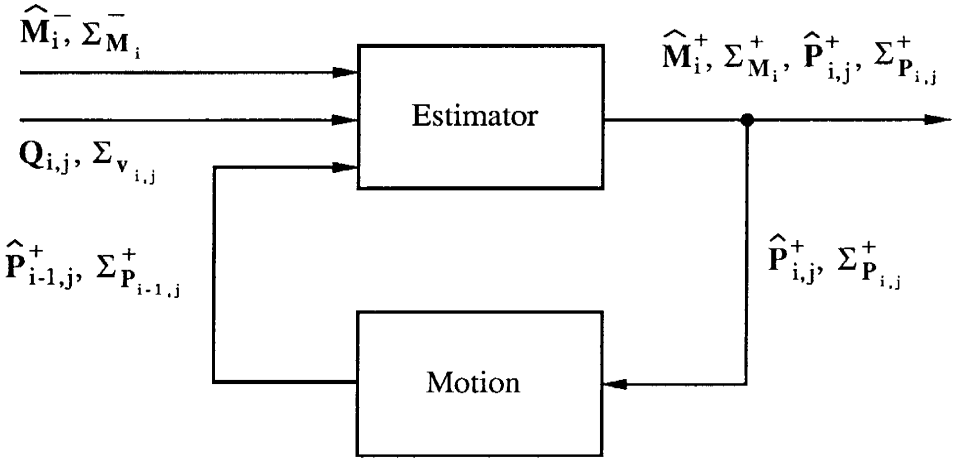
$$\begin{aligned} \Sigma_{\mathbf{P}_{cj}^{+}} &= (\mathbf{W}_{\mathbf{v}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}^{-}})^{-1} \\ &= (\mathbf{W}_{\mathbf{v}_{cj}} + \mathbf{R}_0 \mathbf{W}_{\mathbf{P}_{pj}^{-}} \mathbf{R}_0^T)^{-1} . \end{aligned}$$

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




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

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.

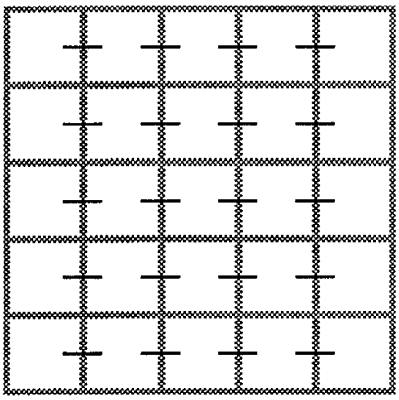
### 2.4.1 Feature Selection

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.

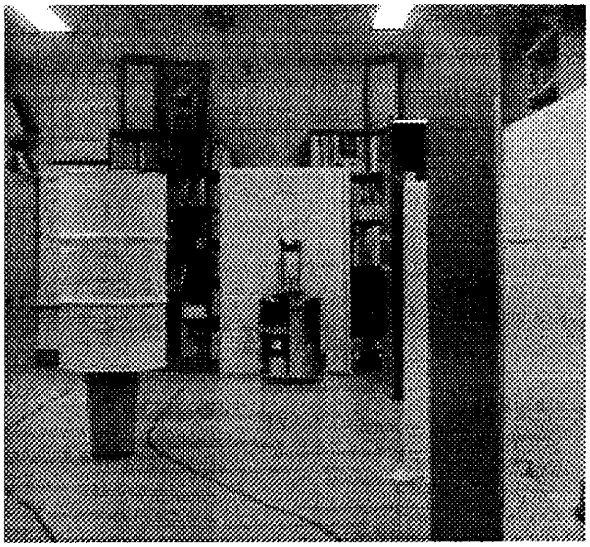
For the experiments described in this chapter, the localizability issue was addressed in an informal manner by using the convolution-like *interest operator* developed by Moravec [Moravec80] to identify regions of the image having high intensity variation in multiple directions. This operator produces 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 $\Omega$ 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

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

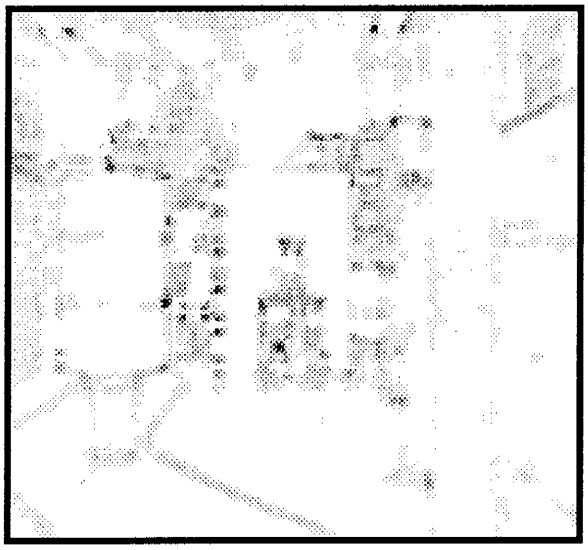
The horizontal measure $m_h$ is illustrated in figure 2.6a. The interest value of the pixel at the center of the region is defined as the minimum of the directional variance measures. This operator accords high interest to regions with strong intensity variation in multiple directions,



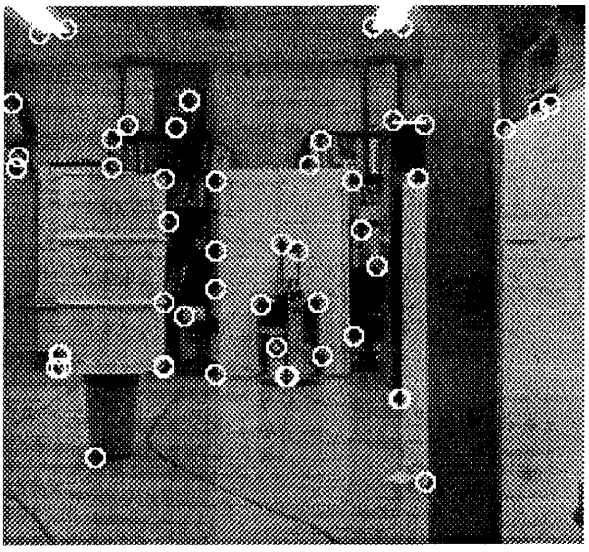
(a)



(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 $\Omega$ is matched in other images. This uncertainty is modelled by the covariance matrices $\Sigma_l$ and $\Sigma_r$. 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 $10 \times 10$ 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,

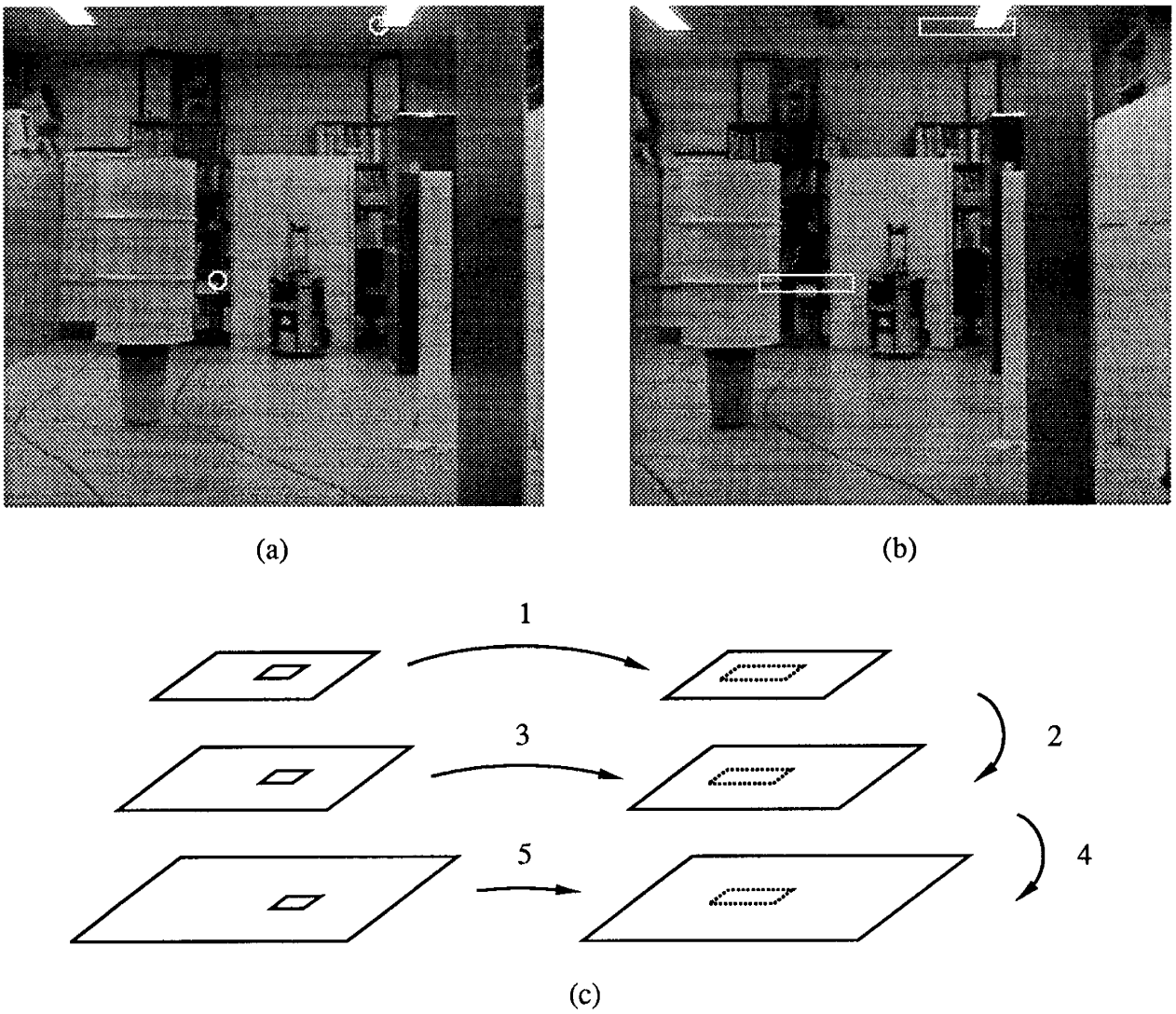



Figure 2.7: Constrained image pyramid correlation for stereo matching

(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 $2 \times 2$ or $4 \times 4$ regions of the higher resolution image, then reducing the resolution by half. Resolutions from $480 \times 512$ down to $15 \times 16$ 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 $5 \times 5$ 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, $\widehat{\mathbf{P}}_{i-1,j}^+$. 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

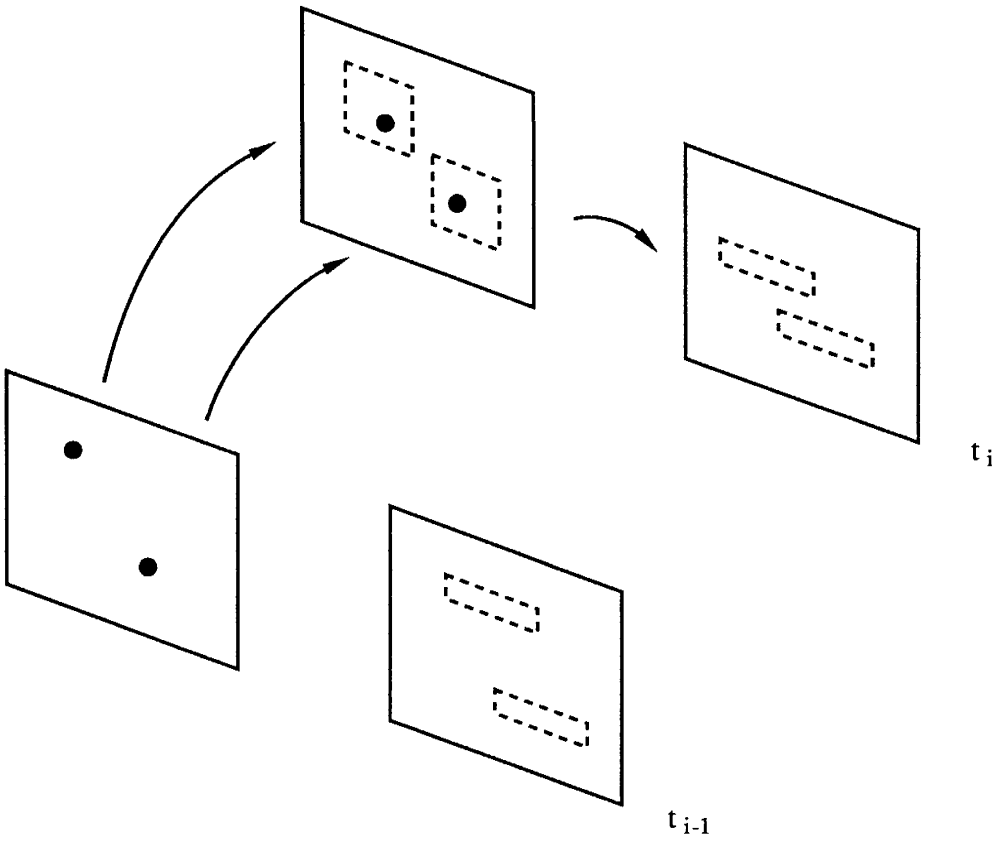



Figure 2.8: Feature tracking

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 $\hat{\mathbf{P}}^{+}_{i-1,j}$ forward in time to obtain the predicted coordinates relative to the next image pair, $\hat{\mathbf{P}}^{-}_{i,j}$. 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 $5 \times 5$ "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, $\widehat{\mathbf{P}}^+_{i-1,j_1}$ and $\widehat{\mathbf{P}}^+_{i-1,j_2}$, 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*

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

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 $\widehat{\mathbf{P}}^+_{0j}$. 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 $\widehat{\mathbf{P}}^+_{1j}$ 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

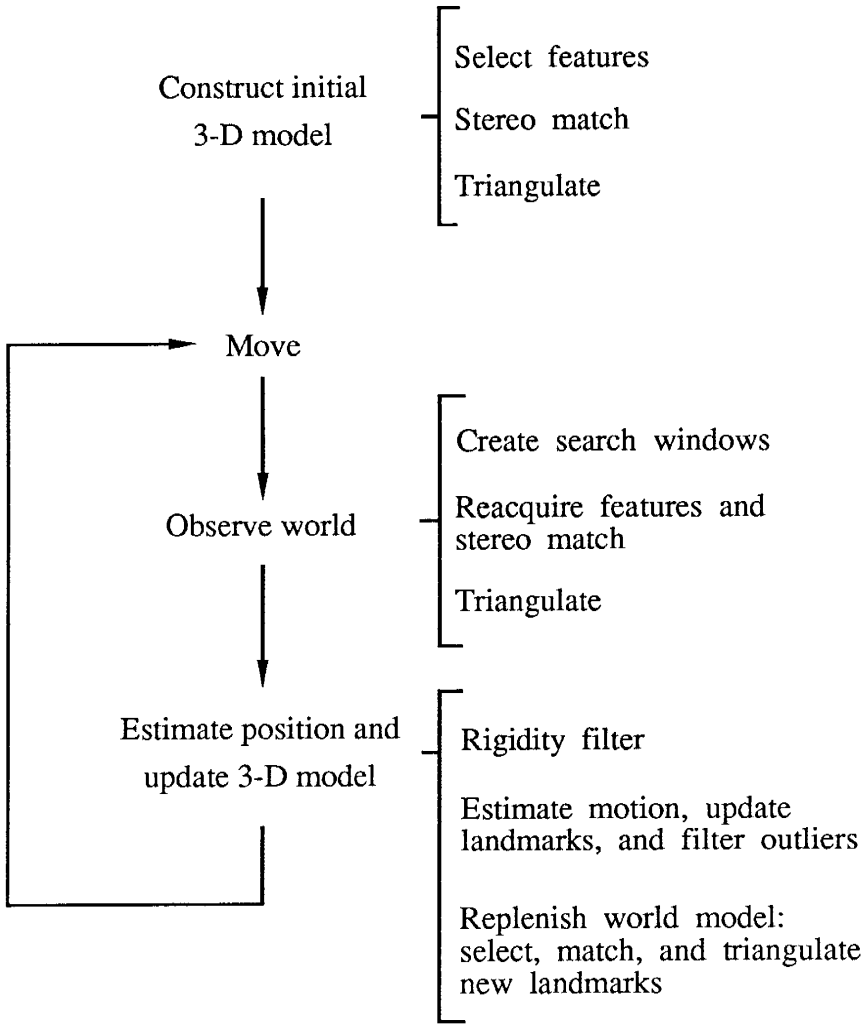



Figure 2.9: Expanded system loop flowchart

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 $480 \times 512$ 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

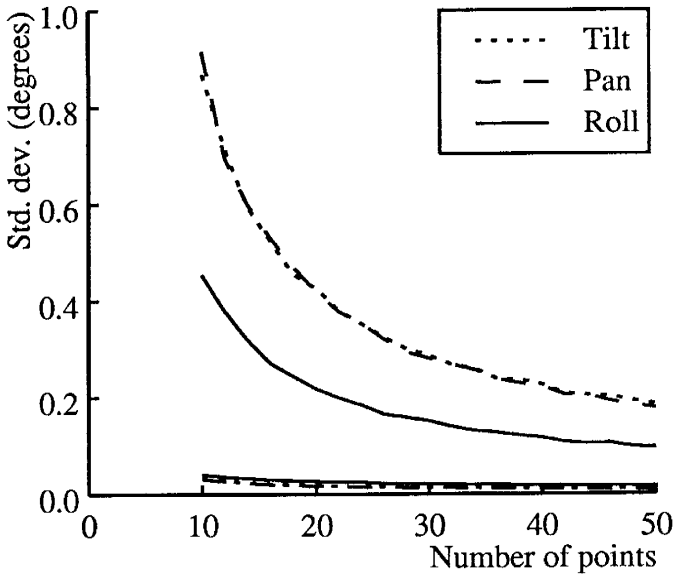



Figure 2.10: Standard deviation vs. number of points for rotations. Top three curves are for the spherical model, bottom three are for the ellipsoidal model (tilt and pan curves overlap).

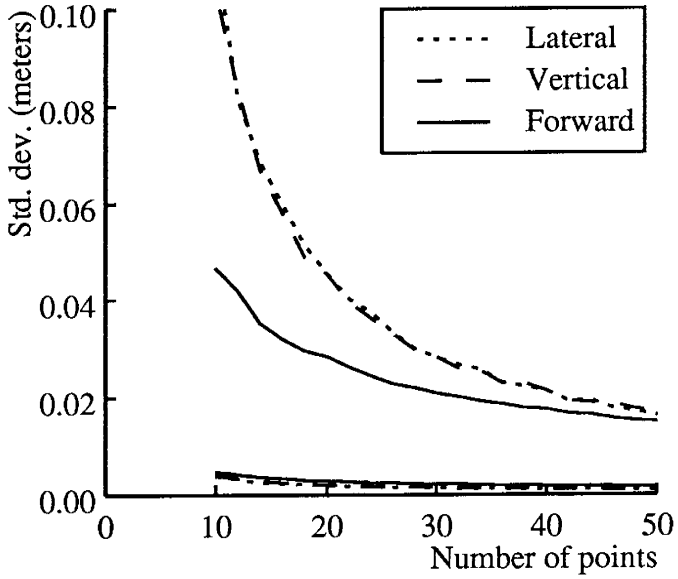



Figure 2.11: Standard deviation vs. number of points for translations. Top three curves are for the spherical model, bottom three are for the ellipsoidal model (lateral and vertical curves overlap).

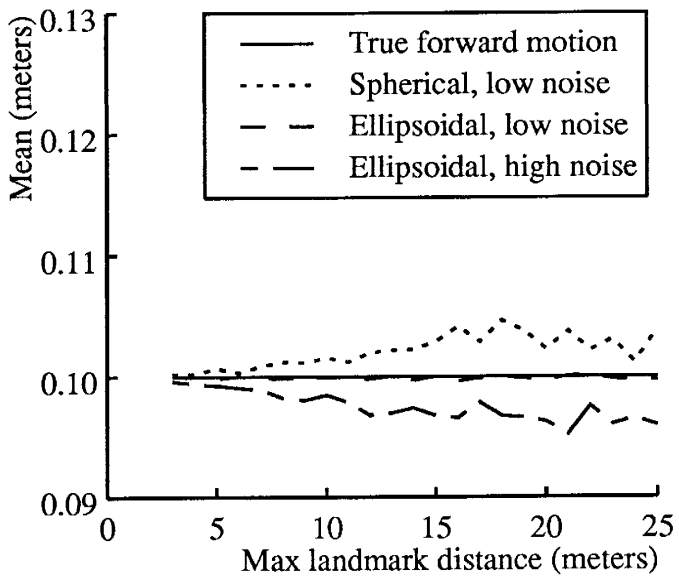



Figure 2.12: Mean estimated forward distance travelled vs. maximum distance to points

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.

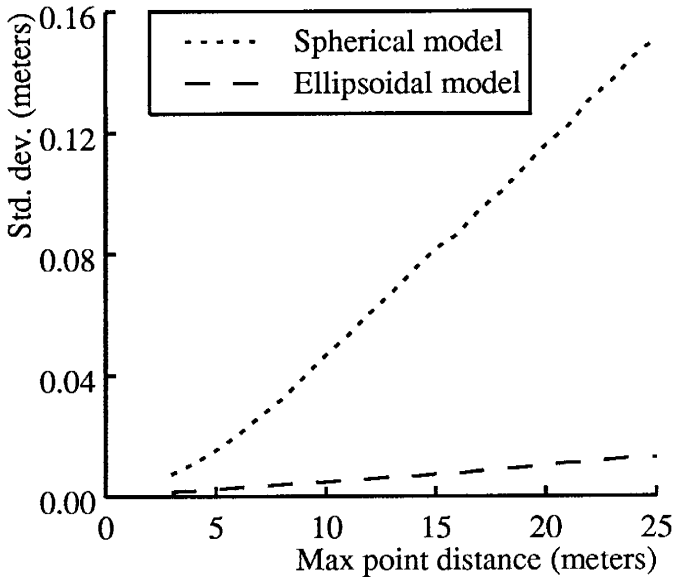



Figure 2.13: Standard deviation of estimated forward distance travelled vs. maximum distance to points

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.

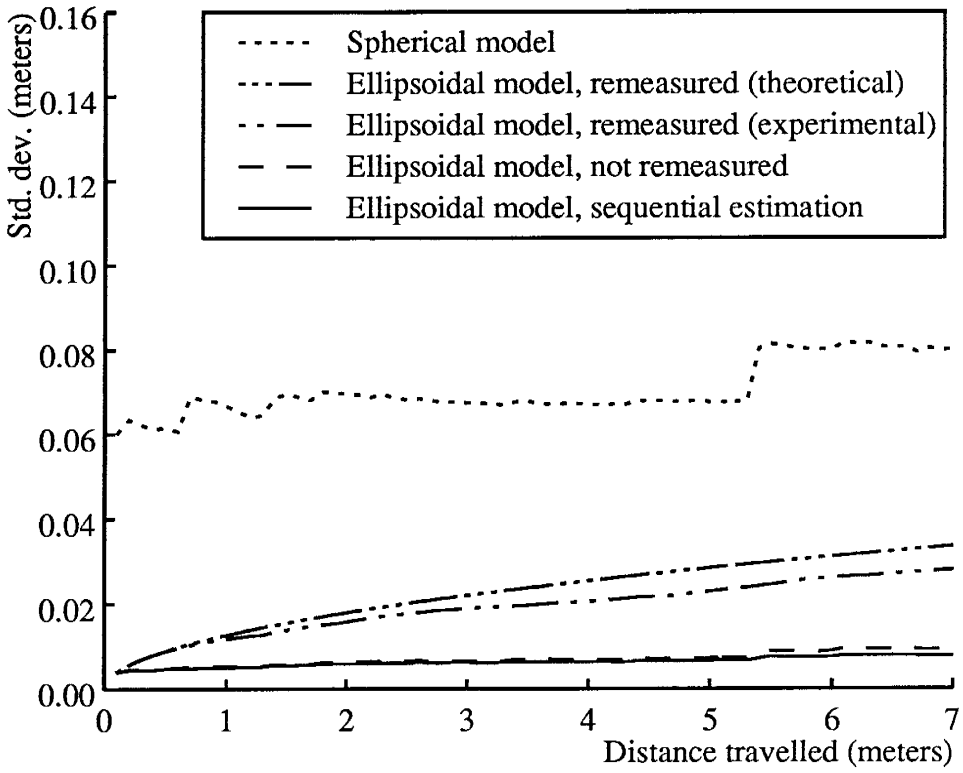



Figure 2.14: Standard deviation of estimated forward distance travelled vs. true distance

- 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 $\sigma_1^2$ 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.

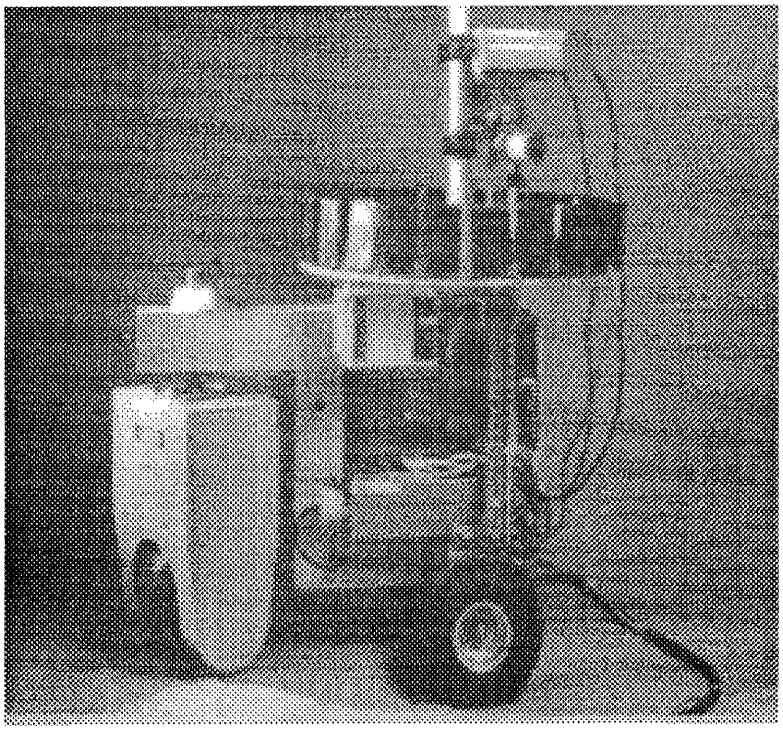


Figure 2.15: The robot vehicle "Neptune" used in the laboratory experiments

### 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

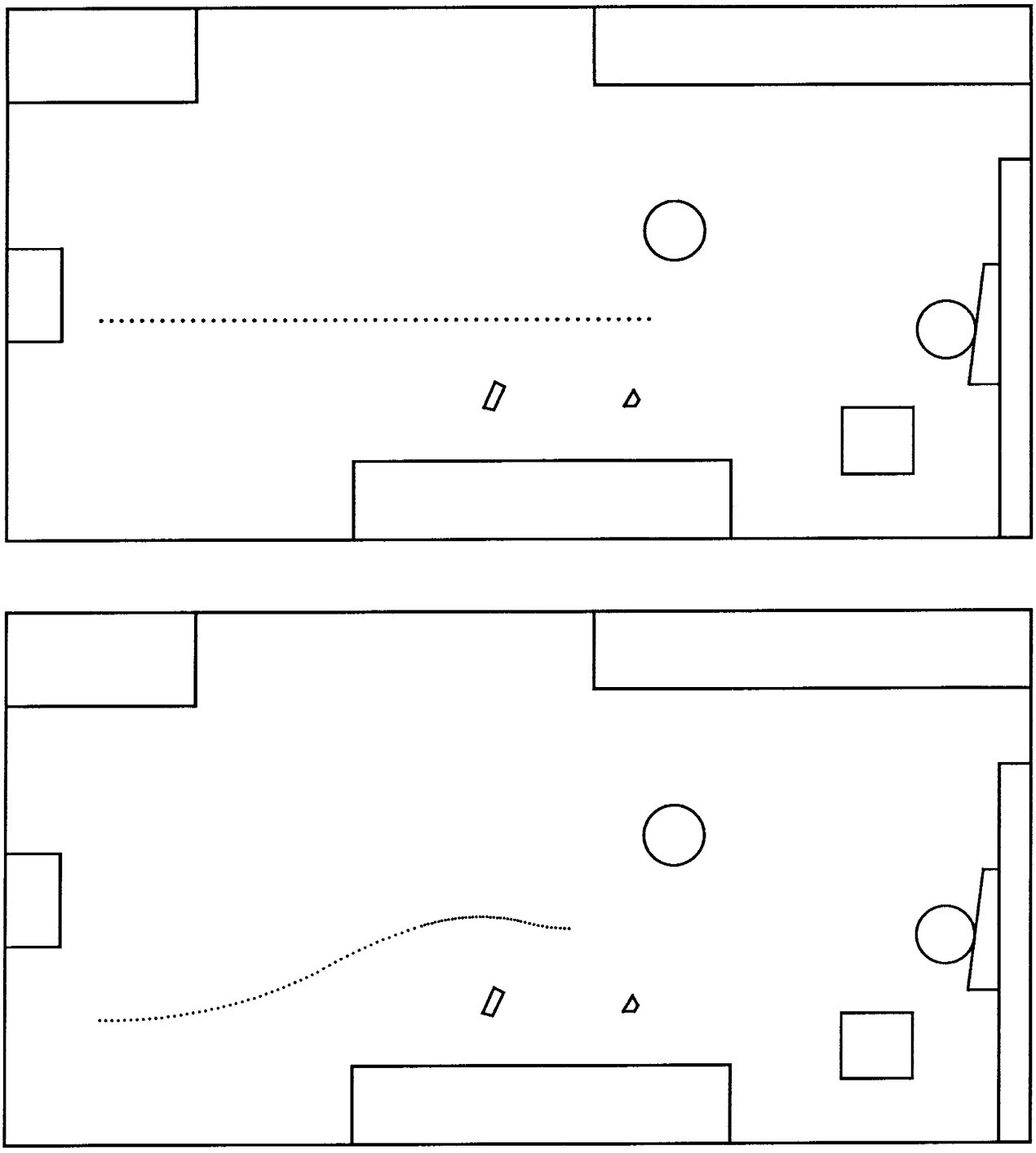


Figure 2.16: Floorplans of the Mobile Robot Lab showing true vehicle trajectories for the experiments with straight and curved motion. Dots mark the vehicle positions where images were acquired.

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

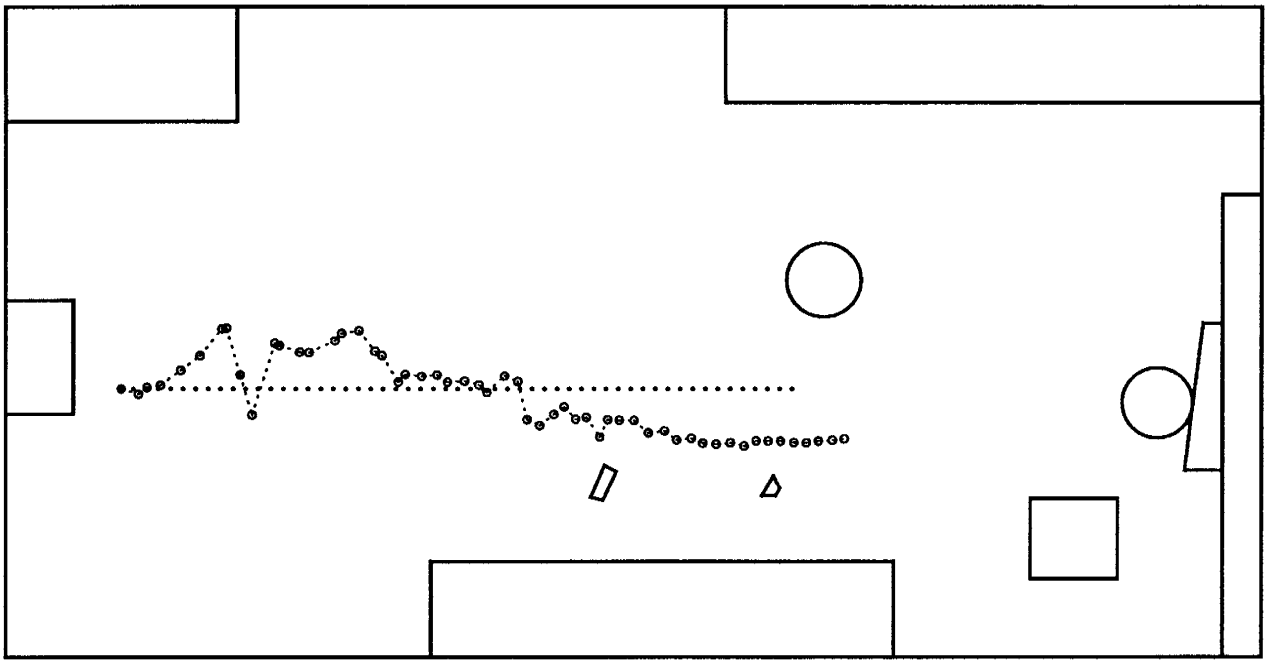
Figure 2.17 shows the vehicle positions estimated by the least-squares (LS) estimator and the maximum-likelihood (ML) estimator for the straight-line trajectory. Both estimators were constrained to estimate only motion in the floor plane; that is, two degrees of translation and one degree of rotation. 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.

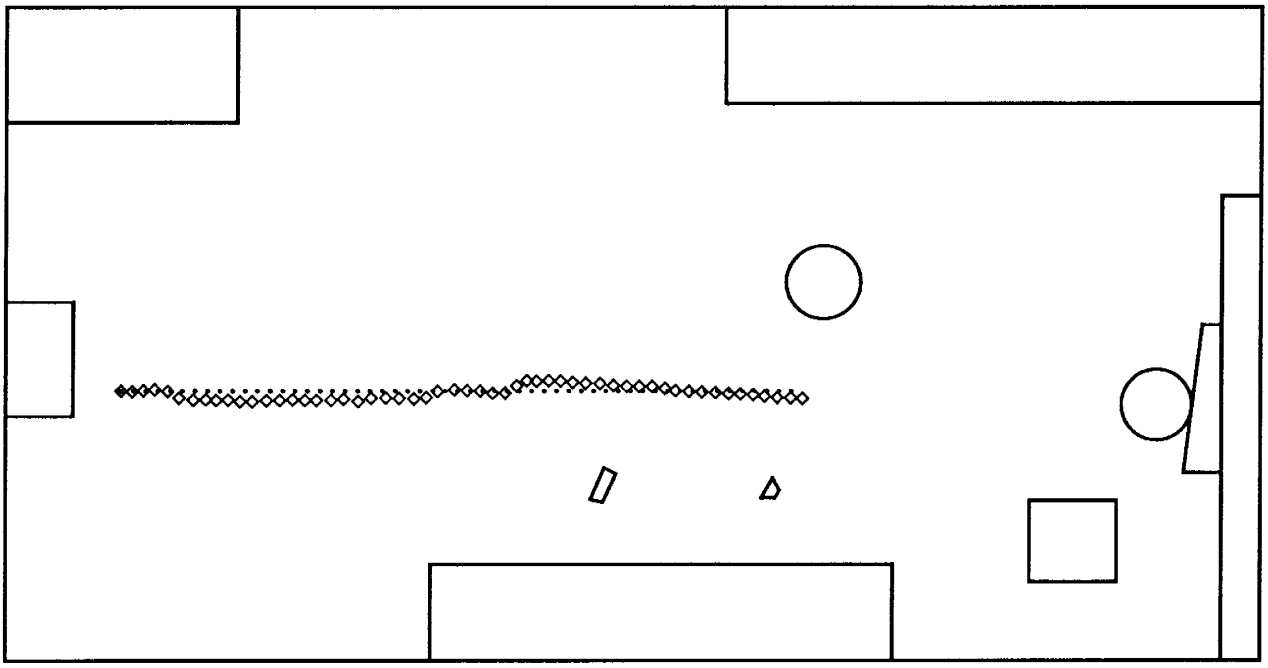
The conditions of this experiment were very similar to those of the multi-step simulation. The results of the experiment and the simulation are in good agreement. In both cases, the LS estimator shows some erratic behavior, whereas the ML estimator is much more stable and precise. Regarding the precision 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

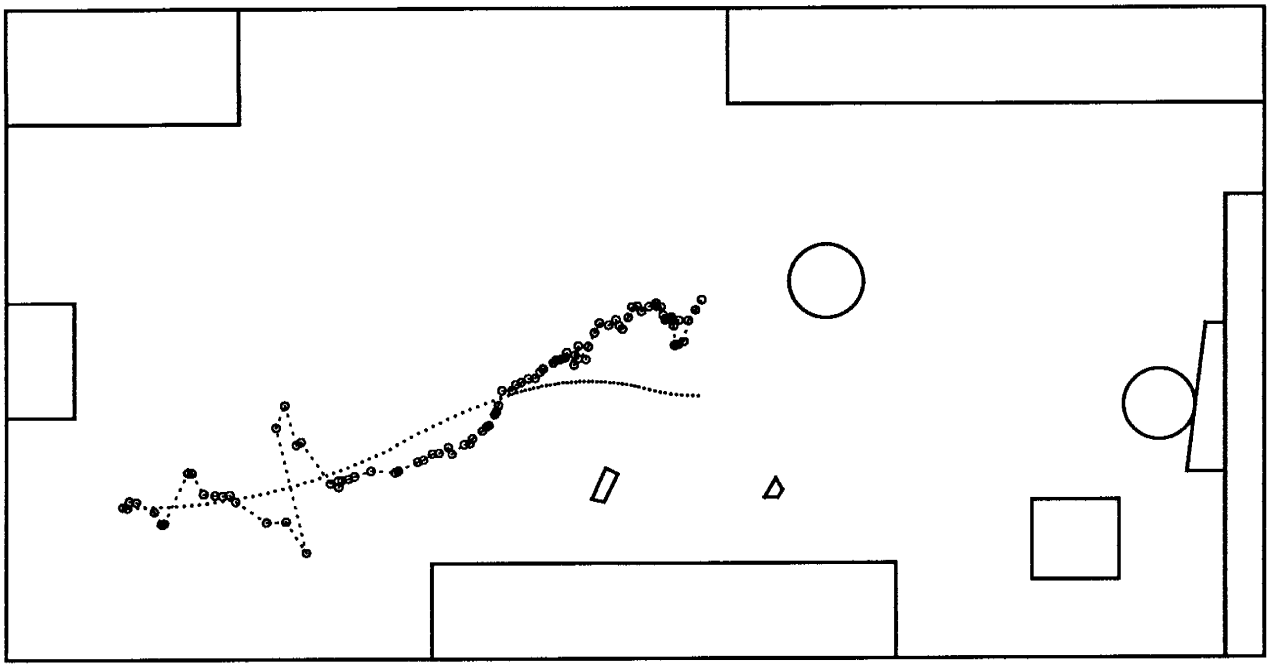


(a) Positions computed with LS estimator

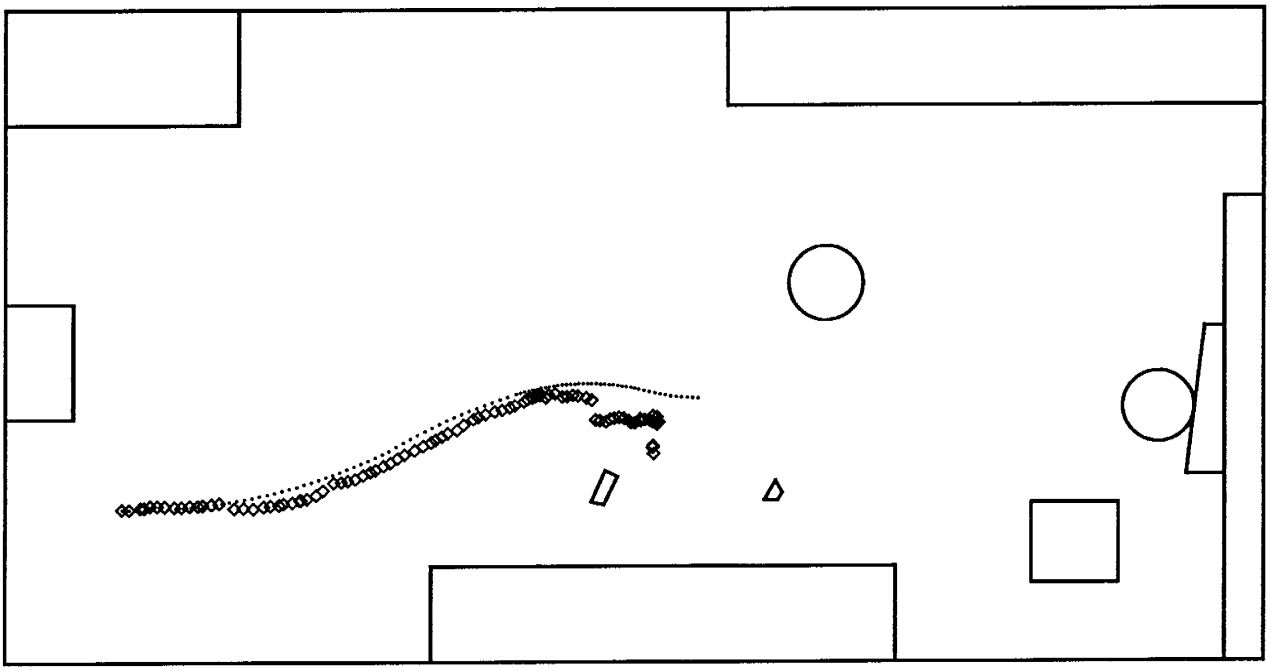


(b) Positions computed with ML estimator

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

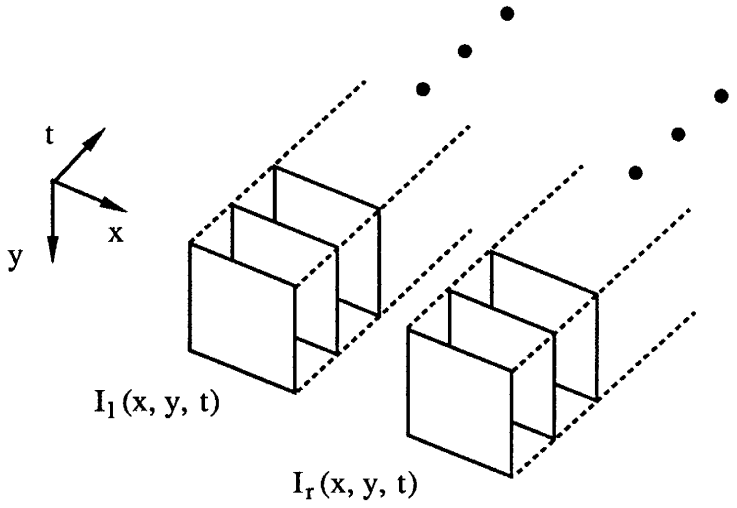



Figure 3.1: Problem: use sampled versions of $I_l$ and $I_r$ to estimate a sampled version of $d$.

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

$$I_l(x,y,t) = I(x,y,t) + n_l(x,y,t) \tag{3.1}$$

$$I_r(x,y,t) = I(x + d(x,y,t),y,t) + n_r(x,y,t)\,, \tag{3.2}$$

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 $\widehat{d}^-(x,y)$ 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 $\widehat{\mathbf{d}}^+$ 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

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

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

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

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 $\lambda$ 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 $\nabla d$. This replaces (3.7) by

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

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

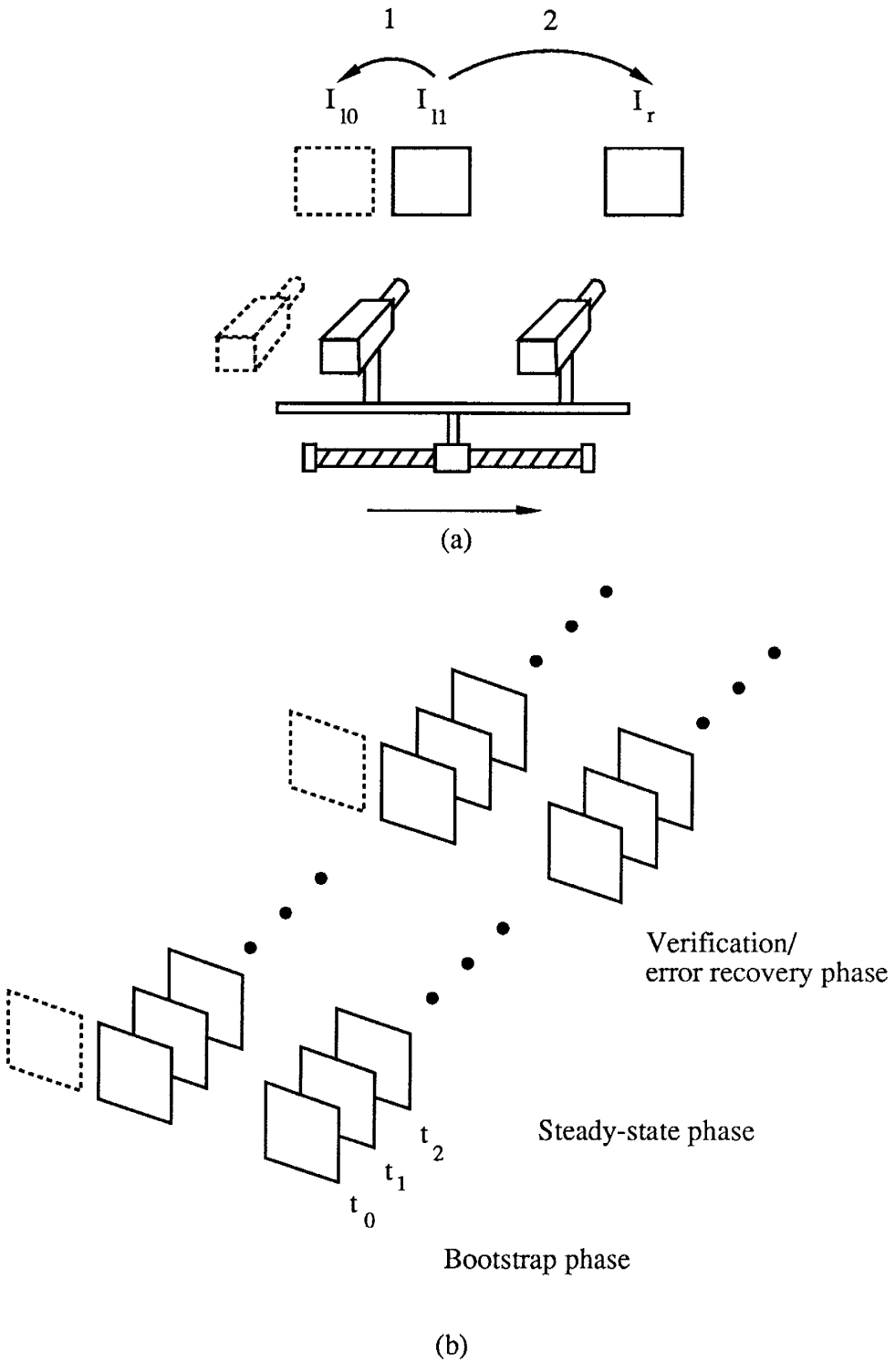



Figure 3.2: Operational framework: (a) bootstrap stage alone, (b) a larger scenario with bootstrap, steady-state, and verification/error recovery modes of operation

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 $\sigma_l^2$ and $\sigma_r^2$, 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 $\Delta x_j$ 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 $\sigma^2$. 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 $\mathbf{e}$ in (4.5), the conditional density of $\mathbf{e}$ 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 $d$:

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

This can be iterated to refine the disparity estimate. In practice, iterating will require estimating the intensity errors $\mathbf{e}$ 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, $E[\tilde{d}^2] = E[(d - \hat{d})^2]$. Assuming $\hat{d}$ is unbiased ($E[\hat{d}] = d$), 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 $\sigma_d^2$ relates the precision of the disparity estimate to the noise level $\sigma^2$ 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 $\sigma_d^2$. 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 $\sigma^2$ is not known in advance, it can be estimated from the residual intensity errors after matching is done. The estimate is given by

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

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 $\hat{\sigma}_d^2$ is a random variable itself, so consideration must be given to the variance of $\hat{\sigma}_d^2$ before it is used. Since $(n-1)\hat{\sigma}_d^2/\sigma^2$ has a $\chi^2$ 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 $\hat{\sigma}_d^2$ will be

$2\sigma^4/(n-1)$. To see what this means in practice, in our experience a typical value for $\sigma^2$ in 8-bit images is about 1.3. For the $5 \times 5$ 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:

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

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 $\widehat{d}_i$ and $\widehat{d}_j$ is $\sigma_{ij} = E[(\widehat{d}_i - d_i)(\widehat{d}_j - d_j)]$. Assuming that the disparity estimates are unbiased, this can be expanded to:

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

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 $\tau$ 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 $\tau$ 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 $\sigma_i$ and $\sigma_j$ 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 scan-lines, 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 $2 \times 2$ 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

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

is evaluated at the initial estimate $\mathbf{d}_0$. As before, this estimate can be iterated until the correction is near zero. The $2 \times 2$ 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 $5 \times 5$ 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 $\chi^2$ goodness of fit test with this data accepts the Gaussian hypothesis at a 70% significance level. Therefore, for a $5 \times 5$ 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.

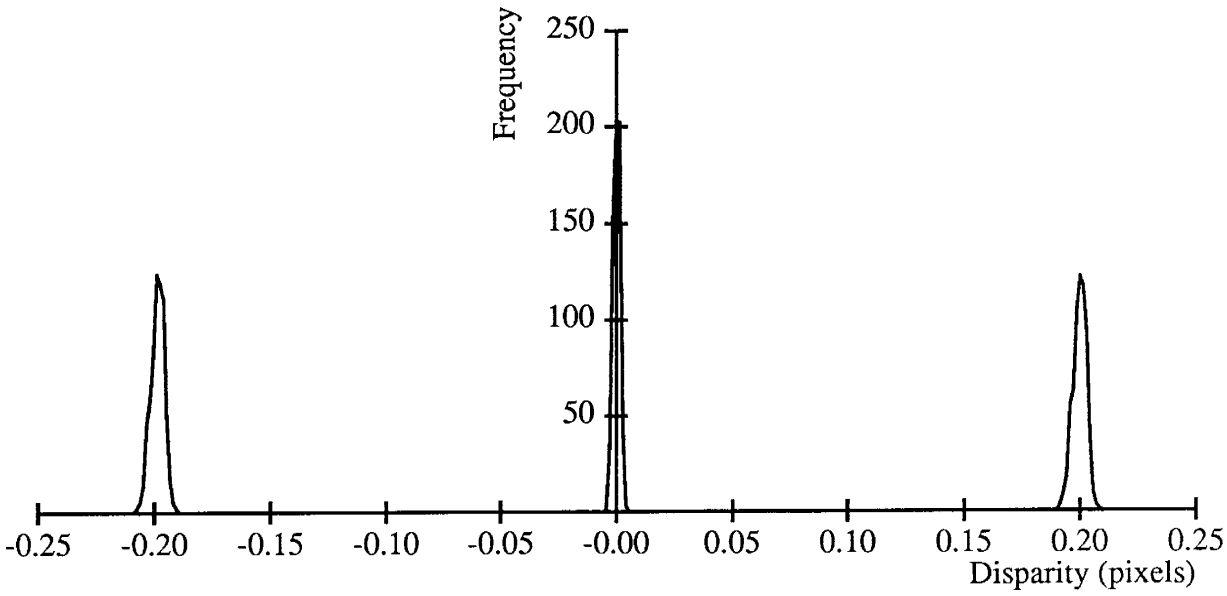



Figure 4.1: Bias plot, linear ramp image

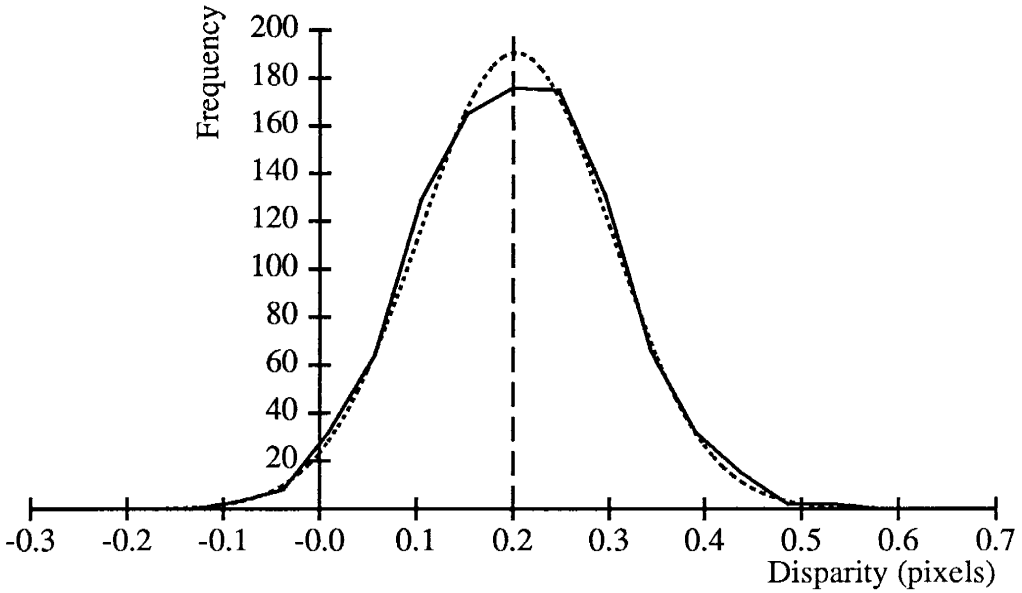



Figure 4.2: Disparity histogram, linear ramp image

### 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 $\chi^2$ 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.

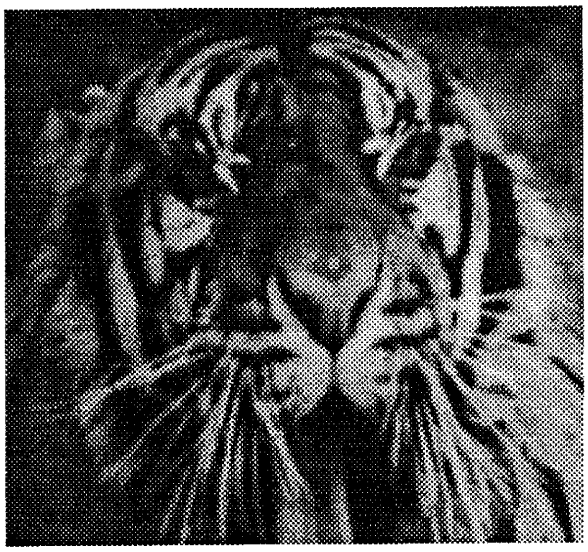


Figure 4.3: Poster used for simulations with realistic data

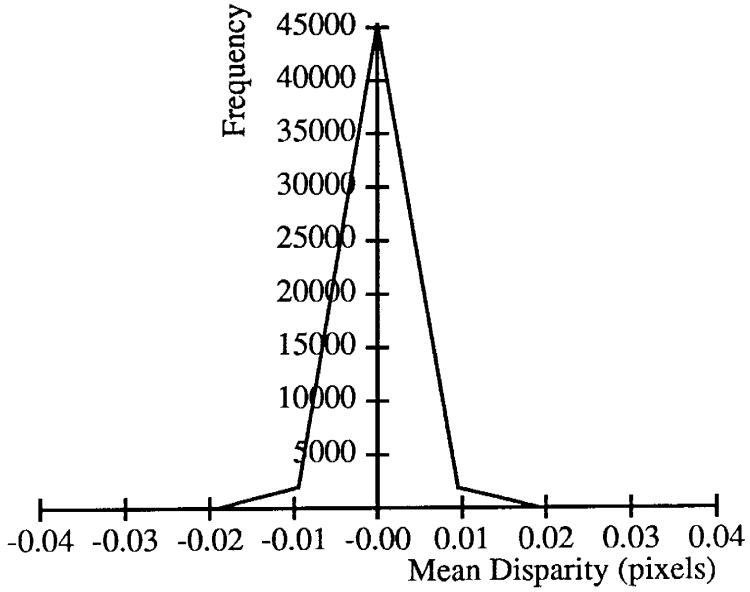



Figure 4.4: Bias plot, tiger poster

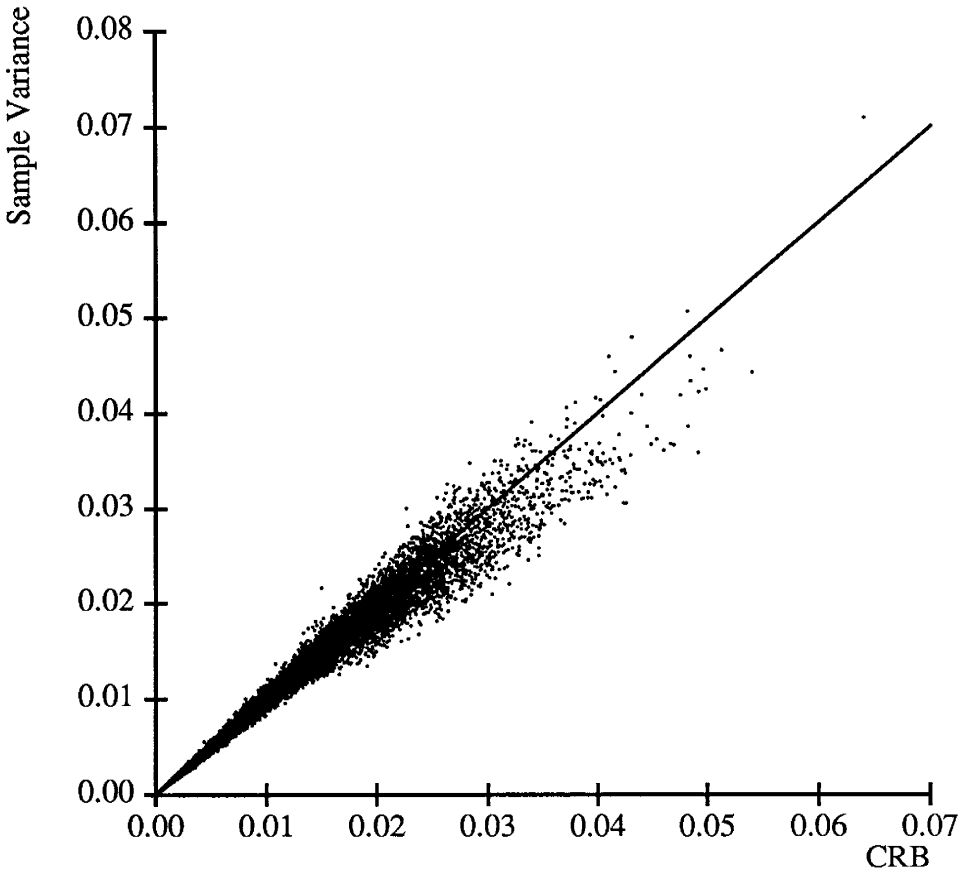



Figure 4.5: Theoretical variance lower bound (CRB) vs. sample variance. The line shows the theoretical variance, the dots show actual variance.

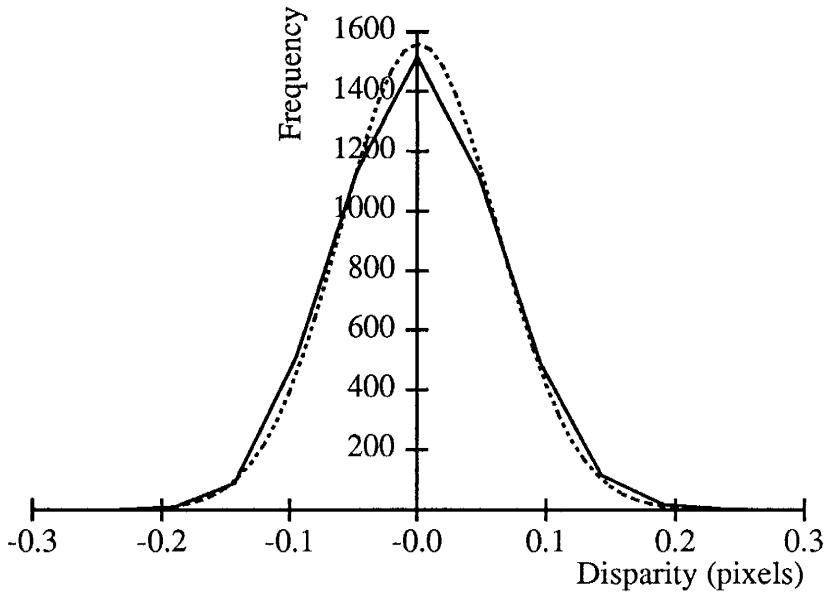



Figure 4.6: Histogram of disparity errors for a single pixel for the matching experiment with the tiger poster and synthetic noise. Superimposed is the Gaussian curve with mean and variance equal to the sample mean and variance.

### 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 $60 \times 70$ 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 $\chi^2$ 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

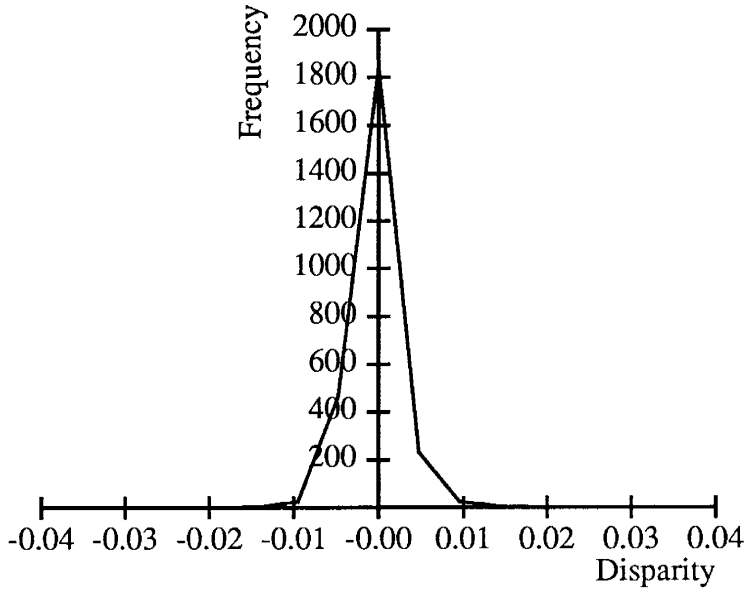



Figure 4.7: Bias histogram for real image with real noise

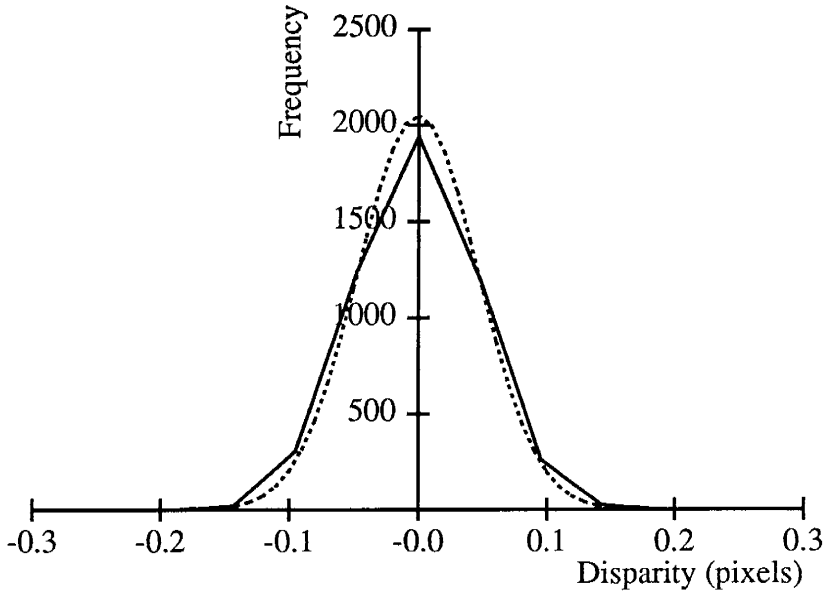



Figure 4.8: Disparity histogram with Gaussian curve for real image with real noise

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, $\sigma^2$ 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 $\sigma$'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 $5 \times 5$ 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 $\sigma^2$, 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

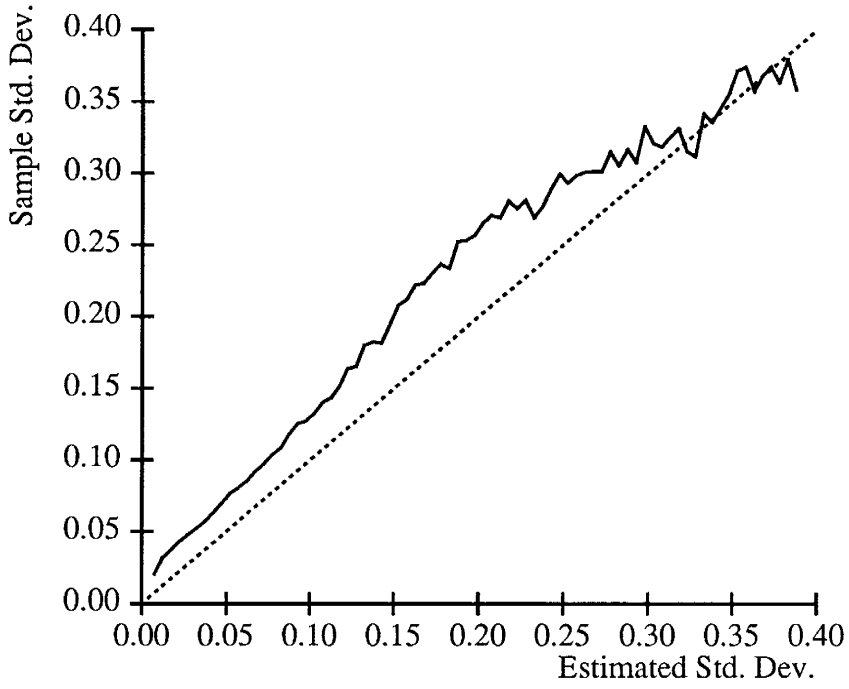



Figure 4.9: Variance scatter plot with spatial averaging

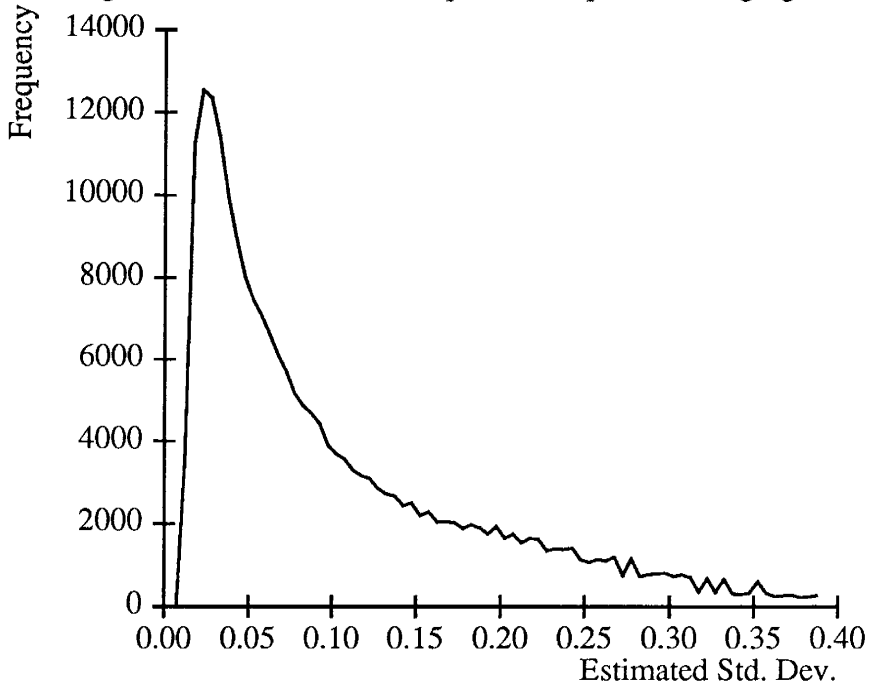



Figure 4.10: Number of pixels in each partition. Partition size is 0.005 sigma.

## 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 $5 \times 5$ 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,

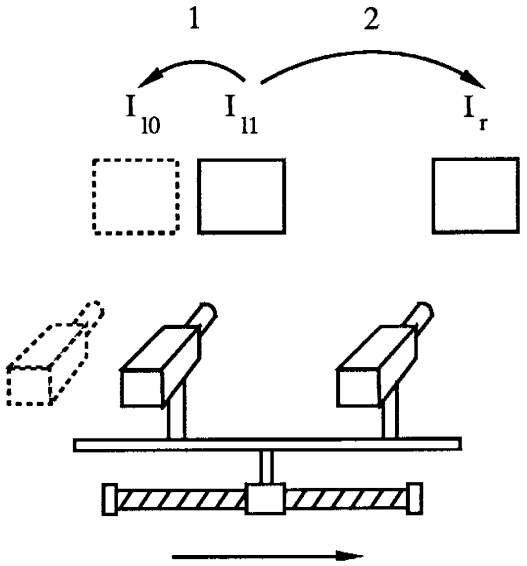



Figure 5.1: Images and matching steps in the bootstrap operation

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 $\mathbf{d}$. We model prior information about $\mathbf{d}$ as jointly Gaussian with mean $\widehat{\mathbf{d}}^-$ and inverse covariance $\mathbf{W}_{\mathbf{d}}^-$. We model intensity differences between the two images as Gaussian conditioned on $\mathbf{d}$. Bayes' theorem is used to obtain an estimate $\widehat{\mathbf{d}}^+$ of $\mathbf{d}$, 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 $\mathbf{d}$ 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,

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

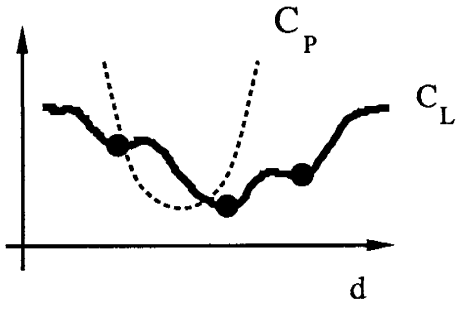



Figure 5.2: Bayesian matching for a single pixel. Curve $C_P$ represents the quadratic cost term from the prior density; curve $C_L$ illustrates the intensity error or "likelihood" term. Local minima of $C_L$ become candidate disparities.

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 $\sigma^2_{ML}$ 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 $\sigma^2_{ML}$, which can be computed in advance. Thresholding $\sigma^2_{ML}$ 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

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

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 $\sigma^2$ 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

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

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

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

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 $\widehat{d}^+_{lr}$ 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 $\sigma^2_d$ 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

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

where $\lambda$ 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),

suppose that an external source provides a prior estimate $\hat{d}^-(x)$ of the disparity field. Then we could replace the derivative constraint term $\lambda(d'(x))^2$ in equation (5.12) with $\lambda(d'(x) - (\hat{d}^-)'(x))^2$. The discrete version of (5.12) becomes

$$\begin{aligned} q(\mathbf{d}) &= \left(\sum_{i=1}^{N} [I_r(x_i - d_i) - I_l(x_i)]^2\right) + \lambda \left(\sum_{i=1}^{N-1} [(d_{i+1} - d_i) - (\hat{d}^-_{i+1} - \hat{d}^-_i)]^2\right) \\ &= \left(\sum_{i=1}^{N} [I_r(x_i - d_i) - I_l(x_i)]^2\right) + \lambda \left(\sum_{i=1}^{N-1} [(d_{i+1} - \hat{d}^-_{i+1}) - (d_i - \hat{d}^-_i)]^2\right) \\ &= \left(\sum_{i=1}^{N} [I_r(x_i - d_i) - I_l(x_i)]^2\right) + \lambda[\mathbf{d} - \hat{\mathbf{d}}^-]^T \mathbf{W}_g [\mathbf{d} - \hat{\mathbf{d}}^-]. \end{aligned} \tag{5.13}$$

This constraint biases the slope of the disparity field estimated from the wide-baseline image pair to be close to that of the prior field measured from the narrow-baseline image pair, rather than close to zero as does the usual gradient constraint. This is very appealing, because it bases the constraint on measured properties of the scene, rather than on heuristic properties of scenes in general. It also suggests that we might do the same with higher-order derivative constraints; however, this will not be pursued here.

We will now make the objective function above more robust and relate it specifically to the Bayesian formulation of the previous section. For each pixel in the scanline, equations (5.12) and (5.13) compare intensities of the two images for that pixel only. Using larger windows, as in the previous section, is more likely to be robust. Recall that we defined the intensity errors in a window around pixel $x_i$ by

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

and that we collected these errors into a vector $\mathbf{e}_i = \mathbf{e}_{lr}(x_i; d)$. With this notation, we replace (5.13) by

$$q(\mathbf{d}) = \left(\sum_{i=1}^{N} \mathbf{e}_i^T \mathbf{e}_i\right) + \lambda[\mathbf{d} - \hat{\mathbf{d}}^-]^T \mathbf{W}_g [\mathbf{d} - \hat{\mathbf{d}}^-]. \tag{5.14}$$

Compared to (5.13) this involves redundant measurements of intensity error; however, this will be a source of robustness. The Bayesian development of the previous section also led to an objective function with a quadratic penalty $(d_i - \hat{d}^-_i)^2/s^-_i$. Incorporating this in our current cost function leads to

$$\begin{aligned} q(\mathbf{d}) &= \left(\sum_{i=1}^{N} w_i \mathbf{e}_i^T \mathbf{e}_i\right) + \left(\sum_{i=1}^{n} \frac{(d_i - \hat{d}^-_i)^2}{s^-_i}\right) + \lambda[\mathbf{d} - \hat{\mathbf{d}}^-]^T \mathbf{W}_g [\mathbf{d} - \hat{\mathbf{d}}^-] \\ &= \left(\sum_{i=1}^{N} w_i \mathbf{e}_i^T \mathbf{e}_i\right) + [\mathbf{d} - \hat{\mathbf{d}}^-]^T \mathbf{W}_s [\mathbf{d} - \hat{\mathbf{d}}^-] + \lambda[\mathbf{d} - \hat{\mathbf{d}}^-]^T \mathbf{W}_g [\mathbf{d} - \hat{\mathbf{d}}^-], \end{aligned} \tag{5.15}$$

where $\mathbf{W}_s = \text{diag}(1/s^-_1, \ldots, 1/s^-_n)$. Previously, the weights $w_i$ were all equal to $1/\sigma^2$; here we allow them to be more general, but leave them unspecified for the moment.

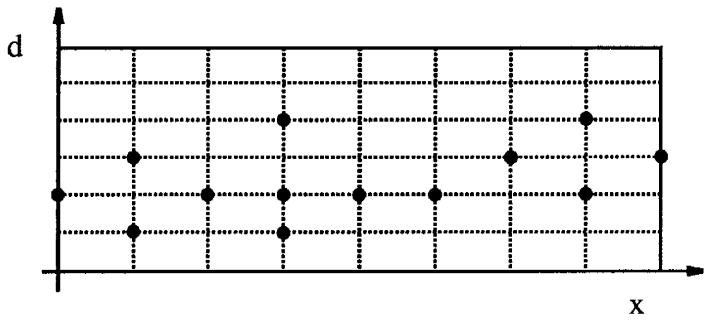
To recapitulate, we have used intuitive reasoning to define an objective function (5.15) of the scanline disparity vector $\mathbf{d}$; an estimate $\hat{\mathbf{d}}^+$ of $\mathbf{d}$ will be one that minimizes this function. The terms of this function measure:

- the similarity of the two images, given the disparity, in a neighborhood around each pixel, summed over all pixels in the scanline;
- the squared deviation of the disparity at each pixel from a prior disparity estimate, summed over all pixels in the scanline; and
- the squared deviation of the first derivative of disparity at each pixel from the prior derivative, summed over all pixels of the scanline.

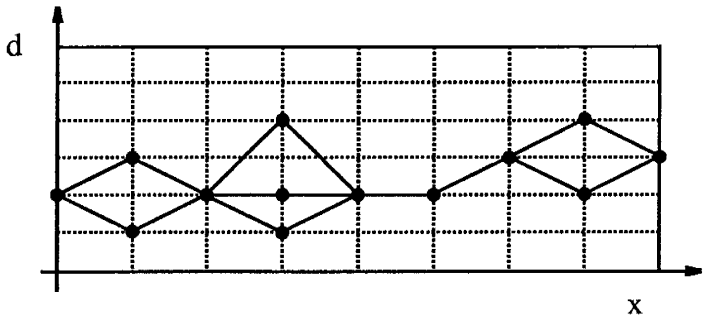
The similarity metric and the per-pixel disparity constraint were motivated in previous sections. The derivative constraint was motivated by smoothness constraints in existing, binocular stereo algorithms; however, the constraint used here is based on deviation from previously measured derivatives, rather than from zero derivative. This is very appealing conceptually, because it uses structural information obtained from the scene at hand, rather than a heuristic that may or may not apply to a given scene. The weights $w_i$, $s_i^-$, and $\lambda$ blend error terms. We have given methods previously for defining $w_i$ and $s_i^-$; $\lambda$ must be chosen heuristically. Note that with $\lambda = 0$, (5.15) decouples such that the disparity at each pixel is estimated independently with the cost function of the previous section.

The next question is how obtain the estimate $\hat{\mathbf{d}}^+$. Treating this as a combinatorial search problem, the question comes in two parts: first, what set of possible vectors $\mathbf{d}$ will be considered, and second, how do we search over this set? The set of possible vectors is obtained as an extension of the method of the previous section. For each pixel in the scanline, we specify a range of possible disparities, compute the similarity measure $\mathbf{e}_i^T\mathbf{e}_i$ for each disparity in this range, and define "candidate" disparities for this pixel to be local minima of the similarity measure. The candidates for all pixels in the scanline can be arranged in an array, as illustrated in figure 5.3a. In this figure, the horizontal axis indexes pixels in the scanline and the vertical axis indexes discrete disparity values; thus, the candidates for each pixel appear as "vertices" in one column. Candidate disparity vectors $\mathbf{d}$ are obtained by selecting one vertex from each column; we now have to search this set to find an optimal vector.

Inspired by the dynamic programming solution developed in [Marroquin85] for a more abstract 1-D matching problem, we note that the cost function (5.15) can be mapped onto the array in figure 5.3a so as to define a weighted, directed graph (figure 5.3b), that candidate vectors $\mathbf{d}$ correspond to paths through this graph, and that a minimum-cost path (equivalent to an optimal estimate of $\mathbf{d}$) can be found very efficiently by dynamic programming. To obtain the graph, note that the first two terms of (5.15), that is the image similarity and the per-pixel constraint terms, associate costs with each candidate (vertex) in the array. The derivative constraint term associates costs with pairs of candidates in adjacent columns. These pairings become edges in the graph. Thus, the value of the objective function for any $\mathbf{d}$ is obtained by adding the vertex and edge costs in the corresponding path through the graph.




(a)




(b)

Figure 5.3: Search graph for joint 1-D matching algorithm: (a) candidate array (b) candidate array with edges, illustrating the graph structure

```
Forward pass:
    For each column i
        For candidate j
            Compute vertex cost
            For all candidates in column i-1
                Compute edge costs
                Compute total path cost as sum of previous total path cost,
                    edge cost, and new vertex cost
            Choose the minimum total cost, assign it as the total cost to this vertex,
                and set a back pointer to the appropriate vertex of the
                previous column.
Reverse pass:
    Find the minimum total cost in column N
    Trace the back pointers to find the optimal path
```

Figure 5.4: Structure of the DP algorithm

For graphs with this structure, it is well-known that minimum-cost paths can be found by dynamic programming (DP) [Sankoff83,Sedgewick83]. In fact, dynamic programming has been applied previously to feature-based approaches to stereo that seek to match extracted edges [Baker82] or intervals between edges [Ohta85]. Our approach differs from these in several respects, in particular by attempting to estimate depth at each pixel and by incorporating cost terms from the prior density. Nevertheless, once the graph and the weights are defined, the optimization algorithm is very similar. The data structure necessary for the algorithm consists of two fields for each vertex. One field stores the total cost of the optimal path up to this node; the other is a back pointer that indicates the immediately preceding vertex on the optimal path (i.e. for column $i$, it points to a vertex in column $i-1$). The algorithm itself consists of a forward pass that computes the path costs and the back pointers, followed by a reverse pass that traces the back pointers to extract the optimal path from the data structure (figure 5.4). The forward pass considers each column of the array in turn from left to right across the scanline. Each column is processed with a nested loop that considers each vertex in that column, and for each vertex in the current column considers the edges to all vertices in the previous column. For each edge, the cost of the optimal path beginning with that edge is the sum of the optimal cost at the previous vertex and the costs for this edge and the current vertex, as given by equation (5.23). The minimum cost over all edges is stored at the current vertex together with a pointer to the appropriate vertex of the previous column. This procedure is repeated for all vertices in the current column, then for all columns across the scanline. The reverse pass locates the end of the optimal path by finding the minimum-cost vertex in the final column, then traces the back pointers from there.

The correctness of this algorithm can be proven by induction on the path length [Sankoff83]. It is easy to see that the complexity of the algorithm is $O(NC^2)$, where $C$ is the maximum number of candidate vertices per column. A deficiency of the algorithm as described is that it contains no mechanism for detecting occluded pixels along the scanline. Approaches to this problem are suggested in [Drumheller86,Ohta85]; we leave this issue for the future.

In this algorithm, the need for "global" optimization via dynamic programming results from the pairwise costs induced by the derivative constraint. As we saw, this constraint can be expressed as the quadratic form $[\mathbf{d} - \hat{\mathbf{d}}^-]^T \mathbf{W}_g [\mathbf{d} - \hat{\mathbf{d}}^-]$, where $\mathbf{W}_g$ is tri-diagonal. In fact, any quadratic form with a tri-diagonal weight matrix can be mapped onto the same graph structure, so that the global optimization can be done with the DP algorithm described above. Constraints based on second derivatives can also be expressed as quadratic forms if second differences are used to approximate the derivatives. However, the resulting weight matrices are penta-diagonal and require a more complex optimization algorithm.

In summary, we have described an approach to jointly estimating the vector $\mathbf{d}$ of disparities for an entire scanline, where the need for joint estimation came from a surface derivative constraint motivated by geometric reasoning. This constraint differs from constraints used by previous, binocular stereo algorithms in that the previous algorithms penalized departure of the derivative from zero, whereas our algorithm penalizes departure from derivative measurements made with the narrow-baseline image pair. This is a valuable conceptual step toward sensing strategies that obtain constraint from previous measurements of the scene at hand, rather than general heuristics. However, the objective function above was still obtained via ad hoc reasoning and contains weighting constants that must be defined heuristically. If joint estimation is appropriate, it is preferable to derive this from first principles, using a statistical formulation of the estimation problem. The observation made in chapter 4 about adjacent disparity estimates being correlated suggests one way in which this may be done. We derive this next.

### Bayesian Formulation with Correlated Prior

We will use Bayes' theorem and the MAP criterion to derive a joint estimate for the vector $\mathbf{d}$ of disparities across the entire scanline. We will make the same simplification as section 5.1.1 of ignoring correlation between the prior density of $\mathbf{d}$ and the noise in the wide-baseline images. Therefore, we begin with the following vector generalization of Bayes' theorem,

$$f(\mathbf{d}|\mathbf{e}) = \frac{f(\mathbf{e}|\mathbf{d})f(\mathbf{d})}{f(\mathbf{e})}, \tag{5.16}$$

where $\mathbf{e}$ is a vector of observations associated with $\mathbf{d}$. As in the single-pixel case, the denominator of (5.16) is a constant. Next, we consider in turn the definition of the prior density $f(\mathbf{d})$, the definition of the conditional density $f(\mathbf{e}|\mathbf{d})$, the objective function obtained via the MAP criterion, and the solution of the resulting optimization problem. The objective function we obtain will be closely related to the objective function obtained with the derivative constraint and will be solvable via the same dynamic programming algorithm.

**Prior Density** The joint prior density for the scanline case is a specialization of (3.4) to one-dimensional images:

$$f(\mathbf{d}) = (2\pi)^{-N/2}|\mathbf{W}_{\mathbf{d}}^{-}|^{1/2}\exp\left\{-\frac{1}{2}\left[\mathbf{d}-\hat{\mathbf{d}}^{-}\right]^{T}\mathbf{W}_{\mathbf{d}}^{-}\left[\mathbf{d}-\hat{\mathbf{d}}^{-}\right]\right\}, \tag{5.17}$$

Within the general scope of the Gaussian model, various specific models are obtained by defining $\mathbf{W}_{\mathbf{d}}^{-}$ accordingly. If there is no prior information, then $\mathbf{W}_{\mathbf{d}}^{-} = 0$; in this case, $\hat{\mathbf{d}}^{-}$ can be left unspecified. For the independent pixel model of the previous section, the inverse covariance matrix $\mathbf{W}_{\mathbf{d}}^{-}$ is diagonal,

$$\mathbf{W}_{\mathbf{d}}^{-} = \left[\text{diag}(s_1^{-},\ldots,s_N^{-})\right]^{-1}, \tag{5.18}$$

so the joint density reduces to the product of the individual prior densities for each pixel:

$$\begin{aligned} f(\mathbf{d}) &= \prod_i f(d_i) \\ &= (2\pi)^{-N/2}\left|\prod_i \sigma_i^{-}\right|^{-1}\exp\left\{-\frac{1}{2}\sum_i \frac{(d_i-\hat{d}_i^{-})^2}{s_i^{-}}\right\}. \end{aligned}$$

This independence in the model allowed us to estimate the disparity independently for each pixel. In this section, we introduce a model for correlation within the scanline that admits efficient search for the optimal disparity estimate $\hat{\mathbf{d}}^{+}$.

As we noted in chapter 4, the use of matching windows with size greater than $1 \times 1$ induces a correlation among pixels of the resulting disparity field[1]. In chapter 4 we showed the correlation between two pixels is roughly a triangular function of distance between the pixels. A tractable approximation to this function is to model the covariance of the field as an exponential function of distance,

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

where $\sigma_i^{-}$ and $\sigma_j^{-}$ are the standard deviations obtained at pixels $x_i$ and $x_j$. For $i = j$, this yields the variance $s^{-}$ at each pixel, as we have used previously. For $i \neq j$, the covariance of $d_i$ and $d_j$ is modelled as the product of the respective (non-stationary) standard deviations and the (stationary) correlation $\rho^{|i-j|}$. Therefore, this model incorporates both the variance estimates obtained earlier and the decay of the correlation with distance. The fact that the correlation in the model tapers exponentially, instead of dropping to zero after a short distance, should not be a serious limitation. The value of the correlation coefficient $\rho$ must be specified to complete the model. From chapter 4 we know that $\rho$ should be between zero and one; beyond that, we leave the choice of good values of $\rho$ as a matter for future research.

From the computational standpoint, the attractiveness of this model stems from structure of

[1]Correlations will also be present if the image noise is not white.

the corresponding covariance matrix $\Sigma_{\mathbf{d}}^-$ and inverse covariance matrix $\mathbf{W}_{\mathbf{d}}^-$:

$$\begin{aligned}
\Sigma_{\mathbf{d}}^- &= \begin{bmatrix} \sigma_1^- & & & \\ & \sigma_2^- & & \\ & & \ddots & \\ & & & \sigma_N^- \end{bmatrix} \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-2} & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-3} & \rho^{N-2} \\ \rho^2 & \rho & 1 & \cdots & \rho^{N-4} & \rho^{N-3} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ \rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \cdots & \rho & 1 \end{bmatrix} \begin{bmatrix} \sigma_1^- & & & \\ & \sigma_2^- & & \\ & & \ddots & \\ & & & \sigma_N^- \end{bmatrix} \\
&= \mathbf{VCV} \\
\mathbf{W}_{\mathbf{d}}^- &= \mathbf{V}^{-1}\mathbf{C}^{-1}\mathbf{V}^{-1},
\end{aligned} \tag{5.19}$$

where $\mathbf{C}^{-1}$ is [Graybill83, p. 201]

$$\frac{1}{1-\rho^2}\begin{bmatrix} 1 & -\rho & 0 & \cdots & 0 & 0 \\ -\rho & 1+\rho^2 & -\rho & \cdots & 0 & 0 \\ 0 & -\rho & 1+\rho^2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -\rho & 1 \end{bmatrix}. \tag{5.20}$$

In other words, the inverse covariance matrix is tri-diagonal with elements

$$w_{i,j}^- = \begin{cases} \dfrac{1}{s_i^-(1-\rho^2)} & i = j = 1 \text{ or } N \\ \dfrac{1+\rho^2}{s_i^-(1-\rho^2)} & 0 < i = j < N \\ \dfrac{-\rho}{\sigma_i\sigma_j(1-\rho^2)} & |i-j| = 1 \\ 0 & \text{otherwise} \end{cases}$$

It is interesting to note that as $\rho$ approaches 1, $\mathbf{W}_{\mathbf{d}}^-$ approaches

$$\frac{1}{1-\rho^2}\mathbf{W}_s.$$

For $\rho = 1$, $\mathbf{W}_{\mathbf{d}}^-$ is undefined.

**Conditional Density** The conditional density $f(\mathbf{e}|\mathbf{d})$ is a vector generalization of the densities used in sections 4.1 and 5.1. In this case, it models the probability of observed intensity errors across the entire scanline, conditioned on the given value of the disparity vector $\mathbf{d}$. For clarity, we will define the density first for the case in which we make a single observation

$$e_{lr}(x_i; d_i) = I_r(x_i - kd_i) - I_{l_1}(x_i)$$

for each pixel, instead of observing errors in a larger window around each pixel. More general cases will be considered subsequently.

Letting $e_i = e_{lr}(x_i; d_i)$, we collect the observations for the entire scanline into the vector $\mathbf{e} = [e_1, \ldots, e_n]^T$. Under our noise model, the conditional density of this vector is jointly Gaussian with zero mean vector and inverse covariance matrix

$$\mathbf{W_e} = \frac{1}{\sigma^2}\mathbf{I},$$

where $\mathbf{I}$ is the $N \times N$ identity matrix. That is,

$$\begin{aligned} f(\mathbf{e}|\mathbf{d}) &= \frac{1}{(2\pi)^{N/2}\sigma} \exp\left\{-\frac{1}{2\sigma^2}\mathbf{e}^T\mathbf{W_e}\mathbf{e}\right\} \\ &= \frac{1}{(2\pi)^{N/2}\sigma} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{N} e_i^T e_i\right\} \\ &= \prod_{i=1}^{N} f(e_i|d_i). \end{aligned}$$

This is a straightforward generalization of the single-pixel case in equation (4.3), restricted to the case of $1 \times 1$ windows.

For better noise immunity, we wish to use more than one obervation per pixel. This leads to considering larger windows around each pixel, we used before. However, windows with a width of more than one pixel embody redundant measurements; that is, the windows for adjacent pixels overlap, so that the intensity error at each pixel is measured more than once. This can produce more observations than there are independent random variables in the original images, which results in the covariance matrix of $\mathbf{e}$ being singular. To avoid this, we can take windows with width of one pixel and vertical extent $v$ of greater than one pixel. Collecting the observations for each window in the vector $\mathbf{e}_i$, the joint observation vector $\mathbf{e} = [\mathbf{e}_1^T, \ldots, \mathbf{e}_N^T]^T$ has dimension $Nv$. The inverse covariance matrix is still $\mathbf{W_e} = (1/\sigma^2)$, but now has dimension $Nv \times Nv$.

For noise immunity we may still want to use windows with both horizontal and vertical extent, say $h \times v$. This implies that each pixel will be overlapped by $h$ windows and the intensity error at each pixel will be measured $h$ times. A reasonable model for this case is to divide the quadratic form in the density by $h$. This yields

$$f(\mathbf{e}|\mathbf{d}) = \frac{1}{(2\pi)^{Nv/2}\sigma} \exp\left\{-\frac{1}{2h\sigma^2}\sum_{i=1}^{N} \mathbf{e}_i^T\mathbf{e}_i\right\}.$$

**MAP Objective Function** Applying the MAP criterion, we obtain a likelihood function by taking the log-probability of f(**d**|**e**) to obtain

$$\begin{aligned} \ell(\mathbf{d}) &= \ln f(\mathbf{d}|\mathbf{e}) \\ &= -\frac{1}{2}\left\{\mathbf{e}^T\mathbf{W_e}\,\mathbf{e} + [\mathbf{d} - \hat{\mathbf{d}}^-]^T\mathbf{W}_{\mathbf{d}}^- [\mathbf{d} - \hat{\mathbf{d}}^-]\right\} + K, \end{aligned} \tag{5.21}$$

where $K$ is a constant. Therefore, with $\mathbf{W_e}$, the MAP estimate of $\mathbf{d}$ to pixel resolution minimizes

$$\frac{1}{\sigma^2}\left(\sum_{i=1}^{N}\mathbf{e}_i^T\mathbf{e}_i\right) + [\mathbf{d}-\hat{\mathbf{d}}^-]^T\mathbf{W}_{\mathbf{d}}^-[\mathbf{d}-\hat{\mathbf{d}}^-]. \tag{5.22}$$

As with the gradient constraint algorithm, this objective function is a sum of the squared intensity errors observed in each window and a quadratic form with a tri-diagonal weight matrix. The only parameter in this function is the correlation coefficient $\rho$. Values for $\rho$ can be determined experimentally or by choosing $\rho$ to satisfy some criterion of fit between the exponential covariance and the triangular covariance derived in chapter 4. If $h \times v$ windows are used, we also multiply the second term by a factor of $1/h$.

**Optimization Algorithm** The structure of (5.22) and the tri-diagonality of $\mathbf{W_e}$ imply that (5.22) can be minimized by the same dynamic programming algorithm used above. The graph is determined by match candidates identified as before. The intensity error terms assign costs to the vertices. The quadratic form from the prior model can be expanded as

$$[\mathbf{d}-\hat{\mathbf{d}}^-]^T\mathbf{W_e}[\mathbf{d}-\hat{\mathbf{d}}^-] = \left(\sum_{i=1}^{N}\frac{1}{\sigma^2}\mathbf{e}(d_i)^T\mathbf{e}(d_i)\right) + \frac{1}{1-\rho^2}\left\{\left[\delta_1^2 - 2\rho\delta_1\delta_2 + \delta_2^2\right] + \sum_{i=3}^{N}\left[\delta_{i-1}\rho - \delta_i\right]^2\right\}, \tag{5.23}$$

where $\delta_i = (d_i - \hat{d}_i^-)/\sigma_i^-$. Each term in braces contains only a consecutive pair of disparities $(d_{i-1}, d_i)$. Therefore, these terms can be associated as edge costs in the graph.

This assignment yields a weighted digraph as before; therefore, DP again suffices to find a minimum-cost path. Such a path defines optimal disparity estimates to pixel resolution for each pixel in the scanline. From this point, sub-pixel disparity estimates and posterior variances can be obtained most simply by using the methods of section 5.1.1.

### Overall Algorithm for the Bootstrap Operation

With the joint 1-D formulations, the entire bootstrap procedure is the same as with the fully independent formulation, except that we use one of the joint algorithms to match the wide-baseline image pair. That is, we use the variance measure computed from $I_{l_1}$ to determine where to match, then use the independent algorithm to estimate depth from the narrow-baseline image pair, then use a coupled model and dynamic programming to match the wide-baseline image pair. Thus, for the wide-baseline pair, the following procedure is applied to each scanline:

- Disparity candidates are obtained for each pixel as local minima of the ML-based intensity error term. The search window for each pixel is obtained either from a global disparity limits fixed for the whole image or from a tolerance ($\approx 0.5$ pixels) applied to the narrow-baseline estimate for that pixel. The latter is generally more effective, but occasionally produces errors if the narrow-baseline estimate is wrong.

- Dynamic programming is used to obtain optimal disparity estimates to pixel resolution.

- Linearization and iteration are used independently at each pixel to obtain a sub-pixel disparity estimate and, if desired, a posterior variance estimate at each pixel.

Some pixels will have no candidates, either because the interest operator rejected them or because the only local minima were at the ends of the search interval, which is rejected as unreliable. This breaks the scanline into sections of contiguous pixels having candidates, separated by pixels with no candidates. The above algorithm processes each contiguous section separately. The entire algorithm is applied independently to each scanline. Therefore, the opportunity exists to process all scanlines in parallel.

### Discussion

The algorithms developed in this section extend the "fully independent" algorithm of section 5.1.1 to joint estimation of disparity for all pixels in the entire scanline. This results from modifying the mathematical formulation of the matching problem to include coupling of the disparity estimates at adjacent pixels. We started on an intuitive basis by looking at the one-dimensional specialization of the disparity gradient constraints employed in the literature. This results in an objective function with a penalty term containing the squared derivatives of disparity across the scanline. The penalty term biases the estimated disparity field to have low derivative and expresses the heuristic assumption that surfaces tend to be smooth.

In the bootstrap scenario, we replace this general heuristic with the more powerful observation that the disparity field estimated from the wide-baseline image pair should be approximately equal to the field estimated from the narrow-baseline image pair. In the fully independent algorithm, this idea is expressed by the prior density for the disparity at each pixel. In the joint 1-D case, we express it in two alternate fashions. The first replaces the smoothness-based derivative penalty with a new penalty for deviation from the disparity derivative measured with the narrow-baseline image pair. This is a useful step toward stereo constraints based on measured properties of the scene at hand, instead of on general heuristics. However, the new constraint still has an *ad hoc* element, because there is no basis for determining the relative weight to accord to different terms of the objective function. Therefore, we derived a second approach that models correlation in the disparity field estimated from the narrow-baseline image pair as exponential. This leads to a new objective function with a better statistical foundation. For both of the joint 1-D formulations, as well as the original derivative constraint from the literature, optimal disparity esitmates for the entire scanline can be found efficiently by dynamic programming.

The algorithms developed here do not account for the possibility of occluded pixels. Doing so reliably is a murky issue that is beyond our scope. Dynamic programming algorithms for edge-based approaches to stereo that have addressed this issue are discussed in [Baker82,Ohta85]. Also, the Bayesian model is not complete since we do not derive joint, sub-pixel disparity estimates or a joint posterior covariance matrix. In fact, applying such a derivation to the narrow-baseline image pair may lead to better correlation models than the one employed here. These issues are left for the future.

### 5.1.3 Joint 2-D Model

So far, we have developed matching algorithms for cases in which depth estimation is either completely uncoupled or coupled in only one direction. A logical extension is to consider coupling in both directions. The reason for doing so is the same as for 1-D coupling: better depth estimates may result from taking into account consistency over neighborhoods. The questions to ask are how to achieve 2-D coupling and how the results compare to 1-D coupling, in terms of quality of the estimates and cost of the algorithm. In this section, we suggest answers to these questions and draw the tentative conclusion that 2-D coupling may not be a valuable extension to the 1-D algorithms described above. Confirming this conclusion will require further study.

In the past, 2-D coupling has been realized by a variety of local support and surface smoothness constraints. These are all motivated by the observation [Marr76] that surfaces in the world tend to be smooth, so that depth in small image neighborhoods tends to be uniform. In stereo algorithms, this observation is embodied in penalty functions that bias the estimated disparity field to exhibit low spatial variation. The overall penalty is a sum of terms for each pixel; for each pixel, the penalty term is a function of the candidate disparities in a small, 2-D neighborhood around the pixel. Local support constraints [Drumheller86,Marr76,Marroquin85,Prazdny85] [Stewart88,Szeliski85] and regularization-based surface smoothness constraints [Barnard89] [Boult88,Horn86,Poggio85,Witkin87] both take this form. In fact, these approaches can be implemented with 1-D or 2-D support; the attraction of 2-D support is that it enforces consistency between scanlines, as well as within scanlines.

One way to model 2-D coupling in our approach is to use a penalty based on the full disparity gradient, rather than just the $x$ derivative. As in the 1-D case, in matching the wide-baseline image pair we would penalize departure from the gradient measured in the narrow-baseline pair, not departure from zero. In the discrete version of the problem, this would be expressed by adding a squared-error term for disparity differences in the vertical direction as well as the horizontal. The overall objective function would take the form

$$q(\mathbf{d}) = \sum_{i,j} \mathbf{e}_{i,j}^T \mathbf{e}_{i,j} + \sum_{i,j} [(d_{i+1,j} - d_{i,j}) - (\bar{d}_{i+1,j} - \bar{d}_{i,j})]^2 + \sum_{i,j} [(d_{i,j+1} - d_{i,j}) - (\bar{d}_{i,j+1} - \bar{d}_{i,j})]^2.$$

This is similar to approaches described in [Barnard89,Poggio85,Witkin87]. An alternate approach is to derive a 2-D covariance model for the prior disparity field, as we did in the 1-D case. Either way, the result is an objective function coupled in both dimensions.

The next question is how to find the global minimum of such an objective function. Past approaches have used gradient descent in scale space [Witkin87], simulated annealing [Barnard89], and various, mostly iterative methods that use the penalty term to decide matches locally [Marroquin85][Prazdny85][Szeliski85]. Some of these algorithms, notably [Barnard89] and [Witkin87], have achieved impressive results on real imagery. However, the cost and complexity of the search algorithms appears to be significantly higher than comparable approaches restricted to 1-D would be, including the specific 1-D algorithms developed here.

At this point, we are led to the following question: since the principal advantage of 2-D coupling appears to be enforcement of coherence across scanlines, as well as within scanlines, is such coupling necessary with area-based matching operators such as that used here? Because

area-based operators use 2-D windows that span several scanlines, they implicitly impose a degree of coherence in disparity estimates for adjacent scanlines. Although the 2-D window represents additional cost when compared with single-pixel intensity comparisons, the robustness this affords, together with the simplicity obtained by avoiding 2-D coupling, suggest that area-based operators with zero or 1-D coupling may be adequate. Our scope does not permit investigating this question, so we leave the issue of 2-D coupling with this tentative conclusion and note that more work is necessary to determine its validity.

### 5.1.4 Summary

So far in this chapter, we have considered three different ways of formulating single-scale depth estimation algorithms for the bootstrap operation. These correspond to estimators that are completely uncoupled, coupled in one dimension, and coupled in two dimensions.

For the uncoupled, or "fully independent" case, we developed a Bayesian algorithm that estimates depth independently for each pixel, using measurements of intensity differences in windows around each pixel. This gives a simple, efficient matching that produces reliable depth maps for appropriate choices of baseline.

For the case of 1-D coupling, we began by proposing a new type of constraint on disparity derivatives by observing that derivatives measured from the wide-baseline image pair should be the same as those measured from the narrow-baseline image pair. This leads to a quadratic penalty function of differences in derivatives between the narrow and wide-baseline estimates. This is a direct extension of common approaches in the literature that penalize departures from zero gradient; however, in the bootstrap scenario the penalty is based on real knowledge of the scene at hand, not on universal heuristics about the nature of surfaces. In one dimension, the constraint leads to an objective function that is optimized efficiently by dynamic programming. Because the derivative constraint still has an *ad hoc* element, we derived a second algorithm that has a firmer statistical basis. This algorithm stems from the observation that disparity estimates from the narrow-baseline image pair are correlated. Modelling the correlation as exponential, we developed a joint Bayesian estimator for disparity along the entire scanline. The resulting objective function can be optimized by the same DP algorithm as the derivative-based approach.

Finally, we briefly examined 2-D coupling. One motivation for such coupling is the hope for better depth estimation by enforcing coherence between scanlines. However, this comes at the cost of a more difficult optimization problem. Moreover, area-based matching algorithms using 2-D windows achieve a degree of inter-line coherence without needing to couple the disparity estimates, simply because the windows span several scanlines. We conclude that, as a practical trade-off, 2-D coupling is probably unnecessary and undesirable, for reasons of cost. More research is necessary to verify this conclusion.

The algorithms developed in this section use error variance estimates computed from image $I_{l_1}$ as an interest operator in a novel sense: instead of matching only local maxima of "interestingness", as feature point methods and even edge-based stereo algorithms do, they set an interest threshold and match all pixels within the threshold. For these pixels, the error variance models the precision of the resulting disparity estimate. Pixels not within the threshold are left

unmatched; in practice, this can be flagged by an inverse variance of zero. Both the explicit model of depth uncertainty at the pixel level and the explicit distinction between matchable and unmatchable regions are useful conceptual advances. The former takes random field models of disparity a step closer to practicality by showing how to compute uncertainty at the pixel level. Moreover, most work to date with random field models has focussed exclusively on using the correlation in the field to enforce neighborhood consistency. For stereo, at least, this is partly missing the point of the model. The explicit distinction between matchable and unmatchable regions is, we believe, a small but useful clarification of the role of the lowest level representation of depth; that is, to represent only what has been measured from the images, instead of blurring the measurements with the effects of surface interpolation.

Finally, our approach to the whole bootstrap operation breaks with the common practice of using smoothness assumptions to constrain matching of binocular and trinocular images. Rather than regularizing an ill-posed optimization problem by applying universal heuristics about the nature of scenes, we use a conservative sensing action that can be expected to succeed, with simple algorithms, based on much weaker assumptions. The knowledge this gains about the scene at hand enables less conservative sensing actions to be taken — in our case, the wide-baseline stereo match. This is another articulation of the notion of active vision [Aloimonos87,Krotkov88].

## 5.2 Reasoning About Camera Motion

The previous section developed algorithms appropriate for a bootstrap operation employing lateral camera translation, as in figure 5.1. We implicitly assumed that lateral translation was the best camera motion to use in acquiring the narrow-baseline image pair and we did not specify the ratio between the narrow and wide baselines. In this section, we examine both the direction and the distance to move the camera in acquiring the narrow-baseline image pair. We begin with a sensitivity analysis that quantitatively compares the precision of depth estimates obtained by moving in the optimal direction (lateral translation) with the precision obtained by moving in a sub-optimal direction (forward along the camera axis). This relates directly to the constraint such depth estimates provide for matching the wide-baseline image pair. We then examine the question of how far to move the camera to obtain the narrow-baseline image pair. This involves issues of sensitivity, complexity, and reliability. For each issue, we derive criteria that can be used to guide the choice of narrow and wide baselines.

### 5.2.1 Direction to Move

In choosing the direction to move the cameras, we seek to optimize the precision of the depth information obtained from the narrow-baseline image pair. Intuitively, it is clear that the optimal motion is translation parallel to the stereo baseline. However, if forward motion is necessary for the robot to accomplish its task, there is reason to use that motion for the bootstrap operation in order to optimize the overall robot motion. We examine the effectiveness of such a strategy by using a sensitivity analysis to compare the precision obtainable from lateral and forward translation.

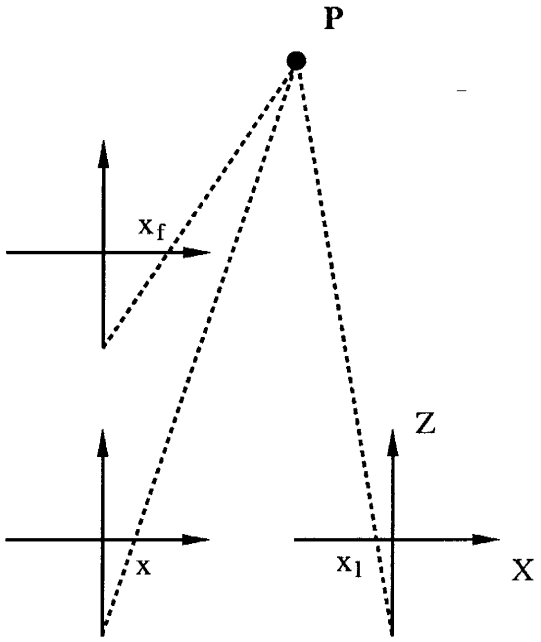



Figure 5.5: Sensitivity analysis for direction of motion: displacements

It is well known that rotating a camera about the focal point of the lens yields no depth information; furthermore, for translational motion the precision of depth estimates increases with increasing distance of image features from the *focus of expansion* (FOE), the point where the translation vector (**T**) pierces the image plane. This implies that the "best" translations are parallel to the image plane and the "worst" are forward along the camera axis. A lengthy examination of the effects of measurement uncertainty in depth from motion is given in [Snyder87]. Here we give a briefer analysis that demonstrates the relative precision obtainable from forward and lateral translation; that is, for $\mathbf{T} = [0\ 0\ T_z]^T$ and $\mathbf{T} = [T_x\ 0\ 0]^T$, respectively. For this comparison, it suffices to examine 1-D images.

For a lateral translation of distance $T_x$, the image coordinates of **P** in $I_{l_0}$ and $I_{l_1}$ are, respectively (see figure 5.5),

$$\begin{aligned} x &= s_x\frac{X}{Z} + c_x \\ x_l &= s_x\frac{X+T_x}{Z} + c_x. \end{aligned}$$

$s_x$ and $c_x$ are coefficients of the pinhole camera model described in appendix A. The displacement of the image feature between frames is

$$\Delta x_l = x - x_l = \frac{-s_x T_x}{Z},$$

which yields the following estimate of the inverse depth:

$$d_l = \frac{1}{Z} = \frac{-\Delta x_l}{s_x T_x}.$$

For forward motion, the projection of **P** in the second image is

$$x_f = s_x \frac{X}{Z + T_z} + c_x.$$

This yields an interframe displacement for forward motion of

$$\begin{aligned} \Delta x_f &= x - x_f \\ &= s_x X \left( \frac{1}{Z} - \frac{1}{Z + T_z} \right) \\ &= x \left( 1 - \frac{1}{1 + T_z/Z} \right) \\ &\approx \frac{x T_z}{Z} \quad \text{for small } T_z, \end{aligned}$$

where the last expression is obtained from a first-order expansion about $T_z = 0$. Therefore, the estimated inverse depth in the case of forward motion is

$$d_f = \frac{\Delta x_f}{x T_z}.$$

Perturbations of $\delta x_l$ and $\delta x_f$ in the displacement measurements $\Delta x_l$ and $\Delta x_f$ yield the following perturbations in the disparity estimates:

$$\begin{aligned} \delta d_l &= \frac{\delta x_l}{|s_x T_x|} \\ \delta d_f &= \frac{\delta x_f}{|x T_z|}. \end{aligned}$$

These equations give the error in the inverse depth as a function of the error in the measured image displacement, the amount of camera motion, and position of the feature in the field of view. Since we are interested in comparing forward and lateral motions, a good way to visualize these equations is to plot the relative depth uncertainty, $\delta d_f / \delta d_l$. Assuming that the displacement perturbations $\delta x_l$ and $\delta x_f$ are equal, the relative uncertainty is

$$\frac{\delta d_f}{\delta d_l} = \frac{\delta x_f / |x T_z|}{\delta x_l / |s_x T_x|} = \left| \frac{s_x T_x}{x T_z} \right|.$$

The image coordinate $x$ indicates where the object appears in the field of view. Since $x/s_x = X/Z$, this ratio equals the tangent of the angle $\theta$ between the object and the camera axis (figure 5.6). The formula for the relative uncertainty is therefore

$$\frac{\delta d_f}{\delta d_l} = \left| \frac{T_x}{T_z \tan\theta} \right|. \tag{5.24}$$

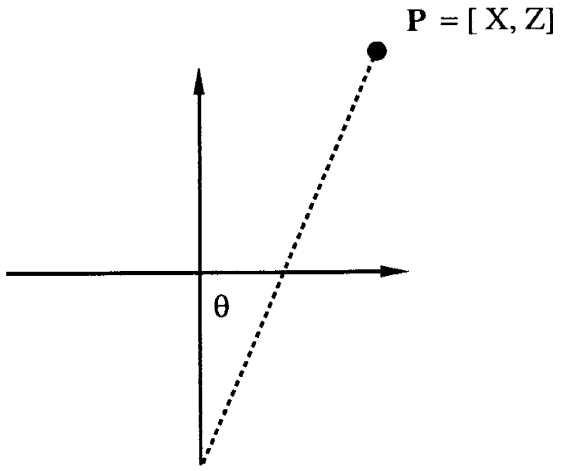



Figure 5.6: Angle between object and camera axis is $\theta$

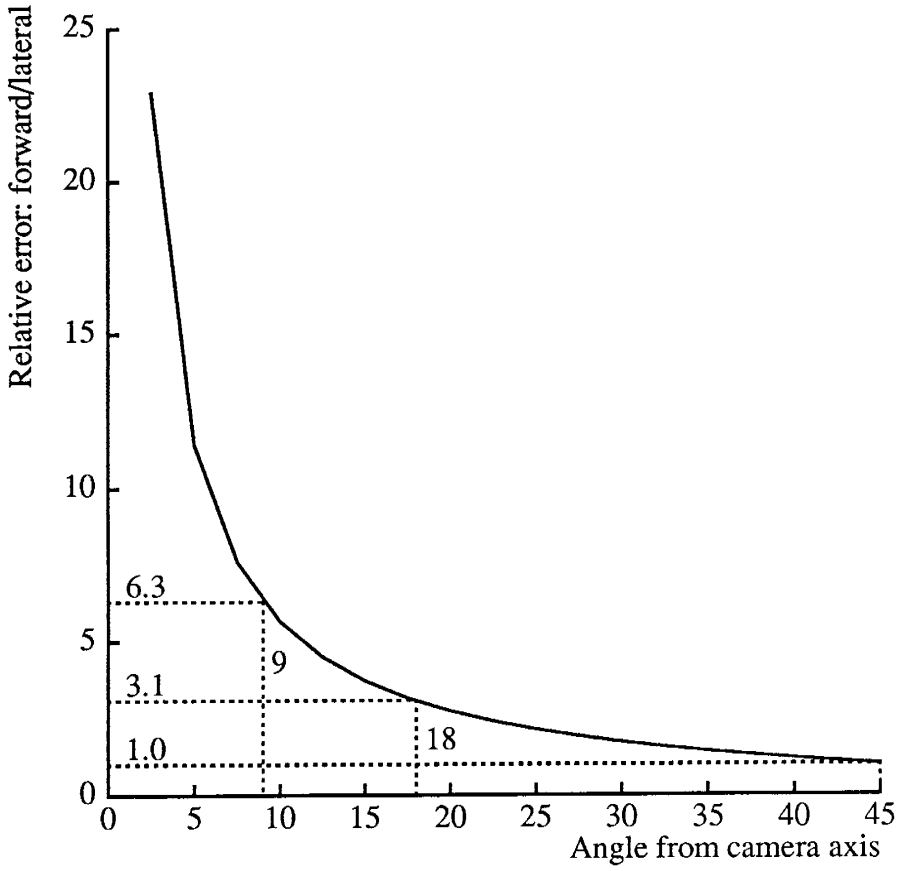



Figure 5.7: Relative depth uncertainty for forward vs. lateral translation

This relationship is plotted in Figure 5.7 for $T_x = T_z$. At 45 degrees from the camera axis, depth uncertainty is equal for forward and lateral motions. At 18 degrees, which is the edge of the image for the experiments in Section 5.3, the ratio of uncertainties is 3.1; at 9 degrees, the ratio is 6.3.

This analysis demonstrates the extreme difference in the precision obtained from equal translations in forward and lateral directions. Since inverse depth $(1/Z)$ is proportional to stereo disparity, the uncertainty in inverse depth relates directly to the constraint available to matching operations guided by use these depth estimates. Therefore, we conclude that the use forward motion to estimate depth, in the bootstrap operation or otherwise, is a very poor strategy.

### 5.2.2 Distance to Move

Having verified that translation parallel to the baseline is indeed the preferred motion, the next question is to determine how far the camera should move relative to the separation between the left and right cameras; that is, we need to choose the sizes of the narrow baseline $B_n$ and the wide baseline $B_w$. In this section, we derive criteria for this choice based on issues of sensitivity, computational complexity, and probability of matching error. It suffices to conduct the analyses for 1-D images.

#### Sensitivity Analysis

The purpose of the narrow-baseline image pair is to constrain matching in the wide-baseline pair. Since depth precision increases with baseline, this implies that better constraint is obtained by increasing $B_n$. However, increasing $B_n$ increases the likelihood of matching errors with the narrow-baseline images. Therefore, we may wish to keep $B_n$ as small as possible, subject to maintaining a given degree of constraint in the wide-baseline images. If search windows for the wide-baseline images are determined by error tolerances applied to the narrow-baseline disparity estimates, then a simple sensitivity analysis can be used to set the baseline ratio $B_w/B_n$.

For the pinhole camera model used here, the perspective projections of a point at $(X, Z)$ onto the left $(I_{l_1})$ and right $(I_r)$ images of the wide-baseline image pair are

$$\begin{aligned} x_{l_1} &= s_x \frac{X}{Z} + c_x \\ x_r &= s_x \frac{X - B_w}{Z} + c_x \\ &= x_{l_1} - s_x B_w d. \end{aligned} \tag{5.25}$$

In the bootstrap scenario, the inverse depth $d$ is determined by matching in images $I_{l_0}$ and $I_{l_1}$ and is used to predict $x_r$ as above. Thus, for a given uncertainty $\delta d$ in $d$, the uncertainty in the predicted position $x_r$ grows linearly with the stereo baseline $B_w$. For the system design being considered here, $B_w$ is assumed to be fixed by some external consideration, while the narrow baseline is specified relative to $B_w$. If we set a maximum search interval size in the right image of $\delta x_{w_{max}}$, we can reason backwards to determine the relative sizes of $B_n$ and $B_w$ that will guarantee

that this bound is met. From the narrow-baseline images, the inverse depth estimate is

$$d = \frac{x_{l_0} - x_{l_1}}{s_x B_n},$$

so the error in $d$ is

$$\delta d = \frac{\delta x_n}{s_x B_n},$$

where $\delta x_n$ is the error in the disparity estimate $x_{l_0} - x_{l_1}$. If we attempt to match only those pixels with disparity error less than some maximum $\delta x_{n_{max}}$, then the smallest $B_n$ that guarantees that search windows are bounded by $\delta x_{w_{max}}$ is

$$B_n = \frac{\delta x_{n_{max}} B_w}{\delta x_{w_{max}}}.$$

In other words, the baseline ratio should be

$$\frac{B_w}{B_n} \leq \frac{\delta x_{w_{max}}}{\delta x_{n_{max}}}.$$

To apply this formula, we need to specify $\delta x_{n_{max}}$ and $\delta x_{w_{max}}$. The sub-pixel matching results in chapter 4 suggest that a precision ($\delta x_{n_{max}}$) of 0.25 pixels or less is readily achievable. The size of $\delta x_{w_{max}}$ is harder to specify rationally. If we fix this at $\pm$ five pixels as an example, we obtain

$$B_n \geq \frac{B_w}{20}.$$

That is, the narrow baseline must be at least 5% of the wide baseline.

### Computational Complexity Analysis

Stereo matching must be performed twice in the bootstrap operation, once for the narrow-baseline image pair and once for the wide-baseline image pair. Since the narrow-baseline pair provides constraint for the wide-baseline pair, the total image area searched for the bootstrap operation may be smaller than the area searched if only the wide-baseline image pair is used. This implies that the bootstrap operation may also be less expensive than matching the wide-baseline pair alone. Furthermore, the baseline ratio can be chosen to minimize the total area searched in both image pairs. To demonstrate these facts, suppose that all objects in the scene are known *a priori* to lie in the distance range $[Z_n, Z_f]$. If matching is done only with the wide-baseline images, then for each pixel the image range $[x_n, x_f]$ corresponding to the entire $Z$ range must be searched. From (5.25), this search range is

$$\begin{aligned} \Delta x_{nc} &= x_f - x_n \\ &= x_{l_1} - s_x B_w / Z_f - (x_{l_1} - s_x B_w / Z_n) \\ &= s_x B_w \left( \frac{1}{Z_n} - \frac{1}{Z_f} \right) \\ &= s_x B_w \Omega, \end{aligned} \tag{5.26}$$

where $\Omega$ denotes the range of inverse depths. We see that the area to be searched grows linearly with the baseline.

If we use the bootstrap operation, then the area to be searched in the narrow-baseline image pair is

$$\Delta x_n = s_x B_n \Omega.$$

If this provides initial depth estimates $\hat{d}^-$ with precision of $\pm \delta d$, then from the previous section the range of disparities to be searched in the wide-baseline image pair is

$$\begin{aligned} \Delta x_w &= x_{l_1} + s_x B_w (\hat{d}^- + \delta d) - [x_{l_1} + s_x B_w (\hat{d}^- - \delta d)] \\ &= 2 s_x B_w \delta d \\ &= 2 \delta x \frac{B_w}{B_n}. \end{aligned}$$

Therefore, the total area searched with constraint from the bootstrap operation is

$$\Delta x_c = \Delta x_n + \Delta x_w = s_x B_n \Omega + 2 \delta x \frac{B_w}{B_n}. \tag{5.27}$$

Comparing equations (5.27) and (5.26) reveals the difference in search area between using versus not using the bootstrap operation. For the experiments conducted at the end of this chapter, $B_w = 0.0254\text{m}$, $\Omega \approx 2.0\text{m}^{-1}$, and $s_x \approx 500$ pixels. The cameras had 30 degree fields of view and the images were processed at $256 \times 240$ resolution. Using the narrow baseline value of $B_n = B_w/20$ and the disparity precision of $\delta x = 0.25$ pixels, as in the previous sensitivity analysis, yields search areas of 25.4 pixels for the wide-baseline only case and 11.3 pixels in the bootstrap case. Thus, there is a reduction of 55% of the search area in this case.

Equation (5.27) can also be viewed as a parameterized function of $B_n$ and used to derive an optimal baseline ratio in the sense of minimizing the total search area. Differentiating (5.27) with respect to $B_n$, setting the result to zero, and solving for $B_n$, we obtain

$$B_n = \sqrt{\frac{2 \delta x B_w}{s_x \Omega}}.$$

Using the same values for the constants $s_x$, $B_w$, $\Omega$, and $\delta x$ as above yields $B_n = 0.00356\text{m}$, or $B_w/B_n = 7.1$. In this case, the narrow baseline is over twice as long as the example we gave during the sensitivity analysis; therefore, it will led to a smaller search area in the wide-baseline image pair. Plugging this value of $B_n$ back into (5.27), the optimal total search area is 7.1 pixels. Thus, the complexity-based analysis has lead to a choice of baseline ratio that provides more constraint for less total search area than the choice arrived at by the sensitivity analysis above. Since computational cost is a function of search area, the cost of matching with the bootstrap operation also may be less than the cost without it.

### Prior Ambiguity Analysis

Given image $I_{l_0}$ and an *a priori* distance range $[Z_n, Z_f]$, it is possible to reason about the level of ambiguity that will be encountered in matching $I_{l_0}$ to $I_{l_1}$. Moreover, the level of ambiguity

will be a function of the size of the baseline $B_n$. Therefore, we can also decide to choose $B_n$ to satisfy a criterion derived from an ambiguity analysis.

Figure 5.8 shows how an ambiguity analysis can be done. Suppose the first image ($I_{l_0}$) contains two similar "features" at image coordinates $x_1$ and $x_2$. Given the prior distance range, we can compute search ranges for both features in image $I_{l_1}$ as shown in figure 5.8a. That is, the minimum and maximum possible depths $Z_n$ and $Z_f$ determine the endpoints of the search intervals for each feature. If the intervals do not overlap, there is no ambiguity in matching these two features; we simply scan the disjoint search intervals to find the position of best match for each feature. However, if the intervals overlap, ambiguity exists because both features might appear in the region of overlap; this would force a decision about the order in which the features appear (figure 5.8b). Since the amount of overlap grows as a function of $B_n$, the "degree" of ambiguity, as measured by the amount of overlap, also grows as a function of $B_n$. The most precise depth estimates obtainable with no ambiguity are obtained with the value of $B_n$ that makes the intervals abut.

This value of $B_n$ can be computed easily by supposing that the feature at $x_2$ corresponds to world point $\mathbf{P}_2$, with distance $Z_n$, supposing that the feature at $x_1$ corresponds to world point $\mathbf{P}_1$, with distance $Z_f$, and finding $B_n$ such that $\mathbf{P}_1$ and $\mathbf{P}_2$ project to the same pixel in the new image. We will carry out this analysis using the pinhole camera model with $s_x = 1$ and $c_x = 0$. In this case, the $X$ coordinates of $\mathbf{P}_1$ and $\mathbf{P}_2$ are

$$X_1 = x_1 Z_f, \quad X_2 = x_2 Z_n. \tag{5.28}$$

Projecting to the same pixel implies that

$$\frac{X_1 - B_n}{Z_1} = \frac{X_2 - B_n}{Z_2}$$

or

$$\frac{x_1 Z_f - B_n}{Z_f} = \frac{x_2 Z_n - B_n}{Z_n}.$$

Solving for $B_n$ yields

$$\begin{aligned} B_n &= \frac{x_2 - x_1}{1/Z_n - 1/Z_f} \\ &= \frac{\Delta x}{1/Z_n - 1/Z_f}, \end{aligned} \tag{5.29}$$

where $\Delta x$ is the distance between the two features. If $Z_f = \infty$, the result is the very simple upper bound

$$B_n = Z_n \Delta x$$

on how far the camera can move without incurring an ambiguous match.

To see how this can be used in a real system, first consider matching discrete features, such as edge tokens, instead of the dense correlation methods we have employed. In this case, the image coordinates $x_1$ and $x_2$ represent the locations of edges in the first image. Overlap in the

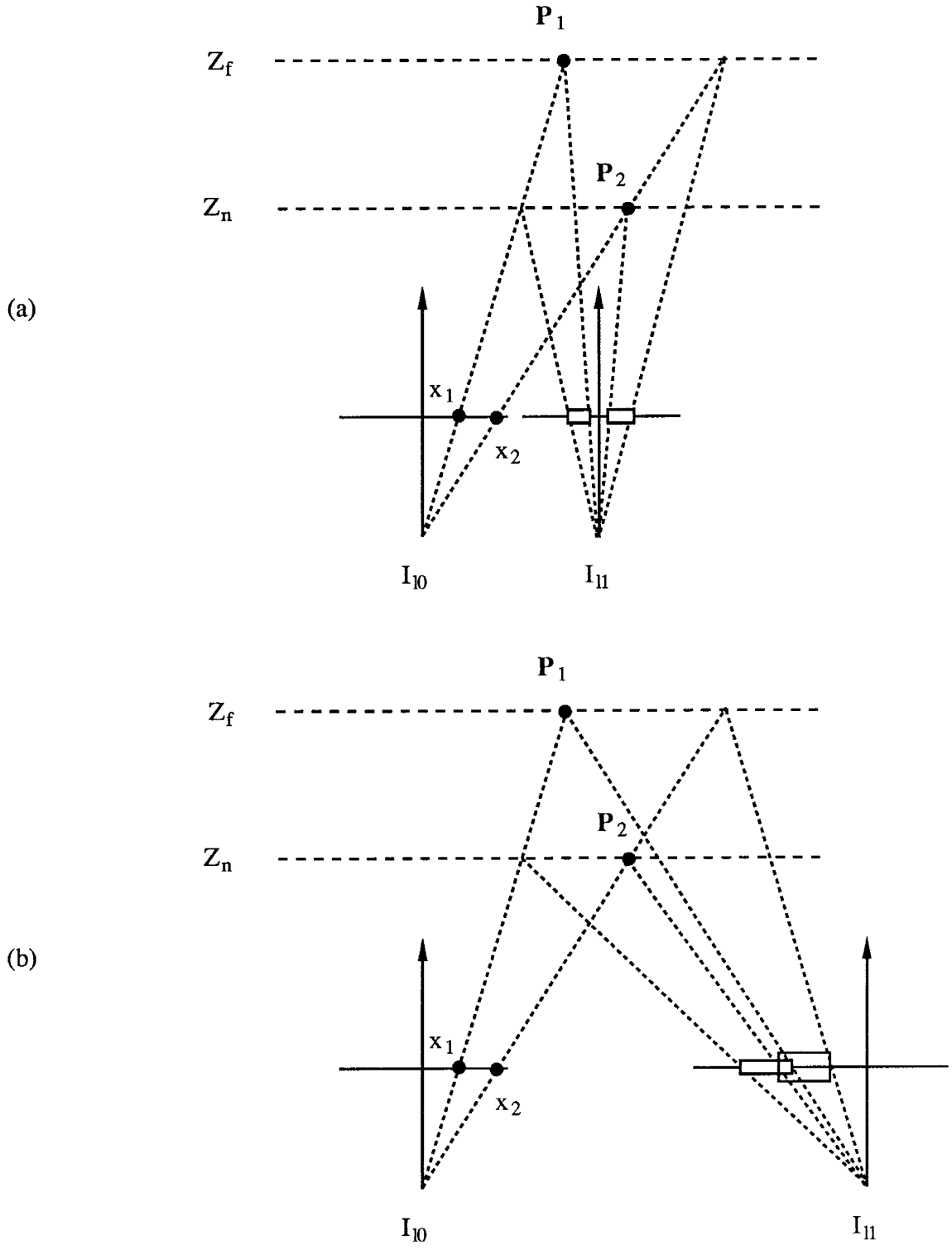



Figure 5.8: Ambiguity analysis: (a) distance range projects onto $I_{l_1}$ such that search intervals do not overlap — no ambiguity; (b) search intervals do overlap — possible ambigous match

search intervals implies that the edges may appear in reversed order in the second image or that one edge may be occluded by the surface containing the other. Both of these circumstances cause difficulty in the matching process and therefore introduce possibilities for erroneous depth estimation. Using the "conservative" baseline defined above allows depth information to be obtained without incurring the possibility of this kind of error. To avoid the possibility of overlap anywhere in the image, we would search the image to find the smallest interval $\Delta x$ between adjacent edges within a single scanline, then use (5.29) to obtain $B_n$. The analogous approach to dense depth estimation is to obtain $\Delta x$ by autocorrelation; that is, we correlate regions of $I_{l_0}$ against the same image to find the nearest signficant autocorrelation peak within the same scanline. The distance to this peak becomes $\Delta x$ and is used to choose the baseline as above.

Thus, given the prior distance range $[Z_n, Z_f]$ and the smallest displacement between features in the image, we have derived the largest baseline that avoids overlap in the search intervals for known features. This use of existing scene knowledge to plan the next sensing action has analogs in other areas of computer vision. There is much more potential to exploit this notion within the stereo problem.

### 5.2.3 Summary

In this section, we examined issues concerning the direction and distance to move the cameras in acquiring the narrow-baseline image pair. For direction, it is intuitively clear that lateral translation (parallel to the stereo baseline) must be optimal, but one still wonders how good other directions can be; for example, since the robot will often be moving forward, can we use depth from forward motion effectively? To find out, we used a sensitivity analysis to show quantitatively the relative precision of disparity estimates obtained from equal translations forward and laterally. The results showed a significant advantage for lateral translation. We concluded that forward translation is very unlikely to be useful. This is particularly true in the autonomous navigation scenario, since uncertainties in the vehicle motion will be much higher than those achievable with an on-board translation stage.

For distance to move, we briefly examined three relevant issues concerning sensitivity, computational cost, and matching ambiguity. The sensitivity analysis derived a relationship between match precision, desired size of the wide-baseline search windows, and the baseline ratio. This relationship can be used in choosing the baseline ratios. The computational cost analysis compares the total search area experienced with and without the use of the bootstrap operation. We found that the bootstrap operation can require a significantly smaller search area than would be needed to match the wide-baseline image pair without the bootstrap operation. Whether this translates into less costly matching remains to be seen. Finally, the ambiguity analysis showed that it is possible to use the first image acquired, together with assumptions about the range of disparities present in the scene, to reason about the possibility of match ambiguity as a function of baseline ratio. Such reasoning will be important in reducing the error rates of future systems.

## 5.3 Evaluation

There are many issues to examine concerning the validity of the statistical models employed in this chapter, as well as the relative performance the algorithms for various parameter settings. Here, we will only show the qualitative performance of the bootstrap operation in contrast with results obtained on the same images without the bootstrap operation. The data for these experiments was obtained from the Calibrated Imaging Lab at CMU and consists of scale models of complex, outdoor scenes. For experimental convenience, all images were obtained by translation of a single camera. Thus, while our main goal is the development of a bootstrap module for binocular stereo *per se*, the experimental results to follow also demonstrate what can be achieved in a single-camera, narrow/wide-baseline scenario.

The camera for these experiments was a Sony XC-37 CCD with a 16mm lens. For both data sets, visible objects in the scene ranged in depth from approximately 20 to 55 inches from the camera. Images were acquired at $480 \times 512$ resolution and reduced to $240 \times 256$ by Gaussian filtering and subsampling. For the first data set, the baselines for the narrow and wide image pairs were 0.15 and 1.0 inches respectively. This gave a disparity range for the scene of 9 to 25 pixels, or between 5 and 10 percent of the image width. The 10 percent maximum disparity is comparable to the maximum disparity experienced by the motion estimation system described in chapter 2.

Figure 5.9 shows a full-page rendition of one image from the first data set. The scene has train-tracks in the foreground, several buildings in the middle ground, a bridge and a hill with trees beyond the buildings, and a calibration grid as a backdrop. Features of the scene that make it difficult to process include the many repeated texture patterns, large depth discontuities, some very fine three-dimensional structure in the form of antennas on two of the buildings, and strong highlights on the train tracks. The row of buildings on the right side of the scene has a steeply receding wall, which normally would be expected to cause difficulty in area-based approaches to matching[2].

Figure 5.10 shows both images of the wide-baseline stereo pair (i.e. images $I_{l_1}$ and $I_r$), an oblique view of the scene, and a floorplan sketch showing the layout of the main buildings. The oblique view and floorplan are intended to illustrate the 3-D structure of the scene for comparison gray-scale depth maps to be shown shortly.

The results of using the ML error variance as an interest operator are shown in figure 5.11. The black areas in the image on the right were deemed too featureless to match. This excludes most of the large, blank areas, but leaves a significant percentage of the image area to match — much more than one would expect with an edge-based approach.

Figure 5.12 shows depth maps produced with and without the bootstrap operation. Black areas were not matched; this includes the areas filtered away by the interest operator and bands on the sides that are not visible in all images. Figure 5.12b shows the depth map obtained

[2]Sampling artifacts also produced moire patterns on the roof of the building in the center of the image. This causes matching problems that are beyond the scope of our methods to deal with. However, it prompted us to defocus the lens somewhat for the second set of data.

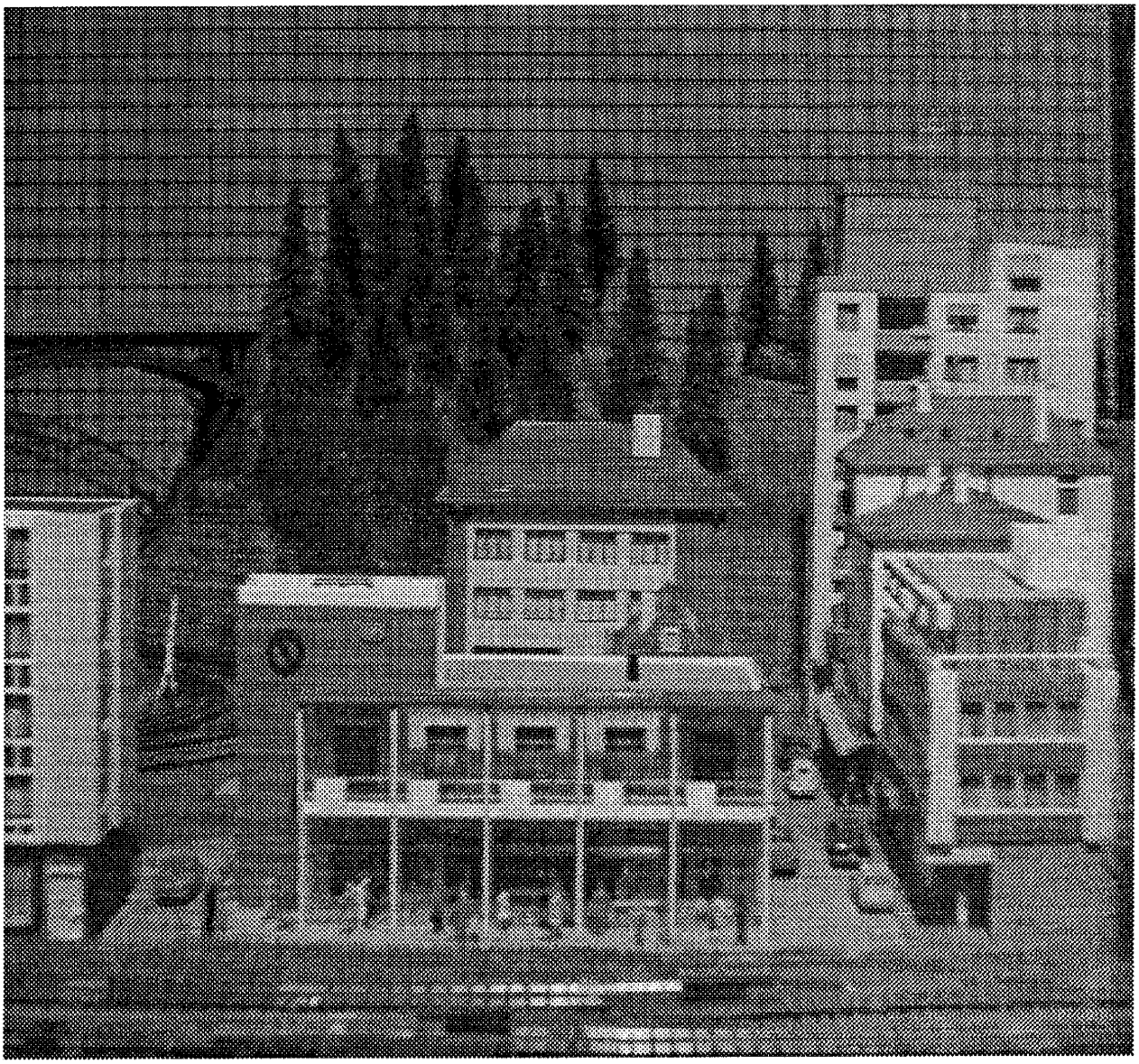


Figure 5.9: Image from "CIL 1" data set

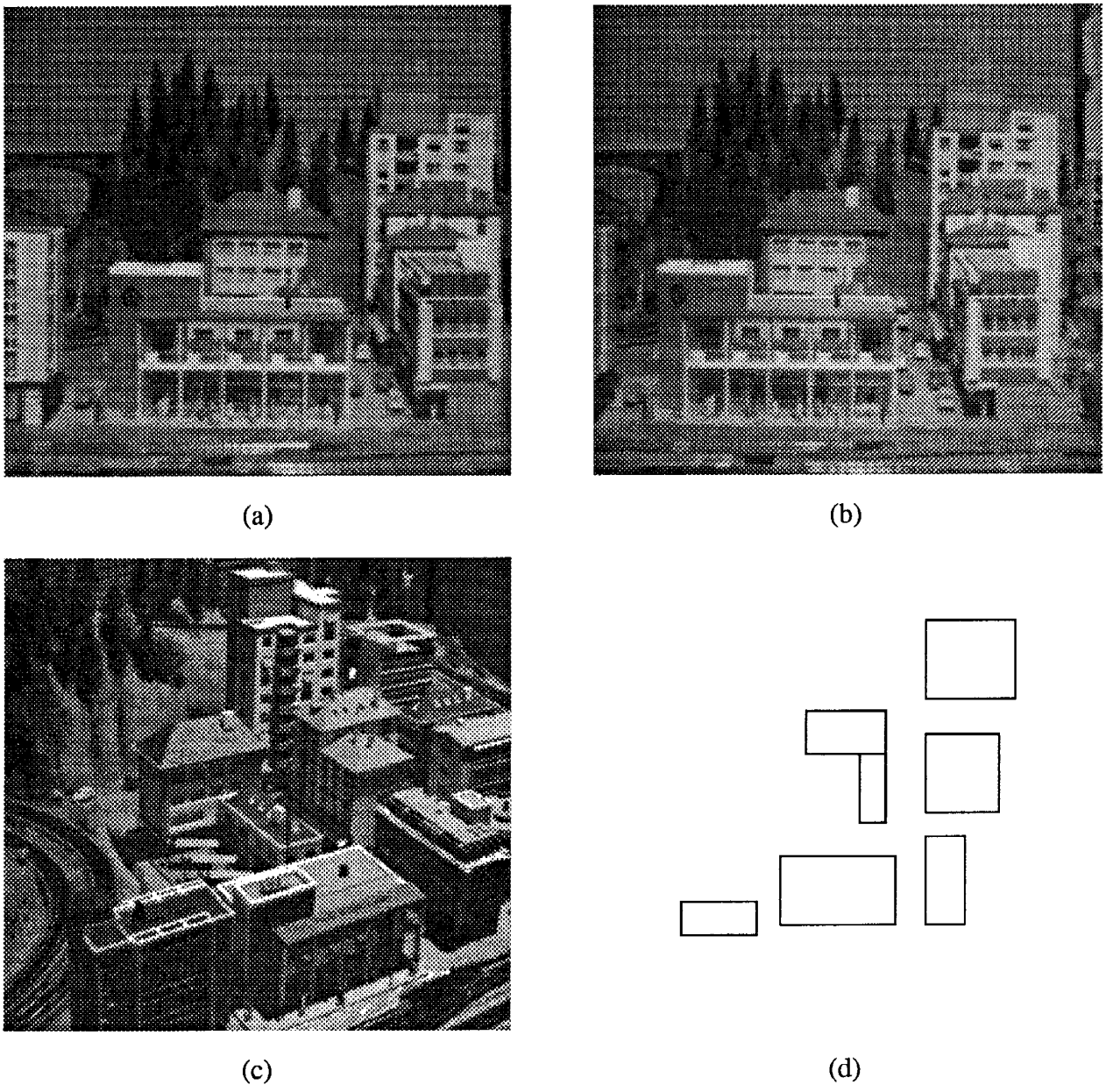


Figure 5.10: Views of CIL 1 data set: (a) left image (b) right image (c) oblique view (d) floorplan of main buildings

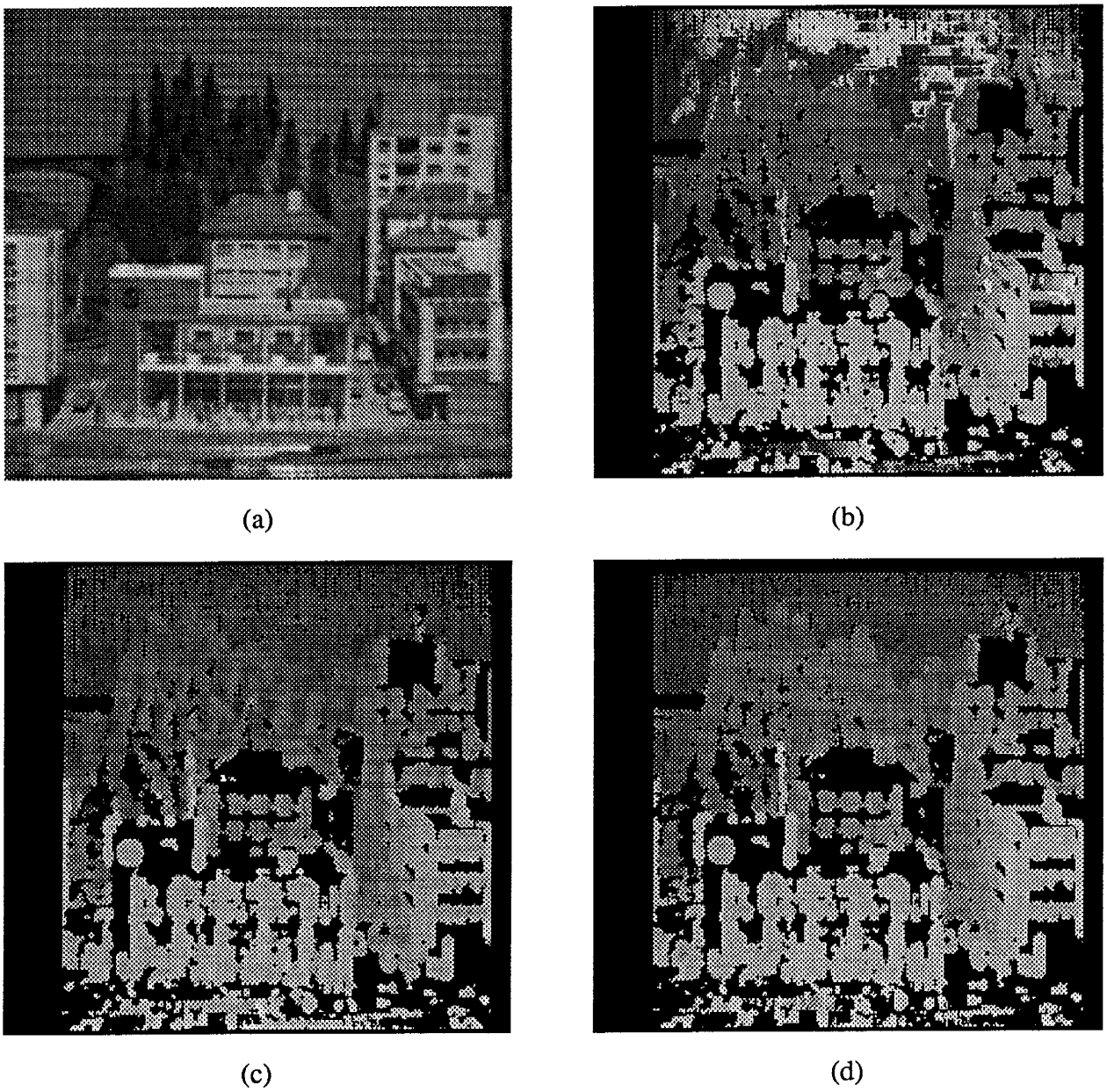
(a) (b)

Figure 5.11: Interest operator result for CIL 1 data set: (a) left image (b) left image masked with interest operator result. The black areas are considered unmatchable.

from the wide-baseline image pair without use of the bootstrap operation. In this case, disparity estimated were obtained simply by using the ML matcher with a $5 \times 5$ window to search slightly more than the full disparity range present in the image; that is, 6 to 28 pixels. Sub-pixel disparity estimates were computed for the position of best match. Many errors occur in areas of repetitive texture, near depth discontinuities, and in the vicinity of the highlight on the train tracks.

Figures 5.12c and 5.12d show depth maps for the narrow and wide-baseline image pairs, respectively, obtained with the bootstrap operation. The narrow-baseline result was obtained using only the ML matcher. The wide-baseline result was obtained with the joint 1-D algorithm with $\rho = 0.9$. Matching for the wide-baseline image pair used windows centered on the predicted disparity and ranging $\pm 0.5$ pixels times the baseline ratio to either side; thus, the total width of the search window was eight pixels. Both depth maps are significantly better than figure 5.12b. The errors that remain occur primarily at depth discontinuities, among the highlights on the train track, and on the building roof that produced moire patterns. For the most part, the same errors are present in the narrow and wide-baseline results. The narrow-baseline depth map does show more depth variation due to the lower signal-to-noise ratio available with the smaller baseline. Finally, the algorithm has performed well on the steeply receding wall on the right side of the scene.

Figure 5.13 shows a full-page rendition of an image from the second data set. The baselines for this set were 0.1 and 0.4 inches, giving a disparity range in the $240 \times 256$ images of 3 to 12 pixels, or a maximum disparity of about 5 percent of the image width. The difficulty of this scene is comparable to the previous scene. Here we have defocussed the lens somewhat; aliasing artifacts are reduced, but the image shows blur.



(a) (b)

(c) (d)

Figure 5.12: Depth maps from CIL 1 data set: (a) intensity image (b) wide-baseline depth map obtained without prior information (c) narrow-baseline depth map (d) wide-baseline depth map obtained with prior information

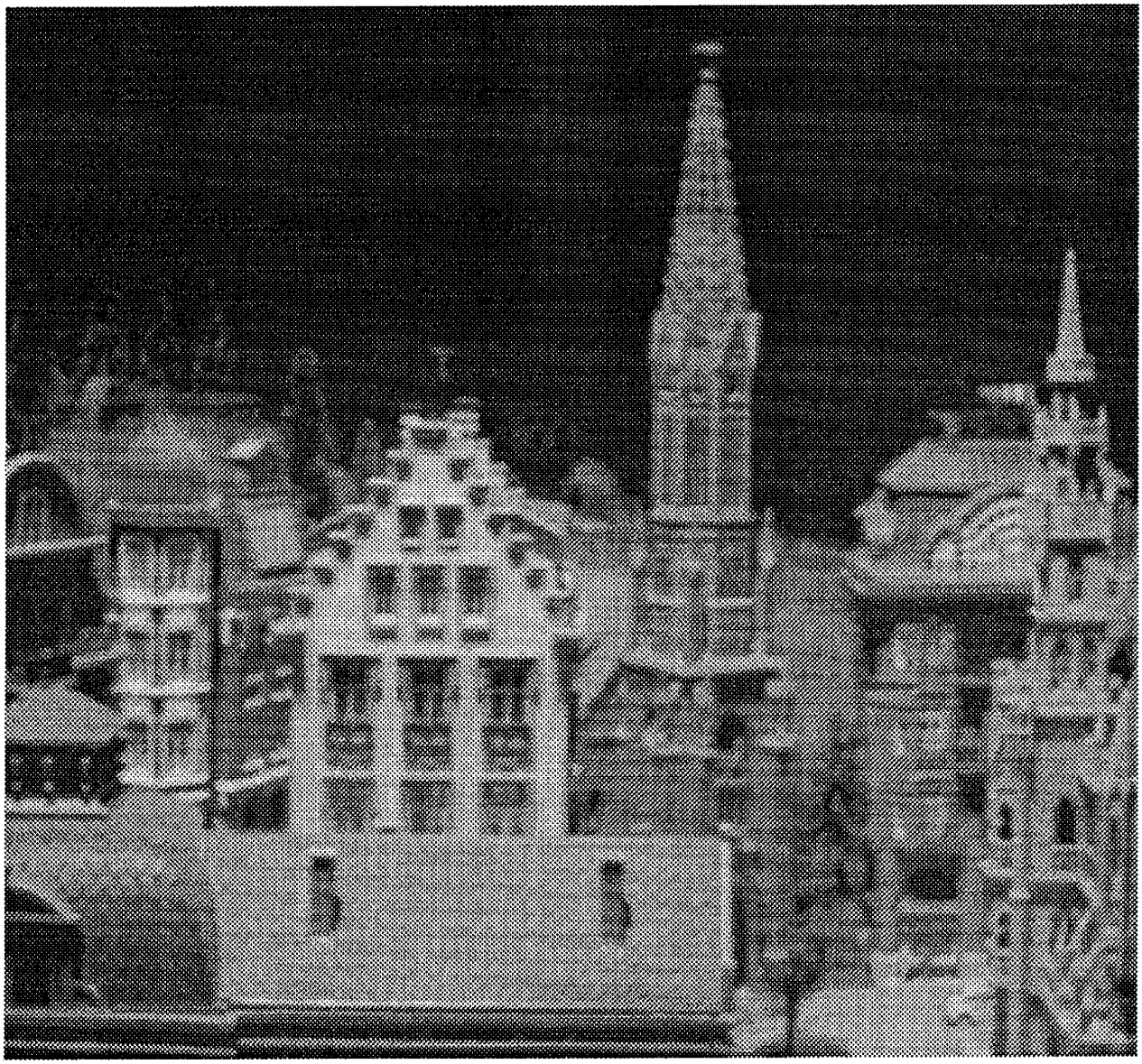
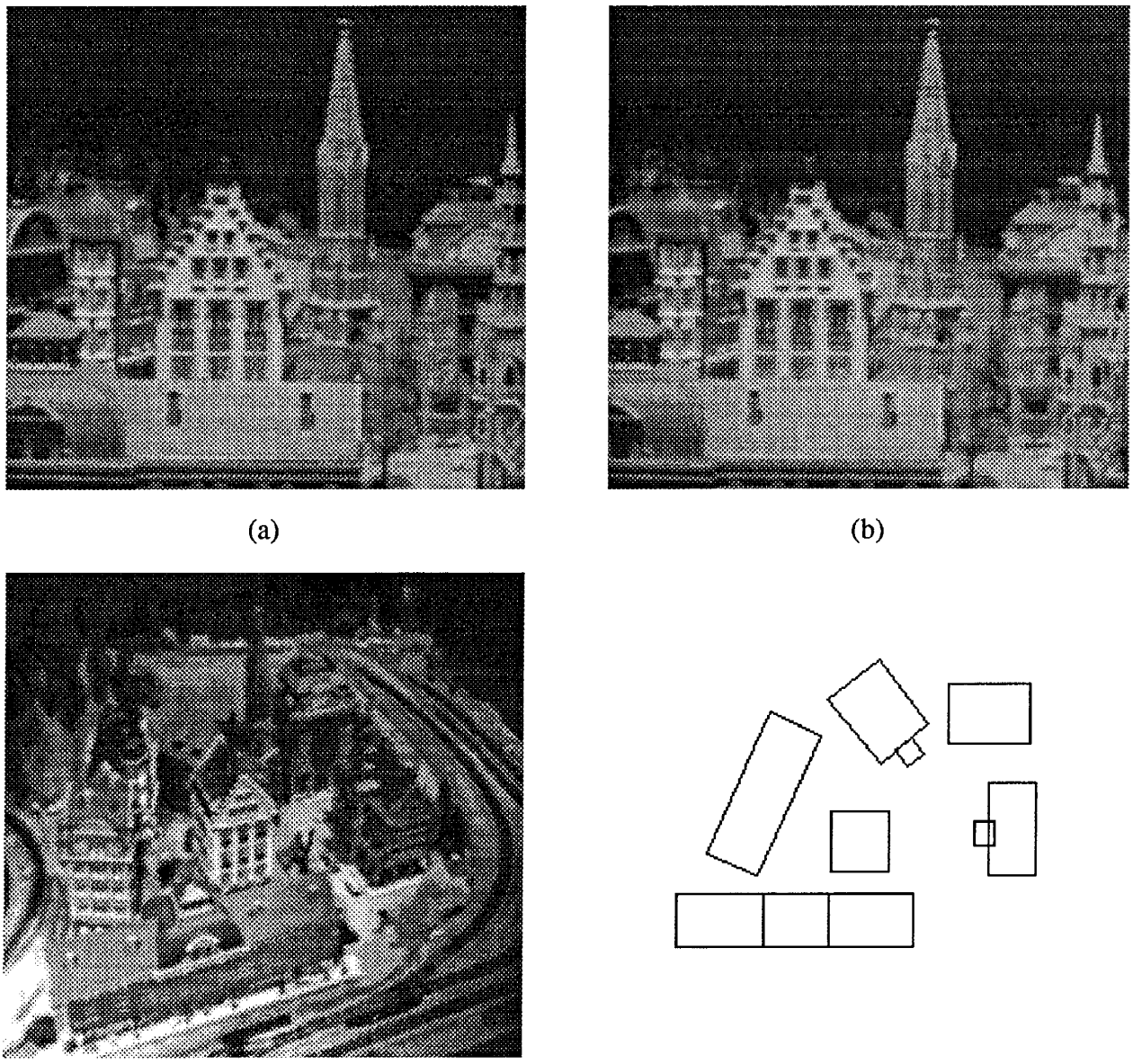


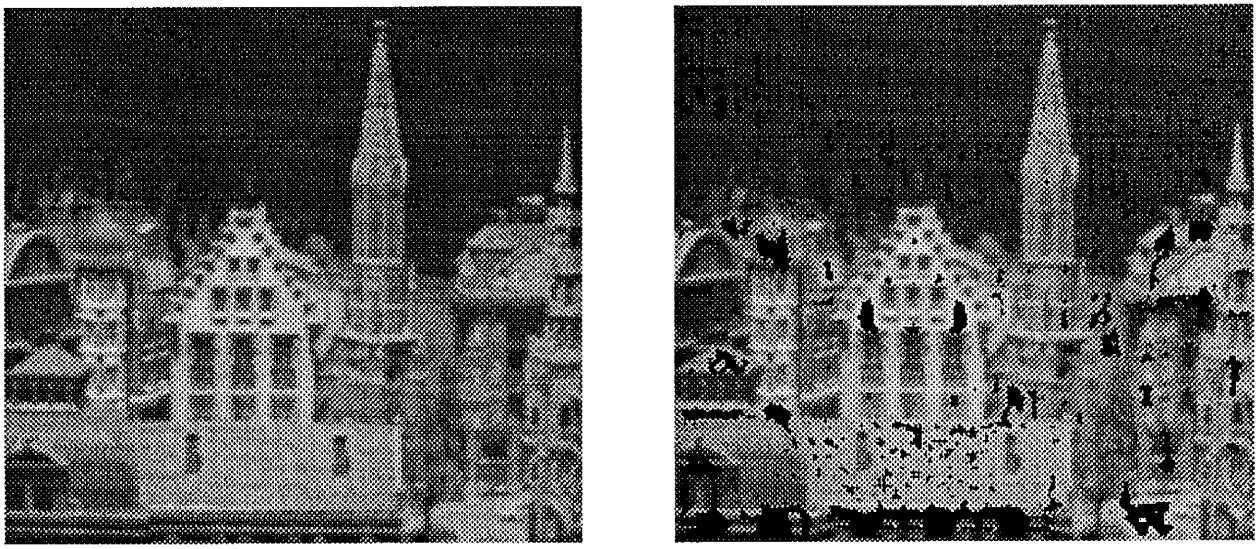
Figure 5.13: Image from "CIL 2" data set



(a) (b)

(c) (d)

Figure 5.14: Views of CIL 2 data set: (a) left image (b) right image (c) oblique view (d) floorplan of main buildings



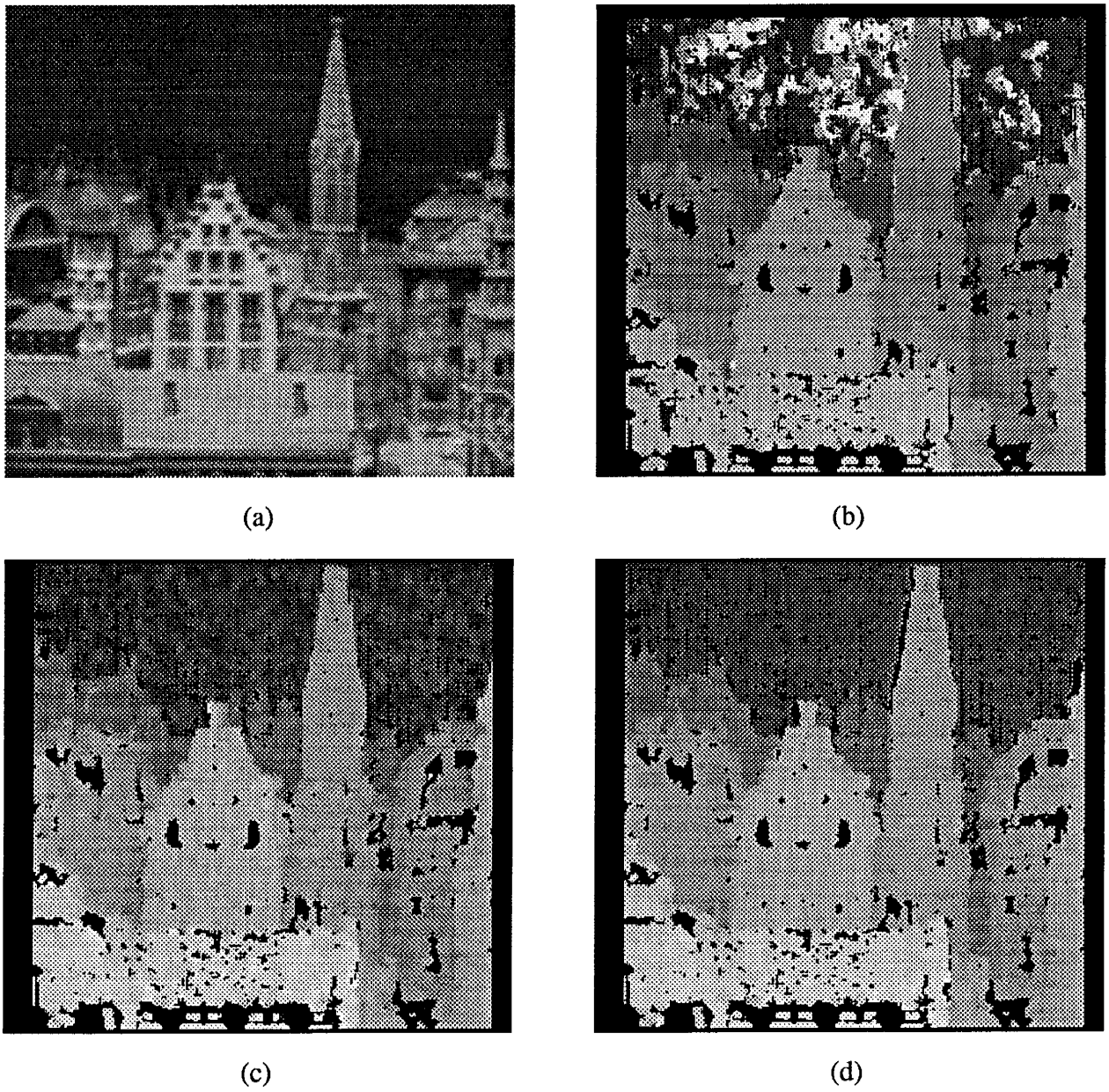
(a) (b)

Figure 5.15: Interest operator result for CIL 2 data set: (a) left image (b) left image masked with interest operator result

The wide-baseline stereo pair, an oblique view, an the floorplan for this data set are shown in figure 5.14. Interest operator results are shown in figure 5.15. Note that much of the roof in the lower foreground is considered matchable, even though strong texture is not apparent. The area around the eves of this roof is not considered matchable; although a strong intensity gradient is present, it is oriented vertically.

Depth maps for the narrow and wide-baseline image pairs, with and without the bootstrap operation are shown in figure 5.16. The results have the same character as before; the results obtained with the narrow/wide baseline combination are quite good, whereas those obtained without the constraint afforded by the narrow-baseline image pair show errors. Since this depth map is fuller than the CIL 1 data set, it is feasible to interpolate it to better visualize the results. Figure 5.17a shows an interpolated version of the wide-baseline depth map of figure 5.12d. All of the main structural units of the scene stand out well. As an experiment, we performed a very simple segmentation of the depth map by thresholding the angle of incidence between the line of sight through each pixel and the local surface normal. Figures 5.17c and 5.17d show the intensity image and the depth map with the removed pixels in black. The major units of the scene are quite well separated from each other, including the buildings, the hills and trees behind the buildings, and calibration grid in the background.

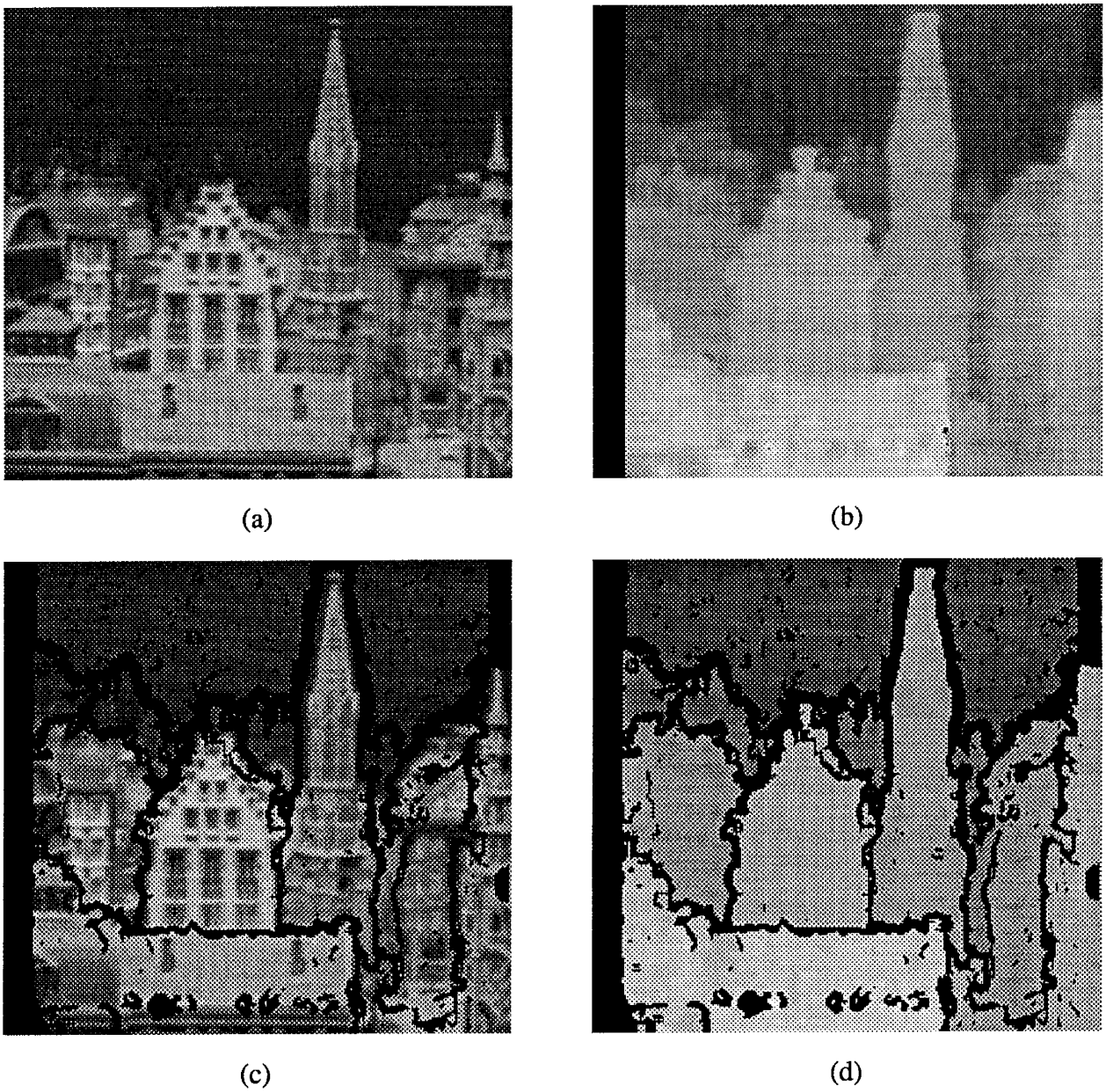
The results with these two data sets demonstrate that the combination of techniques we have employed — the bootstrap operation, area-based matching, and the error variance-based interest operator — succeed at producing very good depth maps for a difficult scene. We have not performed quantitative experiments evaluating the relative effectiveness of the different algorithms described in this chapter. However, our experience is that the joint 1-D algorithm



(a) (b)

(c) (d)

Figure 5.16: Depth maps from CIL 2 data set: (a) intensity image (b) wide-baseline depth map obtained without prior information (c) narrow-baseline depth map (d) wide-baseline depth map obtained with prior information



(a) (b)

(c) (d)

Figure 5.17: Segmentation results for CIL 2 data set: (a) intensity image (b) interpolated depth map (c) segmented intensity image (d) segmented depth map

produces slightly fewer errors than the fully independent algorithm. The baselines for these data sets were chosen by hand. The effectiveness of automatic methods for baseline determination remain to be explored.

The question of how the new algorithms compare to existing, binocular methods is beyond our scope.

## 5.4 Extensions and Related Work

Depth map estimation is a complex problem, so the work here leaves many issues unaddressed. This section examines some of the most signficant ones. The discussion progresses from issues involved in putting the algorithms developed here into practice, to larger issues that represent extensions within the scope of the bootstrap operation, and finally to the question of depth estimation over time — for example, the steady-state phase of chapter 3.

Concerning the specific image models and matching algorithms developed here, at the core of our approach is a purely additive, Gaussian white noise model that justifies the use of intensity differences as the basis for image matching. This model is useful as a first step in deriving the uncertainty models and disparity estimators of chapters 4 and 5; however, it is certainly not a perfect model of noise in real images. In particular, there is a need to examine the degree of bias and gain fluctuation present and to model or filter the images accordingly. Similarly, the experiments in chapter 4 to validate the uncertainty model need to be extended for real images under various lighting, reflectance, and surface orientation conditions. The matching experiments in this chapter used images acquired by motion of a single camera. This was a great convenience experimentally, but it side-stepped several pragmatic issues involved in calibrating stereo cameras. In particular, algorithms that attempt to estimate dense depth maps are more sensitive to epipolar misaligment than approaches using feature-points or line segments. Thorough calibration is one answer to this problem. Another is that this sensitivity can be reduced by extending the interest operator to ensure that the local intensity gradient is not close to vertical. Lastly, we have not quantitatively evaluated the impact of coupling in the joint 1-D algorithms; doing so would be valuable. A reasonable start can be made by considering only scenes in which there are no depth discontinuities.

Larger issues within the scope of the bootstrap operation include automating the choice of baselines, coping with occlusions, and multi-scale matching. We have already considered possibilities for the baseline issue in section 5.2. The occlusion and multi-scale matching issues are both complex. Several existing dynamic programming algorithms incorporate methods for detecting occlusions [Baker82,Ohta85,Sankoff83]; whether or not such methods are adequate is unclear. One interesting possibility is to use the narrow-baseline results to predict occlusions in the wide-baseline images; this idea has already been exploited in work with fine-motion image sequences [Bolles87]. Regarding multi-scale matching, most area-based approaches to multiple scales use gradient descent or local search to progress from coarse to fine resolution [Anandan84,Moravec80,Quam84,Witkin87]. A natural step would be to use the methods of this chapter to estimate depth at a low resolution, then to use existing methods to proceed to high resolution.

In the long term, integrating motion-based bootstrapping with orthogonal, trinocular camera configurations [Hansen88,Milenkovic85,Stewart88] is attractive. This would provide verification of binocular match hypotheses and would allow depth to be measured at locations of horizontal or vertical intensity gradients. Except for calibration issues, doing so should be straightforward for completely uncoupled matching. Beyond this, there are too many ways to structure trinocular objective functions and optimization algorithms to review the possibilities here.

Finally, two extensions beyond the scope of the bootstrap operation are sequential depth estimation and depth estimation from stereo image sequences. Regarding sequential estimation, modelling depth uncertainty at the pixel level introduces the possibility of updating depth estimates at the pixel level, given multiple depth measurements from an image sequence. Approaches to this, for both random field and feature-based representations, are discussed in [Baker88,Heel89,Matthies89,Szeliski88]. For depth estimation from stereo image sequences, we conjecture that the most effective approach will be one foreshadowed in chapter 3; that is, acquiring images very rapidly and tracking disparity changes with gradient descent (or very local search) in an image pyramid.

## 5.5 Summary

In this chapter, we developed depth map estimation algorithms for the stereo bootstrap operation proposed in chapter 3. This operation uses motion of one camera to obtain a narrow-baseline image pair. It then uses depth estimates from the narrow-baseline image pair to constrain matching for a wide-baseline image pair acquired with both cameras of the stereo system.

Our approach was to formulate the bootstrap operation as a statistical estimation problem using the same estimation framework and Bayesian methods that we employed for motion estimation in chapter 2. Accordingly, we modelled the depth map as a random field, formulated single-scale matching algorithms with intensity differences as the basic measurements, and used the depth map estimated from the narrow-baseline image pair as a prior density in matching the wide-baseline image pair. In this probabilistic framework, we identified three classes stereo algorithms: (1) "fully independent" algorithms that model the prior density of the disparity field as completely uncorrelated, (2) joint 1-D algorithms that model the prior density as correlated within scanlines but uncorrelated across scanlines, and (3) joint 2-D algorithms that model correlation in both dimensions. We then developed algorithms for the fully independent and the joint 1-D cases. For the joint 1-D case, we developed two algorithms that represent conceptual advances over current, regularization-based approaches to stereo. This is achieved by deriving surface derivative constraints and disparity correlation models from the narrow-baseline depth map, rather than from universal heuristics about the nature of scenes. By restricting consideration to one-dimension, we were also able to use an efficient, dynamic programming algorithm to find optimal disparity estimates for the entire scanline. Finally, we considered joint 2-D algorithms and concluded that their prime attraction, that of enforcing consistency between scanlines, is already accomplished to some degree by area-based matching operators. Therefore, the additional expense of 2-D coupling may be unnecessary.

After presenting these algorithms, we examined several issues relating to the direction and

distance to move the cameras in the bootstrap operation. Regarding direction, we used a sensitivity analysis to compare the precision of depth estimates obtained from forward and lateral camera motion. The conclusion was that lateral motion is far superior to forward motion; therefore, we are pessimistic about the usefulness of forward motion for estimating depth in general. We then examined issues of sensitivity, computational complexity, and the possibility of match ambiguity that arise in determining the baslines for the narrow and wide-baseline image pairs. For each issue, we derived criteria that may be useful in guiding baseline specification.

This work has made the following contributions:

- It proposed an overall paradigm for depth map estimation from stereo image sequences. The lack of such a paradigm to date has made it unclear how to make progress on this problem.
- It developed specific new algorithms for the first component of the paradigm, which was the bootstrap operation. These algorithms embodied conceptual advances over the use of regularization and expensive optimization algorithms for stereo fusion. They were also demonstrated to perform very well on images of complex, outdoor scene models.
- It demonstrated the potential usefulness of pixel-based models of depth uncertainty, as well as the effectiveness of area-based matching for depth estimation with complex scenes. We believe that stereo research must return to area-based methods to be effective in such domains.

# Chapter 6

# Summary and Conclusions

In chapter 1, we illustrated the potential role of stereo vision as a depth and motion sensor for robotics. We also outlined the scope of the depth and motion estimation problem by classifying sub-problems according to the type of depth or motion model employed. Subsequent chapters developed solutions to two sub-problems that are fundamental in the classification. In this chapter, we summarize the approaches taken and the results obtained, re-iterate our principal conclusions, and review areas for extension.

## 6.1 Motion Estimation

We began by using feature-based methods to estimate the rigid rotation and translation of a robot vehicle between successive image pairs of a stereo image sequence. This was done by tracking 3-D point features or "landmarks" and by using the apparent motion of the landmarks to estimate the actual motion of the vehicle. We decomposed this problem into two sets of issues:

- the mathematical modelling and numerical issues involved in parameter estimation, given noisy measurements of landmark coordinates;
- the system and algorithmic issues involved in reliably tracking the landmarks through the image sequence.

In other words, the first set of issues concerned what measurements to get and how to use them, whereas the second set concerned how to obtain the measurements themselves. We will summarize our approach, then draw conclusions and discuss extensions.

### 6.1.1 Summary

We formulated the estimation part of the problem by employing a standard methodology from the optimal estimation literature [Maybeck79]. This was instantiated as follows. From each stereo pair, our system observed 3-D landmark coordinates relative to the current vehicle position. Uncertainty in these observations was modelled by 3-D Gaussian random vectors with zero means

and with covariance matrices that were determined from the images. With these observations, we derived a series of estimators for the vehicle motion and landmark coordinates. The first was a simple least-squares (LS) estimate of the vehicle rotation and translation between frames. This estimator did not use the full model of measurement uncertainty; however, it had the advantage that a direct solution for the unknown motion parameters was available. We then derived a maximum-likelihood (ML) estimate of the motion parameters. This estimator employed the full uncertainty model, but a direct solution was not available. Therefore, we linearized the equations about the initial estimate provided by the LS solution and iteratively refined the estimate. Finally, we derived sequential Bayesian estimates of both the motion parameters and the landmark coordinates. After processing each stereo pair, this estimator updated the landmark coordinates by combining the new observations with the existing landmark model. The updated coordinates were defined relative to the current coordinate frame. Thus, the sequential estimator continually updated a local world model defined relative to the current coordinate frame.

Landmarks were tracked through stereo image sequences with refined versions of the algorithms developed in [Moravec80]. Landmarks were defined by using an interest operator to detect highly localizable points in one image of a stereo pair; for example, such points typically occur at corners of objects. Coarse-to-fine correlation was used to find the corresponding points in the other image of the stereo pair. The resulting pairs of corresponding points were submitted to triangulation and error-propagation routines that computed the 3-D landmark observations and associated uncertainty model. To locate previously-defined landmarks in a new image pair, a prior estimate of the motion parameters was used to project the landmark model onto the new images. Then, uncertainty in the prior motion estimate was used to determine search windows around the projected location of each feature. The coarse-to-fine correlation procedure was then used to locate each landmark within the respective search window. A rigidity constraint was used to filter out errors in feature matching or tracking.

The entire approach was evaluated by a variance analysis, by simulations, and by laboratory experiments. The variance analysis examined the impact of landmark tracking on the variance of global position estimates obtained by concatenating successive transformations. Over three successive stereo pairs, we found that tracking the same landmarks led to half the variance that results from picking new landmarks for each step. This illustrated quantitatively the benefit of tracking. Next, we used simulations to compare the LS and the ML estimators in order to examine the impact of the 3-D Gaussian model of uncertainty in the observed landmark coordinates. When landmarks are very near the robot, the two estimators performed similarly; however, the performance of the LS estimator degraded rapidly with increasing landmark distance, whereas the ML estimator continued to perform quite well. We also found that the performance of the sequential Bayesian estimator was only marginally better than the ML estimator. Finally, the laboratory experiments confirmed the simulation results regarding the relative performance of the LS and ML estimators. In one trial, the system achieved an accuracy of 2% of distance and one degree of orientation over 55 stereo pairs covering 5.5 meters of vehicle travel. This was the first demonstration of accurate visual motion estimation in unknown environments. The results of a second trial were comparable, except for two poor motion estimates near the end of the run. These were attributed to failures of the feature tracking and error detection mechanisms.

### 6.1.2 Conclusions

The two most important conclusions we draw from this work are

- that accurate visual motion estimation in unknown environments is feasible, and
- that determining the impact of uncertainty is essential in analyzing this, as well as other vision problems.

The system developed here provides a case study that may be useful in developing vision systems for other motion estimation problems. In this respect, it is worthwhile to review lessons learned about the parameter estimation and feature tracking issues mentioned earlier. Regarding parameter estimation, one of the primary concerns is to ensure the accuracy and stability of the computed estimates. This can be achieved with a combination of design-time and run-time techniques. These include using appropriate uncertainty models, using sensitivity analyses to determine the impact of uncertainty, and maintaining adequate numbers and adequate spatial distribution of the landmarks during operation (see [Wunsche86] for a discussion of the latter point). On-line checks of error covariance can be used to detect poor conditioning. Regarding feature tracking, two strategies are important for minimizing both the expense and the probability of correspondence errors. The first is to use data redundancy to verify that correspondence is correct. This is present in our system in the rigidity test and in the over-determined approach to parameter estimation. The second is to acquire images rapidly enough that the world model does not change radically between frames. This allows a substantial part of the existing world model to be used to disambiguate multiple possible interpretations of new images.

### 6.1.3 Extensions

Related work and possible extensions were discussed in chapter 2. We will review the main points here. The discussion will follow the classification of motion problems into single rigid body, multiple rigid body, and deformable situations.

Several unresolved issues exist within the scope of chapter 2. First, the landmark model and the estimators were formulated in terms of 3-D world coordinates $(X, Y, Z)$. This requires a nonlinear triangulation operation to convert disparity measurements into 3-D observations. It may be possible to simplify the estimators by reformulating the landmark model in terms of inverse distance $(1/Z)$. Second, the feature tracking and error detection procedures were not always adequate, as was seen in the experiment with the curved vehicle trajectory. Better error detection methods, such as those described in appendix B, may improve this situation. "Robust estimation" methods [Haralick88] are also of interest. A related point is that the existing system does not explicitly check to ensure that the landmark distribution adequately determines the motion estimate. For example, the estimate will be ill-conditioned if all landmarks nearly collinear. A method for selecting appropriate subsets of features from known 3-D models is discussed in [Wunsche86]; similar methods can be employed here.

Beyond these issues, the most immediate extensions of this work are within the scope of single, rigid-body motion. First, our position estimation formulation can be extended to kine-

matic models including velocity and acceleration. Our approach is useful for a jerky vehicle with slow frame rates. More elaborate kinematic models are useful for estimating motions involving smooth accelerations, where high frame rates are available. Modelling and estimation methods for tracking known objects with one or two cameras are discussed in [Dickmanns88,Gennery86,Tietz82,Young88]. Trajectory models are reviewed in [Wertz78].

Another extension within the single, rigid-body framework is to use geometric features other than points. General, statistical methodologies for estimating feature-based models, as well as specific formulations for line segments and planar or quadric surface patches, are given in [Ayache88,Hung88].

As final extension within the single, rigid-body framework, one may consider robot mapping applications in which the positions of the landmarks are of interest, as well as the motion of the vehicle. In such applications, the robot may return to positions visited previously and see old landmarks again. Therefore, a global network of robot and landmark locations may be estimated from observations spread throughout the network. This problem closely resembles the aerial surveying problems addressed in geodesy [Mikhail76,Vanicek86]. Related methods have been applied to robotics in [DurrantWhyte88,Smith87].

Given our results, related work in computer vision, and the wealth of related literature from other fields, motion problems involving single rigid bodies appear to be well understood from the research standpoint. Less has been done for motion problems involving multiple rigid bodies. For problems involving separate bodies, the key issue distinguishing this from the single body case is the need to segment and group features into distinct objects undergoing different motions. Once this is accomplished, single-body methods apply. Techniques based on rigidity testing appear to be relevant here; an initial attack on this problem using rigidity is made in [Zhang88]. For linked bodies, it is also necessary to model the kinematic constraints between the bodies. Some work on this problem is described in [Mulligan89].

In deformable motion, it is necessary to model the surfaces and the deformations they can undergo. Such models are well-known in areas of mechanical engineering [Shabana89]. Recent work in computer vision that estimates dynamic shape models from stereo image sequences is [Terzopoulos87]. Further discussion of this problem is beyond our scope.

## 6.2 Depth Estimation

As we noted above, feature-based depth and motion estimation problems are becoming well understood. These are problems involving the use of parameterized models of geometric curves and surfaces. Although such models are useful in many applications, they have limited ability to represent depth and intensity variations in complex domains, including outdoor and cluttered indoor environments. In the long term, it appears that pixel-based depth map representations may lead to a more fundamental and general-purpose approach. Therefore, we turned our attention to estimating depth maps from stereo image sequences. As above, we will summarize our approach, then review our conclusions and possible extensions.

### 6.2.1 Summary

Our approach to depth map estimation closely paralleled that for feature-based motion estimation by addressing the same two groups of issues: statistical formulation of the estimation problem and system and algorithm design for reliability. Our formulation was influenced by the general framework for estimation problems outlined in [Maybeck79], the statistical formulations of area-based matching presented in [Forstner86,Gennery80,Ryan80], and the random field models and Bayesian estimation methods employed in [Marroquin85,Szeliski88]. Thus, we modelled the depth map as a Gaussian random field, modelled prior information about depth by simple prior densities for the disparity field, and formulated Bayesian, area-based approaches to matching. Our approach to reliability was to use camera translation to obtain narrow-baseline and wide-baseline image pairs, then to use depth estimates from the narrow-baseline image pair to constrain matching in the wide-baseline image pair. This was referred to as a "bootstrap" operation. This approach recognized the need to obtain reliability through redundant sensing, rather than through heuristic assumptions about scene structure.

To obtain depth estimation algorithms, we began by using intensity differences between two images to derive a classical, maximum-likelihood disparity estimator similar to those in [Forstner86,Gennery80,Ryan80]. Sub-pixel precision was obtained by linearization and iteration. The error variance of the disparity estimate was inversely proportional to the square of the local intensity derivative. We then extended the maximum-likelihood estimator to Bayesian algorithms for the bootstrap operation. In these algorithms, depth estimates from the narrow-baseline image pair determine a prior probability density of the disparity field for matching the wide-baseline image pair. We classified single-scale matching algorithms as fully independent, joint 1-D, or joint 2-D, according to whether the algorithms estimated depth independently for each pixel, jointly for each scanline, or jointly for the entire image. We then developed algorithms for the independent and joint 1-D cases. For the independent case, the Bayesian formulation led to quadratic weights on the search windows for the wide-baseline image pair. For the joint 1-D case, we derived 1-D coupling models from physical intuition and also from statistical considerations. Physical intuition led to constraining disparity derivatives measured from the wide-baseline image pair to be similar to those measured from the narrow-baseline image pair. Statistical considerations led to an alternate approach that modelled correlations in the narrow-baseline depth estimates as having an exponential correlation function. In both approaches, optimal disparity estimates for an entire scanline could be obtained efficiently by dynamic programming and all scanlines could be processed in parallel. A brief consideration of joint 2-D estimators led to the conclusion that these may not offer much improvement over the other estimators, but would increase the cost of matching. Finally, we demonstrated that the algorithms developed here perform very well on images of complex scenes.

### 6.2.2 Conclusions

We draw three main conclusions from this work:

- first, the bootstrap operation produces reliable depth maps with simple, efficient matching algorithms;
- second, the bootstrap operation allows us to replace regularization-based matching constraints, which embody heuristic properties of scenes in general, by more powerful constraints derived from measurements of the scene at hand;
- third, the random field model of depth is a promising tool for integrating stereo depth estimates into multi-sensor robot perception systems, especially for applications in unstructured environments.

Overall, the approach was very successful. We expect that the statistical formulation, the area-based approach to matching, and the bootstrap operation will be important directions in future stereo research.

### 6.2.3 Extensions

Related work and possible extensions were discussed at length in chapter 5. We will review the main issues here.

First, the adequacy of the uncertainty model must be examined further, particularly concerning local fluctuations of mean intensity. Next, there are a large number of unaddressed issues within the scope of the bootstrap operation, including occlusion detection, the use of multiple scales, and trinocular camera configurations. More elaborate reasoning can be done about the results of the matching algorithms to quantify degrees of ambiguity and likelihood of errors. Extensions to process stereo image sequences are clearly desirable, but are beyond our scope to discuss here. Finally, the bootstrap techniques developed here may be useful in other applications, particularly in aerial surveying.

# Bibliography


[Adelson87] E. H. Adelson, E. Simoncelli, and R. Hingorani. Orthogonal pyramid transforms for image coding. In *Proc. SPIE Vol 845: Visual Communications and Image Processing II*, pages 50–58, SPIE, October 1987.

[Aloimonos87] J. Aloimonos, I. Weiss, and A. Bandyopadhyay. Active vision. In *Proc. First International Conference on Computer Vision*, pages 35–54, IEEE Computer Society Press, 1987.

[Anandan84] P. Anandan. Computing dense displacement fields with confidence measures in scenes containing occlusion. In *IUS Workshop*, pages 236–246, DARPA, October 1984.

[Arnold80] R. D. Arnold and T. O. Binford. Geometric constraints in stereo vision. In *Proceedings of SPIE Conference 238: Image Processing for Missile Guidance*, pages 281–291, SPIE, 1980.

[Ayache88] N. Ayache and O. D. Faugeras. Building, registrating, and fusing noisy visual maps. *International Journal of Robotics Research*, 7(6):45–65, December 1988.

[Baker82] H. H. Baker. *Depth from edge and intensity based stereo.* PhD thesis, Stanford Artificial Intelligence Laboratory AIM-347, Stanford University, 1982.

[Baker88] H. H. Baker and R. C. Bolles. Generalizing epipolar-plane image analysis on the spatiotemporal surface. In *Proceedings of the DARPA Image Understanding Workshop*, pages 1022–1030, Morgan Kaufmann Publishers, April 1988.

[Barnard89] S. T. Barnard. Stochastic stereo matching over scale. *International Journal of Computer Vision*, 3(1):17–32, May 1989.

[Barnett84] V. Barnett and T. Lewis. *Outliers in Statistical Data.* John Wiley and Sons, 1984.

[Blostein87] S.D. Blostein and T.S. Huang. Quantization errors in stereo triangulation. In *Proc. 1st Int'l Conf. on Computer Vision*, pages 325–334, IEEE, June 1987.

[Bolles87] R. C. Bolles, H. H. Baker, and D. H. Marimont. Epipolar-plane image analysis: an approach to determining structure from motion. *International Journal of Computer Vision*, 1:7–55, 1987.

[Boult88] L. Chen and T. Boult. An integrated approach to stereo matching, surface reconstruction and depth segmentation using consistent smoothness assumptions. In *Proceedings of the DARPA Image Understanding Workshop*, pages 166–176, Morgan Kaufmann Publishers, April 1988.

[Broida86] T. J. Broida and R. Chellappa. Kinematics and structure of a rigid object from a sequence of noisy images. In *Proc. Workshop on Motion: Representation and Analysis*, pages 95–100, IEEE, May 1986.

[Canny86] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.

[DeGroot70] M. H. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill Book Co., New York, NY, 1970.

[Dickmanns88] E. D. Dickmanns. An integrated approach to feature based dynamic vision. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 820–825, IEEE, June 1988.

[Drumheller86] M. Drumheller and T. Poggio. On parallel stereo. In *IEEE Conf. on Robotics and Automation*, pages 1439–1448, IEEE, April 1986.

[DurrantWhyte88] H. F. Durrant-Whyte. *Integration, Coordination and Control of Multi-Sensor Robot Systems*. Kluwer Academic Publishers, 1988.

[Elfes89] A. Elfes. *Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation*. PhD thesis, Electrical and Computer Engineering Department/Robotics Institute, Carnegie Mellon University, May 1989.

[Forstner86] W. Forstner and A. Pertl. Photogrammetric standard methods and digital image matching techniques for high precision surface measurements. In E. S. Gelsema and L. N. Kanal, editors, *Pattern Recognition in Practice II*, pages 57–72, Elsevier Science Publishers, 1986.

[Forstner88] W. Forstner. Personal communication. 1988.

[Forstner89] W. Forstner. Precision of geometric features derived from image sequences. In K. Linkwitz and U. Hanleiter, editors, *High Precision Navigation: Integration of Navigational and Geodetic Methods*, pages 313–329, Springer-Verlag, 1989.

[Geiger87] D. Geiger and A. Yuille. Stereopsis and eye-movements. In *Proc. 1st Int'l Conf. on Computer Vision*, pages 306–314, IEEE, June 1987.

[Gelb74] A. Gelb, editor. *Applied Optimal Estimation.* MIT Press, Cambridge, MA, 1974.

[Gennery80] D. B. Gennery. *Modelling the environment of an exploring vehicle by means of stereo vision.* PhD thesis, Stanford University, June 1980.

[Gennery86] D. B. Gennery. Stereo vision for the acquisition and tracking of moving three-dimensional objects. In A. Rosenfeld, editor, *Techniques for 3-D Machine Perception*, pages 53–74, Elsevier Science Publishers, 1986.

[Golub83] G. E. Golub and C. F. van Loan. *Matrix Computations.* Johns Hopkins University Press, 1983.

[Graybill83] F. A. Graybill. *Matrices with Applications in Statistics.* Wadsworth International Group, Belmont, CA, 1983.

[Gruen84] A. W. Gruen. Algorithmic aspects in on-line triangulation. In *XVth Int'l Congress of Photogrammetry and Remote Sensing, Commission III, Part 3a*, pages 342–362, Int'l Society for Photogrammetry and Remote Sensing, 1984.

[Hansen88] C. Hansen, N. Ayache, and F. Lustman. Efficient depth estimation using trinocular stereo. In *Proceedings of SPIE Conference 1003, Sensor Fusion: Spatial Reasoning and Scene Interpretation*, pages 124–131, SPIE, November 1988.

[Haralick88] R. M. Haralick and H. Joo. 2D-3D pose estimation. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, pages 385–391, IEEE, 1988.

[Hawkins80] D. M. Hawkins. *Identification of Outliers.* Chapman and Hall, 1980.

[Hebert83] M. Hebert. *Reconnaissance de formes tridimensionelles.* PhD thesis, L'Universite de Paris-Sud, Centre d'Orsay, September 1983.

[Hebert89] M. Hebert, C. Caillas, E. Krotkov, I. S. Kweon, and T. Kanade. Terrain mapping for a roving planetary explorer. In *Proc. IEEE Conf. on Robotics and Automation*, IEEE Computer Society Press, May 1989.

[Heeger88] D. J. Heeger. Optical flow using spatiotemporal filters. *International Journal of Computer Vision*, 1(4):279–302, January 1988.

[Heel89] Joachim Heel. Dynamic motion vision. In *Proceedings of the DARPA Image Understanding Workshop*, Morgan-Kaufman Publishers, May 1989.

[Horn86] B. K. P. Horn. *Robot Vision.* MIT Press, 1986.

[Horn87] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America*, 4(4):629–642, April 1987.

[Horn88] B. K. P. Horn. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America*, 5(7):1127–1135, July 1988.

[Hung88] Y. Hung, D. B. Cooper, and B. Cernushci-Frias. Bayesian estimation of 3-d surfaces from a sequence of images. In *Proc. IEEE Conference on Robotics and Automation*, pages 906–911, IEEE, April 1988.

[Kass84] M. Kass. *Computing visual correspondence.* Master's thesis, MIT, 1984.

[Kass86] M. Kass. Computing visual correspondence. In A. P. Pentland, editor, *From Pixels to Predicates: Recent Advances in Computational and Robotic Vision*, chapter 4, pages 78–92, Ablex Publishing Corp., Norwood, N. J., 1986.

[Kriegman87] T.O. Binford D.J. Kriegman, E. Triendl. A mobile robot: sensing, planning and locomotion. In *Proc. Conf. on Robotics and Automation*, pages 402–408, IEEE, March 1987.

[Krotkov88] E. Krotkov and R. Kories. Adaptive control of cooperating sensors: focus and stereo ranging with an agile camera system. In *Proc. IEEE Conf. on Robotics and Automation*, pages 548–553, Philadelphia, April 1988.

[Lucas84] B. D. Lucas. *Generalized Image Matching by the Method of Differences.* PhD thesis, Carnegie Mellon University, July 1984.

[Mallat87] S. G. Mallat. A compact multiresolution representation: the wavelet model. In *Proceedings of the IEEE Workshop on Computer Vision*, pages 2–7, Miami, Fl, December 1987.

[Marce86] L. Marce. Personal communication. 1986.

[Marr76] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194:283–287, 1976.

[Marroquin85] J. L. Marroquin. *Probabilistic Solution of Inverse Problems.* PhD thesis, MIT, September 1985.

[Marroquin87] J. Marroquin, S. Mitter, and T. Poggio. Probabilistic solution of ill-posed problems in computational vision. *Journal of the American Statistical Association*, 82(397):76–89, March 1987.

[Matthies89] L. H. Matthies, R. Szeliski, and T. Kanade. Kalman filter-based algorithms for estimating depth from image sequences. *Internationl Journal of Computer Vision*, 3:209–236, 1989.

[Maybeck79] P. S. Maybeck. *Stochastic Models, Estimation, and Control.* Volume 1, Academic Press, New York, NY, 1979.

[Mikhail76] E. M. Mikhail. *Observations and Least Squares.* University Press of America, Lanham, MD, 1976.

[Milenkovic85] V. J. Milenkovic and T. Kanade. Trinocular vision using photometric and edge orientation constraints. In *Proc. DARPA Image Understanding Workshop*, December 1985.

[Moravec80] H. P. Moravec. *Obstacle avoidance and navigation in the real world by a seeing robot rover.* PhD thesis, Stanford University, September 1980.

[Mulligan89] I. J. Mulligan, A. K. Mackworth, and P. D. Lawrence. *A model-based vision system for manipulator position sensing.* Technical Report 89-13, University of British Columbia, 1989.

[Nagel86] H.-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(5):565–593, September 1986.

[Nalwa86] V. Nalwa. On detecting edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):699–714, Nov. 1986.

[Ohta85] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7(2):139–154, March 1985.

[Paul81] R. P. Paul. *Robot Manipulators.* MIT Press, 1981.

[Podnar84] G. Podnar, K. Dowling, and M. Blackwell. *A functional vehicle for autonomous mobile robot research.* Technical Report CMU-RI-TR-84-28, Carnegie Mellon University, April 1984.

[Poggio85] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317(n):314–319, September 1985.

[Prazdny85] K. Prazdny. Detection of binocular disparities. *Biological Cybernetics*, 52:93–99, 1985.

[Quam84] L. H. Quam. Hierarchical warp stereo. In *Proc. Image Understanding Workshop*, Science Applications International, 1984.

[Requicha80] A. Requicha. Representations for rigid solids: theory, methods, and systems. *Computing Surveys*, 12(4):437–464, December 1980.

[Rives86] P. Rives, E. Breuil, and B. Espiau. Recursive estimation of 3d features using optical flow and camera motion. In *Proceedings Conference on Intelligent Autonomous Systems*, pages 522–532, Elsevier Science Publishers, December 1986. (also appeared in Proc. 1987 IEEE Int'l Conf. on Robotics and Automation).

[Rogers76] D. F. Rogers and J. A. Adams. *Mathematical Elements for Computer Graphics*. McGraw-Hill Book Co., New York, 1976.

[Ryan80] T. W. Ryan, R. T. Gray, and B. R. Hunt. Prediction of correlation errors in stereo-pair images. *Optical Engineering*, 19(3):312–322, May/June 1980.

[Sankoff83] D. Sankoff and J. B. Kruskal. *Time Warps, String Edits, and Macromolecules: the Theory and Practice of Sequence Comparison*. Addison-Wesley, 1983.

[Schonemann66] P. H. Schonemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, March 1966.

[Schonemann70] P. H. Schonemann and R. M. Carroll. Fitting one matrix to another under choice of a central dilation and a rigid motion. *Psychometrika*, 35(2):245–255, June 1970.

[Sedgewick83] R. Sedgewick. *Algorithms*. Addison-Wesley, 1983.

[Shabana89] A. A. Shabana. *Dynamics of Multibody Systems*. John Wiley and Sons, 1989.

[Slama80] C. C. Slama. *Manual of Photogrammetry*. Volume 4th ed., American Society of Photogrammetry, Falls Church, Va., 1980.

[Smith87] R. C. Smith, M. Self, and P. Cheeseman. Estimating uncertain spatial relationships in robotics. In *Uncertainty in Artificial Intelligence*, North-Holland, 1987.

[Snyder87] M. A. Snyder. Uncertainty analysis of image measurements. In *Image Understanding Workshop*, pages 681–693, DARPA, Los Angeles, February 1987.

[Sobel74] I. Sobel. On calibrating computer controlled cameras for perceiving 3-d scenes. *Artificial Intelligence*, 5:185–198, 1974.

[Stewart88] C. V. Stewart and C. R. Dyer. The trinocular general support algorithm: a three-camera stereo algorithm for overcoming binocular matching errors. In *Proc. Second Int'l Conf. on Computer Vision*, pages 134–138, IEEE, December 1988.

[Szeliski85] R. Szeliski and G. Hinton. Solving random-dot stereograms using the heat equation. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 284–288, IEEE, 1985.

[Szeliski88] R. Szeliski. *Bayesian Modeling of Uncertainty in Low Level Vision.* PhD thesis, Carnegie Mellon University, August 1988.

[Terzopoulos87] D. Terzopoulos, A. Witkin, and M. Kass. Energy constraints on deformable models: recovering shape and non-rigid motion. In *Proceedings of AAAI-87*, pages 755–760, AAAI, 1987.

[Thorpe84] C. E. Thorpe. *FIDO: Vision and Navigation for a Robot Rover.* PhD thesis, Carnegie Mellon University, December 1984.

[Tietz82] J.C. Tietz and J.H. Kelly. *Development of an Autonomous Video Rendezvous and Docking System.* Technical Report MCR-82-569, Martin Marietta Aerospace, Denver, CO, June 1982.

[Tsai87] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4):323–345, August 1987.

[Vanicek86] P. Vanicek and E. J. Krakiwsky. *Geodesy: The Concepts.* Elsevier Science Publishing Co., New York, 1986.

[Vanmarcke83] E. Vanmarcke. *Random Fields: Analysis and Synthesis.* MIT Press, 1983.

[VanTrees68] H. L. Van Trees. *Detection, Estimation, and Modulation Theory.* Volume Part I, John Wiley and Sons, New York, 1968.

[Wertz78] J. R. Wertz. *Spacecraft Attitude Determination and Control.* D. Reidel Publishing Company, 1978.

[Wilkinson71] J. H. Wilkinson and C. Reinsch. *Linear Algebra.* Springer-Verlag, 1971.

[Witkin87] A. Witkin, D. Terzopoulos, and M. Kass. Signal matching through scale space. *International Journal of Computer Vision*, 1(2):133–144, 1987.

[Wunsche86] H.-J. Wunsche. Detection and control of mobile robot motion by real-time computer vision. In *Proc. Conf. on Mobile Robots*, SPIE, October 1986.

[Xu85] G. Xu, S. Tsuji, and A. Minoru. Coarse-to-fine control strategy for matching motion stereo pairs. In *Proceedings of IJCAI*, pages 892–894, 1985.

[Young88] G. Young and R. Chellappa. 3-D motion estimation using a sequence of noisy stereo images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 710–716, IEEE, June 1988.

[Zhang88] Z. Zhang, O. Faugeras, and N. Ayache. Analysis of a sequence of stereo scenes containing multiple moving objects using rigidity constraints. In *Proceedings of the Second International Conference on Computer Vision*, pages 177–186, IEEE, December 1988.

# Appendix A

# Camera Model and Calibration Procedure

In chapter 2, we defined the stereo observation model in terms of an idealized camera system. This is adequate for simulation; however, the experiments conducted with the actual robot vehicle required triangulation and error propagation procedures appropriate for the real camera system. In this appendix, we describe the camera model employed, the triangulation and error propagation procedures, and the procedures used to calibrate the camera model.

## A.1 Camera Model

Following the practice in photogrammetry [Slama80], we divide the camera model into two components: the *exterior* model, which specifies the location and orientation of the camera relative to a world coordinate frame, and the *interior* model, which specifies geometric characteristics of the imaging process within the camera.

The interior model used for the implementation in chapter 2 was a simple pinhole model similar to that used in [Sobel74]. This model defines the relationship between the 3-D coordinates of a point $\mathbf{P} = [X\ Y\ Z]^T$ and its image coordinates $\mathbf{p} = [x\ y]^T$ by the equations (figure A.1a)

$$\begin{aligned} x &= s_x \frac{X}{Z} + c_x \\ y &= s_y \frac{Y}{Z} + c_y. \end{aligned}$$

Defining the *collimation matrix* $\mathbf{C}$ to be

$$\mathbf{C} = \begin{bmatrix} s_x & 0 & c_x \\ 0 & s_y & c_y \end{bmatrix},$$

the interior orientation equation can be abbreviated to

$$\mathbf{p} = \frac{1}{Z}\mathbf{CP}.$$

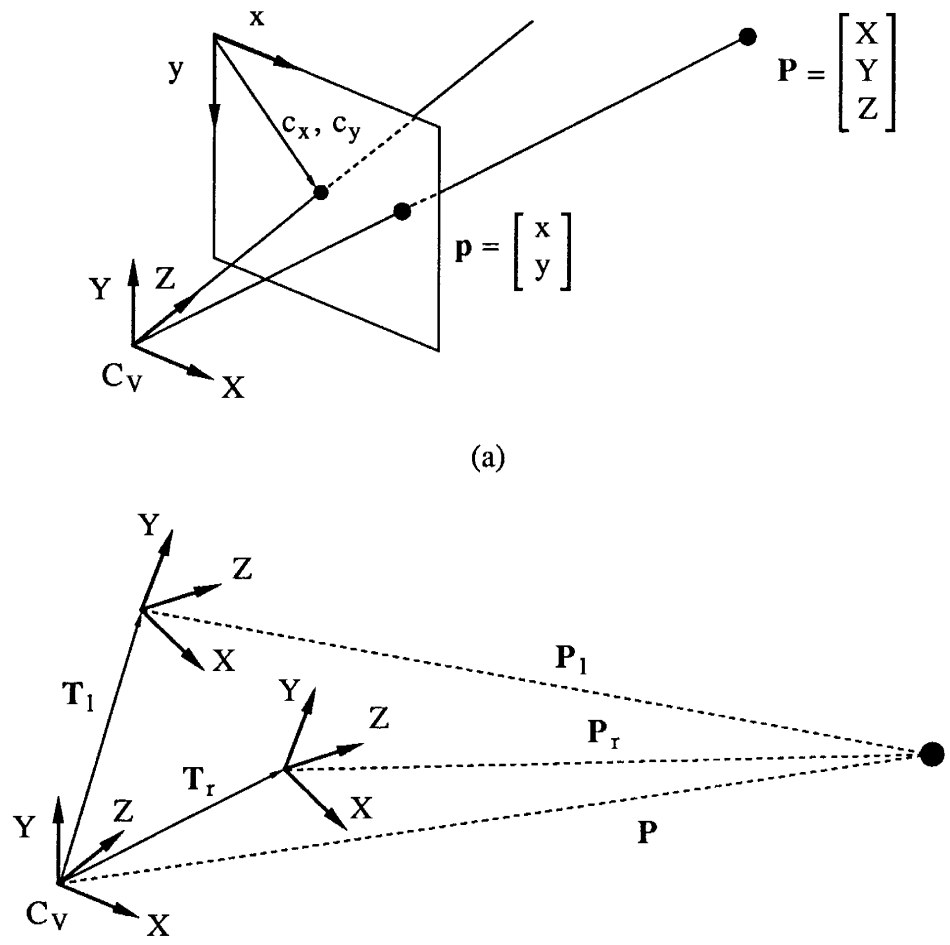



Figure A.1: Coordinate system conventions: (a) interior orientation, (b) exterior orientation (camera coordinate frames relative to vehicle coordinate frame)

For the exterior camera models, we take the origin of the 3-D camera coordinate system to be at the center of projection. For the simulations in chapter 2, the exterior models simply placed the left and right camera coordinate systems at offsets of $\pm b$ along the $X$ axis of the vehicle coordinate system, with the camera coordinate axes parallel to the vehicle coordinate axes. For experiments with real images, the coordinate systems were established as illustrated in figure A.1b. The projection of a 3-D point $\mathbf{P}$ onto a camera therefore is defined by the transformation

$$\mathbf{p} = \frac{1}{Z'}\mathbf{CR}(\mathbf{P}+\mathbf{T}), \tag{A.1}$$

where $\mathbf{T}$ specifies the translational offset of the center of projection in the world coordinate frame, $\mathbf{R}$ specifies the rotation of the camera coordinate frame relative to the world frame, and $Z'$ is the $Z$ component of the transformed vector $\mathbf{R}(\mathbf{P}+\mathbf{T})$. One such equation is defined for each camera; therefore, the parameters of the collimation matrix $\mathbf{C}$, rotation matrix $\mathbf{R}$, and translation vector $\mathbf{T}$ must be calibrated separately for each camera.

## A.2 Triangulation

The camera model of equation (A.1) is instantiated as a measurement equation for each camera,

$$\begin{aligned}\mathbf{q}_l &= \frac{1}{Z'_l}\mathbf{C}_l\mathbf{R}_l(\mathbf{P}+\mathbf{T}_l)+\mathbf{v}_l \\ \mathbf{q}_r &= \frac{1}{Z'_r}\mathbf{C}_r\mathbf{R}_r(\mathbf{P}+\mathbf{T}_r)+\mathbf{v}_r,\end{aligned}$$

where $\mathbf{v}_l$ and $\mathbf{v}_r$ denote zero-mean, Gaussian noise vectors as described in chapter 2. These equations can be abbreviated as a set of nonlinear measurement equations

$$\mathbf{q} = \begin{bmatrix}\mathbf{q}_l \\ \mathbf{q}_r\end{bmatrix} = \mathbf{h}(\mathbf{P}) + \mathbf{v}_\mathbf{q}. \tag{A.2}$$

Here $\mathbf{v}_\mathbf{q} = [\mathbf{v}_l^T \ \mathbf{v}_r^T]^T$, so the covariance of $\mathbf{v}_\mathbf{q}$ is

$$\Sigma_{\mathbf{v}_\mathbf{q}} = \begin{bmatrix}\Sigma_l & 0 \\ 0 & \Sigma_r\end{bmatrix}.$$

The triangulation problem is to estimate $\mathbf{P}$ from (A.2) and to model the uncertainty in the estimate. As described in chapter 2, we denote the estimate as the *observation* $\mathbf{Q}$ and model its uncertainty by the additive, random noise vector $\mathbf{v}$:

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

Since (A.2) is nonlinear, the estimation procedure is not completely straightforward and the uncertainty in the estimate is not Gaussian. We deal with these issues in the following manner:

- We obtain an initial estimate of **P** by solving a set of linear equations.
- We linearize (A.2) in order to obtain an iterative approach to computing a maximum likelihood estimate of **P**.
- We model the estimation error (i.e. **v**) as zero-mean Gaussian and approximate its covariance by error propagation.

By expanding (A.2), temporarily ignoring the noise term, and rearranging, the measurement equations can be rewritten as follows:

$$\begin{bmatrix} \mathbf{q}_l \mathbf{R}_{l3} \mathbf{T}_l - \mathbf{C}_l \mathbf{R}_l \mathbf{T}_l \\ \mathbf{q}_r \mathbf{R}_{r3} \mathbf{T}_r - \mathbf{C}_r \mathbf{R}_r \mathbf{T}_r \end{bmatrix} = \begin{bmatrix} \mathbf{C}_l \mathbf{R}_l - \mathbf{q}_l \mathbf{R}_{l3} \\ \mathbf{C}_r \mathbf{R}_r - \mathbf{q}_r \mathbf{R}_{r3} \end{bmatrix} \mathbf{P},$$

where $\mathbf{R}_{l3}$ and $\mathbf{R}_{r3}$ denote the third rows of the rotation matrices $\mathbf{R}_l$ and $\mathbf{R}_r$, respectively. We abbreviate this equation to

$$\mathbf{A} = \mathbf{H}\mathbf{P}.$$

This equation can be used to estimate **P** via linear least-squares; the result is

$$\hat{\mathbf{P}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathbf{A}.$$

This estimate was adequate for our purposes. However, to obtain a maximum likelihood estimate this can be used as an initial estimate, denoted $\mathbf{P}_0$, and (A.2) can be linearized about the initial estimate:

$$\mathbf{q} \approx \mathbf{h}(\mathbf{P}_0) + \mathbf{H}_0(\mathbf{P} - \mathbf{P}_0) + \mathbf{v}_\mathbf{q},$$

where

$$\mathbf{H}_0 = \left[\frac{d(\mathbf{h}(\mathbf{P}))}{d\mathbf{P}}\right]_{\mathbf{P}=\mathbf{P}_0}.$$

Applying standard methods [VanTrees68] yields

$$\hat{\mathbf{P}} = (\mathbf{H}_0^T\Sigma_{\mathbf{v}_\mathbf{q}}^{-1}\mathbf{H}_0)^{-1}\mathbf{H}_0^T\Sigma_{\mathbf{v}_\mathbf{q}}^{-1}(\mathbf{q} - \mathbf{h}(\mathbf{P}_0) + \mathbf{H}_0\mathbf{P}_0) \tag{A.3}$$

as the landmark estimate and

$$\Sigma = (\mathbf{H}_0^T\Sigma_{\mathbf{v}_\mathbf{q}}^{-1}\mathbf{H}_0)^{-1}. \tag{A.4}$$

as its error covariance. This can be iterated to refine the estimate. In the notation of chapter 2, (A.3) defines the estimate **Q** and (A.4) determines the noise covariance $\Sigma_\mathbf{v}$.

The derivation above can be cast in Bayesian terms by using a non-informative prior distribution for **P**; in our case, that is a Gaussian distribution with arbitrary mean and zero inverse covariance. Using MAP estimation then leads to the same estimate and error covariance as above. This gives a Bayesian justification for defining the initial world model as equivalent to the observations made at $t_0$, as we do in chapter 2.

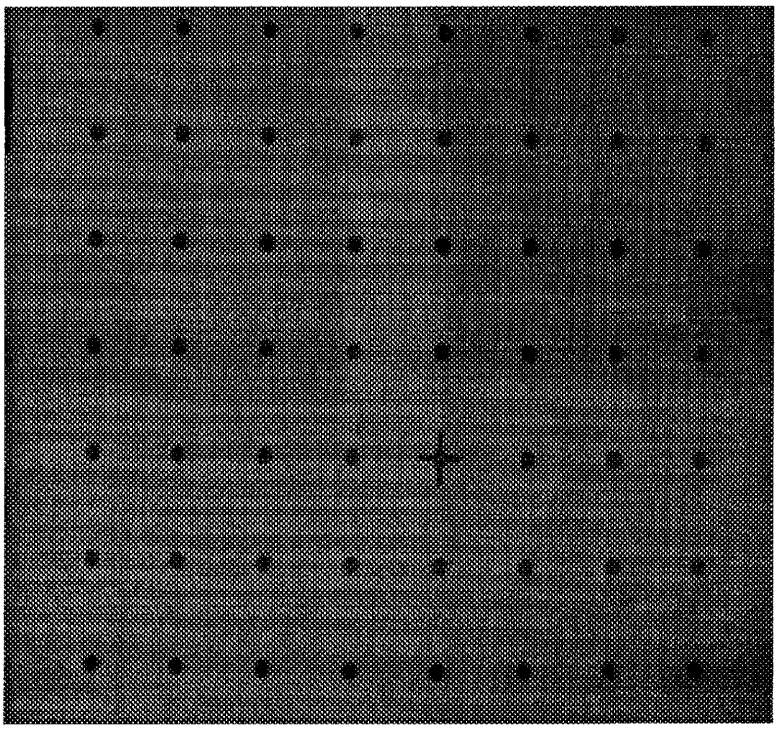


Figure A.2: Calibration grid. This consists of black circles spaced six inches apart on a white background.

## A.3 Calibration Procedure

The camera model was calibrated using a target consisting of a flat, white board painted with an array of black, circular dots (figure A.2). The dots were one inch in diameter and were spaced six inches apart. This target was placed from six to fifteen feet from the cameras. The 3-D coordinates $\mathbf{P}_j$ of the dots in the vehicle coordinate frame were determined manually by tape measure; for each camera, the 2-D image coordinates $\mathbf{q}_j$ of the dot centroids were determined automatically with software developed by Moravec [Moravec80]. The calibration problem then was to determine the interior and exterior orientation parameters for each camera from measurements $\mathbf{P}_j$, $\mathbf{q}_j$ and the camera model equations

$$\mathbf{q}_j = \frac{1}{Z'_j}\mathbf{CR}(\mathbf{P}_j + \mathbf{T}) + \mathbf{v}_{q_j} . \tag{A.5}$$

This was treated as a least-squares problem in which the camera parameters were chosen to minimize the sum of squared residuals

$$\sum_j \mathbf{e}_j^T \mathbf{e}_j ,$$

with

$$\mathbf{e}_j = \mathbf{q}_j - \frac{1}{Z'_j}\mathbf{CR}(\mathbf{P}_j + \mathbf{T}) .$$

Since (A.5) is nonlinear in the camera parameters, the least-squares problem is nonlinear. This was solved with the following two-step procedure:

1. Initial estimates of R and T were obtained by manual measurement of the camera position and orientation. Treating these quantities as known, (A.5) reduces to a linear equation in the interior orientation parameters. This allows initial estimates of the interior orientation to be obtained via linear least squares.

2. Treating all of the parameters as unknown, (A.5) was then linearized to refine the camera parameters iteratively. Convergence was generally obtained within five iterations.

This procedure was applied separately for each camera. The resulting camera models were sufficiently accurate to obtain the results presented in chapter 2; further quantitative evaluation of the camera model itself was not undertaken.

Potential limitations of the camera model and calibration procedure used here include the lack of a model of lens distortion, the lack of a model of the precision of the input measurements $\mathbf{P}_j$ and $\mathbf{q}_j$, and the manual component of the calibration procedure. These issues are addressed in [Slama80] and [Tsai87].

# Appendix B

# Mathematics of Motion Estimation

This appendix gives detailed derivations of the motion estimators and error detection procedures outlined in chapter 2. Section B.1 derives the least-squares estimates of the motion parameters, section B.2 provides additional details for the derivation of the maximum-likelihood estimates of the motion parameters, and section B.3 completes the derivation of the sequential Bayesian estimator. Sections B.4 and B.5 discuss the posterior estimate of the reference variance and the error detection procedures, respectively.

## B.1 Least-squares Estimation of $\Theta$ and T

Given point sets $\mathbf{Q}_{pj}$ and $\mathbf{Q}_{cj}$, related by an unknown rigid rotation and translation, which we express by the matrix $\mathbf{R}$, the vector $\mathbf{T}$, and the equation

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

our problem is to find a best-fit estimate for the rotation and translation. We approach this by forming the residual

$$\mathbf{e}_j = \mathbf{Q}_{cj} - \mathbf{R}\,\mathbf{Q}_{pj} - \mathbf{T}$$

and seeking the transformation that minimizes the weighted, sum-squared-error

$$q(\mathbf{R}, \mathbf{T}) = \sum_j w_j \mathbf{e}_j^T \mathbf{e}_j. \tag{B.1}$$

The weights $w_j$ are related to the observation errors in $\mathbf{Q}_{pj}$ and $\mathbf{Q}_{cj}$; details are given in chapter 2.

This optimization problem is non-linear in the rotation angles. However, two direct solutions for the optimal transformation are known. One uses quaternions to express the rotation and obtains the best-fit transformation via convenient properties of quaternion algebra. This method is well-known in the aerospace literature [Wertz78, p. 426] and has also been presented in the computer vision literature [Hebert83,Horn87]. The second solution, which we employ here, expresses the rotation in terms of an orthogonal matrix and obtains the best-fit transformation

via properties of matrix algebra. This method was developed in the psychometric literature [Schonemann66,Schonemann70]. Variants of this algorithm are also presented in [Horn88].

This solution solves for the elements of the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{T}$. The requirement that $\mathbf{R}$ be orthogonal is expressed by constraint equations that stipulate that the columns $\mathbf{r}_i$ of $\mathbf{R}$ be mutually orthogonal and of unit length. These constraints are given by the expressions

$$\mathbf{r}_i^T\mathbf{r}_i = 1, \quad \mathbf{r}_i^T\mathbf{r}_j = 0 \text{ where } i,j \in \{1,2,3\} \text{ and } i \neq j.$$

These are incorporated in the objective function via Lagrange multipliers as follows:

$$q(\mathbf{R},\mathbf{T},l_i,m_i) = \left\{\sum_j w_j \mathbf{e}_j^T\mathbf{e}_j\right\} + \sum_{i=1}^{3} l_i(\mathbf{r}_i^T\mathbf{r}_i - 1) + \sum_{\substack{i,j=1 \\ i\neq j}}^{3} m_i\mathbf{r}_i^T\mathbf{r}_i, \tag{B.2}$$

where $l_i$ and $m_i$ are unknown Lagrange multipliers. The constraint equations can be re-expressed more neatly in matrix notation by letting

$$\mathbf{L} = \begin{bmatrix} l_1 & m_1/2 & m_2/2 \\ m_1/2 & l_2 & m_3/2 \\ m_2/2 & m_3/2 & l_3 \end{bmatrix}$$

and rewriting $q$ as

$$q(\mathbf{R},\mathbf{T},\mathbf{L}) = \left\{\sum_j w_j \mathbf{e}_j^T\mathbf{e}_j\right\} + \text{tr}\{\mathbf{L}(\mathbf{R}^T\mathbf{R} - \mathbf{I})\},$$

where $I$ is the $3 \times 3$ identity matrix and $\text{tr}\{\cdot\}$ is matrix trace.

To minimize this expression, we follow the usual route of taking the derivatives with respect to the unknown parameters, setting these to zero, and solving for the unknowns. However, the resulting equations are still difficult to solve; the method employed in [Schonemann66] involves a clever way to avoid solving for the Lagrange multipliers. First, differentiating $q$ with respect to the translation vector $\mathbf{T}$, we obtain

$$\begin{aligned} \frac{dq}{d\mathbf{T}} &= \sum_j w_j(-2\mathbf{Q}_{cj} + 2\mathbf{R}\mathbf{Q}_{pj} + 2\mathbf{T}) \\ &= 2(-\mathbf{Q}_1 + \mathbf{R}\mathbf{Q}_2 + w\mathbf{T}) \\ &= 0, \end{aligned}$$

where $\mathbf{Q}_1 = \sum_j w_j\mathbf{Q}_{cj}$, $\mathbf{Q}_2 = \sum_j w_j\mathbf{Q}_{pj}$, and $w$ is the sum of the weights, or $w = \sum_j w_j$. Thus, the optimal translation vector is given in terms of the optimal rotation as

$$\mathbf{T} = \frac{1}{w}[\mathbf{Q}_1 - \mathbf{R}\mathbf{Q}_2]. \tag{B.3}$$

The hard part of this problem is to estimate $\mathbf{R}$. To do this, we differentiate $q$ with respect to $\mathbf{R}$, obtaining

$$\begin{aligned} \frac{dq}{d\mathbf{R}} &= \left\{\sum_j w_j(-2\mathbf{Q}_{cj}\mathbf{Q}_{pj}^T + 2\mathbf{R}\mathbf{Q}_{pj}\mathbf{Q}_{pj}^T + 2\mathbf{T}\mathbf{Q}_{pj}^T)\right\} + (\mathbf{L} + \mathbf{L}^T)\mathbf{R} \\ &= 0. \end{aligned}$$

Making the abbreviations

$$\begin{aligned}\mathbf{A} &= \sum_j w_j \mathbf{Q}_{cj} \mathbf{Q}_{pj}^T \\ \mathbf{B} &= \sum_j w_j \mathbf{Q}_{pj} \mathbf{Q}_{pj}^T \\ \mathbf{D} &= \frac{1}{2}(\mathbf{L} + \mathbf{L}^T)\end{aligned}$$

and substituting in for $\mathbf{T}$ from (B.3), we obtain

$$0 = -\mathbf{A} + \mathbf{RB} + \frac{1}{w}(\mathbf{Q}_1\mathbf{Q}_2^T - \mathbf{RQ}_2\mathbf{Q}_2^T) + \mathbf{DR}.$$

Since $\mathbf{R}$ is orthogonal, though unknown, this can be rewritten as

$$\begin{aligned}\mathbf{D} &= \left(\mathbf{A} - \frac{1}{w}\mathbf{Q}_1\mathbf{Q}_2^T\right)\mathbf{R}^T + \mathbf{R}\left(\frac{1}{w}\mathbf{Q}_2\mathbf{Q}_2^T - \mathbf{B}\right)\mathbf{R}^T \\ &= \mathbf{ER}^T + \mathbf{RFR}^T.\end{aligned}$$

Matrices $\mathbf{D}$ and $\mathbf{RFR}^T$ are symmetric; therefore, $\mathbf{ER}^T$ is also symmetric. This implies that

$$\begin{aligned}\mathbf{ER}^T &= \mathbf{RE}^T \\ \mathbf{E} &= \mathbf{RE}^T\mathbf{R} \\ \mathbf{EE}^T &= \mathbf{RE}^T\mathbf{ER}^T.\end{aligned} \tag{B.4}$$

Matrices $\mathbf{EE}^T$ and $\mathbf{E}^T\mathbf{E}$ are symmetric with identical singular values [Golub83, p. 285]. Letting the singular value decompositions of these matrices be

$$\begin{aligned}\mathbf{EE}^T &= \mathbf{USU}^T \\ \mathbf{E}^T\mathbf{E} &= \mathbf{VSV}^T,\end{aligned}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices, equation (B.4) becomes

$$\mathbf{USU}^T = \mathbf{RVSV}^T\mathbf{R}^T.$$

Finally, this gives the solution for the rotation matrix as

$$\mathbf{R} = \mathbf{UV}^T. \tag{B.5}$$

By observing that the singular value decomposition of $\mathbf{E}$ itself is $\mathbf{E} = \mathbf{USV}^T$, we find that the optimal rotation can be computed from the singular value decomposition of $\mathbf{E}$. Using arguments similar to those given in [Schonemann66], it can be shown that this solution corresponds to a minimum of the objective function (B.2) and that this solution is unique. The solution can be ill-conditioned or degenerate if the points are not well-distributed in space; for example, this

happens if the points are nearly collinear. This can be detected by checking for near-zero singular values in **S**.

To summarize the results and to illustrate the steps necessary to implement this solution, let

$$\begin{aligned} w &= \sum_j w_j \\ \mathbf{Q}_1 &= \sum_j w_j \mathbf{Q}_{cj} \\ \mathbf{Q}_2 &= \sum_j w_j \mathbf{Q}_{pj} \\ \mathbf{A} &= \sum_j w_j \mathbf{Q}_{pj} \mathbf{Q}_{cj}^T \\ \mathbf{E} &= \mathbf{A} - \frac{1}{w}\mathbf{Q}_1\mathbf{Q}_2^T \end{aligned}$$

and let

$$\mathbf{E} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

be the singular value decomposition of **E**. Then

$$\begin{aligned} \hat{\mathbf{R}} &= \mathbf{U}\mathbf{V}^T \\ \hat{\mathbf{T}} &= \frac{1}{w}\left[\mathbf{Q}_1 - \hat{\mathbf{R}}\mathbf{Q}_2\right] . \end{aligned}$$

For the implementation used here, the singular value decomposition was computed with the subroutine described in [Wilkinson71]. Rotation angles were extracted from $\hat{\mathbf{R}}$ with algorithms given in [Paul81, chapter 3].

## B.2 Maximum-likelihood Estimation of $\Theta$ and T

In this section, we supply additional details of the iterative solution to the maximum-likelihood estimate of the motion parameters. We will first complete the details of the linearization, then discuss the calculation of the estimate. From equation (2.12), we have

$$\mathbf{Q}_{cj} - \mathbf{R}_0\mathbf{Q}_{pj} + \mathbf{J}_j\Theta_0 = \begin{bmatrix} \mathbf{J}_j & \mathbf{I} \end{bmatrix} \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix} + \mathbf{v}_j . \tag{B.6}$$

The $3 \times 3$ Jacobian matrix $\mathbf{J}_j$ is given by

$$\begin{aligned} \mathbf{J}_j &= \left[\frac{d(\mathbf{R}\mathbf{Q}_{pj})}{d\Theta}\right]_0 \\ &= \left[\mathbf{R}_x\mathbf{Q}_{pj} \;\; \mathbf{R}_y\mathbf{Q}_{pj} \;\; \mathbf{R}_z\mathbf{Q}_{pj}\right]_0 , \end{aligned}$$

where $\mathbf{R}_x$, $\mathbf{R}_y$, and $\mathbf{R}_z$ are the partial derivatives of the rotation matrix with respect to the rotation angles $\theta_x$, $\theta_y$, $\theta_z$, respectively, and $\mathbf{R}_x\mathbf{Q}_{pj}$, $\mathbf{R}_y\mathbf{Q}_{pj}$, $\mathbf{R}_z\mathbf{Q}_{pj}$ form columns of the Jacobian. The

initial estimates of $\mathbf{T}$, $\mathbf{R}$, and therefore $\Theta$ are obtained by the method of the previous section. Efficient methods for computing the partial derivatives of the rotation matrix will be discussed at the end of this section.

As shown in chapter 2, application of the maximum-likelihood method to (B.6) leads to the following linear system,

$$\left[\sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{H}_j\right] \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix} = \left[\sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{Q}_j\right], \tag{B.7}$$

where

$$\begin{aligned} \mathbf{H}_j &= [\mathbf{J}_j \ \mathbf{I}] \\ \mathbf{Q}_j &= \mathbf{Q}_{cj} - \mathbf{R}_0 \mathbf{Q}_{pj} + \mathbf{J}_j \Theta_0 . \end{aligned}$$

From this, we obtain the parameter estimates as

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

with error covariance

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

The linearization and solution process are iterated until $\widehat{\Theta} \approx \Theta_0$.

Given the iteration, a more efficient solution can be obtained by expanding the linear system of (B.7) to

$$\begin{bmatrix} \overline{\mathbf{J}_j^T \mathbf{W}_j \mathbf{J}_j} & \overline{\mathbf{J}_j^T \mathbf{W}_j} \\ \overline{\mathbf{W}_j \mathbf{J}_j} & \overline{\mathbf{W}_j} \end{bmatrix} \begin{bmatrix} \Theta \\ \mathbf{T} \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{J}_j^T \mathbf{W}_j \mathbf{Q}_j} \\ \overline{\mathbf{W}_j \mathbf{Q}_j} \end{bmatrix} . \tag{B.8}$$

The overline notation denotes summation over all landmark points. This $2 \times 2$ partitioned matrix system can be solved for $\mathbf{T}$ in terms of $\Theta$,

$$\hat{\mathbf{T}} = \overline{\mathbf{W}_j}^{-1} [\overline{\mathbf{W}_j \mathbf{Q}_j} - \overline{\mathbf{W}_j \mathbf{J}_j} \Theta] ,$$

and then for $\Theta$ alone,

$$\begin{aligned} \widehat{\Theta} &= [\overline{\mathbf{J}_j^T \mathbf{W}_j \mathbf{J}_j} - \overline{\mathbf{J}_j^T \mathbf{W}_j}\ \overline{\mathbf{W}_j}\ \overline{\mathbf{W}_j \mathbf{J}_j}]^{-1} [\overline{\mathbf{J}_j^T \mathbf{W}_j \mathbf{Q}_j} - \overline{\mathbf{J}_j^T \mathbf{W}_j}\ \overline{\mathbf{W}_j}\ \overline{\mathbf{W}_j \mathbf{Q}_j}] \\ &= \mathbf{N}^{-1} [\overline{\mathbf{J}_j^T \mathbf{W}_j \mathbf{Q}_j} - \overline{\mathbf{J}_j^T \mathbf{W}_j}\ \overline{\mathbf{W}_j}\ \overline{\mathbf{W}_j \mathbf{Q}_j}] . \end{aligned} \tag{B.9}$$

As in the least-squares case, we obtain a solution for the optimal translation in terms of the optimal rotation. Since this solution is based on linearizing about an initial set of angles $\Theta_0$, we iterate the solution for $\Theta$ until this converges, then compute $\mathbf{T}$ as the final step of the solution. Expanding $\mathbf{Q}_j$ in (B.8), the solution for $\mathbf{T}$ can be re-expressed as

$$\begin{aligned} \hat{\mathbf{T}} &= \overline{\mathbf{W}_j}^{-1} [\overline{\mathbf{W}_j (\mathbf{Q}_{cj} - \mathbf{R}_0 \mathbf{Q}_{pj} + \mathbf{J}_j \Theta_0)} - \overline{\mathbf{W}_j \mathbf{J}_j} \widehat{\Theta}] \\ &= \overline{\mathbf{W}_j}^{-1} [\overline{\mathbf{W}_j (\mathbf{Q}_{cj} - \mathbf{R}_0 \mathbf{Q}_{pj})} + \overline{\mathbf{W}_j \mathbf{J}_j} (\widehat{\Theta} - \Theta_0)] . \end{aligned}$$

If that $\widehat{\Theta}$ has converged, then $\widehat{\Theta} - \Theta_0 \approx 0$ and $\hat{\mathbf{T}}$ reduces to

$$\hat{\mathbf{T}} = \overline{\mathbf{W}_j}^{-1}\overline{\mathbf{W}_j(\mathbf{Q}_{cj} - \hat{\mathbf{R}}\mathbf{Q}_{pj})}. \tag{B.10}$$

The covariance matrix can be computed from terms formed in the solution for $\Theta$ and $\mathbf{T}$. The expression for the covariance matrix expands to

$$\Sigma_{\mathbf{M}} = \begin{bmatrix} \overline{\mathbf{J}_j^T\mathbf{W}_j\mathbf{J}_j} & \overline{\mathbf{J}_j^T\mathbf{W}_j} \\ \overline{\mathbf{W}_j\mathbf{J}_j} & \overline{\mathbf{W}_j} \end{bmatrix}^{-1}. \tag{B.11}$$

The components of this matrix all appear as terms in (B.9) and (B.10). Alternatively, if only some elements of the covariance matrix are desired, the inverse can be obtained symbolically using identities for inverting $2 \times 2$ block matrices [Graybill83]. The result is

$$\Sigma_{\mathbf{M}} = \begin{bmatrix} \mathbf{N}^{-1} & -\mathbf{N}^{-1}\overline{\mathbf{J}_j^T\mathbf{W}_j}\,\overline{\mathbf{W}_j}^{-1} \\ -\overline{\mathbf{W}_j}^{-1}\overline{\mathbf{W}_j\mathbf{J}_j}\mathbf{N}^{-1} & \overline{\mathbf{W}_j}^{-1} + \overline{\mathbf{W}_j}^{-1}\overline{\mathbf{W}_j\mathbf{J}_j}\mathbf{N}^{-1}\overline{\mathbf{J}_j^T\mathbf{W}_j}\,\overline{\mathbf{W}_j}^{-1} \end{bmatrix}. \tag{B.12}$$

Note that $\mathbf{N}$ appeared in the solution for $\Theta$. Therefore, if only components of the covariance matrix are needed, this gives a simpler form for computing them.

In summary, from (B.6),

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

we obtain the rotation via (B.9)

$$\begin{aligned} \widehat{\Theta} &= [\overline{\mathbf{J}_j^T\mathbf{W}_j\mathbf{J}_j} - \overline{\mathbf{J}_j^T\mathbf{W}_j}\,\overline{\mathbf{W}_j}\,\overline{\mathbf{W}_j\mathbf{J}_j}]^{-1}[\overline{\mathbf{J}_j^T\mathbf{W}_j\mathbf{Q}_j} - \overline{\mathbf{J}_j^T\mathbf{W}_j}\,\overline{\mathbf{W}_j}\,\overline{\mathbf{W}_j\mathbf{Q}_j}] \\ &= \mathbf{N}^{-1}[\overline{\mathbf{J}_j^T\mathbf{W}_j\mathbf{Q}_j} - \overline{\mathbf{J}_j^T\mathbf{W}_j}\,\overline{\mathbf{W}_j}\,\overline{\mathbf{W}_j\mathbf{Q}_j}]. \end{aligned}$$

This is iterated until $\widehat{\Theta} \approx \Theta_0$. Then the optimal translation is obtained from (B.10):

$$\hat{\mathbf{T}} = \overline{\mathbf{W}_j}^{-1}\overline{\mathbf{W}_j(\mathbf{Q}_{cj} - \hat{\mathbf{R}}\mathbf{Q}_{pj})}.$$

The error covariance matrix is given by (B.11) or (B.12).

Finally, an efficient method for computing the partial derivatives of $\mathbf{R}$ is given in [Mikhail76, p. 212]. Letting the rotations about each axis be

$$\begin{aligned} \mathbf{R}(\theta_x) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \\ \mathbf{R}(\theta_y) &= \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix} \\ \mathbf{R}(\theta_z) &= \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}, \end{aligned}$$

the partial derivatives are as follows:

$$\mathbf{R}_x = \mathbf{R}(\theta_z)\,\mathbf{R}(\theta_y)\,\frac{d(\mathbf{R}(\theta_x))}{d\theta_x} = \mathbf{R}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & +1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\mathbf{R}_y = \mathbf{R}(\theta_z)\,\frac{d(\mathbf{R}(\theta_y))}{d\theta_y}\,\mathbf{R}(\theta_x) = \mathbf{R}\begin{bmatrix} 0 & \sin\theta_x & \cos\theta_x \\ -\sin\theta_x & 0 & 0 \\ \cos\theta_x & 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & -\cos\theta_z \\ 0 & 0 & \sin\theta_z \\ \cos\theta_z & -\sin\theta_z & 0 \end{bmatrix}\mathbf{R}$$

$$\mathbf{R}_z = \frac{d(\mathbf{R}(\theta_z))}{d\theta_z}\,\mathbf{R}(\theta_y)\,\mathbf{R}(\theta_x) = \begin{bmatrix} 0 & +1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\mathbf{R}.$$

## B.3 Sequential Bayesian Estimation of Θ, T, and P

In this section we complete the derivation of the sequential Bayesian estimator presented in chapter 2. Definitions for notation used here can be found in chapter 2.

From equation (2.17), the MAP estimates of **M** and **P** maximize

$$\ell = \ln f = -\frac{1}{2}\left(\sum_j \mathbf{e}_{\mathbf{v}_j}^T\,\mathbf{W}_{\mathbf{v}_{cj}}\,\mathbf{e}_{\mathbf{v}_j} + \sum_j \mathbf{e}_{\mathbf{P}_j}^T\,\mathbf{W}_{\mathbf{P}_{cj}}^-\,\mathbf{e}_{\mathbf{P}_j} + \mathbf{e}_{\mathbf{M}}^T\,\mathbf{W}_{\mathbf{M}}^-\,\mathbf{e}_{\mathbf{M}}\right) + K, \tag{B.13}$$

where $K$ is constant. We will put this equation into matrix notation to simplify the derivation. First, we expand the prior motion term to reflect the linearization:

$$\begin{aligned}\mathbf{e}_{\mathbf{M}}^T\,\mathbf{W}_{\mathbf{M}}^-\,\mathbf{e}_{\mathbf{M}} &= \left[\mathbf{M} - \mathbf{M}_0 + \mathbf{M}_0 - \widehat{\mathbf{M}}^-\right]^T \mathbf{W}_{\mathbf{M}}^- \left[\mathbf{M} - \mathbf{M}_0 + \mathbf{M}_0 - \widehat{\mathbf{M}}^-\right] \\ &= [\mathbf{M} - \mathbf{M}_0]^T\,\mathbf{W}_{\mathbf{M}}^-\,[\mathbf{M} - \mathbf{M}_0] + 2\,[\mathbf{M} - \mathbf{M}_0]^T\,\mathbf{W}_{\mathbf{M}}^- \left[\mathbf{M}_0 - \widehat{\mathbf{M}}^-\right] + \\ &\quad \left[\mathbf{M}_0 - \widehat{\mathbf{M}}^-\right]^T \mathbf{W}_{\mathbf{M}}^- \left[\mathbf{M}_0 - \widehat{\mathbf{M}}^-\right].\end{aligned}$$

Then, by grouping the first of these terms with the second summation in (B.13), it can be shown that

$$\begin{aligned}\ell = -\frac{1}{2}\Big([\mathbf{Q} - \mathbf{P}]^T\,\mathbf{W}_{\mathbf{v}}\,[\mathbf{Q} - \mathbf{P}] + 2\,[\mathbf{M} - \mathbf{M}_0]^T\,\mathbf{W}_{\mathbf{M}}^- \left[\mathbf{M}_0 - \widehat{\mathbf{M}}^-\right] + \\ \left[\mathbf{M}_0 - \widehat{\mathbf{M}}^-\right]^T \mathbf{W}_{\mathbf{M}}^- \left[\mathbf{M}_0 - \widehat{\mathbf{M}}^-\right] + \begin{bmatrix} \mathbf{P} - \widehat{\mathbf{P}}^- \\ \mathbf{M} - \mathbf{M}_0 \end{bmatrix}^T \begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{P} - \widehat{\mathbf{P}}^- \\ \mathbf{M} - \mathbf{M}_0 \end{bmatrix}\Big)\end{aligned} \tag{B.14}$$

where

$$\mathbf{W}_{\mathbf{v}} = \begin{bmatrix} \mathbf{W}_{\mathbf{v}_{c1}} & & \\ & \ddots & \\ & & \mathbf{W}_{\mathbf{v}_{cn}} \end{bmatrix}$$

is the inverse covariance matrix of $\mathbf{Q}$ and

$$\begin{aligned}
\hat{\mathbf{P}}^- &= \begin{bmatrix} \hat{\mathbf{P}}_{c1}^- \\ \vdots \\ \hat{\mathbf{P}}_{cn}^- \end{bmatrix} \\
\mathbf{N}_{11} &= \begin{bmatrix} \mathbf{W}_{\mathbf{P}_{c1}}^- & & \\ & \ddots & \\ & & \mathbf{W}_{\mathbf{P}_{cn}}^- \end{bmatrix} \\
\mathbf{N}_{12} &= \begin{bmatrix} -\mathbf{W}_{\mathbf{P}_{c1}}^- \mathbf{H}_1 \\ \vdots \\ -\mathbf{W}_{\mathbf{P}_{cn}}^- \mathbf{H}_n \end{bmatrix} \\
\mathbf{N}_{21} &= \mathbf{N}_{12}^T \\
\mathbf{N}_{22} &= \mathbf{W}_{\mathbf{M}}^- + \sum_j \mathbf{H}_j^T \mathbf{W}_{\mathbf{P}_{cj}}^- \mathbf{H}_j .
\end{aligned}$$

Since (B.14) is quadratic in the unknowns, the MAP estimate is obtained by setting its derivates with respect to $\mathbf{P}$ and $\mathbf{M}$ to zero:

$$\begin{aligned}
\begin{bmatrix} \partial \ell / \partial \mathbf{P} \\ \partial \ell / \partial \mathbf{M} \end{bmatrix} &= -\begin{bmatrix} \mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{P} \\ \mathbf{M} \end{bmatrix} + \begin{bmatrix} \mathbf{W}_{\mathbf{v}}\mathbf{Q} \\ \mathbf{W}_{\mathbf{M}}^-(\widehat{\mathbf{M}}^- - \mathbf{M}_0) \end{bmatrix} + \begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{P}}^- \\ \mathbf{M}_0 \end{bmatrix} \\
&= 0 .
\end{aligned}$$

As with the maximum-likelihood formulation for $\mathbf{M}$ alone, the resulting linear system

$$\begin{aligned}
\begin{bmatrix} \mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{P} \\ \mathbf{M} \end{bmatrix} &= \begin{bmatrix} \mathbf{W}_{\mathbf{v}}\mathbf{Q} \\ \mathbf{W}_{\mathbf{M}}^-(\widehat{\mathbf{M}}^- - \mathbf{M}_0) \end{bmatrix} + \begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{P}}^- \\ \mathbf{M}_0 \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \end{bmatrix}
\end{aligned}$$

can be solved as a partitioned matrix to yield the following updated estimates of the unknowns:

$$\begin{aligned}
\widehat{\mathbf{M}}^+ &= \left[\mathbf{N}_{22} - \mathbf{N}_{21}(\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1}\mathbf{N}_{12}\right]^{-1} \left[\mathbf{K}_2 - \mathbf{N}_{21}(\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1}\mathbf{K}_1\right] \\
\hat{\mathbf{P}}^+ &= (\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1} \left[\mathbf{K}_1 - \mathbf{N}_{12}\mathbf{M}\right] .
\end{aligned}$$

It can be shown that the respective covariance matrices are

$$\begin{aligned}
\Sigma_{\mathbf{M}}^+ &= \left[\mathbf{N}_{22} - \mathbf{N}_{21}(\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1}\mathbf{N}_{12}\right]^{-1} \\
\Sigma_{\mathbf{P}}^+ &= (\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1} + (\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1}\mathbf{N}_{12}\Sigma_{\mathbf{M}}^+\mathbf{N}_{21}(\mathbf{W}_{\mathbf{v}} + \mathbf{N}_{11})^{-1} .
\end{aligned}$$

Note that $\hat{\mathbf{P}}^+$ is a vector containing *all* of the landmark coordinates and that $\Sigma_{\mathbf{P}}^+$ is the joint covariance of all of the landmarks. We will expand these expressions into estimates for individual landmarks shortly.

By application of the matrix inversion lemma [Maybeck79], it can be shown that the solutions for the motion parameters $\widehat{\mathbf{M}}^+$ and their covariance $\Sigma_{\mathbf{M}}^+$ are equal to

$$\widehat{\mathbf{M}}^+ = \left[\mathbf{W}_{\mathbf{M}}^- + \sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{H}_j\right]^{-1} \left[\mathbf{W}_{\mathbf{M}}^- \widehat{\mathbf{M}}^- + \sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{Q}_j\right]$$

$$\Sigma_{\mathbf{M}}^+ = \left[\mathbf{W}_{\mathbf{M}}^- + \sum_j \mathbf{H}_j^T \mathbf{W}_j \mathbf{H}_j\right]^{-1},$$

where $\mathbf{W}_j = (\Sigma_{\mathbf{P}_{cj}}^- + \Sigma_{\mathbf{v}_{cj}})^{-1}$ and $\mathbf{Q}_j = \mathbf{Q}_{cj} - \mathbf{R}_0 \hat{\mathbf{P}}_{pj}^+ + \mathbf{J}_j \Theta_0$. These expressions differ from the counterpart expressions in the maximum-likelihood motion estimator only by the addition of the terms for the prior information. The partitioned solution applied in the maximum-likelihood case can be applied here as well.

Assuming that the solution for $\widehat{\mathbf{M}}^+$ is iterated to convergence, the solution for $\hat{\mathbf{P}}^+$ can be expanded into the following expression for the individual landmark coordinates:

$$\hat{\mathbf{P}}_{cj}^+ = (\mathbf{W}_{\mathbf{v}_{cj}} + \mathbf{W}_{\mathbf{P}_{cj}}^-)^{-1} (\mathbf{W}_{\mathbf{v}_{cj}} \mathbf{Q}_{cj} + \mathbf{W}_{\mathbf{P}_{cj}}^- \hat{\mathbf{P}}_{cj}^-),$$

where $\hat{\mathbf{P}}_{cj}^- = \hat{\mathbf{R}}^+ \hat{\mathbf{P}}_{pj}^+ + \hat{\mathbf{T}}^+$ is computed using the optimal motion estimate $\widehat{\mathbf{M}}^+$. Finally, the covariances of individual landmark estimates appear as $3 \times 3$ sub-matrices on the diagonal of $\Sigma_{\mathbf{P}}^+$. Expanding $\Sigma_{\mathbf{P}}^+$, the landmark covariances are

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

These equations are interpretted in chapter 2.

## B.4 Posterior Estimate of the Reference Variance

The estimation and error detection procedures described here rely on an accurate model of the statistical uncertainty in the measured image coordinates $\mathbf{q}_{l_{i,j}}$ and $\mathbf{q}_{l_{i,j}}$. As described in chapter 4, the covariance matrices of these measurements are directly proportional to the variance $\sigma^2$ of the noise in the image. Moreover, the covariance matrices of estimated parameters will also be directly proportional to $\sigma^2$; therefore, $\sigma^2$ functions as a global scale factor on the entire uncertainty model. In photogrammetry, this is referred to as the *reference variance*. Knowledge of $\sigma^2$ is not necessary for computing the parameter estimates, but it is needed whenever correctly scaled covariance matrices are necessary, such as in the error detection procedures discussed in appendix B.5 or in motion planning algorithms that incorporate uncertainty models. Therefore, methods for automatically estimating (or verifying previous estimates) of $\sigma^2$ are desirable.

In this section, we will show how techniques discussed in [Mikhail76] can be used to estimate $\sigma^2$ from the results of the maximum-likelihood formulation of the motion estimation problem. In section 2.3.2, we showed that an estimate $\widehat{\mathbf{M}}$ of the motion parameters can be computed from observations of the form

$$\mathbf{Q}_j = \mathbf{H}_j \mathbf{M} + \mathbf{v}_j,$$

where $\mathbf{v}_j$ is modelled as zero-mean, Gaussian noise with covariance $\Sigma_j$. Having $\widehat{\mathbf{M}}$, the *residual* observation error is

$$\hat{\mathbf{v}}_j = \mathbf{Q}_j - \mathbf{H}_j\widehat{\mathbf{M}}.$$

It can be shown that the quadratic form

$$q = \sum_j \hat{\mathbf{v}}_j^T \Sigma_j^{-1} \hat{\mathbf{v}}_j$$

is a random variable with the $\chi^2$ distribution with $r = N - m$ degrees of freedom, where $N$ is the total number of observations and $m$ is the number of parameters estimated. Since we observe 3-D coordinate vectors for $n$ landmarks in two successive images, $N = 6n$, $m = 6$, and $r = 6n - 6$. From the properties of the $\chi^2$ distribution, $q$ has mean $\mathrm{E}[q] = r$ and variance $\mathrm{Var}[q] = 2r$. If an incorrect estimate of $\sigma^2$ is used, say $\sigma_0^2 = \sigma^2/k$, then $\mathrm{E}[q] = kr$ and $\mathrm{Var}[q] = 2k^2r$. Therefore, an estimate of $\sigma^2$ is given by

$$\hat{\sigma}^2 = \hat{k}\sigma_0^2 = \frac{q}{r}\sigma_0^2.$$

In practice, this can be used as follows. If $\sigma^2$ is initially unknown, we can define $\sigma_0^2 = 1$. We then process the first two stereo pairs of the image sequence, compute $\hat{\sigma}^2$, and rescale the covariance matrices as necessary. In doing so, note that $\hat{\sigma}^2$ is itself a random variable with variance $2\sigma^4/r$. Therefore, if some prior knowledge of $\sigma^2$ is available, this can be used to determine the number of landmarks necessary to make $r$ large enough to obtain a statistically reliable estimate $\hat{\sigma}^2$. An additional difficulty occurs if the observations contain correspondence errors, since these will cause an over-estimate of the noise. This issue is addressed in [Gennery80]; however, we will not consider it here.

## B.5 Error Detection

The estimators derived in chapter 2 and in this appendix assume that statistical uncertainty in the obervations is due only to zero-mean, Gaussian noise in the measured image coordinates. Gross errors in tracking landmarks (i.e. *correspondence errors*) violate this assumption. In this section we describe two procedures that can be used to filter out such errors. The first is a *rigidity test* that does not require knowledge of the motion parameters; therefore, this test is used to filter out wild observations before invoking the estimation procedures. The second is an *outlier test* that examines residual observation errors after the motion has been estimated. This test is used as part of a data editting loop that repeatedly estimates the motion parameters, then deletes the observation with the largest residual, until all residuals are within a threshold.

### B.5.1 Rigidity Test

Because we assume that the robot's environment is stationary, distances between pairs of landmarks cannot change over time. The rigidity test enforces this constraint by rejecting landmarks

that appear to move relative to other landmarks. The test used in the implementation was developed by Moravec [Moravec80]; however, it did not embody the statistical model developed for the rest of the system. We will describe Moravec's test first, then describe extensions that incorporate the statistical model.

As before, let $\mathbf{Q}_{pj}$ and $\mathbf{Q}_{cj}$ denote the landmark observations made relative to the previous and the current coordinate frames, respectively. For a given pair of landmarks $i$ and $j$, the rigidity constraint implies that the observed distance

$$d_{pij} = \sqrt{(\mathbf{Q}_{pi} - \mathbf{Q}_{pj})^T(\mathbf{Q}_{pi} - \mathbf{Q}_{pj})} \tag{B.15}$$

between the landmarks at the previous time must be the same as the observed distance

$$d_{cij} = \sqrt{(\mathbf{Q}_{ci} - \mathbf{Q}_{cj})^T(\mathbf{Q}_{ci} - \mathbf{Q}_{cj})} \tag{B.16}$$

between them currently. This can be checked without knowing the transformation between the previous and current coordinate frames; therefore, correspondence errors or true non-rigidity (caused by the presence of moving objects, for example) can be detected before estimating motion by checking the inter-landmark distances. This test can be confounded by observation noise and by the fact that one landmark can rotate about another without changing the distance between them. Moravec moderated these effects requiring consistency between each landmark and all the others. Letting the change in distance between landmarks $i$ and $j$ be

$$c_{ij} = d_{pij} - d_{cij},$$

a measure of agreement between landmark $i$ and all others was obtained by summing the squared changes over all other landmarks:

$$a_i = \sum_j w_{ij} c_{ij}^2. \tag{B.17}$$

As in chapter 2, the weights $w_{ij}$ were scalar models of uncertainty in the relevant observations. The landmark $i$ for which $a_i$ was maximum was judged to be most discordant; if this $a_i$ also exceeded a threshold, the landmark was rejected. This was repeated until all $a_i$ were within the threshold. Since $O(n)$ distances are computed for each landmark, each iteration of this procedure requires $O(n^2)$ time.

This test is effective, but it does not explicitly incorporate the model of observation uncertainty developed in chapter 2. This makes it difficult to choose the threshold. It also suggests that the test may not work well for distant landmarks because it fails to model eccentric error distrubtions, just as the motion estimator failed in this case. Deriving a probability distribution for $a_i$ can resolve both of these concerns. We take a step in this direction by deriving a Gaussian approximation for $c_{ij}$ and basing a test on this distribution. Questions about the performance of the test, or the existence of better tests, will be left for the future.

Let the difference vectors between pairs of observations be

$$\begin{aligned} \mathbf{D}_{pij} &= \mathbf{Q}_{pi} - \mathbf{Q}_{pj} \\ \mathbf{D}_{cij} &= \mathbf{Q}_{ci} - \mathbf{Q}_{cj} \end{aligned}$$

with covariance $\Sigma_{pij} = \Sigma_{\mathbf{v}_{pi}} + \Sigma_{\mathbf{v}_{pj}}$ and $\Sigma_{pij} = \Sigma_{\mathbf{v}_{pi}} + \Sigma_{\mathbf{v}_{pj}}$, respectively. Given that the observations are noisy, the measured distances $d_{pij}$ and $d_{cij}$ are samples from probability distributions. We approximate these distributions as Gaussian, with variances $\sigma^2_{pij}$ and $\sigma^2_{cij}$ obtained by linearizing about the observed difference vectors. From (B.15) and (B.16), we obtain

$$\sigma^2_{pij} \approx \frac{1}{d^2_{pij}} \mathbf{D}^T_{pij} \Sigma_{pij} \mathbf{D}_{pij} \tag{B.18}$$

$$\sigma^2_{cij} \approx \frac{1}{d^2_{cij}} \mathbf{D}^T_{cij} \Sigma_{cij} \mathbf{D}_{cij} . \tag{B.19}$$

If the two landmarks are rigid, then the means of the distributions will be approximately equal. Therefore, the distance change $c_{ij}$ is approximately zero-mean Gaussian with variance $\sigma^2_{ij} = \sigma^2_{pij} + \sigma^2_{cij}$. Making the further approximation that $c_{ij_1}$ and $c_{ij_2}$ are independent for $j_1 \neq j_2$, the sum

$$a_i = \sum_j \frac{c^2_{ij}}{\sigma^2_{ij}} = \sum_j \frac{c^2_{ij}}{\sigma^2_{pij} + \sigma^2_{cij}} \tag{B.20}$$

is approximately $\chi^2$ with $n-1$ degrees of freedom, where $n$ is the number of landmarks. For given levels of significance, thresholds for the acceptability of computed values of $a_i$ can be obtained from standard tables for the $\chi^2$ distribution. Because the derivation is approximate, thresholds obtained this way may need to be tuned by hand.

Comparing (B.17) and (B.20), the new test is equivalent to Moravec's with $w_{ij} = 1/(\sigma^2_{pij} + \sigma^2_{cij})$ and with the threshold obtained with the aid of a $\chi^2$ table. Since $w_{ij}$ now reflects the 3-D observation uncertainty through the error propagations of equations (B.18) and (B.19), the new test should be more effective. By employing the previous landmark estimates $\hat{\mathbf{P}}^+_{pj}$ instead of the previous observations $\mathbf{Q}_{pj}$, it also lets us take advantage of the increasing precision of the landmark model over time. Further experiments will be needed to determine how well the test performs in practice.

## B.5.2 Outlier Test

If the number of correspondence errors is a small percentage of the number of landmarks, errors can be detected from outlying residuals after estimating the motion parameters. This leads to a test a procedure that estimates the motion, rejects the observation with the largest residual, and iterates these two steps until the largest residual is within a fixed threshold. The test described here employs *studentized* residuals [Barnett84]. To make the presentation clear, we will first review the basic mathematical development in [Barnett84], then show how this applies to our problem.

### Studentized residuals

This technique is appropriate in linear estimation problems with Gaussian noise. The basic procedure involves estimating the covariance of the residuals, then applying a threshold to the

quadratic forms of the residuals weighted by their inverse covariance matrices. Suppose an unknown $m \times 1$ parameter vector $\mathbf{M}$ is related to an $n \times 1$ vector of observations $\mathbf{Q}$ via the $n \times m$ observation matrix $\mathbf{H}$ and an $n \times 1$ noise vector $\mathbf{v}$ as follows:

$$\mathbf{Q} = \mathbf{HM} + \mathbf{v}, \tag{B.21}$$

where $\mathbf{v} \sim N(\mathbf{0}, \Sigma_{\mathbf{v}})$. The maximum likelihood estimate of $\mathbf{M}$ is given by

$$\widehat{\mathbf{M}} = (\mathbf{H}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{Q}. \tag{B.22}$$

The covariance of $\widehat{\mathbf{M}}$ is $\Sigma_{\mathbf{M}} = (\mathbf{H}^T \Sigma_{\mathbf{v}}^{-1}$. The residuals are obtained from (B.21) and (B.22) as

$$\begin{aligned} \hat{\mathbf{v}} &= \mathbf{Q} - \mathbf{H}\widehat{\mathbf{M}} \\ &= \mathbf{Q} - \mathbf{H}(\mathbf{H}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{Q} \\ &= [\mathbf{I} - \mathbf{H}(\mathbf{H}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \Sigma_{\mathbf{v}}^{-1}] \mathbf{Q} \\ &= [\mathbf{I} - \mathbf{A}] \mathbf{Q}. \end{aligned} \tag{B.23}$$

Using (B.23) and the fact that $\mathbf{Q}$ is Gaussian with mean $\mathbf{HM}$ and covariance $\Sigma_{\mathbf{v}}$, it is easy to show that the distribution of $\hat{\mathbf{v}}$ is zero-mean Gaussian with covariance

$$\begin{aligned} \Sigma_{\hat{\mathbf{v}}} &= \Sigma_{\mathbf{v}} - \mathbf{H}(\mathbf{H}^T \Sigma_{\mathbf{v}}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \\ &= \Sigma_{\mathbf{v}} - \mathbf{H} \Sigma_{\mathbf{M}} \mathbf{H}^T. \end{aligned}$$

The variances of individual components $\hat{v}_i$ of the residual vector are the diagonal elements of $\Sigma_{\hat{\mathbf{v}}}$; these are given by

$$\sigma_{\hat{v}_i}^2 = \sigma_{\hat{v}_i}^2 - \mathbf{H}_i \Sigma_{\mathbf{M}} \mathbf{H}_i^T, \tag{B.24}$$

where $\mathbf{H}_i$ is the $i^{th}$ row of $\mathbf{H}$. When we consider the observation vector $\mathbf{Q}$ to consist of scalar observations $q_i$, the *studentized residuals* $e_i$ are obtained by dividing components of the residual vector by their standard deviations,

$$e_i = \frac{\hat{v}_i}{\sigma_{\hat{v}_i}}.$$

To test for outliers, we find the studentized residual of largest magnitude and reject the corresponding measurement if the residual exceeds a threshold. The threshold is often chosen to be some factor $t$ times the standard deviation of the residual. The intuitive rationale for studentizing is now clear: the residuals will have different standard deviations, so dividing these out allows a constant threshold $t$ to be used for all residuals. The test is to reject measurement $i$ if

$$\tau = \max_i |e_i| = \max_i \frac{\hat{v}_i}{\sigma_{\hat{v}_i}} > t.$$

Barnett [Barnett84, p. 299] justifies this procedure formally in terms of a likelihood ratio test. Briefly, rejecting the observation with the largest studentized residual leads to the largest increase in the likelihood of the remaining sample.

When the individual observations constituting $\mathbf{Q}$ are not scalars, but k-dimensional vectors $\mathbf{Q}_i$, the foregoing procedure can be generalized by using the corresponding k-dimensional components $\hat{v}_i$ of the residual vector and the respective submatrices $\Sigma_{\hat{v}_i}$ of $\Sigma_{\hat{v}}$. The submatrices $\Sigma_{\hat{v}_i}$ can be computed, without first computing all of $\Sigma_{\hat{v}}$, by extending (B.24) to the k-D case:

$$\Sigma_{\hat{v}_i} = \Sigma_{v_i} - \mathbf{H}_i \Sigma_{\mathbf{M}} \mathbf{H}_i^T, \tag{B.25}$$

where $\mathbf{H}_i$ is $k \times m$. Now the vector $\hat{v}_i$ is normalized (studentized) by taking the quadratic form

$$e_i^2 = \hat{v}_i^T \Sigma_{\hat{v}_i}^{-1} \hat{v}_i \tag{B.26}$$

and the rejection test applied is

$$e_i^2 > t^2.$$

This form of the test was used for 2-D vectors of image coordinates to detect correspondence errors in Gennery's stereo camera calibration algorithm [Gennery80]. A particular value for $t$ is chosen based on the probability that $e_i^2$ will exceed $t^2$. If we choose to reject all points such that

$$P(t^2 < e_i^2) < \alpha,$$

then $t = \chi^2_{\alpha,k}$ is the desired value of $t$, where $\chi^2_{\alpha,k}$ is the $\alpha$ significance level of the $\chi^2$ distribution with $k$ degrees of freedom.

### Application to motion estimation

From the foregoing discussion, it is clear how the method applies to outlier detection in the maximum-likelihood formulation of the motion estimation problem. The linearized motion equation (B.6),

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

has the same structure as we used in describing the outlier test. The observations $\mathbf{Q}_j$ are three-dimensional with error covariance matrices $\Sigma_{\mathbf{v}_j} = \Sigma_{\mathbf{v}_{cj}} + \mathbf{R}_0 \Sigma_{\mathbf{v}_{pj}} \mathbf{R}_0^T$, as computed in chapter 2 and appendix A. Expressions for $\widehat{\mathbf{M}}$ and $\Sigma_{\widehat{\mathbf{M}}}$ are given in section B.2. The residuals are simply

$$\begin{aligned} \hat{\mathbf{v}}_j &= \mathbf{Q}_j - \mathbf{H}_j \widehat{\mathbf{M}} \\ &= \mathbf{Q}_{cj} - \hat{\mathbf{R}} \mathbf{Q}_{pj} - \hat{\mathbf{T}} \end{aligned}$$

and the respective covariances are exactly as given in equation (B.25):

$$\Sigma_{\hat{v}_j} = \Sigma_{v_j} - \mathbf{H}_j \Sigma_{\mathbf{M}} \mathbf{H}_j^T.$$

The data editting loop then consists of the following steps:

- Compute $\widehat{\mathbf{M}}$ and $\Sigma_{\widehat{\mathbf{M}}}$.
- Compute the residuals $\hat{v}_i$ and covariance matrices $\Sigma_{\hat{v}_i}$ as above.
- Compute the quadratic forms $e_i$ of (B.26).
- If the largest $e_i$ exceeds the threshold, delete the observation and repeat the loop. Otherwise, terminate.

Extensions to the sequential estimation formulation are straightforward.

See [Gennery80] for a more elaborate editting loop [Barnett84,Hawkins80] for additional discussion of the outlier detection problem.