Gauge Freedoms and Uncertainty Modeling for 3D Computer Vision

Daniel D. Morris

March 2001

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

Thesis Committee:
Takeo Kanade, Chair
Martial Hebert
Robert Collins
Kenichi Kanatani, Okayama University
Carlo Tomasi, Stanford University

Keywords: Gauge Orbits, Structure from Motion, Computer Vision, Indeterminacy, Covariance, Estimation

Abstract

Parameter indeterminacies are inherent in 3D computer vision. We show in this thesis that how they are treated can have significant impact on the accuracy of the estimated 3D structure. However, there has not been a general and convenient method available for representing and analyzing the indeterminacies and their effects on accuracy. Consequently, up to the present their effects are usually ignored in uncertainty modeling research.

In this work we a develop gauge-based uncertainty representation for 3D estimation that includes indeterminacies. We represent indeterminacies with orbits in the parameter space and model local linearized parameter indeterminacies as gauge freedoms. Combining this formalism with first order perturbation theory, we are able to model uncertainties along with parameter indeterminacies.

The key to our work is a geometric interpretation of the parameters and gauge freedoms. We solve the problem of how to compare parameter uncertainties despite indeterminacies and added constraints. This permits us to extend the Cramer-Rao lower bound to problems that include parameter indeterminacies.

In 3D computer vision the basic quantities that often cannot be recovered include scale, rotation and translation. We use our method to analyze the local effects of these indeterminacies on the estimated shape, and find all the local gauge freedoms. This enables us to express the uncertainties when additional information is available from measurements that constrain the gauge freedoms.

Through analytical and empirical means we gain intuition into the effects of constraining the gauge freedoms, for both general Structure from Motion and stereo shape estimation. We include, in our uncertainty model, measurement errors and feature localization errors. These results along with our theory allow us to find optimal constraints on the gauge freedoms that maximize the accuracy of the part of the object we seek to estimate.

Acknowledgements

Foremost I want to thank my wife, Felicia, for her support and encouragement during the last year of thesis work and for her interest and patience in listening to all my latest ideas. She is the perfect partner in so many ways.

I would like to thank my committee members for all their input. Takeo Kanade's enthusiasm and positive outlook was a source of inspiration, and his insightful comments helped guide my path. Working for a summer with professor Kanatani and collaboration since then has lead directly to large portions of the thesis. He provided many ideas and was a great help in mathematically formalizing them. Martial Hebert, Bob Collins and Carlo Tomasi all gave helpful comments and suggestions.

Others with whom I have had good and enlightening conversations during the thesis years include Jianbo Shi, David Larose, Mei Han, Chris McGlone, Santosh Mathan, Naoki Chiba, Teck Khim Ng, Sidd Puri, Greg Aist and Kerry Ojakian.

Finally, I want to recognize my parents, David and Angela, and my brothers, Nathan and Nigel, for always being a great support and family.

Contents

List of Figures

List of Tables

1. Introduction

Reconstruction of the three dimensional (3D) shape of an object from multiple images is one of the central areas of computer vision. Many new opportunities and tasks become possible when 3D information about the world is available. Being able to estimate the 3D structure enables an observer to assess its environment and perform tasks, such as navigation and obstacle avoidance, that depend on knowing distances between objects. It enables virtual 3D models to be created automatically and used by other tools such as CAD programs, or for mixed reality where real and virtual models are superimposed. In these and other applications, the more accurate the 3D estimates, the more powerfully they can be used. The focus of this thesis is to investigate how to evaluate the accuracy of 3D estimates when indeterminacies play a role, and to find out how to remove indeterminacies in a way that maximizes accuracy.

For our applications, 3D reconstruction can be seen as a means of obtaining accurate measurements of the world. In this context, 3D imaging is viewed as a Euclidean measurement tool for physical objects. The two key properties that make computer vision attractive for this are that imaging can be performed at a distance from the subject being measured, and secondly that it is a passive tool that is non-invasive and does not alter the object being measured. Ancient civilizations noticed this benefit of imaging, and used clever techniques to make quantitative measurements of the world. The classic example is trigonometry, where elevation angles and/or distance measurements are used to calculate the height of a difficult-to-measure object as illustrated in Figure 1.1. The field of photogrammetry has been developed with the goal of using images to obtain measurements of the world. Cartography is a clear example of images being used directly to obtain maps and hence measurements of the world. With the advent of digital imaging, more powerful 3D modeling tools have become available enabling complicated 3D structures to be modeled, including ancient archeological sites [58] to high accuracy. Thus 3D vision is interpreted as a tool for obtaining *accurate* 3D estimates or measurements of the world from images.

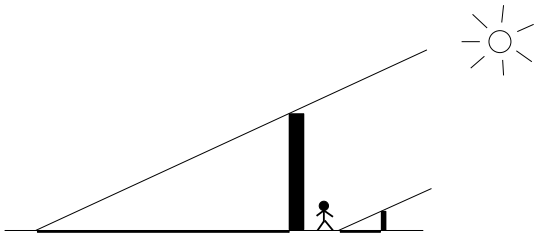


Fig. 1.1. Imaging methods such as that illustrated here have been used since ancient civilizations to obtain 3D measurements of the world. The shadow of an object acts as an image of it. An estimate of the height of a tall tower can be obtained by comparing the length of its shadow to the length of the shadow of an object whose height is known. We would like to use modern computer vision methods to obtain more accurate shape and size estimation.

1.1 Uncertainties and Indeterminacies in Computer Vision

Assume we are able to make 3D measurements of the world, and that these measurements are not perfectly accurate but contain errors. Of importance almost as great, if not as great, as the measurement itself, is an accuracy measure of the measurements. The accuracy measure tells us how to interpret the measurement. For instance, if someone were to report his height as seven and a half feet, one might recommend him to a basketball team. But if he added that this measurement is only accurate up to plus or minus three feet, then one would probably first ask for a more precise measurement. In the worst case if we know nothing about the accuracy and we assume all errors are equally likely, then the measurement gives us no information. Thus on a most basic level, an accuracy measure, or synonymously *uncertainty model*, tells us the "meaningfulness" of a measurement.[1]

Beyond a basic meaningfulness question, there are further reasons for wanting an uncertainty model. The usefulness of a measurement for different applications depends very strongly on the accuracy of the measurement. This is clearly the case in the construction and architecture industries which require precise measurements. But we can only say that a measurement is precise if we have an idea of its uncertainty. Thus implicitly all precise measurements contain an uncertainty model, even if it is as rough as an accuracy up to the number of significant figures.

[1] Photogrammetrists often distinguish "precision," which is measured by variance, from true "accuracy" which includes effects of systematic and gross errors. In this work, however, we assume systematic and gross errors have been removed, and so use precision and accuracy synonymously.

Uncertainty models are also useful in comparing reliability of different measurements. They let us select the most accurate measurement, or they may enable us to appropriately combine multiple measurements and obtain a more accurate estimate.

Let us say our goal is to most accurately estimate a dimension of an object given a set of measurements of the object and the uncertainties of these. There are various methods for combining the measurements to obtain a final estimate. Let us define the optimal solution to be the one that combines these to obtain the most accurate final estimate. Now there may be many optimal solution techniques, but there is only one accuracy bound, and this accuracy bound is the uncertainty of any optimal solution method. Hence in this sense uncertainty modeling is independent of and more general than any particular solution technique.

In imaging there are some inherent aspects that make uncertainty representation of 3D reconstruction difficult. This is due to a number of key quantities either being lost in the projection, or else being unknown. Light travels in straight lines and so every pixel in an image corresponds to a ray incident on the image plane. In a single image, the source of each ray of light can be at any distance from the image plane along the ray. We say that this 3D point has a position *indeterminacy* along this ray. When two or more images of these light sources, which we call features, are observed from different positions, this projection indeterminacy is reduced, and the 3D points can be determined. That is, they can almost be determined; some basic indeterminacies still remain. The classic indeterminacy is scale. It is always possible for the images to have been formed by a smaller object closer to the cameras or a larger object further from the cameras, as illustrated in Figure 1.2. In addition, while relative positions and orientations can be obtained for the camera and object, the absolute position and orientation remain unknown. These lost or unknown quantities show up as indeterminacies in the parameters.

The standard procedure for evaluating accuracy is first order error propagation. A local linearization of the equations is performed around the solution, and bias and covariance terms are estimated. But in order to perform error propagation, indeterminacies in the parameters must be removed by constraining the solution. For example, we can affix the coordinate system to points on the object and normalize the size of the object. However, the choice of normalizing constraints will significantly affect the final uncertainty model. Figure 1.3 illustrates in 2D how choice of coordinate systems and scale affects the accuracy of the estimated object shape and camera motions. This is a problem also faced in other domains, including 3D registration, where uncertainties or errors need to be compared, but where normalizing can act to dramatically increase or decrease

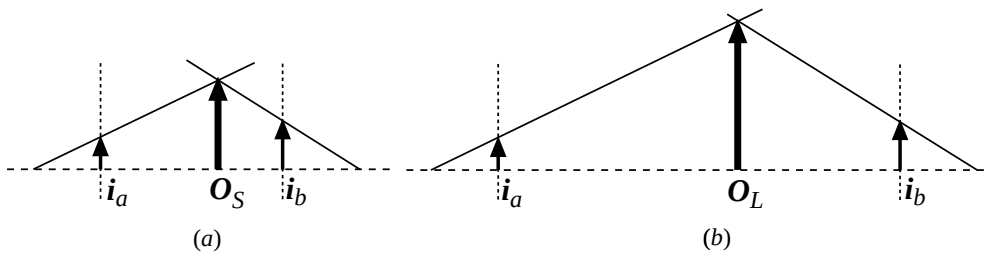


Fig. 1.2. (a) A small object, $\boldsymbol{O}_S$, projects onto the image planes of two cameras to form images, $\boldsymbol{i}_a$ and $\boldsymbol{i}_b$. (b) A large object $\boldsymbol{O}_L$, but further from the two cameras, projects onto the image planes to form the same images, $\boldsymbol{i}_a$ and $\boldsymbol{i}_b$. This ambiguity holds for all objects of all sizes and for multiple cameras, and thus results in the basic scale indeterminacy of Structure from Motion.

the error depending how it is performed. There is the potential to cleverly choose a coordinate system that normalizes away much of the error and makes comparisons between different methods unfair.

In this thesis we will look into how indeterminacies affect the solution, and the accuracy of the solution, for 3D shape reconstruction. We will borrow the term "Gauge Theory" from physics, and extend and apply the ideas to the computer vision domain, as was first done by Triggs [70]. This will allow us to treat indeterminacies in a formal way, and extract the "true quantities" inherent in 3D models that contain indeterminacies. We will show that these true quantities are invariant to the indeterminacies. Furthermore, we will encode constraints in this formalism, and demonstrate their consequences on the uncertainty. Since constraints can be obtained by making measurements of the real world, our representation will give us a quantitative guide for how best to eliminate indeterminacies.

The approach in this thesis will be to first give a theoretical derivation and explanation of gauge theory in Chapter 2. This will be sufficiently general to apply to many estimation problems containing indeterminacies. Then in Chapter 3 we will apply this to the domain of Structure from Motion, (SFM), and show how its indeterminacies are encoded. In Chapter 4 we will derive the consequences of adding constraints to the solution, and show how these affect accuracy. This will allow us to look at the stereo problem in Chapter 5, and make quantitative statements about its accuracy. Finally we will sum up our theory and its consequences.

Before we embark on this endeavor, we will define and explain some of our assumptions and formalisms. We will also place this work in its context of computer vision research.

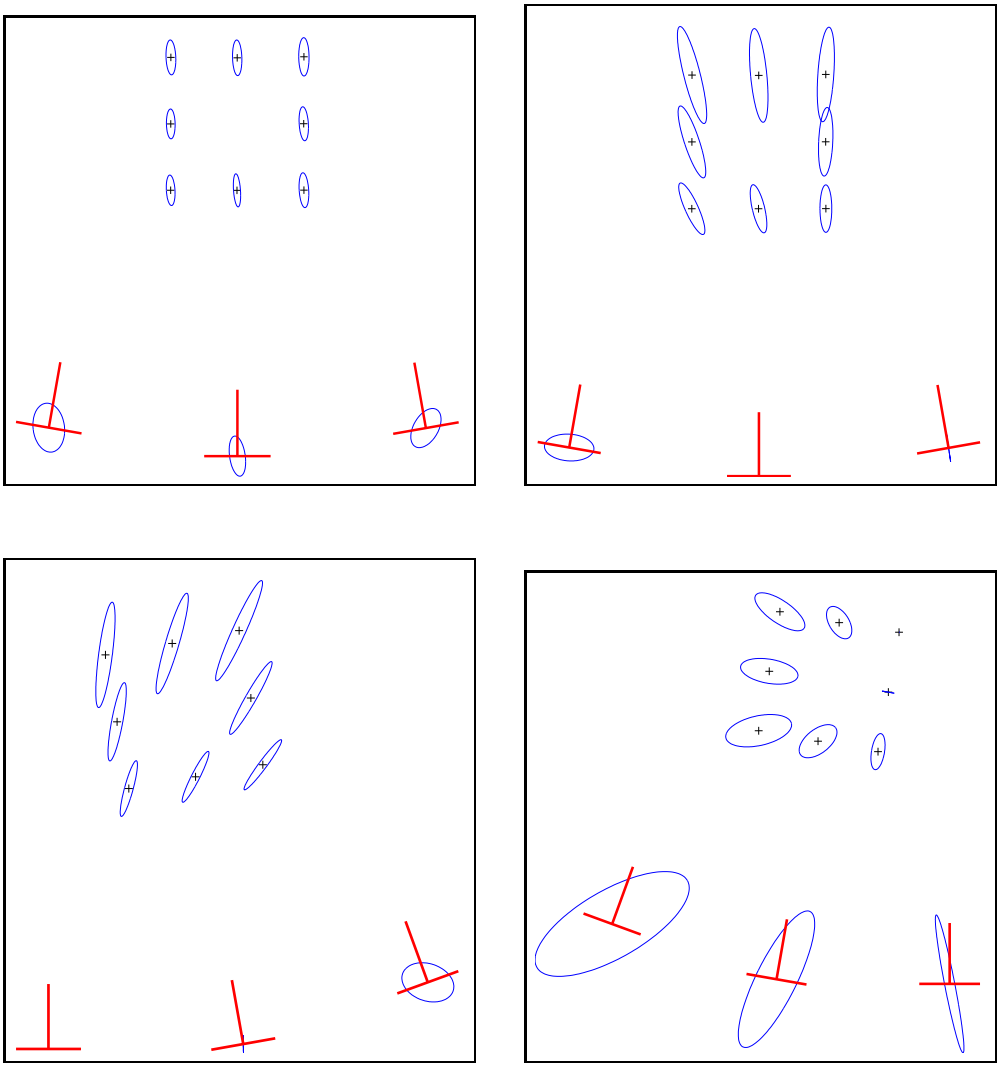


Fig. 1.3. Illustrated are eight points whose positions in the 2D plane are estimated using 3 cameras. The camera positions and orientations, shown by the "$\perp$" symbol, are also estimated. The ellipses show the translational uncertainty of each feature point and each camera position. This 2D reconstruction problem has 4 degrees of freedom corresponding to translation in the plane, scale and rotation. To plot the reconstruction and uncertainties requires us to choose a reference by constraining these degrees of freedom. The choice of coordinates affects the relative positions of the features, as we would expect, however, the uncertainties may be affected in non-intuitive ways, as illustrated by how the ellipses vary between plots. (a) Shown is the "normal covariance", (b) here the coordinate system is fixed to the middle camera, (c) the coordinate system is fixed to the left camera, and (d) the coordinate system is attached to a point on the object. In this thesis we analyze how the coordinate system, or more generally the reference frame, affects these uncertainties, and how all four of these plots can be said to be equivalent.

1.2 Object Modeling

Our starting point is to use multiple images to estimate the 3D structure of the world. For convenience we shall call the portion of the world in which we are interested, the *object*. We shall use object *shape* to mean the 3D geometry of a set of points on its surface. Since we will use multiple images of a static object from different view-points, we will typically speak of a single camera moving between the view-points, although for our purposes it would be equivalent to have multiple cameras at the different view-points. Camera *motion* will thus mean the set of positions and orientations of the camera at which it took each image of the object. We note that our equations can equally well describe an object moving in front of a camera, and so we will interchangeably use camera-centered coordinates with a moving object, and object-centered coordinates with a moving camera, as is convenient.

Of primary importance is to define what we mean by "3D modeling." We take a *model* of an object to mean a set of statements about the properties of the object. This can include statements about the dimensions of the object, the position or relative position of features on it, surface specifications including orientation and texture, and so forth. In this thesis we will deal with purely geometric quantities, and use point features as our primitives. The work can be extended to other primitives such as lines and planes [51, 60, 65]. Surface properties including color and texture will not affect our geometric statements, and so we ignore these. Since we will be using a camera to image the object, we will include the camera, and its motion and orientation, in our 3D model. Our model will thus consist of a set of geometric statements about feature points on the object, and the motion of the camera.

The more restrictive the assumptions we make about our model, the easier it will be to estimate, but also the less general it will be. In order to balance these concerns, we will follow the classic Structure from Motion approach and not assume any geometric properties of the object surface, like smoothness, planarity or other shape models. We will assume simply that the object is rigid and has a finite set of feature points on its surface. Our model will thus contain a 3D point-cloud of these feature points. The object will be viewed through multiple view-points, of initially unknown positions and orientations. We will not assume any particular prior arrangement of view-points, and so part of the model will include specifying the position and orientation of each view-point.

1.2.1 Parametrization

It is natural to assume a parametrized model structure. The parameters permit an efficient encoding of the physical properties of the object and camera and their interaction. The parameters can also be used to create an uncertainty measure for the estimated shape and motion. The model itself consists of the parameters plus a physical interpretation of them. A model configuration will be specified by the values of all of the parameters.

There are many ways parameters can be chosen for a model, but since ultimately the model must be interpreted physically, it is natural to choose parameters that directly correspond to physical quantities, or from which physical quantities can easily be inferred. For example lengths along a coordinate axis may represent the 3D position of part of the model.

Our object rigidity assumption naturally leads to two types of parameters: shape and motion parameters. The shape parameters describe quantities that remain constant over all the images, while the motion parameters restrict the changes between images to those resulting from rotations and translations.

Parameters, however, are simply intermediate quantities. They need to be interpreted. Let us say that our model is represented by $\boldsymbol{\theta}$, a large vector of parameters, and let $\mathcal{T}$ be our parameter space such that any point in this space, $\boldsymbol{\theta} \in \mathcal{T}$, is a valid model. The 3D estimation task corresponds to parametric fitting; namely using the image data to find the point in our space, $\boldsymbol{\theta} \in \mathcal{T}$, that best describes this data. While the actual parameters, $\boldsymbol{\theta}$, may not have physical meaning, our model interpretation, denoted $\mathcal{M}(\boldsymbol{\theta})$, makes our set of geometric statements about the object from the solution parameters. The 3D estimates are thus obtained by interpreting the solution using the model: $\mathcal{M}(\boldsymbol{\theta})$. This is illustrated in Figure 1.4.

1.2.2 Over-parametrization

The number of parameters must be considered. Given a fixed-size problem, the more parameters used, typically the more computationally expensive the optimization procedure. Thus we would like to use a minimal number of parameters to specify a model.

An important question is what is the minimal number of parameters for a model? This is lower-bounded by the number of physical degrees of freedom of the object along which a perturbation would result in a new object. Since each degree of freedom can be changed independently, there must be at least that many parameters specifying the model otherwise there could be a change in the object that is not modeled. Of course we

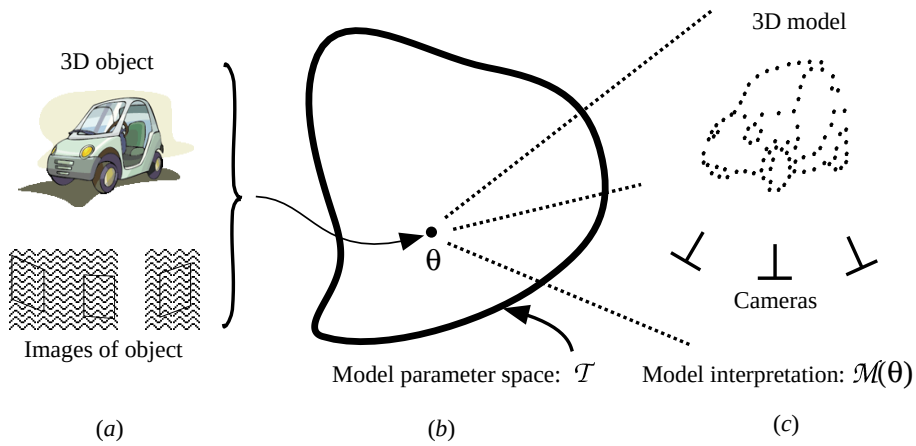


Fig. 1.4. An illustration of 3D vision as a parametric modeling task. (a) A set of images are taken of an object. (b) This whole measurement is encoded into a parameter vector, $\boldsymbol{\theta}$, which must lie in our model parameter space, $\boldsymbol{\theta} \in \mathcal{T}$. ($c$) For this to be useful, we must have an interpretation of the point, $\boldsymbol{\theta}$, in our model space. In this case our model interpretation consists of a set of 3D points and the positions and orientations of the camera view-points. We denote our interpretation as: $\mathcal{M}(\boldsymbol{\theta})$. It identifies which parameters correspond to which 3D quantities, for example, 3 elements of $\boldsymbol{\theta}$ may correspond to the 3D position of one of the feature points, and so forth.

may not wish to model every detail of an object, but for all the aspects we wish to model there must be at least as many parameters as local degrees of freedom. We are interested in systems that can be linearized around the current point, hence, at least locally, the behavior can be modeled by parameters equal in number to the degrees of freedom. It will suffice for us to consider a system as a collection of linear systems defined at each point in the global parameter space, and so a minimally parametrized system will have exactly the same number of parameters as degrees of freedom.

It would thus be nice, for a given modeling task, if we could use a set of parameters whose number is equal to the number of degrees of freedom of the model. There are, however, three main reasons why this may not be possible, or why we may prefer to over-parametrize the model.

The first reason for over-parametrizing is to achieve simplicity in the model. There may be situations where an over-parametrized system is much simpler than a minimally parametrized one. Imagine first that we have a minimally parametrized system with a single parameter for each degree of freedom. Now if a constraint is added that removes a degree of freedom it is over-parametrized. However, there may be no simple way to re-

move a parameter from our equations, and so to maintain simplicity we keep the system over-parametrized.

The second reason follows from the first. If we do find a more complicated minimal parametrization, we may introduce non-linearities and even singularities that make optimization difficult. A common example of over-parametrization is in describing rotations, which have three degrees of freedom, with rotation matrices, having nine parameters, or quaternions, having four parameters. These over-parametrizations are often preferred to the three Euler angles because of their simplicity and the elimination of singularities.

Finally, it is often very helpful that the parameters be physically meaningful, or that we can easily obtain the desired physically properties from them. This allows us to reason about the behavior of an algorithm operating on the parameters, and to easily interpret the results. However, making the parameters meaningful may cause us to describe some physical aspects that are not included in the model. So, for example, giving 3D coordinates of a set of points is physically meaningful, but if our model does not include the overall scale of these points, then this is an over-parametrization.

1.3 Objectives

In this thesis we will not propose a new 3D estimation method, as many adequate methods and algorithms have been proposed for 3D estimation including SFM algorithms, and stereo algorithms. Rather we will assume that an appropriate 3D estimation method has been used to obtain an optimal solution.

Our focus will concern indeterminacies. We will ask: when there are indeterminacies, what does it mean to obtain an optimal solution? This will involve extending the Cramer-Rao lower bound. But then we will address the more fundamental problem of how we represent and interpret parameter uncertainties when there are indeterminacies. We will develop a theory to describe these uncertainties. We will show practical consequences of the theory for SFM and for stereo, and how it can be used to improve the accuracy of 3D estimation methods.

1.4 Related Work

Our goal is to analyze and understand the effects of indeterminacies on 3D estimation by viewing the results as the solution to a non-linear parametric fitting problem. Parametric fitting is, of course, a well studied area. It was Gauss and Legendre [16, 17, 44]

who around 1800 invented the least squares method for estimation problems. Gauss showed that least squares gives the best unbiased linear estimator, and his work included many new aspects of probability theory including the Gaussian distribution. The Gauss-Newton iteration for non-linear optimization is based on his matrix elimination work (see [59] for an overview). In our case we are interested in the 3D geometry of the camera-world system, and so pose the problem as geometric fitting, with parameters to represent the geometry. Works by Kanatani in geometric fitting [32, 34] provide much of the framework for our analysis.

Two fields in computer vision that have focussed on estimating 3D shape are Structure from Motion and Stereo vision. The research in SFM can be divided into three areas. The first area has the goal of simplicity of solution, which includes finding closed form approximate solutions such as the Factorization method introduced by Tomasi and Kanade and extended by others [30, 43, 51, 57, 60, 62, 68]. The second area of interest is efficiency, which includes finding fast or robust numerical schemes [1, 21, 29, 69, 76]. The third area, in which we are interested, is uncertainty analysis. Early work by Weng *et al.* [73] found optimal motion and motion covariance estimates. Young and Chellappa [75] and Szeliski and Kang [64] analyzed ambiguities in 3D shape and motion estimation along with statistical properties of the estimates. Thomas, Hanson and Oliensis [67] looked at the effects of cross correlation in recursive shape estimation. Daniilidis and Spetsakis [10] analyzed noise sensitivity in as part of the visual navigation problem. Morris and Kanade [51], and Sun *et al.* [63] investigated uncertainty models for the Factorization algorithm. While many aspects of the 3D uncertainty problem have been researched, the area that has not been considered is the effect of indeterminacies. All of these works have made arbitrary choices for coordinate systems and scale and ignored the effect this choice has on the final accuracy and uncertainty.

Stereo vision also aims to estimate the 3D structure of the world, but usually with just two images. Numerous algorithms have been proposed, many based on the *essential matrix* of Longuet-Higgins [45] or the *fundamental matrix*, for the uncalibrated case, introduced by Faugeras *et al.* [14, 15] and Hartley *et al.* [20, 26]. Some early error modeling was done by Matthies and Shafer [46] on the fully calibrated case with known baseline and coordinate system. In this case there are no indeterminacies. Weng *et al.* [74] estimated a similar measure. More recently the uncertainties in the uncalibrated case have been considered by Csurka *et al.* [9] and Kanatani [33, 37]. But without calibration, the resulting covariances for the fundamental matrix do not lead to Euclidean uncertainties, and rather are restricted to use in epipolar line matching and calibration.

Our interest lies in the cases where Euclidean structure is estimated and where scale is not known and needs to be estimated using the baseline or other measurement. In these cases the unknown scale is a gauge freedom and it plays an important role in the uncertainty modeling. Yet none of the past works in stereo have considered this issue.

The third area of related work is in geodesy and photogrammetry. Photogrammetrists have long been interested in estimating 3D structure from a sequence of images. Typical applications include using aerial or satellite imagery and GPS to create topographic or elevation maps. The standard bundle-adjustment algorithm was developed by Brown and coworkers in the 1950's [5, 6]. It gives the Maximum Likelihood estimate for independent, isotropic Gaussian noise. The solution is obtained using least squares, but in an efficient way that takes advantage of the block-structure in the equations. Some recent descriptions are reported in the Manual of Photogrammetry (Slama *ed.*) [61] and by Hartley [21, 24].

The indeterminacy issue of translation, rotation and scale have been recognized in photogrammetry as early as 1965, when Meissl [48] introduced so called "inner constraints" to remove these indeterminacies. In 1973, Baarda [2] developed the S-Transformation as a means to minimize these constraints. Further work by Ebner [13], Granshaw [18], Koch [42], Pabo [54] and Dermanis [12, 11] investigated possible inner constraints. Teunissen [66] gives an overview of the "zero order design" problem for creating geodetic networks along with an uncertainty analysis. As initially proposed by Baarda, the indeterminant parameters are typically chosen to optimize a global measure such as the trace of a covariance matrix. While this defines the minimum uncertainty under certain criteria, it does not help in the practical choice of which physical measurements to make in order to remove indeterminacies and at the same time maximize accuracy.

These works were made possible by the development of generalized inverses, initially by Moore [49, 50], Bjerhammar [3] and Penrose [55, 56] with their application to solving linear equations. A good overview and description is provided by Bjerhammar [4], and their applications to photogrammetry by Cooper and Cross [7, 8].

Gauge theory has recently entered the vernacular of the Computer Vision community. Introduced by Triggs [70, 71], gauge orbits were shown to provide a convenient way to speak of indeterminacies in 3D estimation. McLauchlan [47] showed that the choice of gauge in projective SFM can affect convergence rates. The first analysis on how gauge freedoms and constraints jointly affect uncertainties was our work with Kanatani [39, 40,

53]. This thesis will present the development of these ideas and progress of this theory into optimizing gauge constraints.

Our goal is to develop a method for understanding and evaluating uncertainty when there are indeterminacies in 3D estimation. We build a gauge based theory for modeling the indeterminacies and for incorporating constraints. The underlying mathematics is linear algebra and estimation theory, and some of the results that come out of our theory correspond to the geodetic results. In particular, Baarda's S-Transformations follow from our analysis. In the second half of our thesis we analyze various consequences of our theory for SFM and stereo, and how our theory can guide the choice of measurements for optimizing accuracy.

2. Model Fitting and Gauge Theory

Many computer vision tasks are formulated as parametric model fitting. In this paradigm the goal is to find the point, in a potentially large parameter space, that is closest to the true point. Both the resulting solution point and the local uncertainty of that point are important properties. While there have been been many formulations of problems like this, there has not been a careful analysis of the case when indeterminacies exist. We will consider parametric fitting and uncertainty analysis when the parameter space contains indeterminacies.

By an indeterminacy we mean that the solution is not just a single point in parameter space, but rather a whole set of points, or in our case, a manifold of points. All points on this manifold are equivalent and together constitute the solution. The question we are faced with then, is how do we describe and represent the uncertainty of this manifold of points? A thorough treatment of this uncertainty representation problem is the aim of this chapter.

We first introduce our notation and general framework by describing parametric model fitting and first order error propagation. Then the problem of gauge freedoms is described. We develop gauge theory to model uncertainties when there are gauge freedoms, and use many tools from Kanatani's works [31, 32, 34]. Our key results include an oblique projection operator for perturbations and a geometric equivalence relationship for covariances. These were developed in joint work with Kanatani [39, 40, 53]. This allows us to extend the Cramer-Rao lower bound to cases with indeterminacies. The treatment in this chapter is general and thus the theory and interpretations developed here can be applied in any geometric problem that satisfies our basic assumptions. The next chapter will put SFM in this framework enabling us to obtain results there.

2.1 Parametric Fitting

Many estimation tasks can be formulated as parametric fitting to data obtained by a measurement device. In order to evaluate the resulting uncertainty of the resulting estimates, it is necessary to analyze the parametric fitting process. For simplicity we will first look at cases when there are no indeterminacies. Then in later sections of this chapter these results will be generalized to apply to cases containing indeterminacies.

2.1.1 Assumptions

The parametric modeling and fitting task is illustrated in Figure 1.4. Our goal is to estimate the properties of some physical object or process. To do this we define a potentially large parameter space, $\mathcal{T}$, and a model function, $\mathcal{M}(\boldsymbol{\theta})$, where $\boldsymbol{\theta} \in \mathcal{T}$ is a point in the parameter space. The model function is really an interpretation of the parameters, specifying what they imply about the physical object. For example, three parameters could be interpreted by the model function as the 3D position of the corner of an object. A particular model is an instantiation of this model function. If, for instance, we estimate a parameter vector $\hat{\boldsymbol{\theta}}$, then we can also say we have estimated the model $\mathcal{M}(\hat{\boldsymbol{\theta}})$, which tells us all the dimensions and other properties we wish to know about the object of interest. The parameter vector, $\hat{\boldsymbol{\theta}} \in T[]$, completely specifies the model, $\mathcal{M}(\hat{\boldsymbol{\theta}})$. Thus the parameter space and model function must be sufficiently large and general enough to capture all the quantities we wish to know about the objects we model.

Usually, only indirect access to the object model is available, and this is through the measurement process. A measurement occurs when, for example, an image is taken of an object. We assume that we can describe this measurement process with a known set of equations. First the measured data are collected into a vector, $\boldsymbol{p}$, containing all of the individual measurements. A measurement is then described by a vector valued function that takes an object model and produces the measurement data:

$$\boldsymbol{p} = \boldsymbol{\Pi}_{\mathcal{M}}[\mathcal{M}(\boldsymbol{\theta})]. \tag{2.1}$$

In 3D vision, $\boldsymbol{\Pi}_{\mathcal{M}}[\cdot]$ encodes the camera projection equations that take 3D points onto the image plane. For simplicity of both notation and computation we typically merge the object model and the measurement model into a single model to obtain:

$$\boldsymbol{p} = \boldsymbol{\Pi}[\boldsymbol{\theta}]. \tag{2.2}$$

The projection then operates directly on whatever parameters we use in describing the object. However, for indeterminacy analysis it will be useful to distinguish the measurement and object models, and to do this we will refer back to equation (2.1).

Now real world measurements are much more complicated and may have many minor effects that change the value of our measurement. Since these effects often cannot be exactly predicted, we will use a stochastic model to take all of these into account. Let us assume that the data formation process can be adequately described as being generated by a "true" data term, $\bar{\boldsymbol{p}}$, given by our equations, and a noise term, $\Delta\boldsymbol{p}$, namely:

$$\boldsymbol{p} = \bar{\boldsymbol{p}} + \Delta\boldsymbol{p}. \tag{2.3}$$

Throughout the thesis we will assume that the noise is small, and hence that first order perturbation analysis will capture the important effects. While most real world vision applications include the chance of outliers due to tracking or registration errors, we will assume that a pre-processing step has been applied to remove outliers. This step could be an interactive operation, or an automated robust-statistic method as described in [69, 76].

We will assume that the small noise term is unbiased, and thus that its expectation is zero:

$$E[\Delta\boldsymbol{p}] = 0\,. \tag{2.4}$$

The covariance will be denoted as:

$$\begin{aligned} E[\Delta\boldsymbol{p}\Delta\boldsymbol{p}^\top] &= \boldsymbol{V}_{\boldsymbol{p}} \\ &\equiv \epsilon^2 \boldsymbol{V}_{\boldsymbol{po}}. \end{aligned} \tag{2.5}$$

We assume that we have at least a normalized version of the covariance, $\boldsymbol{V}_{\boldsymbol{po}}$, equal to the true covariance, $\boldsymbol{V}_{\boldsymbol{p}}$, apart from an unknown scale factor, ϵ^2.

Part of our small noise assumption is that the first non-zero term in the Taylor expansion of $\Delta\boldsymbol{p}$ will dominate, namely the covariance term. Hence we will ignore higher order terms.

2.1.2 Optimal Parametric Fitting

If there were no noise in our measurement process, then given a sufficient number of true measurement data, $\bar{\boldsymbol{p}}$, we could determine the true parameter vector, $\bar{\boldsymbol{\theta}}$, by inverting equation (2.2). However, there will inevitably be noise in the measurements as given in equation (2.3). Using the noise-corrupted data, we will obtain a parameter estimate $\hat{\boldsymbol{\theta}}$

which we would like to be as close to the true value, $\bar{\boldsymbol{\theta}}$, as possible. This can be formalized by saying that wish to achieve an optimal parameter estimate where optimality is defined to mean the solution achieves the Cramer-Rao lower bound. Under Gaussian noise and without indeterminacies, the optimal solution for this is known to correspond to the Maximum Likelihood solution, and we present it here. Since all of the data are assumed to be available before solution estimation, we will use a batch estimation process rather than a recursive method, such as the Extended Kalman Filter, which would give suboptimal results.

In order to derive an optimal solution, we will assume zero-mean, Gaussian noise for our measurements. When the noise is not Gaussian, our optimality statement will not apply, but the rest of the statistics we derive, namely the bias and covariance results, will still be valid descriptions of our solution under small noise.

The probability of our data and noise is given by:

$$\begin{aligned}\rho(\Delta\boldsymbol{p}) &= ke^{-\frac{1}{2}\Delta\boldsymbol{p}^\top \boldsymbol{V}_{\boldsymbol{p}}^{-1}\Delta\boldsymbol{p}} \\ &= ke^{-\frac{1}{2}(\bar{\boldsymbol{p}}-\boldsymbol{p})^\top \boldsymbol{V}_{\boldsymbol{p}}^{-1}(\bar{\boldsymbol{p}}-\boldsymbol{p})}\,,\end{aligned} \tag{2.6}$$

where k is the appropriate constant to make this a Gaussian. This can be parametrized in terms of our model parameters as:

$$\rho(\boldsymbol{\theta}) = ke^{-\frac{1}{2}(\boldsymbol{p}-\boldsymbol{\Pi}[\boldsymbol{\theta}])^\top \boldsymbol{V}_{\boldsymbol{p}}^{-1}(\boldsymbol{p}-\boldsymbol{\Pi}[\boldsymbol{\theta}])}\,. \tag{2.7}$$

The Maximum Likelihood solution is thus obtained by minimizing the cost:

$$\begin{aligned}J(\boldsymbol{\theta}) &= (\boldsymbol{p}-\boldsymbol{\Pi}[\boldsymbol{\theta}])^\top \boldsymbol{V}_{\boldsymbol{p}}^{-1}(\boldsymbol{p}-\boldsymbol{\Pi}[\boldsymbol{\theta}]) \\ &\equiv \|\boldsymbol{p}-\boldsymbol{\Pi}[\boldsymbol{\theta}]\|^2_{\boldsymbol{V}_{\boldsymbol{p}}}\end{aligned} \tag{2.8}$$

to get a parameter estimate $\hat{\boldsymbol{\theta}}$. Here we denote the L_2 norm, weighted by the inverse of the covariance $\boldsymbol{V}_{\boldsymbol{p}}$, as $\|\cdot\|^2_{\boldsymbol{V}_{\boldsymbol{p}}}$. In 3D computer vision, depending on the camera model involved, various methods can be used to minimize this cost function. In the affine case the camera model is bilinear in shape and motion, and a globally optimal solution can be obtained using SVD[68]. Perspective camera models are nonlinear, and so iterative nonlinear optimization is typically needed. This opens the hazard of being trapped by local minima, or even that the minimum around the true solution has higher cost than other minima of incorrect solutions. Nevertheless, we shall assume in this work that we have obtained an estimate $\hat{\boldsymbol{\theta}}$ at a global minimum of J which is close to the true solution, $\bar{\boldsymbol{\theta}}$, and can be written:

$$\hat{\boldsymbol{\theta}} = \bar{\boldsymbol{\theta}} + \Delta\boldsymbol{\theta} \tag{2.9}$$

The quality of the solution will be measured by the statistics of the error or noise term: $\Delta\boldsymbol{\theta}$.

Conditions for Optimality

We now wish to estimate the bias and covariance of this solution, $\hat{\boldsymbol{\theta}}$, and compare these to the Cramer-Rao lower bound. Our final results in this section will apply only when there are no indeterminacies, but these results are extended in section 2.5 to cases with indeterminacies.

The lower bound requires the Fisher information matrix, which we obtain as follows. The score, $\boldsymbol{l}_{\boldsymbol{\theta}}$, is defined as:

$$\begin{aligned}\boldsymbol{l}_{\boldsymbol{\theta}} &= \nabla_{\boldsymbol{\theta}} \log \rho(\boldsymbol{\theta}) \\ &= -\left(\nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}]\right)^{\top} \boldsymbol{V}_{\boldsymbol{p}}^{-1} \left(\boldsymbol{p} - \boldsymbol{\Pi}[\boldsymbol{\theta}]\right).\end{aligned} \tag{2.10}$$

where the notation $\nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}]$ indicates the gradient matrix:

$$\nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}] = \begin{pmatrix} \partial \boldsymbol{\Pi}_1[\boldsymbol{\theta}]/\partial\theta_1 & \dots & \partial \boldsymbol{\Pi}_1[\boldsymbol{\theta}]/\partial\theta_n \\ \vdots & \ddots & \vdots \\ \partial \boldsymbol{\Pi}_m[\boldsymbol{\theta}]/\partial\theta_1 & \dots & \partial \boldsymbol{\Pi}_m[\boldsymbol{\theta}]/\partial\theta_n \end{pmatrix}, \tag{2.11}$$

given that $\boldsymbol{\theta}$ has n elements and $\boldsymbol{\Pi}[\boldsymbol{\theta}]$ has m elements.

Then the Fisher information matrix is then obtained as:

$$\boldsymbol{J}_{\boldsymbol{\theta}} = E[\boldsymbol{l}_{\boldsymbol{\theta}} \boldsymbol{l}_{\boldsymbol{\theta}}^{\top}]. \tag{2.12}$$

Expanding the right hand side of this equation and substituting $\Delta\boldsymbol{p} = \boldsymbol{p} - \boldsymbol{\Pi}[\boldsymbol{\theta}]$ we obtain

$$\begin{aligned}\boldsymbol{J}_{\boldsymbol{\theta}} &= \nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top} \boldsymbol{V}_{\boldsymbol{p}}^{-1} E[\Delta\boldsymbol{p}\Delta\boldsymbol{p}^{\top}] \boldsymbol{V}_{\boldsymbol{p}}^{-1} \nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}] \\ &= \nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top} \boldsymbol{V}_{\boldsymbol{p}}^{-1} \boldsymbol{V}_{\boldsymbol{p}} \boldsymbol{V}_{\boldsymbol{p}}^{-1} \nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}] \\ &= \nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top} \boldsymbol{V}_{\boldsymbol{p}}^{-1} \nabla_{\boldsymbol{\theta}} \boldsymbol{\Pi}[\boldsymbol{\theta}].\end{aligned} \tag{2.13}$$

Hence we see that $\boldsymbol{J}_{\boldsymbol{\theta}}$ is symmetric.

Cramer-Rao theory states that the inverse of the Fisher information matrix, $\boldsymbol{J}_{\boldsymbol{\theta}}^{-1}$, provides us with a lower bound on the covariance of our solution. Hence if we can show that a solution is unbiased and that its covariance is equal to the inverse Fisher information matrix, it must be optimal.

Maximum Likelihood Solution

We now seek to find expressions for the bias and covariance of the Maximum Likelihood solution, which we will refer to as $\hat{\boldsymbol{\theta}}$. To do this we first expand the cost in equation (2.8) around the true value:

$$J(\hat{\boldsymbol{\theta}}) = J(\bar{\boldsymbol{\theta}} + \Delta\boldsymbol{\theta}) = \bar{J} + \nabla_{\boldsymbol{\theta}}\bar{J}^{\top}\Delta\boldsymbol{\theta} + \frac{1}{2}\Delta\boldsymbol{\theta}^{\top}\nabla^2_{\boldsymbol{\theta}}\bar{J}\Delta\boldsymbol{\theta} + \dots , \tag{2.14}$$

where $\bar{J}$ represents the cost at the true parameter value: $J(\bar{\boldsymbol{\theta}})$. We would like to find the bias and variance of $\boldsymbol{\theta}$ around the true value $\bar{\boldsymbol{\theta}}$. Ignoring higher order terms of the noise, and differentiating with respect to $\Delta\boldsymbol{\theta}$, we obtain:

$$\nabla_{\boldsymbol{\theta}}\bar{J} + \nabla^2_{\boldsymbol{\theta}}\bar{J}\Delta\boldsymbol{\theta} = 0 . \tag{2.15}$$

Inverting the Hessian[1], we obtain an expression for $\Delta\boldsymbol{\theta}$:

$$\Delta\boldsymbol{\theta} = -(\nabla^2_{\boldsymbol{\theta}}\bar{J})^{-1}\nabla_{\boldsymbol{\theta}}\bar{J} . \tag{2.16}$$

To solve for this we will need expressions for the gradient and Hessian terms. These can be obtained as:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\bar{J} &= -2\nabla_{\boldsymbol{\theta}}\boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top}\boldsymbol{V}_{\boldsymbol{p}}^{-1}(\boldsymbol{p} - \boldsymbol{\Pi}[\boldsymbol{\theta}]) \\ &= 2\boldsymbol{l}_{\boldsymbol{\theta}} ,\end{aligned} \tag{2.17}$$

and as we would expect the gradient is just the score. The Hessian is the gradient of the score:

$$\nabla^2_{\boldsymbol{\theta}}\bar{J} = 2\nabla_{\boldsymbol{\theta}}\boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top}\boldsymbol{V}_{\boldsymbol{p}}^{-1}\nabla_{\boldsymbol{\theta}}\boldsymbol{\Pi}[\boldsymbol{\theta}] \; - \; 2\nabla^2_{\boldsymbol{\theta}}\boldsymbol{\Pi}[\boldsymbol{\theta}]\boldsymbol{V}_{\boldsymbol{p}}^{-1}(\boldsymbol{p} - \boldsymbol{\Pi}[\boldsymbol{\theta}]) . \tag{2.18}$$

Now if we assume that our noise term $\Delta\boldsymbol{p} = \boldsymbol{p} - \boldsymbol{\Pi}[\boldsymbol{\theta}]$ is small and of order $\mathcal{O}(\epsilon)$, then the first term on the right of this equation is $\mathcal{O}(1)$ and the second term is $\mathcal{O}(\epsilon)$. Thus to first order we can drop the second term and obtain:

$$\begin{aligned}\nabla^2_{\boldsymbol{\theta}}\bar{J} &= 2\nabla_{\boldsymbol{\theta}}\boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top}\boldsymbol{V}_{\boldsymbol{p}}^{-1}\nabla_{\boldsymbol{\theta}}\boldsymbol{\Pi}[\boldsymbol{\theta}] \\ &= 2E[\boldsymbol{l}_{\boldsymbol{\theta}}\boldsymbol{l}_{\boldsymbol{\theta}}^{\top}] \\ &= 2\boldsymbol{J}_{\boldsymbol{\theta}} .\end{aligned} \tag{2.19}$$

Thus the perturbation of our solution, $\hat{\boldsymbol{\theta}}$, from the true solution, $\bar{\boldsymbol{\theta}}$, caused by noise in the measurement, $\Delta\boldsymbol{p}$, as described by equation (2.16), is found to be:

$$\Delta\boldsymbol{\theta} = -\boldsymbol{J}_{\boldsymbol{\theta}}^{-1}\boldsymbol{l}_{\boldsymbol{\theta}} . \tag{2.20}$$

[1] When there are indeterminacies the Hessian will be singular and the generalized inverse must be used as described in section 2.5.

We can now find the expectation of the error component in our solution, $\Delta\boldsymbol{\theta}$. From our assumption that the measurement noise is unbiased in equation (2.4), we obtain

$$\begin{aligned} E[\Delta\boldsymbol{\theta}] &= -E[\boldsymbol{J}_{\boldsymbol{\theta}}^{-1}\boldsymbol{l}_{\boldsymbol{\theta}}] \\ &= \boldsymbol{J}_{\boldsymbol{\theta}}^{-1}\, 2\boldsymbol{\Pi}[\boldsymbol{\theta}]^{\top}\, \boldsymbol{V}_{\boldsymbol{p}}^{-1}\, E[\boldsymbol{p}] \\ &= 0\,, \end{aligned} \tag{2.21}$$

and conclude that the Maximum Likelihood solution is unbiased to first order. The covariance of the solution can be estimated as follows:

$$\begin{aligned} V_{\boldsymbol{\theta}} &= E[\Delta\boldsymbol{\theta}\Delta\boldsymbol{\theta}^{\top}] \\ &= \boldsymbol{J}_{\boldsymbol{\theta}}^{-1} E[\boldsymbol{l}_{\boldsymbol{\theta}}\boldsymbol{l}_{\boldsymbol{\theta}}^{\top}]\boldsymbol{J}_{\boldsymbol{\theta}}^{-\top} \\ &= \boldsymbol{J}_{\boldsymbol{\theta}}^{-1}\,. \end{aligned} \tag{2.22}$$

We conclude that with no indeterminacies, and up to first order, the covariance of the Maximum Likelihood solution is just the inverse of the Fisher information matrix and so achieves the Cramer-Rao lower bound. It is thus an optimal estimate.

Estimating the Unknown Covariance Scale Factor

In many cases the data covariance may be known only up to a scale factor. For example the noise may be *i.i.d.*, but with unknown global magnitude. Instead of knowing the true covariance, $\boldsymbol{V}_{\boldsymbol{p}} = E[\Delta\boldsymbol{p}\Delta\boldsymbol{p}^{\top}]$, we assume that we know a normalized form of it, $\boldsymbol{V}_{\boldsymbol{po}}$, where $\boldsymbol{V}_{\boldsymbol{p}} = \epsilon^2\boldsymbol{V}_{\boldsymbol{po}}$ and ϵ^2 is an unknown scale factor. Hence to obtain a solution we minimize the modified residual:

$$J_o(\boldsymbol{\theta}) = \Delta\boldsymbol{p}^{\top}\boldsymbol{V}_{\boldsymbol{po}}^{-1}\Delta\boldsymbol{p}\,, \tag{2.23}$$

where we have omitted the unknown ϵ^2 from $J(\boldsymbol{\theta})$ in equation (2.8). Minimizing this of course gives us the same solution as minimizing the original residual $J(\boldsymbol{\theta})$. But if we want to obtain the covariance of our solution we need to recover this unknown scale factor ϵ^2.

If the noise is Gaussian then the residual J will be a χ^2 variable. The expectation of a χ^2 variable is equal to its number of degrees of freedom. Let m be the length of $\Delta\boldsymbol{p}$ and assume its covariance, $\boldsymbol{V}_{\boldsymbol{p}}$, has full rank. We fit our measurements with a set of parameters in our parameter vector $\boldsymbol{\theta}$ which we assume has length n. Thus the total number of degrees of freedom in the residual are: $m - n$, and so we can write:

$$E[J] = \frac{E[J_o]}{\epsilon^2} \;=\; m - n\,. \tag{2.24}$$

When we actually run experiments, we take our recovered residual, $\hat{J}_o$, as our estimate for $E[J_o]$, and then obtain the unknown scale factor as:

$$\epsilon^2 = \frac{\hat{J}_o}{m-n}\,. \tag{2.25}$$

When there are indeterminacies, the effective number of parameters is reduced. If the indeterminacies involve r degrees of freedom, then the effective number of fitted parameters is $n-r$, and the residual, J, is then a χ^2 variable with $m-n+r$ degrees of freedom.

2.2 Gauge Transformations: their Orbits and Freedoms

For various reasons there may not be a unique solution to the parameter estimate obtained from equation (2.2). Rather there may be a range of solutions that all satisfy this equation. The range of solutions for a given problem can be distinguished into two general classes: discrete ambiguities, and continuous indeterminacies. For discrete ambiguities, there is not an arbitrarily small neighborhood of solutions around a given solution, but rather these ambiguities correspond to fixed transformations of the solution such as reflections. These ambiguities do not affect perturbation analysis and so are not considered in this work. When these ambiguities exist, other cues beyond the measurement data must be used to select an appropriate solution. Our interest, however, lies in the continuous indeterminacies that define a manifold of points around any given point. These correspond to cases where the measurement equation (2.2) does not fully constrain the parameter vector, $\boldsymbol{\theta}$. Solutions to this equation will have local degrees of freedom of $\boldsymbol{\theta}$ which we will refer to as the gauge freedoms.

In physics, gauge freedoms are used to describe symmetries found in the equations of quantum mechanics and electricity and magnetism. Their use is limited, and not directly parallel to our work. So while we choose to borrow the term, we will freely redefine it for use in computer vision.

As is the case for discrete ambiguities, gauge freedoms result in the solution not being uniquely defined, and so additional information must be employed in specifying the solution. But unlike discrete ambiguities, perturbations of the solution are affected by the gauge freedoms and by how a unique solution is determined. Our goal in this research is to analyze how gauge freedoms affect perturbation uncertainties of model parameters.

2.2.1 Gauge Orbits

Imagine that there are two points, $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ in the parameter space, $\mathcal{T}$, that both give the same measurement: $\boldsymbol{\Pi}[\boldsymbol{\theta}'] = \boldsymbol{\Pi}[\boldsymbol{\theta}]$. We say that these point are *geometrically equivalent*, since they cannot be distinguished by our equations. Let these points be related to each other by a gauge transformation, g, such that

$$\boldsymbol{\theta}' = g\boldsymbol{\theta}. \tag{2.26}$$

Now we assume that all points satisfying this property lie on a manifold, and hence the set of all such gauge transformations form a group, G, which we denote as our *gauge group*.[2] Then for any two points, $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$, related by a gauge transformation, $g \in G$, we denote their geometric equivalence by the expression: $\boldsymbol{\theta}' \sim \boldsymbol{\theta}$.

Let us call the manifold in $\mathcal{T}$, over which we have defined our equivalence relationship for all $g \in G$, a *gauge orbit*, $\mathcal{G}_{\boldsymbol{\theta}}$. The gauge orbit contains all geometrically equivalent values of $\boldsymbol{\theta}$, namely: $\mathcal{G}_{\boldsymbol{\theta}} = \{\boldsymbol{\theta}' | \boldsymbol{\theta}' \sim \boldsymbol{\theta}\}$. We further assume that all the points of the gauge orbit make up a single connected component. Any point in the gauge orbit is completely equivalent to any other point for an optimization routine. All points in the parameter space belong to a gauge orbit, and so the optimization process only needs to operate in the quotient space: $\mathcal{T}/G$ of $\mathcal{T}$. Each element of this quotient space is a gauge orbit. Mathematicians call gauge orbits *leafs*, and the parameter space, $\mathcal{T}$, they call a *foliated space*, or a *foliation*. Figure 2.1 illustrates such a foliation. We note that perturbations along the gauge orbit do not change the cost, and so all points on the gauge orbit must be regarded as equivalent.

2.2.2 Gauge Freedoms

We will be interested in doing perturbation analysis in the neighborhood of a point in the parameter space. For this we need a linearization of the gauge transformation. Let us select a gauge transformation $\gamma \in G$ close to the identity transformation, where the identity transformation maps a point onto itself. We call this an infinitesimal gauge transformation and can expand it in a Taylor series as:

$$\gamma\boldsymbol{\theta} = \boldsymbol{\theta} + \boldsymbol{D}(\boldsymbol{\theta}) + \ldots, \tag{2.27}$$

[2] From the definition of a manifold it is easy to see that G must be a group. If $\boldsymbol{x}$ and $\boldsymbol{y}$ are patches of our manifold such that $\boldsymbol{\theta}$ is in the image of $\boldsymbol{x}$ and $\boldsymbol{\theta}'$ is in the image of $\boldsymbol{y}$, then g and its inverse exist. Moreover a sequence of gauge transformations will be associative.

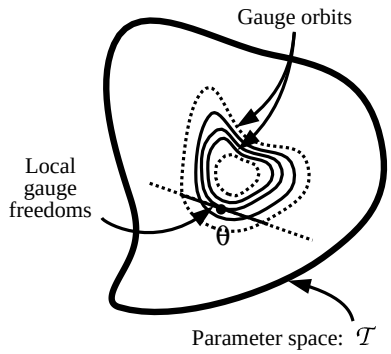


Fig. 2.1. We illustrate a foliated parameter space, $\mathcal{T}$. It is filled with gauge orbits, or leafs, such that every point is on a gauge orbit. The gauge freedoms at a given point span the tangent space to the gauge orbit.

where $\boldsymbol{D}(\cdot)$ is a vector operator which we term the infinitesimal gauge generator of γ. The set of all such infinitesimal gauge generators form a linear space, $\mathcal{D}_\theta(\mathcal{T})$, around point $\boldsymbol{\theta}$. It can be proved that every point in this space, $\boldsymbol{D}(\cdot) \in \mathcal{D}_\theta(\mathcal{T})$, corresponds to an infinitesimal gauge transformation, γ, with $\boldsymbol{D}(\cdot)$ as its generator.

We will call the linear space, $\mathcal{D}_\theta(\mathcal{T})$, the collection of *gauge freedoms* at point $\boldsymbol{\theta} \in \mathcal{T}$. If r is the dimension of this space, then the gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, is a r-dimensional manifold in $\mathcal{T}$, and the space, $\mathcal{T}$, has r gauge freedoms.

2.2.3 Causes of Gauge Freedoms

There are two sources of gauge freedoms that we distinguish. The first source is an over-parametrization of the model. A gauge freedom in this case exists when we can define a non-identity gauge transformation, g_m, on the parameter vector $\boldsymbol{\theta}$ to be any transformation in $\mathcal{T}$ that does not change the model interpretation:

$$\mathcal{M}(\boldsymbol{\theta}) \equiv \mathcal{M}(g_m \boldsymbol{\theta}), \ \forall \boldsymbol{\theta} \in \mathcal{T}. \tag{2.28}$$

A common example of this occurs in the use of homogeneous vectors. A homogeneous vector $\boldsymbol{a} = (a_1, a_2, a_3, a_4)$ under a model interpretation may define a point: $\mathbf{X} = (a_1/a_4, a_2/a_4, a_3/a_4)$. A gauge transformation corresponding to a rescaling of $\boldsymbol{a}$ will not change the model interpretation, revealing a gauge freedom. We note, however, that not all over-parametrizations lead to to gauge freedoms. For example, the space of rotation matrices, often denoted as $SO(3)$, has three dimensions and is described by 9 parameters, but points not on $SO(3)$ are not defined, and do not correspond to any valid rotation matrices.

This type of gauge freedom is created for representational or numerical convenience. In certain problems these may be introduced to reduce nonlinearities or avoid singularities in the estimation process. However, typically the extra parameters are purely non-physical, and hence do not affect the model. Thus we do not expect these to contribute to changes in the uncertainties of physical aspects of the model. We do, however, need a mathematical formalism for treating these freedoms in the overall uncertainty model.

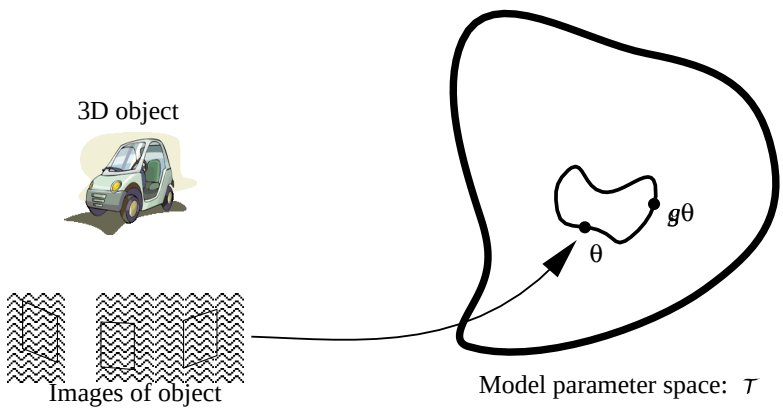


Fig. 2.2. 3D vision can be understood as model estimation. Given a set of image measurements, and the equations that govern these measurements, the goal is to find the point in the model space that corresponds to the 3D object. Ideally a single point will be found. If, however, there are gauge freedoms in the measurements, then a possibly large subset of the models will correspond to the same measurements. From these measurements alone, all of these models are equally likely; if the measurements have gauge freedom g, then $\boldsymbol{\theta} \sim g\boldsymbol{\theta}$.

The second source of gauge freedoms occurs in the measurement process. Under certain gauge transformations the model changes but the measurements remain the same. Let g_p be such a gauge transformation where $\mathcal{M}(\boldsymbol{\theta}) \neq \mathcal{M}(g_p\boldsymbol{\theta})$, but:

$$\boldsymbol{\Pi}[\boldsymbol{\theta}] = \boldsymbol{\Pi}[g_p\boldsymbol{\theta}], \quad \forall \boldsymbol{\theta} \in \mathcal{T}. \tag{2.29}$$

So long as our projection can be decomposed into pure model and pure projection components, as in equation (2.1), both model and projection gauge transformations, g_m and g_p, will be covered by our analysis.

This second type of gauge freedom captures basic indeterminacies resulting from the measurement process. Due to the physics of the measurement process, certain physical properties of the model are lost and indeterminant. The classic example of this in computer vision is scale. A small object close to a camera may appear identical to a large

object farther from the camera. If the parameter vector $\boldsymbol{\theta}$ represents the shape of an object, and the gauge transformation g_p is a rescaling, then we can write $\boldsymbol{\theta} \sim g_p\boldsymbol{\theta}$. We shall show that how these gauge freedoms are dealt with can have significant physical effects on the solution.

2.2.4 Maximum Likelihood with Gauge freedoms

To achieve our modeling task we optimize a cost function over the parameter space. To achieve a Maximum Likelihood solution we will choose the weighted reprojected error denoted as $J(\boldsymbol{\theta})$ in equation (2.8). It could alternatively be expressed as: $J(\boldsymbol{\Pi}[\boldsymbol{\theta}])$, but typically we incorporate the projection function into the cost and call it: $J(\boldsymbol{\theta})$. We define a gauge transformation, g, on the parameters to be any transformation in $\mathcal{T}$ that does not change the cost function:

$$J(\boldsymbol{\theta}) = J(g\boldsymbol{\theta}), \ \forall \boldsymbol{\theta} \in \mathcal{T}. \tag{2.30}$$

Changing the parameter vector by a gauge transformation will not affect the cost. The set of all such transformations form a group, G, which we call the group of gauge transformations. So long as all degrees of freedom are captured by $J(\boldsymbol{\Pi}[\boldsymbol{\theta}])$, and thus the cost function itself does not introduce any new freedoms, this group will consist of the set of all gauge transformations due to model over-parametrization, g_p, and those due to the measurement process g_m.

Optimization will proceed in a very similar manner to that done when there are no gauge freedoms.

2.3 Invariants

While the values of our parameters will vary over gauge orbits, it is in general possible to define functions of the parameters that do not change over the gauge orbits. We call these *invariants* and define them as follows:

Definition 2.3.1. *A function, $\mathcal{I}(\cdot) : \mathcal{T} \to \mathcal{R}$ is a gauge invariant if:*

$$\mathcal{I}(\boldsymbol{\theta}) = \mathcal{I}(g\boldsymbol{\theta}) \quad \forall g \in G, \tag{2.31}$$

We are assuming a single connected component for the gauge orbit.

From our definition we can conclude that $\mathcal{I}(\cdot)$ is an invariant if and only if it is also invariant to infinitesimal gauge transformations. If γ is an infinitesimal gauge transformation and $\boldsymbol{D}(\cdot)$ is its generator, then we can write:

$$\begin{aligned}\mathcal{I}(\boldsymbol{\theta}) &= \mathcal{I}(\gamma\boldsymbol{\theta}) \\ &= \mathcal{I}(\boldsymbol{\theta}) + \nabla_\theta \mathcal{I}^\top \boldsymbol{D}(\boldsymbol{\theta}) + \dots .\end{aligned} \tag{2.32}$$

From this equation we can see that the first order term of this expansion must be zero. If we let $\boldsymbol{D}_i(\boldsymbol{\theta})$ for $i = 1, \dots, r$ be a basis set of vectors for the space $\mathcal{D}_\theta(\mathcal{T})$, then these vectors span the tangent space, $T_\theta[\mathcal{G}_{\boldsymbol{\theta}}]$, to the gauge orbit $\mathcal{G}_{\boldsymbol{\theta}}$. From our equation we conclude that $\nabla_\theta \mathcal{I}^\top \boldsymbol{D}_i(\boldsymbol{\theta}) = 0$, for $i = 1, \dots, r$. This leads to the following theorem:

Theorem 2.3.1. *A function $\mathcal{I}(\cdot) : \mathcal{T} \to \mathcal{R}$ is gauge invariant if and only if*

$$\nabla_\theta \mathcal{I} \in T_\theta[\mathcal{G}_{\boldsymbol{\theta}}]^\perp, \tag{2.33}$$

where "$\perp$" refers to the orthogonal complement.

Intuitively this says that the gradient of an invariant is always orthogonal to the gauge orbit. This orthogonality constraint will be useful in our perturbation analysis.

Invariants are important since they do not contain indeterminacies or gauge freedoms, and so they are the "real" geometric properties that are estimated by our equations. By using these as our basic entities, in the next section we will be able to derive expressions for how parameters and their perturbations vary along gauge orbits.

2.4 Perturbation Analysis

A solution, $\boldsymbol{\theta}$, of our optimization procedure will lie somewhere on a gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$. The solution can be transformed to any other point on the gauge orbit by selecting the appropriate gauge transformation, g, and the cost from equation (2.30) will remain the same. This makes perturbation analysis in the full parameter space around a solution point meaningless since an arbitrarily large transformation of parameters on the gauge orbit will result in no change in cost. To perform perturbation analysis we will need our solution to be at least a locally unique. This is achieved by applying constraints in a process we will call gauge fixing, as is illustrated in Figure 2.3.

2.4.1 Gauge Fixing

We can achieve a unique solution on the gauge orbit by introducing a set of r constraint equations where r is the number of gauge freedoms. Let

$$c_1(\boldsymbol{\theta}) = 0, \quad \dots, \quad c_r(\boldsymbol{\theta}) = 0 \tag{2.34}$$

denote our r constraints with the following assumptions.

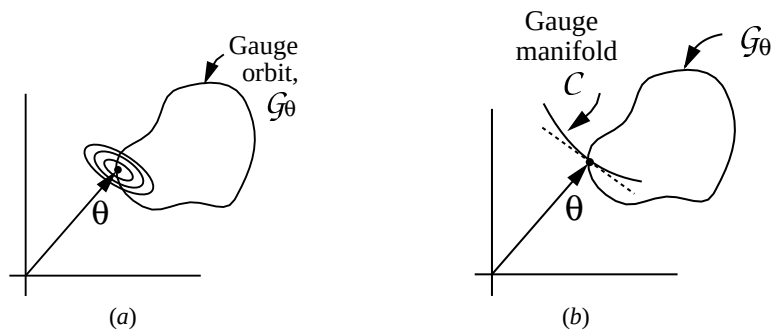


Fig. 2.3. Plot (a) shows the contradiction we obtain when we use a full rank covariance to describe perturbations of a point on a gauge orbit. Ellipses of different probability density intersect the gauge orbit, all of whose points are equivalent and hence have equal probability density. Instead, we must choose a gauge by imposing a constraint that reduces our parameter space to the gauge manifold, $\mathcal{C}$, which intersects the gauge orbit at a single point, as shown in (*b*). Perturbations are now restricted to the tangent plane to the gauge manifold, shown by the dashed line. (These plots are two dimensional for clarity only. In our case, the vector, $\boldsymbol{\theta}$, is n dimensional, the gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, is r dimensional, and the gauge, $\mathcal{C}$, is $n - r$ dimensional.)

- The equations are independent and define a manifold, $\mathcal{C}$, of codimension r in $\mathcal{T}$,
- The manifold $\mathcal{C}$ intersects all of the leafs, $\mathcal{G}_{\boldsymbol{\theta}}$, transversally and at a single point, and
- For any element $\boldsymbol{\theta} \in \mathcal{G}_{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}_{\mathcal{C}} = \mathcal{C} \cup \mathcal{G}_{\boldsymbol{\theta}}$, there exists a unique element $g \in G$ such that $\boldsymbol{\theta}_{\mathcal{C}} = g\boldsymbol{\theta}$.

Enforcing this set of constraints we call *gauge fixing* or *choosing a gauge*. We let $\mathcal{C}$ denote our *gauge manifold*, and for short call it simply a *gauge*.

Gauge fixing achieves two important factors which will affect the solution and the perturbation analysis. Firstly it determines a unique solution point, $\boldsymbol{\theta}_{\mathcal{C}}$, on the gauge orbit at the intersection of the gauge manifold and the gauge orbit, $\boldsymbol{\theta}_{\mathcal{C}} = \mathcal{C} \cap \mathcal{G}_{\boldsymbol{\theta}}$. Secondly it determines the tangent plane to the gauge, $T_\theta[\mathcal{C}]$, and this will differ depending on the gauge constraints. Even when two gauges intersect the gauge orbit at the same point the tangent spaces may be different. This is important since perturbations of our solution, $\Delta\boldsymbol{\theta}_{\mathcal{C}}$, to first order must remain in our gauge, and hence must be the tangent plane to the gauge: $\Delta\boldsymbol{\theta}_{\mathcal{C}} \in T_\theta[\mathcal{C}]$. We would like to compare perturbations in different gauges around the solution manifold and derive a relationship between them.

2.4.2 Jacobian Transformation

There are two factors that will affect perturbations at different points on a gauge orbit. The first factor depends on the gauge transformation, g, between the local parameter space, $\mathcal{T}$, at each point. Let $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ be two points on a gauge orbit such that

$$\boldsymbol{\theta}' = g\boldsymbol{\theta}\,. \tag{2.35}$$

We can think of this $\boldsymbol{\theta}'$ as a function of $\boldsymbol{\theta}$. If we perform a first order expansion of both sides of this relationship we obtain:

$$\Delta\boldsymbol{\theta}' = \frac{\partial\boldsymbol{\theta}'}{\partial\boldsymbol{\theta}}\Delta\boldsymbol{\theta}\,, \tag{2.36}$$

where $\partial\boldsymbol{\theta}'/\partial\boldsymbol{\theta}$ is our Jacobian matrix that transforms perturbations at one point on the gauge orbit, $\boldsymbol{\theta}$, to perturbations at another point, $\boldsymbol{\theta}'$. This is illustrated in Figures 2.4 and 2.5. For example, if g rotates the parameters, then the Jacobian matrix rotates the perturbations, and hence covariances, accordingly.

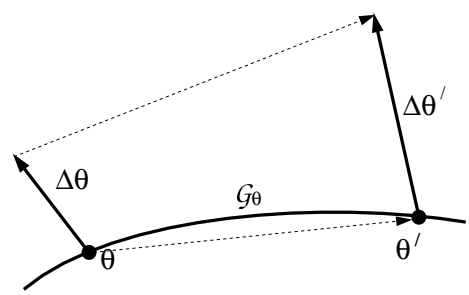


Fig. 2.4. Let $\boldsymbol{\theta}' = g\boldsymbol{\theta}$ be a gauge transformation of point $\boldsymbol{\theta}$ onto $\boldsymbol{\theta}'$. This can also be thought of as a change of variables. The change of perturbations from point $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$ are described the Jacobian matrix: $\partial\boldsymbol{\theta}'/\partial\boldsymbol{\theta}$, such that $\Delta\boldsymbol{\theta}' = \partial\boldsymbol{\theta}'/\partial\boldsymbol{\theta}\,\Delta\boldsymbol{\theta}$. These affects are quite intuitive in real examples as illustrated in Figure 2.5.

2.4.3 Geometric Equivalence I

The second factor affecting the covariance when choosing a gauge is less intuitive but may have even stronger effects. This is the tangent space to the gauge orbit. In order to derive this effect, we must first consider what constitutes an equivalence between perturbations. As we argued in section 2.3, it is invariants, rather than actual parameters, that capture the essential geometric information in the measurements. Furthermore it is the perturbations of invariants that capture the geometric information in parametric perturbations.

Let us assume we have two gauges, $\mathcal{C}$ and $\mathcal{C}'$ that intersect a gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, at $\boldsymbol{\theta}_{\mathcal{C}}$ and $\boldsymbol{\theta}_{\mathcal{C}'}$ respectively. These points are related by a gauge transformation, denoted as $\boldsymbol{\theta}_{\mathcal{C}} = g\boldsymbol{\theta}_{\mathcal{C}'}$.

We can parametrize an invariant $\mathcal{I}$ using points in either of these gauges:

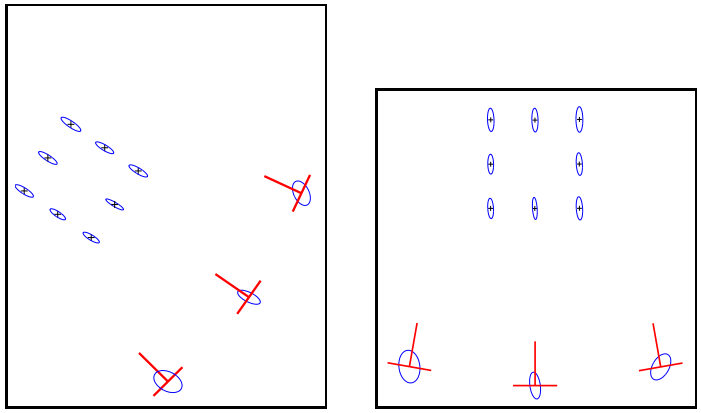


Fig. 2.5. Here we illustrate the effect of a Jacobian matrix $\partial\boldsymbol{\theta}'/\partial\boldsymbol{\theta}$ on a synthetic example. The gauge transformation, $\boldsymbol{\theta}' = g\boldsymbol{\theta}$, is a rotation of feature points and camera positions. The Jacobian thus also acts as a rotation, and appropriately rotates the ellipses around each point.

$$\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}}) = \mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}'}) \,. \tag{2.37}$$

Furthermore we can parametrize perturbations of the invariant $\mathcal{I}(\boldsymbol{\theta})$ in terms of perturbations along either of these gauge tangent spaces. We obtain:

$$\Delta\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}}) = \nabla_{\theta}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^{\top}\Delta\boldsymbol{\theta}_{\mathcal{C}} \,, \tag{2.38}$$

for gauge $\mathcal{C}$ where $\Delta\boldsymbol{\theta}_{\mathcal{C}} \in T_{\theta}[\mathcal{C}]$. On the other hand, in gauge $\mathcal{C}'$, we have:

$$\begin{aligned}\Delta\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}'}) &= \nabla_{\theta}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}'})^{\top}\Delta\boldsymbol{\theta}_{\mathcal{C}'} \\ &= \nabla_{\theta}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^{\top}\frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}_{\mathcal{C}'}}\Delta\boldsymbol{\theta}_{\mathcal{C}'} \,,\end{aligned} \tag{2.39}$$

where $\Delta\boldsymbol{\theta}_{\mathcal{C}'} \in T_{\theta}[\mathcal{C}']$. If we think of the gauge transformation $\boldsymbol{\theta} = g\boldsymbol{\theta}_{\mathcal{C}}$, as a change-of-variables, then term $\partial\boldsymbol{\theta}_{\mathcal{C}}/\partial\boldsymbol{\theta}_{\mathcal{C}'}$ is the Jacobian matrix transforming perturbations according to this change-of-variables.

Now we know that the perturbations of an invariant are the same irrespective of which gauge they are expressed in, $\Delta\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}}) = \Delta\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}'})$, and so subtracting equation (2.39) from (2.38), we obtain:

$$\nabla_{\theta}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^{\top}(\Delta\boldsymbol{\theta}_{\mathcal{C}} - \frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}_{\mathcal{C}'}}\Delta\boldsymbol{\theta}_{\mathcal{C}'}) = 0 \,. \tag{2.40}$$

This equation expresses an equivalence between perturbations $\Delta\boldsymbol{\theta}_{\mathcal{C}}$ and $\Delta\boldsymbol{\theta}_{\mathcal{C}'}$. Both resulted in the same perturbation of the invariant $\mathcal{I}$. We say that two perturbations are *geometrically equivalent* if they give the same perturbation for any invariant. We note

that in the special case, where the gauge manifolds intersect the gauge orbit at the same point and so $\boldsymbol{\theta}_{\mathcal{C}} = \boldsymbol{\theta}_{\mathcal{C}'}$, the Jacobian matrix, $\partial\boldsymbol{\theta}_{\mathcal{C}}/\partial\boldsymbol{\theta}_{\mathcal{C}'}$, is the identity.

From equation (2.40) and Theorem 2.3.1 we can deduce the following theorem:

Theorem 2.4.1. *Given a point $\boldsymbol{\theta}$ in two gauges $\mathcal{C}$ and $\mathcal{C}'$, perturbations in these gauges, $\Delta\boldsymbol{\theta}_{\mathcal{C}}$ and $\Delta\boldsymbol{\theta}_{\mathcal{C}'}$, are geometrically equivalent if and only if*

$$\boldsymbol{u}^{\top}(\Delta\boldsymbol{\theta}_{\mathcal{C}} - \frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}_{\mathcal{C}'}}\Delta\boldsymbol{\theta}_{\mathcal{C}'}) = 0\,, \qquad \forall\boldsymbol{u} \in T_{\theta}[\mathcal{G}_{\boldsymbol{\theta}}]^{\perp}\,. \tag{2.41}$$

We denote this relationship as:

$$\Delta\boldsymbol{\theta}_{\mathcal{C}} \equiv \Delta\boldsymbol{\theta}_{\mathcal{C}'} \mod \mathcal{G}_{\boldsymbol{\theta}} \tag{2.42}$$

Intuitively this states that the difference between two geometrically equivalent perturbations is in the tangent plane to the gauge orbit at that point. This is illustrated in Figure 2.6.

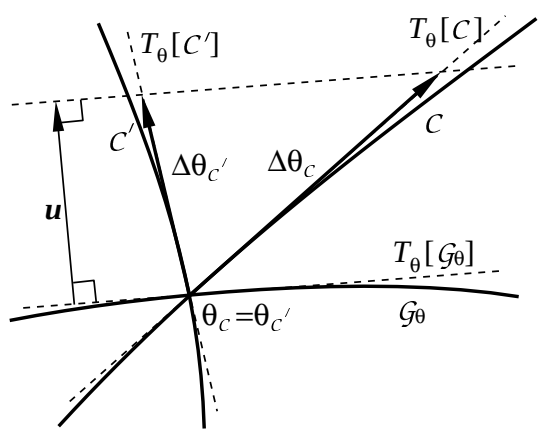


Fig. 2.6. Perturbations in the tangent planes to two gauges, $\mathcal{C}$ and $\mathcal{C}'$, are shown. These perturbations are geometrically equivalent when their difference, $\Delta\boldsymbol{\theta}_{\mathcal{C}} - \Delta\boldsymbol{\theta}_{\mathcal{C}'}$, is orthogonal to all vectors $\boldsymbol{u} \in T_{\theta}[\mathcal{G}_{\boldsymbol{\theta}}]^{\perp}$ as stated in Theorem 2.4.1. For simplicity, only the special case where the gauges intersect on the gauge orbit, and so the Jacobian matrix is the identity, is illustrated. For the full case see Figure 2.7.

2.4.4 Oblique Projection

Now that we have a measure for comparing perturbations and determining if they are geometrically equivalent, we would like a tool for transforming perturbations from one gauge to another in a way that they remain geometrically equivalent. If we can do this,

then it will not matter in which gauge we perform an uncertainty analysis, since we can always transform it into any other gauge.

We pose our problem as follows. Given an arbitrary perturbation, $\Delta\boldsymbol{\theta}$, at a point $\boldsymbol{\theta}$, but in an unknown gauge, we would like to calculate the geometrically equivalent perturbation, $\Delta\boldsymbol{\theta}_{\mathcal{C}}$, in a known gauge, $\mathcal{C}$.

Let $\mathcal{G}_{\boldsymbol{\theta}}$ be the gauge orbit to which point $\boldsymbol{\theta}$ belongs. We can calculate the point $\boldsymbol{\theta}_{\mathcal{C}}$ at the intersection of the orbit and the gauge: $\boldsymbol{\theta}_{\mathcal{C}} \in \mathcal{G}_{\boldsymbol{\theta}} \cap \mathcal{C}$. This gives us the gauge transformation g to the new point: $\boldsymbol{\theta}_{\mathcal{C}} = g\boldsymbol{\theta}$, and the Jacobian matrix, $\partial\boldsymbol{\theta}_{\mathcal{C}}/\partial\boldsymbol{\theta}$, that maps perturbation $\Delta\boldsymbol{\theta}$ to a new perturbation which we denote as $\Delta\boldsymbol{\theta}'$:

$$\Delta\boldsymbol{\theta}' = \frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}\Delta\boldsymbol{\theta}\,. \tag{2.43}$$

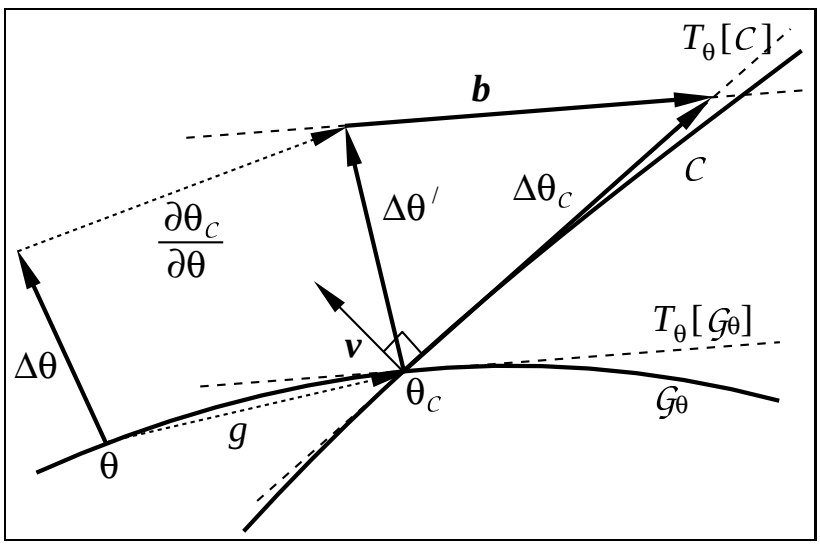


Fig. 2.7. Given a perturbation, $\Delta\boldsymbol{\theta}$, at a point $\boldsymbol{\theta}$ on gauge orbit $\mathcal{G}_{\boldsymbol{\theta}}$, we want to find the geometrically equivalent perturbation, $\Delta\boldsymbol{\theta}_{\mathcal{C}}$, in gauge $\mathcal{C}$. First we find the gauge transformation: $\boldsymbol{\theta}_{\mathcal{C}} = g\boldsymbol{\theta}$ so that $\boldsymbol{\theta}_{\mathcal{C}}$ is in gauge $\mathcal{C}$, and then we find the Jacobian transformation: $\partial\boldsymbol{\theta}_{\mathcal{C}}/\partial\boldsymbol{\theta}$. Then we apply an oblique projection onto the tangent space to the gauge manifold $\mathcal{C}$: $\Delta\boldsymbol{\theta}_{\mathcal{C}} = \Delta\boldsymbol{\theta}' + \boldsymbol{b} = \boldsymbol{Q}\Delta\boldsymbol{\theta}'$, to give us our final perturbation.

We can now express our unknown perturbation in terms of the original perturbation and an unknown vector $\boldsymbol{b}$:

$$\begin{aligned}\Delta\boldsymbol{\theta}_{\mathcal{C}} &= \frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}\Delta\boldsymbol{\theta} + \boldsymbol{b}\\ &= \Delta\boldsymbol{\theta}' + \boldsymbol{b}\end{aligned} \tag{2.44}$$

In order to ensure that $\Delta\boldsymbol{\theta}_{\mathcal{C}}$ is geometrically equivalent to $\Delta\boldsymbol{\theta}'$, we conclude from Theorem 2.4.1 that we can only alter the perturbation $\Delta\boldsymbol{\theta}'$ by components in the tangent space

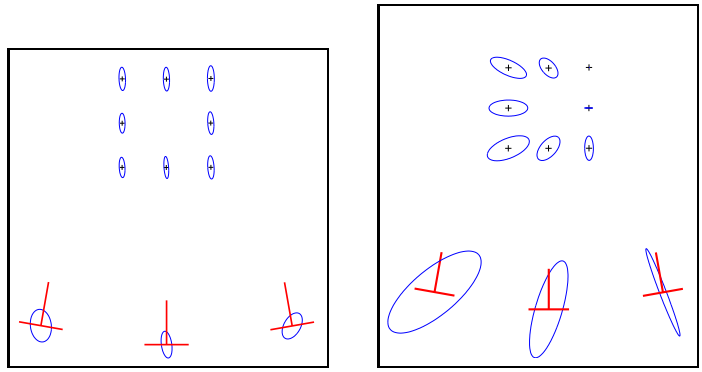


Fig. 2.8. Here we illustrate the effect of an oblique projection matrix, $\boldsymbol{Q}$, on perturbations defined by a covariance matrix. The state vector, $\boldsymbol{\theta}$, which includes shape and motion, is the same in both plots. In the right plot, we have defined a gauge, $\mathcal{C}$, which consists of coordinate system and scale on these points. The oblique projection, $\boldsymbol{Q}$, transforms the ellipses in the left plot to those in the right plot.

to the gauge orbit at $\boldsymbol{\theta}_{\mathcal{C}}$. Thus $\boldsymbol{b}$ is a vector in the tangent space to the gauge orbit. Let matrix $U_{\boldsymbol{\theta}_{\mathcal{C}}}$ have r columns that span the linear space of infinitesimal gauge generators $\mathcal{D}_{\theta}(\boldsymbol{\theta})$ from equation (2.27). Vector $\boldsymbol{b}$ must be in the column space of $U_{\boldsymbol{\theta}_{\mathcal{C}}}$, and hence can be written as:

$$\boldsymbol{b} = U_{\boldsymbol{\theta}_{\mathcal{C}}} \boldsymbol{x} \tag{2.45}$$

where $\boldsymbol{x} = (x_1, \ldots, x_r)^{\top}$ is a vector of r unknown coefficients.

If our gauge $\mathcal{C}$ is defined by r equations, $c_1(\boldsymbol{\theta}) = 0, \ldots, c_r(\boldsymbol{\theta}) = 0$, then the orthogonal complement to the tangent space to the gauge is spanned by the gradients of each of these: $\{\nabla_{\theta} c_1(\boldsymbol{\theta}), \ldots, \nabla_{\theta} c_r(\boldsymbol{\theta})\}$. Let $V_{\boldsymbol{\theta}_{\mathcal{C}}}$ be a matrix whose r columns are made up from these r gradients:

$$V_{\boldsymbol{\theta}_{\mathcal{C}}} = \Big(\nabla_{\theta} c_1(\boldsymbol{\theta}), \ldots, \nabla_{\theta} c_r(\boldsymbol{\theta}) \Big) , \tag{2.46}$$

and so it follows that:

$$V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} \Delta \boldsymbol{\theta}_{\mathcal{C}} = 0 . \tag{2.47}$$

Substituting expressions for $\Delta \boldsymbol{\theta}_{\mathcal{C}}$ and $\boldsymbol{b}$ from equations (2.44) and (2.45), we get:

$$V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} \Delta \boldsymbol{\theta}' + V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} U_{\boldsymbol{\theta}_{\mathcal{C}}} \boldsymbol{x} = 0. \tag{2.48}$$

Now solving for $\boldsymbol{x}$ we find

$$\boldsymbol{x} = -(V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} U_{\boldsymbol{\theta}_{\mathcal{C}}})^{-1} V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} \Delta \boldsymbol{\theta}', \tag{2.49}$$

and finally our new perturbation

$$\begin{aligned}\Delta\boldsymbol{\theta}_{\mathcal{C}} &= \Delta\boldsymbol{\theta}' - U_{\boldsymbol{\theta}_{\mathcal{C}}}(V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top}U_{\boldsymbol{\theta}_{\mathcal{C}}})^{-1}V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top}\Delta\boldsymbol{\theta}' \\ &= \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}\Delta\boldsymbol{\theta}' \\ &= \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}\frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}\Delta\boldsymbol{\theta} \end{aligned} \tag{2.50}$$

where

$$\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}} = I - U_{\boldsymbol{\theta}_{\mathcal{C}}}(V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top}U_{\boldsymbol{\theta}_{\mathcal{C}}})^{-1}V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top}. \tag{2.51}$$

The matrix $\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}$ is thus our oblique projection operator. It takes any vector at $\boldsymbol{\theta}_{\mathcal{C}}$ and obliquely projects it along the tangent space to the gauge orbit, $T_{\theta}[\mathcal{G}_{\boldsymbol{\theta}}]$, and onto the tangent space to the gauge, $T_{\theta}[\mathcal{C}]$.

It turns out that Baarda's S-Transformations [2] for free-network analysis are equivalent to to this oblique projection. Our gauge-based derivation of this transformation gives us intuitive insight into the need for this transformation, and into the effects of it on parameter perturbations.

The oblique projection operator is useful for covariance transformation as well. If we multiply both sides of equation (2.50) by its transpose and take expectations we obtain:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}\frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}\,\boldsymbol{V}_{\boldsymbol{\theta}}\,\frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}^{\top}\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}\top}. \tag{2.52}$$

This equation will serve as our basic transformation between gauges. In order to confirm that it is the correct transformation and that it does not vary the inherent geometric properties of the variance, we will create an equivalence test.

2.4.5 Geometric Equivalence II

We have derived a geometric equivalence relationship for perturbations, and now we would like one for covariances. Assume that we know the covariance of the solution in a particular gauge, $\mathcal{C}$. The covariance, $\boldsymbol{V}_{\mathcal{C}} = E[\Delta\boldsymbol{\theta}_{\mathcal{C}}\,\Delta\boldsymbol{\theta}_{\mathcal{C}}{}^{\top}]$, at a point $\boldsymbol{\theta}_{\mathcal{C}} \in \mathcal{C}$ is defined in the tangent plane to the gauge $\Delta\boldsymbol{\theta}_{\mathcal{C}} \in T_{\theta}[\mathcal{C}]$. Let $\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})$ be an invariant to the gauge orbit. The covariance of this invariant is given by:

$$\begin{aligned}\boldsymbol{V}_{\boldsymbol{I}} &= E[\Delta\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})\,\Delta\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^{\top}] \\ &= \nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^{\top}\,E[\Delta\boldsymbol{\theta}_{\mathcal{C}}\,\Delta\boldsymbol{\theta}_{\mathcal{C}}{}^{\top}]\,\nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}}) \\ &= \nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^{\top}\,\boldsymbol{V}_{\mathcal{C}}\,\nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})\,. \end{aligned} \tag{2.53}$$

Now let $\boldsymbol{V}_{\mathcal{C}'}$ be the covariance of the solution in another gauge, $\mathcal{C}'$ containing the point $\boldsymbol{\theta}_{\mathcal{C}'} = g\boldsymbol{\theta}_{\mathcal{C}}$. We can derive in an identical way the covariance for the invariant:

$$\begin{aligned} \boldsymbol{V_I} &= \nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}'})^\top \boldsymbol{V}_{\mathcal{C}'} \nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}'}) \\ &= \nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}})^\top \frac{\partial \boldsymbol{\theta}_{\mathcal{C}}}{\partial \boldsymbol{\theta}_{\mathcal{C}'}} \boldsymbol{V}_{\mathcal{C}'} \frac{\partial \boldsymbol{\theta}_{\mathcal{C}}}{\partial \boldsymbol{\theta}_{\mathcal{C}'}}^\top \nabla_{\boldsymbol{\theta}}\mathcal{I}(\boldsymbol{\theta}_{\mathcal{C}}) \end{aligned} \tag{2.54}$$

We see that covariances defined in any gauge on the gauge orbit are equivalent in the sense that from any of them we can obtain the same covariance of any invariant. From this and Theorem 2.4.1 we obtain the following corollary:

Corollary 2.4.1. *Given a point $\boldsymbol{\theta}$ in two gauges $\mathcal{C}$ and $\mathcal{C}'$, their covariances, $\boldsymbol{V}_{\mathcal{C}}$ and $\boldsymbol{V}_{\mathcal{C}'}$ respectively, are geometrically equivalent if and only if*

$$\boldsymbol{u}^\top (\boldsymbol{V}_{\mathcal{C}} - \frac{\partial \boldsymbol{\theta}_{\mathcal{C}}}{\partial \boldsymbol{\theta}_{\mathcal{C}'}} \boldsymbol{V}_{\mathcal{C}'} \frac{\partial \boldsymbol{\theta}_{\mathcal{C}}}{\partial \boldsymbol{\theta}_{\mathcal{C}'}}^\top) \boldsymbol{u} = 0 \,, \qquad \forall \boldsymbol{u} \in T_\theta[\mathcal{G}_{\boldsymbol{\theta}}]^\perp \,. \tag{2.55}$$

We denote this relationship for covariance matrices as:

$$\boldsymbol{V}_{\mathcal{C}} \equiv \boldsymbol{V}_{\mathcal{C}'} \mod \mathcal{G}_{\boldsymbol{\theta}} \,. \tag{2.56}$$

We call this our geometric equivalence relationship for covariances.

This criterion can be used to prove that our oblique projection operator, defined in equation (2.51), maintains geometric equivalence for covariances. Let $\boldsymbol{V}$ be a covariance of our parameter vector $\boldsymbol{\theta}$ defined in an unknown gauge. Let us define an arbitrary gauge $\mathcal{C}$ that passes through this point so that $\boldsymbol{\theta}_{\mathcal{C}} = \boldsymbol{\theta}$ and so the Jacobian matrix is the identity. The obliquely projected covariance in gauge $\mathcal{C}$ is then obtained as:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}} \boldsymbol{V} \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}{}^\top \,. \tag{2.57}$$

We can show that these covariances are geometrically equivalent, $\boldsymbol{V}_{\mathcal{C}} \equiv \boldsymbol{V} \mod \mathcal{G}_{\boldsymbol{\theta}}$, as follows. Expanding this expression for $\boldsymbol{V}_{\mathcal{C}}$, using our definition for $\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}$ in equation (2.51), we obtain:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{V} - U_{\boldsymbol{\theta}_{\mathcal{C}}} A - B U_{\boldsymbol{\theta}_{\mathcal{C}}}^\top + U_{\boldsymbol{\theta}_{\mathcal{C}}} C U_{\boldsymbol{\theta}_{\mathcal{C}}}^\top \,, \tag{2.58}$$

where A, B, and C are matrix combinations of $\boldsymbol{V}$ and components of $\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}$. The matrix $U_{\boldsymbol{\theta}_{\mathcal{C}}}$ spans the tangent plane to the gauge orbit, and hence the right three elements of this equation, which are linear combinations of these tangent vectors, will be eliminated when multiplied by all $\boldsymbol{u} \in T_\theta[\mathcal{G}_{\boldsymbol{\theta}}]^\perp$. Thus when we substitute into equation (2.55), we find that $\boldsymbol{V}_{\mathcal{C}}$ and $\boldsymbol{V}$ are equivalent, and so conclude that the oblique projection maintains equivalence. This confirms that our basic transformation-of-covariances equation (2.52) gives the geometrically equivalent covariance within a desired gauge.

2.4.6 The Normal Covariance

As we shall see in subsequent chapters, there are still reasons for calculating the covariance of the parameters directly (rather than just invariants), and to do this we must work in particular gauges. We assume then that we have achieved the Maximum Likelihood solution for the residual $J(\boldsymbol{\theta})$. We can expand this around the minimum and take derivatives with respect to perturbations $\Delta\boldsymbol{\theta}$ as done in equations (2.14) and (2.15) to get the following first order equation governing perturbations:

$$\nabla_{\boldsymbol{\theta}} J + \nabla^2_{\boldsymbol{\theta}} J \Delta\boldsymbol{\theta} = 0\,. \tag{2.59}$$

In solving for the perturbations, $\Delta\boldsymbol{\theta}$, we must face the issue of indeterminacies which cause the Hessian, $\nabla^2_{\boldsymbol{\theta}} J$, to be singular. If $\boldsymbol{\theta}$ contains n parameters, and there are r local degrees of freedom, the Hessian will have rank $n - r$. A perturbation will have two components, one perpendicular to the tangent plane to the gauge orbit and one parallel to the tangent plane,

$$\Delta\boldsymbol{\theta} = \Delta\boldsymbol{\theta}_{||\mathcal{G}} + \Delta\boldsymbol{\theta}_{\perp\mathcal{G}}\,, \tag{2.60}$$

where $\Delta\boldsymbol{\theta}_{||\mathcal{G}} \in T_\theta[\mathcal{G}]$ and $\Delta\boldsymbol{\theta}_{\perp\mathcal{G}} \in T_\theta[\mathcal{G}]^\perp$. Now for any vector $\boldsymbol{u} \in T_\theta[\mathcal{G}]^\perp$, we have $\boldsymbol{u}^\top \Delta\boldsymbol{\theta}_{||\mathcal{G}} = 0$. And so from Theorem 2.4.1 we deduce the following corollary:

Corollary 2.4.2. *Any perturbation in the form: $\Delta\boldsymbol{\theta} = \Delta\boldsymbol{\theta}_{||\mathcal{G}} + \Delta\boldsymbol{\theta}_{\perp\mathcal{G}}$, where $\Delta\boldsymbol{\theta}_{||\mathcal{G}} \in T_\theta[\mathcal{G}]$ and $\Delta\boldsymbol{\theta}_{\perp\mathcal{G}} \in T_\theta[\mathcal{G}]^\perp$, is geometrically equivalent to $\Delta\boldsymbol{\theta}_{\perp\mathcal{G}}$. Namely we have:*

$$\Delta\boldsymbol{\theta} \equiv \Delta\boldsymbol{\theta}_{\perp\mathcal{G}} \bmod \mathcal{G}_{\boldsymbol{\theta}}\,. \tag{2.61}$$

The gradient of J must be orthogonal to the tangent space to the gauge orbit, and so applying the Moore-Penrose generalized inverse[3] to invert equation (2.59) we can solve for the orthogonal perturbation:

$$\begin{aligned}\Delta\boldsymbol{\theta}_{\perp\mathcal{G}} &= (\nabla^2_{\boldsymbol{\theta}} J)^-_{n-r} \nabla_{\boldsymbol{\theta}} J \\ &= (\boldsymbol{J}_{\boldsymbol{\theta}})^-_{n-r}\, \boldsymbol{l}_{\boldsymbol{\theta}}\end{aligned} \tag{2.63}$$

[3] The Moore-Penrose generalized inverse was defined by Penrose [56] for a real matrix A of rank N as being the unique matrix A^-_N satisfying:

$$\begin{aligned} AA^-_N A &= A, \\ A^-_N A A^-_N &= A^-_N, \\ (AA^-_N)^\top &= AA^-_N, \\ (A^-_N A)^\top &= A^-_N A. \end{aligned} \tag{2.62}$$

It can be obtained by the following procedure. If $A = U\Lambda V^\top$ by SVD, then the Moore-Penrose inverse is given as $A^-_N = V\Lambda^-_N U^\top$, where Λ^-_N has the first N singular values inverted on the diagonal, and the rest ensured to be zero.

We call the covariance of this perturbation the *normal covariance*:

$$\begin{aligned}\boldsymbol{V}_{\perp\mathcal{G}} &= E[\Delta\boldsymbol{\theta}_{\perp\mathcal{G}}\,\Delta\boldsymbol{\theta}_{\perp\mathcal{G}}{}^{\top}] \\ &= (\boldsymbol{J}_{\boldsymbol{\theta}})^{-}_{n-r}\,E[\boldsymbol{l}_{\boldsymbol{\theta}}\,\boldsymbol{l}_{\boldsymbol{\theta}}^{\top}]\,(\boldsymbol{J}_{\boldsymbol{\theta}})^{-}_{n-r} \\ &= (\boldsymbol{J}_{\boldsymbol{\theta}})^{-}_{n-r}\boldsymbol{J}_{\boldsymbol{\theta}}(\boldsymbol{J}_{\boldsymbol{\theta}})^{-}_{n-r} \\ &= (\boldsymbol{J}_{\boldsymbol{\theta}})^{-}_{n-r}\,. \end{aligned} \tag{2.64}$$

The normal covariance can be thought of as a gauge-independent covariance. It depends only on the parametrization of the solution, not on any particular choice of gauge. Or more correctly we could say the normal covariance is in the gauge orthogonal to the gauge orbit. The normal covariance is geometrically equivalent to any covariance defined in any gauge manifold.

We can calculate the covariance in any gauge by obliquely projecting the normal covariance onto the gauge tangent space, as explained in section 2.4.4. If $\boldsymbol{V}_{\perp\mathcal{G}}$ is defined at point $\boldsymbol{\theta}$, then performing this projection into gauge $\mathcal{C}$ we find:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}\frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}\,\boldsymbol{V}_{\perp\mathcal{G}}\,\frac{\partial\boldsymbol{\theta}_{\mathcal{C}}}{\partial\boldsymbol{\theta}}^{\top}\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}{}^{\top}\,, \tag{2.65}$$

but if $\boldsymbol{V}_{\perp\mathcal{G}}$ is defined at point $\boldsymbol{\theta}_{\mathcal{C}}$, then the Jacobian becomes the identity and we have simply:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}\,\boldsymbol{V}_{\perp\mathcal{G}}\,\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}{}^{\top}\,. \tag{2.66}$$

Alternatively, if we wish to obtain the normal covariance at point $\boldsymbol{\theta}_{\mathcal{C}}$, we simply need to project any covariance at that point onto the orthogonal complement of the tangent to the gauge orbit:

$$\boldsymbol{V}_{\perp\mathcal{G}} = \boldsymbol{P}_{\boldsymbol{\theta}}\,\boldsymbol{V}_{\mathcal{C}}\,\boldsymbol{P}_{\boldsymbol{\theta}}^{\top}\,, \tag{2.67}$$

where $\boldsymbol{P}_{\boldsymbol{\theta}} = \boldsymbol{I} - U_{\boldsymbol{\theta}_{\mathcal{C}}}(U_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top}U_{\boldsymbol{\theta}_{\mathcal{C}}})^{-1}U_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top}$.

We see that if we choose a gauge whose tangent space is orthogonal to the gauge orbit, $T_{\theta}[\mathcal{C}] = T_{\theta}[\mathcal{G}]^{\perp}$, then $\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}} = \boldsymbol{P}_{\boldsymbol{\theta}}$, and the covariance in this gauge is just the normal covariance. This is illustrated in Figure 2.9.

The normal covariance gives us a lower bound on the global accuracy of the solution. This is stated in the following theorem:

Theorem 2.4.2. *For any positive semidefinite covariance, $\boldsymbol{V}_{\mathcal{C}}$, and normal covariance, $\boldsymbol{V}_{\perp\mathcal{G}}$, defined in equation (2.67), we have:*

$$\mathrm{Tr}[\boldsymbol{V}_{\perp\mathcal{G}}] \leq \mathrm{Tr}[\boldsymbol{V}_{\mathcal{C}}] \qquad \forall\mathcal{C} \;\; \textit{such that}\; \mathcal{C}\cap\mathcal{G}_{\boldsymbol{\theta}} = \boldsymbol{\theta}_{\mathcal{C}} \tag{2.68}$$

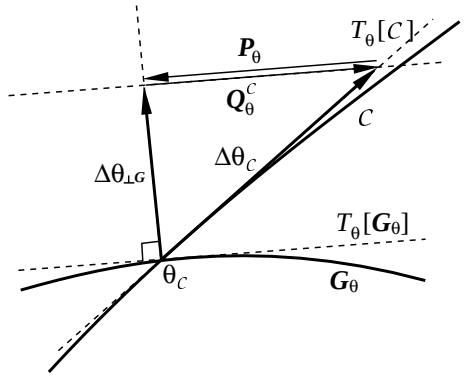


Fig. 2.9. A perturbation is projected onto the tangent plane to gauge $\mathcal{C}$ with $\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}$, whereas it can be projected onto the normal gauge with $\boldsymbol{P}_{\boldsymbol{\theta}}$.

Proof. Matrix $\boldsymbol{V}_{\mathcal{C}}$ can be decomposed in the form: $\boldsymbol{V}_{\mathcal{C}} = \sum_{i=1}^{n} \boldsymbol{v}_i \boldsymbol{v}_i^\top$, and then its trace is given by:

$$\mathrm{Tr}[\boldsymbol{V}_{\mathcal{C}}] = \sum_{i=1}^{n} \boldsymbol{v}_i^\top \boldsymbol{v}_i. \tag{2.69}$$

In a similar way, the trace of $\boldsymbol{V}_{\perp\mathcal{G}}$ can be expressed as:

$$\mathrm{Tr}[\boldsymbol{V}_{\perp\mathcal{G}}] = \sum_{i=1}^{n} (\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i)^\top (\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i). \tag{2.70}$$

Now any projection matrix $\boldsymbol{P}_{\boldsymbol{\theta}}$ can be decomposed in the form:

$$\boldsymbol{P}_{\boldsymbol{\theta}} = \boldsymbol{I} - \sum_{j=1}^{r} \frac{\boldsymbol{a}_j \boldsymbol{a}_j^\top}{\|\boldsymbol{a}_j\|^2}, \tag{2.71}$$

where $\boldsymbol{a}_j^\top \boldsymbol{a}_i = 0$ for $i \neq j$. From this we obtain:

$$\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i = \boldsymbol{v}_i - \sum_{j=1}^{r} \frac{\boldsymbol{v}_i^\top \boldsymbol{a}_j \boldsymbol{a}_j}{\|\boldsymbol{a}_j\|^2}. \tag{2.72}$$

and then expanding the product $(\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i)^\top (\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i)$ we get:

$$(\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i)^\top (\boldsymbol{P}_{\boldsymbol{\theta}} \boldsymbol{v}_i) = \boldsymbol{v}_i^\top \boldsymbol{v}_i - \sum_{j=1}^{r} \frac{(\boldsymbol{v}_i^\top \boldsymbol{a}_j)^2}{\|\boldsymbol{a}_j\|^2}. \tag{2.73}$$

We see that this is always less than or equal to $\boldsymbol{v}_i^\top \boldsymbol{v}_i$ for all i, and so conclude that $\mathrm{Tr}[\boldsymbol{V}_{\perp\mathcal{G}}] \leq \mathrm{Tr}[\boldsymbol{V}_{\mathcal{C}}]$. □

We note that the trace includes translation and rotation terms, and hence if we use this lower bound we should parametrize them such that the relative weighting of rotations and translations is appropriate.

2.4.7 A Test for Geometric Equivalence

A useful property of the normal covariance is that its nullspace is the same as the tangent space to the gauge orbit. This is not the case for an arbitrary covariance matrix, but is true for the normal covariance since it is defined in a gauge orthogonal to the gauge orbit. We can use this fact to derive a geometric equivalence test for normal covariances.

Now suppose we wish to test whether maxtrix, $\boldsymbol{V}_A$ is geometrically equivalent to $\boldsymbol{V}_{\perp\mathcal{G}}$. Assume first that they are calculated at the same point, $\boldsymbol{\theta}_{\mathcal{C}}$, on the gauge orbit, and hence that the Jacobian matrix is unity. If they are not, then transform $\boldsymbol{V}_A$ using the Jacobian so it is defined at $\boldsymbol{\theta}_{\mathcal{C}}$. The Moore-Penrose inverse of a positive semi-definite matrix, $\boldsymbol{V}_{\perp\mathcal{G}}^{-}$, has the same nullspace as $\boldsymbol{V}_{\perp\mathcal{G}}$, and we note that it is exactly the Hessian: $\boldsymbol{V}_{\perp\mathcal{G}}^{-} = \boldsymbol{J}_{\boldsymbol{\theta}}$.

Theorem 2.4.3. *Covariance $\boldsymbol{V}_A$ defined at point $\boldsymbol{\theta}_{\mathcal{C}}$ is geometrically equivalent to $\boldsymbol{V}_{\perp\mathcal{G}}$ if and only if*

$$\boldsymbol{V}_{\perp\mathcal{G}}^{-}\boldsymbol{V}_A\boldsymbol{V}_{\perp\mathcal{G}}^{-} = \boldsymbol{V}_{\perp\mathcal{G}}^{-}\,. \tag{2.74}$$

Proof. We can write matrix $\boldsymbol{V}_A = \boldsymbol{V}_{\perp\mathcal{G}} + D$, where $D = \boldsymbol{V}_A - \boldsymbol{V}_{\perp\mathcal{G}}$. Substituting this expansion into equation (2.74) we obtain

$$\boldsymbol{V}_{\perp\mathcal{G}}^{-}D\boldsymbol{V}_{\perp\mathcal{G}}^{-} = 0\,. \tag{2.75}$$

This is true only if all the rows or all the columns of D are in the nullspace of $\boldsymbol{V}_{\perp\mathcal{G}}$. If this is the case, then we see that our geometric equivalence relationship, equation (2.55), is satisfied. Conversely we see that the geometric equivalence relationship is only satisfied when $\boldsymbol{u}^\top D\boldsymbol{u} = 0$ for all $\boldsymbol{u}$ in the nullspace of $\boldsymbol{V}_{\perp\mathcal{G}}$. □

Using the Hessian, $\boldsymbol{J}_{\boldsymbol{\theta}}$, for $\boldsymbol{V}_{\perp\mathcal{G}}^{-}$ in equation (2.74) gives us the equation:

$$\boldsymbol{J}_{\boldsymbol{\theta}}\boldsymbol{V}_A\boldsymbol{J}_{\boldsymbol{\theta}} = \boldsymbol{J}_{\boldsymbol{\theta}}. \tag{2.76}$$

This means that any generalized inverse $\boldsymbol{V}_A = \boldsymbol{J}_{\boldsymbol{\theta}}^{\dagger}$ defined by this relationship gives a covariance that is geometrically equivalent to $\boldsymbol{V}_{\perp\mathcal{G}}$. This was pointed out by Triggs [71].

2.4.8 Alternative Constrained Covariance Methods

There are alternative methods for calculating the covariance in a gauge. One approach is to find a minimal re-parametrization by eliminating r parameters. This, however, may not be possible for the desired constraint, or if it is possible, it may involve very complicated re-arrangements of the equations.

Some standard approaches involve use of Lagrange multipliers to enforce the constraints during error propagation as described by Haralick [19]. It turns out that if the Fisher information matrix $\boldsymbol{J}_{\boldsymbol{\theta}}$ is calculated at $\boldsymbol{\theta}_{\mathcal{C}}$, and $V_{\boldsymbol{\theta}_{\mathcal{C}}}$ whose domain is orthogonal to the gauge $\mathcal{C}$ is known, then to calculate the covariance $\boldsymbol{V}_{\mathcal{C}}$ in this gauge we can use the expression:

$$\left(\begin{array}{c|c} \boldsymbol{V}_{\mathcal{C}} & A \\ \hline A^{\top} & B \end{array}\right) = \left(\begin{array}{c|c} \boldsymbol{J}_{\boldsymbol{\theta}} & V_{\boldsymbol{\theta}_{\mathcal{C}}} \\ \hline V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} & 0 \end{array}\right)^{-1} . \tag{2.77}$$

Here A and B are padding related to the Lagrange multipliers.

Thus while this and other computational methods can find the constrained covariances, it is difficult to obtain an explicit geometric interpretation of them. But with the gauge freedom approach we can interpret the constraints as an oblique projection along the gauge freedoms, and so intuitively understand the effect of the indeterminacies and the constraints. We will examine this more closely in the following section.

2.4.9 An Interpretation of the Oblique Projection

We have illustrated the action of the oblique projection operator, $\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}$, on a single vector $\Delta\boldsymbol{\theta}'$, in Figure 2.7. What we would like to understand better is how this operator transforms covariance matrices as in our basic transformation equation (2.52) which we summarize as:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}\, \boldsymbol{V}\, \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}\,\top} . \tag{2.78}$$

The covariance matrix $\boldsymbol{V}$ is positive semi-definite, and hence can be decomposed in the form:

$$\begin{aligned} \boldsymbol{V} &= \begin{pmatrix} \boldsymbol{u}_1 \cdots \boldsymbol{u}_n \end{pmatrix} \begin{pmatrix} \sigma_1^2 & & & & \\ & \ddots & & & \\ & & \sigma_{n-r}^2 & & \\ & & & 0 & \\ & & & & \ddots \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_1 \cdots \boldsymbol{u}_n \end{pmatrix}^{\top} \\ &= \begin{pmatrix} \boldsymbol{v}_1 \cdots \boldsymbol{v}_{n-r} \end{pmatrix} \begin{pmatrix} \boldsymbol{v}_1 \cdots \boldsymbol{v}_{n-r} \end{pmatrix}^{\top} , \end{aligned} \tag{2.79}$$

where matrix $(\boldsymbol{u}_1 \cdots \boldsymbol{u}_n)$ is unitary. The vectors $\boldsymbol{v}_i$ are all orthogonal and their magnitudes are given by the standard deviation σ_i. They are the radii of the hyper-ellipsoid defined by covariance $\boldsymbol{V}$. We see that when we transform the covariance with the oblique

projection operator as in equation (2.78), we actually individually project each of these radii:

$$\boldsymbol{v}_i' = \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}} \boldsymbol{v}_i \,, \tag{2.80}$$

onto the tangent space $T_\theta[\mathcal{C}]$. Doing this to all the radii will form a new hyper-ellipsoid, representing $\boldsymbol{V}_{\mathcal{C}}$, in this tangent space. The radii of this projected ellipoid are obtained using an analogous expansion to that in equation (2.79), but for $\boldsymbol{V}_{\mathcal{C}}$.

Figure 2.10 illustrates the oblique projection of covariances along the gauge freedoms. All of the ellipses representing geometrically equivalent covariances lie on the same cylinder, and are formed by planes, intersecting this cylinder. These planes are the tangent spaces to the gauge constraints. The ellipse with smallest radii is the one on the plane orthogonal to the gauge freedoms, and it represents the normal covariance.

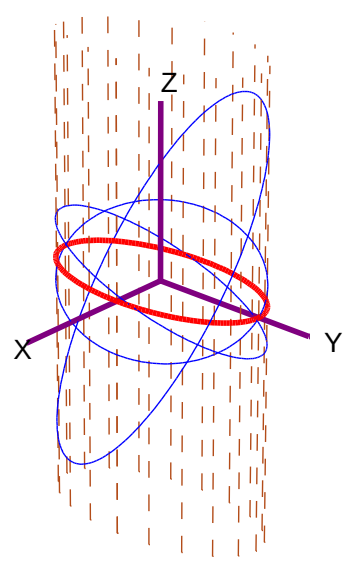


Fig. 2.10. We show the action of the oblique projection operator along the gauge freedoms, $T_\theta[\mathcal{G}_{\boldsymbol{\theta}}]$, indicated by the dashed lines in the Z direction. Ellipses representing covariances are formed on planes intersecting a cylinder. The thick–ellipse corresponds to the normal covariance.

2.5 Optimality with Gauge Freedoms

The optimality criterion for parametric fitting when there are gauge freedoms is similar to that without gauge freedoms. But now instead of there being a true solution point, there is a true gauge orbit or leaf. Any point on this gauge orbit is a "true" set of parameters.

2.5.1 Free-Gauge Optimization

Our cost function, $J(\boldsymbol{\theta})$, is a non-linear function of $\boldsymbol{\theta}$ and optimization requires a non-linear method. We used Levenberg-Marquardt, which combines gradient descent and Gauss-Newton, to minimize $J(\boldsymbol{\theta})$ starting from an initial approximate solution provided by a suboptimal method such as factorization [68].

A choice to be made in optimization is whether or not to apply the gauge constraints during optimization. If indeterminacies have been eliminated from the parameters, there is no choice but to minimize the cost in this constrained space. If, however, we keep the full parameter space, then it is possible to ignore the gauge constraints and proceed orthogonally to the gauge orbits in minimizing $J(\boldsymbol{\theta})$, as illustrated in Figure 2.11). An advantage of this is that minimization steps are not constrained to be in the tangent space to the constrained space. Staying in the constrained space could slow convergence if this space was incident at an oblique angle to the solution manifold. A Gauss-Newton step[4] along the tangent to the constrained space is given by:

$$\Delta\boldsymbol{\theta}_{\mathcal{C}} = -\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}(\nabla_{\boldsymbol{\theta}}^2\bar{J})_{n-r}^{-}\nabla_{\boldsymbol{\theta}}\bar{J}\,, \tag{2.81}$$

whereas a step orthogonal to the gauge orbit is simply:

$$\Delta\boldsymbol{\theta}_{\mathcal{C}} = -(\nabla_{\boldsymbol{\theta}}^2\bar{J})_{n-r}^{-}\nabla_{\boldsymbol{\theta}}\bar{J}\,. \tag{2.82}$$

This turns out to be of more theoretical interest that practical interest, since we found from our examples that the number of convergence steps using free–gauge minimization and using a fixed gauge minimization were usually the same, being around 6 or 7. Hence for our 3D Computer Vision problem the constraints do not sufficiently alter the parameter space to slow convergence, and so either method suffices. There may be other examples in which this is not the case, and free-gauge minimization may be advantageous.

[4] Levenberg-Marquardt modifies this slightly by adding a diagonal matrix to the Hessian before inverting it [59]. The actual implementation does not require inversion of the full Hessian as explained by Hartley [21].

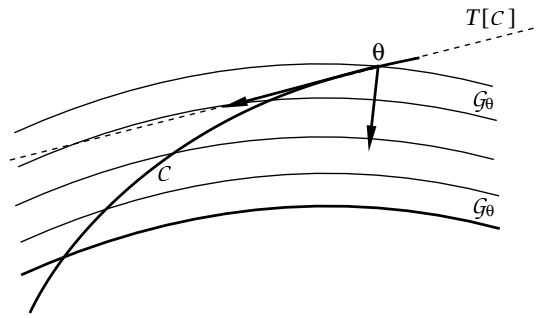


Fig. 2.11. In minimizing the cost $J(\boldsymbol{\theta})$, one could apply the gauge constraints and proceed in the tangent plane to the gauge manifold $T_\theta[\mathcal{C}]$. Or one could take a free-gauge approach and take steps orthogonally to the gauge orbit.

2.5.2 Extension of the Cramer-Rao Bound

To test whether our solution is optimal or not, we must have a measure of optimality when gauge freedoms exist. To do this we will extend the Cramer-Rao lower bound for estimators with unique solutions to the case where there are indeterminacies.

Let us work in gauge $\mathcal{C}$, and assume we have an unbiased estimator:

$$\hat{\boldsymbol{\theta}}_{\mathcal{C}} = \bar{\boldsymbol{\theta}}_{\mathcal{C}} + \Delta\boldsymbol{\theta}_{\mathcal{C}} \tag{2.83}$$

where we assume $\Delta\boldsymbol{\theta}_{\mathcal{C}}$ is small and so lies in the tangent space $T_\theta[\mathcal{C}]$ of dimension $n-r$. Our goal is to find a bound on the variance of this estimator.

First we notice that in this gauge we can locally reparametrize the model points with a $n-r$ dimensional vector $\boldsymbol{\phi}$,

$$\hat{\boldsymbol{\phi}} = \bar{\boldsymbol{\phi}} + \Delta\boldsymbol{\phi}\,, \tag{2.84}$$

where $\bar{\boldsymbol{\phi}}$ corresponds to the true solution point $\bar{\boldsymbol{\theta}}$, and $\hat{\boldsymbol{\phi}}$ is the estimator in this space. In this new local parametrization the estimator, $\hat{\boldsymbol{\phi}}$, has no gauge freedoms, and hence its Cramer-Rao lower bound is simply:

$$\boldsymbol{V}_{\boldsymbol{\phi}} \succ \boldsymbol{J}_{\boldsymbol{\phi}}^{-1}\,, \tag{2.85}$$

where $\boldsymbol{J}_{\boldsymbol{\phi}}$ is the Fisher information matrix in this parametrization, and $\boldsymbol{A} \succ \boldsymbol{B}$ means $\boldsymbol{A}$ - $\boldsymbol{B}$ is positive semi-definite.

We also note that there is a linear mapping between the parametrizations of our perturbations, $\Delta\boldsymbol{\phi}$ and $\Delta\boldsymbol{\theta}_{\mathcal{C}}$, in gauge $\mathcal{C}$. We denote it as:

$$\Delta\boldsymbol{\theta}_{\mathcal{C}} = \frac{\partial\boldsymbol{\theta}}{\partial\boldsymbol{\phi}}\Delta\boldsymbol{\phi} = M\Delta\boldsymbol{\phi}\,, \tag{2.86}$$

where M will denote the Jacobian matrix between these perturbations. Since the columns of M span the tangent space $T_\theta[\mathcal{C}]$, we can also include the oblique projection in this expression: $\Delta\boldsymbol{\theta}_\mathcal{C} = \boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}} M \Delta\boldsymbol{\phi}$.

The probability density, $\rho(\boldsymbol{\theta})$, is a gauge invariant, and so it can be expressed in any gauge, for example $\rho(\boldsymbol{\theta}_\mathcal{C})$, and also in the local gauge parametrization, $\rho(\boldsymbol{\phi})$. We can express the score, $\boldsymbol{l}_{\boldsymbol{\phi}}$, in this new parametrization and relate it to the score of the full parametrization, $\boldsymbol{l}_{\boldsymbol{\theta}}$ as follows:

$$\begin{aligned} \boldsymbol{l}_{\boldsymbol{\phi}} &= \nabla_{\boldsymbol{\phi}} \log \rho(\boldsymbol{\phi}) \\ &= \frac{\partial \boldsymbol{\theta}}{\partial \boldsymbol{\phi}}^\top \nabla_{\boldsymbol{\theta}} \log \rho(\boldsymbol{\theta}_\mathcal{C}) \\ &= M^\top \boldsymbol{l}_{\boldsymbol{\theta}_\mathcal{C}} \,. \end{aligned} \tag{2.87}$$

The Fisher information matrix, $\boldsymbol{J}_{\boldsymbol{\phi}}$ can be related to the full parametrization in a similar way:

$$\begin{aligned} \boldsymbol{J}_{\boldsymbol{\phi}} &= E[\boldsymbol{l}_{\boldsymbol{\phi}} \boldsymbol{l}_{\boldsymbol{\phi}}^\top] \\ &= M^\top E[\boldsymbol{l}_{\boldsymbol{\theta}_\mathcal{C}} \boldsymbol{l}_{\boldsymbol{\theta}_\mathcal{C}}^\top] M \\ &= M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M \,. \end{aligned} \tag{2.88}$$

The inverse of this is then simply:

$$(\boldsymbol{J}_{\boldsymbol{\phi}})^{-1} = (M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M)^{-1} \,. \tag{2.89}$$

Using our reparametrization of perturbations in gauge $\mathcal{C}$ from equation (2.86), we can express the covariance of perturbations $\Delta\boldsymbol{\theta}_\mathcal{C}$ as:

$$\boldsymbol{V}_{\boldsymbol{\theta}_\mathcal{C}} = E[\Delta\boldsymbol{\theta}_\mathcal{C} \Delta\boldsymbol{\theta}_\mathcal{C}{}^\top] = M \boldsymbol{V}_{\boldsymbol{\phi}} M^\top \,. \tag{2.90}$$

Now from the Cramer-Rao bound in equation (2.85), $(\boldsymbol{V}_{\boldsymbol{\phi}}$ - $\boldsymbol{J}_{\boldsymbol{\phi}}^{-1})$ is positive semi-definite, and so the quantity $M(\boldsymbol{V}_{\boldsymbol{\phi}} - \boldsymbol{J}_{\boldsymbol{\phi}}^{-1}) M^\top$ must be positive semi-definite. Expanding this we obtain:

$$\begin{aligned} M(\boldsymbol{V}_{\boldsymbol{\phi}} - \boldsymbol{J}_{\boldsymbol{\phi}}^{-1}) M^\top &= \boldsymbol{V}_{\boldsymbol{\theta}_\mathcal{C}} - M(\boldsymbol{J}_{\boldsymbol{\phi}}^{-1}) M^\top \\ &= \boldsymbol{V}_{\boldsymbol{\theta}_\mathcal{C}} - M(M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M)^{-1} M^\top \,, \end{aligned} \tag{2.91}$$

and the last expression is thus also positive semi-definite. We can show that the right-most term of equation (2.91) is a generalized inverse of $\boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}}$ if the following expression is true:

$$\boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} \left(M(M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M)^{-1} M^\top \right) \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} = \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} \,. \tag{2.92}$$

If we left multiply by $M^\top$ we obtain without loss of rank:

$$(M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M)(M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M)^{-1} M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} = M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} \,, \tag{2.93}$$

and so conclude that it is a generalized inverse of $\boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}}$. But there are many possible generalized inverses, and this expression is not an arbitrary one. We note that its column space and row space are spanned by the columns of M and hence it must lie in the tangent space to the gauge, $T_\theta[\mathcal{C}]$. We have seen that all the generalized inverses, $\boldsymbol{J}^\dagger_{\boldsymbol{\theta}_\mathcal{C}}$ are geometrically equivalent and we can obtain the one in the tangent space to the gauge $\mathcal{C}$ by obliquely projecting an arbitrary one to become: $\boldsymbol{Q}^\mathcal{C}_{\boldsymbol{\theta}} \boldsymbol{J}^\dagger_{\boldsymbol{\theta}_\mathcal{C}} \boldsymbol{Q}^{\mathcal{C}\,\top}_{\boldsymbol{\theta}}$. Thus we deduce that

$$(M(M^\top \boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}} M)^{-1} M^\top) = \boldsymbol{Q}^\mathcal{C}_{\boldsymbol{\theta}} \boldsymbol{J}^\dagger_{\boldsymbol{\theta}_\mathcal{C}} \boldsymbol{Q}^{\mathcal{C}\,\top}_{\boldsymbol{\theta}} \,, \tag{2.94}$$

for an arbitrary generalized inverse $\boldsymbol{J}^\dagger_{\boldsymbol{\theta}_\mathcal{C}}$.

Finally from equation (2.91) we obtain a bound on our covariance as:

$$\boldsymbol{V}_{\boldsymbol{\theta}_\mathcal{C}} \succ \boldsymbol{Q}^\mathcal{C}_{\boldsymbol{\theta}} \boldsymbol{J}^\dagger_{\boldsymbol{\theta}_\mathcal{C}} \boldsymbol{Q}^{\mathcal{C}\,\top}_{\boldsymbol{\theta}} \,, \tag{2.95}$$

where $\boldsymbol{A} \succ \boldsymbol{B}$ implies $\boldsymbol{A} - \boldsymbol{B}$ is positive semi-definite. This then is the Cramer-Rao lower bound for perturbations of an estimator $\hat{\boldsymbol{\theta}}_\mathcal{C}$ in gauge $\mathcal{C}$. The Moore-Penrose inverse is a particular generalized inverse and can be substituted in this expression:

$$\boldsymbol{V}_{\boldsymbol{\theta}_\mathcal{C}} \succ \boldsymbol{Q}^\mathcal{C}_{\boldsymbol{\theta}} (\boldsymbol{J}_{\boldsymbol{\theta}_\mathcal{C}})^-_{n-r} \boldsymbol{Q}^{\mathcal{C}\,\top}_{\boldsymbol{\theta}} \,. \tag{2.96}$$

2.5.3 Optimality of Maximum Likelihood Solution

Finally we would like to confirm that a Maximum Likelihood estimate is unbiased up to first order and achieves the Cramer-Rao lower bound even when there are indeterminacies. We will work in gauge $\mathcal{C}$, and start by expanding the cost function in the tangent space of this gauge:

$$J(\hat{\boldsymbol{\theta}}_\mathcal{C}) = J(\bar{\boldsymbol{\theta}}_\mathcal{C} + \Delta\boldsymbol{\theta}_\mathcal{C}) = \bar{J} + \nabla_{\boldsymbol{\theta}} \bar{J}^\top \Delta\boldsymbol{\theta}_\mathcal{C} + \frac{1}{2} \Delta\boldsymbol{\theta}_\mathcal{C}{}^\top \nabla^2_{\boldsymbol{\theta}} \bar{J} \Delta\boldsymbol{\theta}_\mathcal{C} + \ldots \,, \tag{2.97}$$

where we have denoted $\bar{J}$ to represent the cost at the true parameter value: $J(\bar{\boldsymbol{\theta}})$. If a perturbation of the data, $\Delta\boldsymbol{p}$, has noise $\mathcal{O}(\epsilon)$, then $\Delta\boldsymbol{\theta}_\mathcal{C}$ is also $\mathcal{O}(\epsilon)$, and the first three terms of the expansion in equation (2.97) are $\mathcal{O}(\epsilon^2)$. The rest of the terms are higher order, so ignoring them and differentiating with respect to $\Delta\boldsymbol{\theta}_\mathcal{C}$, we obtain:

$$\nabla_{\boldsymbol{\theta}} \bar{J} + \nabla^2_{\boldsymbol{\theta}} \bar{J} \Delta\boldsymbol{\theta}_\mathcal{C} = 0 \,. \tag{2.98}$$

The Hessian has rank $n - r$, and so in order to solve for a perturbation in the gauge tangent space, we multiply by the generalized inverse and then use $\boldsymbol{Q}^\mathcal{C}_{\boldsymbol{\theta}}$ to transform the solution:

$$\Delta\boldsymbol{\theta}_{\mathcal{C}} = -\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}(\nabla^2_{\boldsymbol{\theta}}\bar{J})^{-}_{n-r}\nabla_{\boldsymbol{\theta}}\bar{J}\,. \tag{2.99}$$

We can calculate the gradient and the Hessian of $\bar{J}$ as in section 2.1.2, and we find:

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\bar{J} &= 2\boldsymbol{l}_{\boldsymbol{\theta}_{\mathcal{C}}}\,,\\ \nabla^2_{\boldsymbol{\theta}}\bar{J} &= 2\boldsymbol{J}_{\boldsymbol{\theta}_{\mathcal{C}}}\,.\end{aligned} \tag{2.100}$$

Here $\boldsymbol{l}_{\boldsymbol{\theta}_{\mathcal{C}}}$ is $\mathcal{O}(\epsilon)$ and $\boldsymbol{J}_{\boldsymbol{\theta}_{\mathcal{C}}}$ is $\mathcal{O}(1)$. It is thus clear that the Maximum Likelihood solution is unbiased,

$$E[\Delta\boldsymbol{\theta}_{\mathcal{C}}] = -E[\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}(\boldsymbol{J}_{\boldsymbol{\theta}_{\mathcal{C}}})^{-}_{n-r}\boldsymbol{l}_{\boldsymbol{\theta}_{\mathcal{C}}}] = 0\,, \tag{2.101}$$

and this is true up to $\mathcal{O}(\epsilon)$, namely first order. The covariance is given by

$$\begin{aligned}V_{\boldsymbol{\theta}} &= E[\Delta\boldsymbol{\theta}_{\mathcal{C}}\,\Delta\boldsymbol{\theta}_{\mathcal{C}}{}^{\top}]\\ &= \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}(\boldsymbol{J}_{\boldsymbol{\theta}_{\mathcal{C}}})^{-}_{n-r}E[\boldsymbol{l}_{\boldsymbol{\theta}_{\mathcal{C}}}\boldsymbol{l}^{\top}_{\boldsymbol{\theta}_{\mathcal{C}}}](\boldsymbol{J}_{\boldsymbol{\theta}_{\mathcal{C}}})^{-}_{n-r}\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}{}^{\top}\\ &= \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}(\boldsymbol{J}_{\boldsymbol{\theta}_{\mathcal{C}}})^{-}_{n-r}\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}{}^{\top}\end{aligned} \tag{2.102}$$

and hence it achieves the Cramer-Rao lower bound.

2.6 Summary

Some parameter fitting tasks have inherent indeterminacies that cannot be eliminated simply by collecting more data. As a result some aspect of the model cannot be estimated. When we solve for a solution parameter set, we could let this aspect take on a random value, or constrain its value, or eliminate it from the set of parameters. We are interested in cases where the last option is not possible or not desired because it introduces non-physical parameters. Hence our solution then will contain an arbitrary component. The problem we are thus faced with is how to describe the uncertainty of our estimated parameters given this arbitrary component in the solution.

In this chapter we created a geometric interpretation for parametric constraints that eliminate gauge freedoms. The constraints permit us to perform standard covariance analysis, but they also affect the solution and its uncertainty. We investigated the nature of these constraints on the uncertainties, and showed that they involve a Jacobian transformation to account for the change of position on the gauge orbit, and an oblique projection to locally satisfy all the constraints. Nevertheless this transformation does not affect uncertainties of gauge invariants derived from these parameters, which are the basic entities obtained in the measurement. This fact allowed us to derive a geometric

equivalence relationship that enables us to compare covariances solely on their gauge independent components.

The normal covariance was derived as a convenient uncertainty description that does not require us to specify gauge constraints, and is a lower bound on all covariances defined at that point on the gauge orbit. Finally we extended the Cramer-Rao lower bound to cases that include gauge freedoms. When working in a particular gauge, we showed that the Maximum Likelihood solution is unbiased and achieves the Cramer-Rao lower bound to first order.

3. Gauge Freedoms in Structure from Motion

From a set of images of an object, taken at different positions and orientations, Structure from Motion (SFM) seeks to estimate the shape of an object and the camera positions and orientations from which the images were taken. The method operates on a set of feature points that are registered in all the images. Using these known feature points as data and the known projection equations, the object shape and camera motion can be represented as parameters of a function that is fit to the data. SFM can thus be directly posed as a parametric fitting task. In this chapter we will take a parametric formulation of SFM and derive the gauge freedoms corresponding to the indeterminacies inherent in SFM. The theory developed in the previous chapter will then allow us to express the uncertainty of SFM estimates despite the gauge freedoms. We will leave applications of this theory to subsequent chapters.

3.1 Camera Equations and Assumptions

Suppose we track N rigidly moving 3D feature points $\boldsymbol{P}_\alpha$, $\alpha = 1, \ldots, N$, in M images. Let $\boldsymbol{p}_{\kappa\alpha} = (x_{\kappa\alpha}, y_{\kappa\alpha})^\top$ be the image coordinates of $\boldsymbol{P}_\alpha$ in the κth image.

We identify the camera coordinate system with the XYZ world coordinate system, and choose an object coordinate system fixed relative to the object. Let $\boldsymbol{t}_\kappa$ be the origin of the object coordinate system in the κth frame, $\boldsymbol{R}_\kappa$ be a 3×3 rotation matrix which specifies its orientation, and $\boldsymbol{s}_\alpha$ be the coordinates of the feature point $\boldsymbol{P}_\alpha$ in the object coordinate system. Thus

$$\boldsymbol{t}_\kappa = \begin{pmatrix} t_{\kappa x} \\ t_{\kappa y} \\ t_{\kappa z} \end{pmatrix}, \quad \boldsymbol{R}_\kappa = \begin{pmatrix} \boldsymbol{r}_{\kappa x} \\ \boldsymbol{r}_{\kappa y} \\ \boldsymbol{r}_{\kappa z} \end{pmatrix}, \quad \boldsymbol{s}_\alpha = \begin{pmatrix} s_{\alpha x} \\ s_{\alpha y} \\ s_{\alpha z} \end{pmatrix}, \tag{3.1}$$

where $\boldsymbol{r}_{\kappa x}$, $\boldsymbol{r}_{\kappa y}$ and $\boldsymbol{r}_{\kappa z}$ are the rows of $\boldsymbol{R}_\kappa$ and are unit orthonormal vectors. The 3D position of feature a point $\boldsymbol{P}_\alpha$ expressed in a camera–centered coordinate system, and hence relative to the κth camera, is:

$$\boldsymbol{P}_\alpha = \boldsymbol{R}_\kappa \boldsymbol{s}_\alpha + \boldsymbol{t}_\kappa \ , \tag{3.2}$$

and this is illustrated in Figure 3.1.

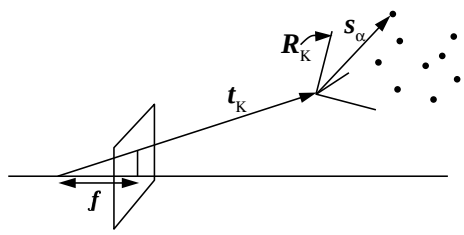


Fig. 3.1. Illustrated is our camera coordinate system. A 3D point is located with respect to the κth camera's center of projection by the formula: $\boldsymbol{P}_\alpha = \boldsymbol{R}_\kappa \boldsymbol{s}_\alpha + \boldsymbol{t}_\kappa$.

We chose to work in a camera–centered coordinate system, although we could also have worked in an object–centered system. If we were to do this, a convenient parametrization for a 3D feature would be given by: $\boldsymbol{P}_\alpha = \boldsymbol{R}_\kappa(\boldsymbol{s}_\alpha - \boldsymbol{t}'_\kappa)$, with $\boldsymbol{t}'_\kappa = -\boldsymbol{R}_\kappa^\top \boldsymbol{t}_\kappa$ now representing the camera translation rather than the object origin. All the equations we derive in this thesis can be written using either coordinate system, although for simplicity we work only in one, the camera–centered coordinate system. It is straightforward to transform results to an object–centered system.

Assume we have a projection operator $\boldsymbol{\Pi}_\kappa[\cdot] : \mathcal{R}^3 \rightarrow \mathcal{R}^2$ which projects a point in 3D to the 2D image plane. We can then express the image coordinates, $\boldsymbol{p}_{\kappa\alpha}$, of feature $\boldsymbol{P}_\alpha$ as:

$$\begin{aligned} \boldsymbol{p}_{\kappa\alpha} &= \boldsymbol{\Pi}_\kappa[\boldsymbol{K}_\kappa \boldsymbol{P}_\alpha] \\ &= \boldsymbol{\Pi}_\kappa[\boldsymbol{K}_\kappa(\boldsymbol{R}_\kappa \boldsymbol{s}_\alpha + \boldsymbol{t}_\kappa)] \, , \end{aligned} \tag{3.3}$$

where $\boldsymbol{\Pi}_\kappa[\cdot]$ is our projection operator for the κth camera and $\boldsymbol{K}_\kappa$ is a 3×3 matrix containing internal camera parameters.

We define the following projection operator on a vector $(X, Y, Z)^\top$ for perspective camera models as:

$$\boldsymbol{\Pi}_\kappa^P[\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}] = \begin{pmatrix} X/Z \\ Y/Z \end{pmatrix} \ . \tag{3.4}$$

For affine projection, including orthography and weak perspective, we use the following projection operator:

$$\boldsymbol{\Pi}_{\kappa}^{A}[\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}] = \begin{pmatrix} X \\ Y \end{pmatrix} . \tag{3.5}$$

The internal camera parameters encoded in $\boldsymbol{K}_{\kappa}$ may vary between cameras. For perspective cameras a typical form of this matrix is [22]:

$$\boldsymbol{K}_{\kappa} = \begin{pmatrix} f_{\kappa} & s_{\kappa} & c_{x\kappa} \\ 0 & f_{\kappa} a_{\kappa} & c_{y\kappa} \\ 0 & 0 & 1 \end{pmatrix} . \tag{3.6}$$

where f_{κ} is the focal length, s_{κ} is the skew, a_{κ} is the aspect ratio, and $c_{x\kappa}$ and $c_{y\kappa}$ are the coordinates of the center of projection.

For an orthographic camera $\boldsymbol{K}_{\kappa}$ is just the 3×3 identity matrix:

$$\boldsymbol{K}_{\kappa} = \boldsymbol{I} . \tag{3.7}$$

In some cases we may with to add an overall unknown scale factor to orthography, since this more accurately reflects the case that even when we use orthography we do not know the actual scale of the object. Nevertheless for the analysis in this chapter we keep to the standard orthographic model without this additional scale factor.

Under weak perspective we let

$$\boldsymbol{K}_{\kappa} = \mu_{\kappa} \boldsymbol{I} , \tag{3.8}$$

where μ_{κ} is a scale factor equal to the inverse of the depth of the centroid of the object in each image. Hence we can express the inverse of this factor as:

$$\frac{1}{\mu_{\kappa}} = t_{\kappa z} + \frac{1}{N} \sum_{\alpha}^{N} \boldsymbol{r}_{\kappa z} \boldsymbol{s}_{\alpha} , \tag{3.9}$$

where $t_{\kappa z}$ is the depth component of translation, and $\boldsymbol{r}_{\kappa z}$ is the third row of the rotation matrix. Since we are primarily interested in the 3D shape rather than the camera translation, for simplicity we will find it convenient to use this depth directly and eliminate $t_{\kappa z}$ rather than eliminate it in favor of $t_{\kappa z}$.

Some or all of these projection parameters in $\boldsymbol{K}_{\kappa}$ may be known, and the rest can be solved for during shape and motion estimation. Also, more complicated camera models can also be created. For instance wide angle lenses may have radial distortion which can be modeled using Tsai's or other methods [28, 72]. It turns out, however, that so long as these and any additional parameters are correctly modeled they will not affect our indeterminacy analysis, which depends only on structure and motion components. Since in this work we are most interested in effects of indeterminacies and less interested in

camera parameters which do not affect the indeterminacies, we assume that most or all of these parameters have been pre-calibrated and so are known. For those that are not known, such as focal lengths, we include them in our parameter vector $\boldsymbol{\theta}$.

Now equation (3.3) can be applied to all features in all images, and then combined to form one large set of equations:

$$\boldsymbol{p} = \boldsymbol{\Pi}[\boldsymbol{\theta}] \tag{3.10}$$

where $\boldsymbol{p} = (\boldsymbol{p}_{11}^\top, \boldsymbol{p}_{12}^\top, \boldsymbol{p}_{13}^\top, \ldots, \boldsymbol{p}_{MN}^\top)^\top$ is a vector containing all the image feature coordinates in all images, and $\boldsymbol{\theta}$ is a vector containing the shape and motion parameters, $\boldsymbol{R}_\kappa$, $\boldsymbol{s}_\alpha$, $\boldsymbol{t}_\kappa$, and possibly unknown internal camera parameters, for all object features and images, and $\boldsymbol{\Pi}[\cdot]$ is the appropriate combination of the projection matrices from equations (3.4) and (3.5) for all features in all images.

3.2 The Parameter Space

The shape and translation components of the parameter vector, $\boldsymbol{\theta}$, provide a straightforward linear parameter space. The rotation parameters contained in the rotation matrices, $\boldsymbol{R}_\kappa$, however, are strongly constrained as the columns of each rotation matrix must remain unit orthogonal vectors. Each rotation matrix has locally only three degrees of freedom. Let $\mathcal{T}$ be the parameter manifold containing all valid vectors $\boldsymbol{\theta}$. However, since all of our operations, including the uncertainty analysis, will be performed in the tangent space to $\mathcal{T}$, we will use $\mathcal{T}$ also to refer to the local parameter space defined as the tangent to manifold $\mathcal{T}$, and only distinguish these when necessary. The manifold and the parameter space, $\mathcal{T}$, have dimension n, where n is the number of parameters needed to locally specify the shape and motion, 3 for each rotation, 3 for each translation, and 3 for each 3D feature point, plus any internal camera parameters that must be estimated. So in general for just motion and shape, the number of unknown parameters is: $n = 3N + 6M$.

3.2.1 Operator Definitions

We will find tensor products, and some identites based on them, useful and so define them here. The product $\mathbf{a} \times \boldsymbol{A}$ of a three dimensional vector $\boldsymbol{a} = (a_1, a_2, a_3)^\top$ and a 3×3 matrix $\boldsymbol{A}$ is defined to be the matrix whose columns are the vector product of $\boldsymbol{a}$

and the columns of $\boldsymbol{A}$. In particular the product, $\boldsymbol{a} \times \boldsymbol{I}$, with the unit matrix $\boldsymbol{I}$ is given by[1]:

$$\boldsymbol{a} \times \boldsymbol{I} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix}. \tag{3.11}$$

Using this we can define the following products for vectors $\boldsymbol{a}$ and $\mathbf{b}$, and matrix $\boldsymbol{A}$:

$$\begin{aligned} \boldsymbol{a} \times \boldsymbol{A} &= (\boldsymbol{a} \times \boldsymbol{I})\boldsymbol{A}, \\ \boldsymbol{A} \times \boldsymbol{b} &= \boldsymbol{A}(\boldsymbol{b} \times \boldsymbol{I})^\top, \\ \boldsymbol{a} \times \boldsymbol{A} \times \boldsymbol{b} &= (\boldsymbol{a} \times \boldsymbol{I})\boldsymbol{A}(\boldsymbol{b} \times \boldsymbol{I})^\top. \end{aligned} \tag{3.12}$$

The following identities which we will use later are derived from these formulae:

$$\begin{aligned} (\boldsymbol{a} \times \boldsymbol{I})\boldsymbol{b} &\equiv -(\boldsymbol{b} \times \boldsymbol{I})\boldsymbol{a} \\ \boldsymbol{R}(\boldsymbol{a} \times \boldsymbol{I})\boldsymbol{R}^\top &\equiv (\boldsymbol{R}\boldsymbol{a}) \times \boldsymbol{I} \end{aligned} \tag{3.13}$$

for a rotation matrix $\boldsymbol{R}$.

3.2.2 Rotation Parametrization

Our parameter vector $\boldsymbol{\theta}$ will contain a number of rotation matrices; one for each image. The 9 parameters in a rotation matrix are highly constrained with only 3 degrees of freedom. Perturbations of a rotation matrix must be, to first order, in the 3 dimensional tangent space to the parameter manifold. We choose not to use quaternions since these have 4 parameters and so do not give a minimal representation of perturbations. Rather we encode rotation perturbations with a vector, $\Delta\boldsymbol{\Omega}$, containing three elements, defined as follows.

If a rotation matrix perturbation is written as $\boldsymbol{R} : \boldsymbol{R} \to \boldsymbol{R} + \Delta\boldsymbol{R}$, then $\Delta\boldsymbol{R}$ can be expressed as:

$$\Delta\boldsymbol{R} = \Delta\boldsymbol{\Omega} \times \boldsymbol{R}. \tag{3.14}$$

Thus while our parameter vector $\boldsymbol{\theta}$ will include rotation matrices, perturbations of it, $\Delta\boldsymbol{\theta}$, will need only three components for each rotation matrix, and are encoded in terms of $\Delta\boldsymbol{\Omega}$. For a fuller explanation of these perturbations, see Appendix A.

[1] The more common notation for this is: $[\boldsymbol{a}]_\times \boldsymbol{I}$, but we believe our notation is simpler and easier to understand.

3.3 Optimal Solution Estimation

In order to achieve the optimal solution, as defined by the Cramer-Rao lower bound (see section 2.5), we must model the noise appropriately, and then use a non-linear optimization method to find it. We describe our choices for these in this section.

3.3.1 Noise Modeling

We assume that the inverse covariance of our measurement data is known at least up to a scale factor and called $\boldsymbol{V}_{\boldsymbol{po}}^{-1}$. Each feature measurement, $\boldsymbol{p}_{\kappa\alpha}$ is a 2D vector, and if all the feature measurements are independent, then $\boldsymbol{V}_{\boldsymbol{po}}^{-1}$ is a block diagonal matrix with block-size of 2. If the noise is $i.i.d.$, then $\boldsymbol{V}_{\boldsymbol{po}}^{-1}$ is the identity matrix (or a scaled version of it). Alternatively, more sophisticated covariance estimates of feature image positions can be obtained as done in [51] and these included in $\boldsymbol{V}_{\boldsymbol{po}}$. Futhermore, a limited number of missing features can be modeled as having infinite uncertainty by making the appropriate terms in the inverse covariance, $\boldsymbol{V}_{\boldsymbol{po}}^{-1}$, zero.

Using this noise model we can achieve the optimal solution by minimizing the residual:

$$J(\boldsymbol{\theta}) = \|\boldsymbol{p} - \boldsymbol{\Pi}[\boldsymbol{\theta}]\|^2_{\boldsymbol{V}_{\boldsymbol{po}}} \tag{3.15}$$

where we substitute our projection equations and the appropriate parameters for $\boldsymbol{\Pi}[\boldsymbol{\theta}]$. Once we find this solution, we can obtain the unknown scale factor in the covariance using the method described in section 2.1.2.

3.3.2 Non-linear optimization

The gradient of the residual is a non-linear function of the parameters, and so we must rely on non-linear optimization techniques. In our work we implemented Levenberg-Marquardt to minimize the residual. This requires an intitial starting point. For scenes with only small or moderate amounts of perspective distortion, we used the factorization algorithm [68, 51] to give us a rough shape and motion estimate. With large perspetive effects, we used the eight-point algorithm [23, 45] on pairs of images to obtain rough motion estimates. Combining these along with the images we can linearly estimate shape, and then these estimates are fed into a global non-linear optimization procedure that used an appropriate camera model, (orthographic, weak perspective or perspective), and also solved for focal lengths if unknown.

3.4 Gauge Freedoms

The camera equations (3.3) and (3.10) contain a number of indeterminacies. The result of these is to fill our parameter space $\mathcal{T}$ with gauge orbits. In this section we will first define the gauge orbits. Then we will use infinitesimal gauge generators to identify all the gauge freedoms at any point on a gauge orbit. These will depend on the camera projection model, and so we will derive the gauge freedoms for perspective, orthographic and weak perspective models.

3.4.1 Gauge Transformations and Gauge Orbits

There are two reasons for indeterminacies in the camera equations: first the object coordinate system can be selected arbitrarily, and second the camera projection model maps many possible scaled versions of the 3D points onto the same set of 2D points in the images. These are specified as follows:

Coordinate System Transformations

If we rotate and then translate the coordinate system by $\boldsymbol{R}$ and $\boldsymbol{t}$ respectively, we obtain the following transformed shape and motion parameters:

$$\begin{aligned} \boldsymbol{s}'_\alpha &= \boldsymbol{R}^\top(\boldsymbol{s}_\alpha - \boldsymbol{t}), \\ \boldsymbol{R}'_\kappa &= \boldsymbol{R}_\kappa \boldsymbol{R}, \\ \boldsymbol{t}'_\kappa &= \boldsymbol{R}_\kappa \boldsymbol{t} + \boldsymbol{t}_\kappa\,. \end{aligned} \tag{3.16}$$

We note that $\boldsymbol{R}'_\kappa \boldsymbol{s}'_\alpha + \boldsymbol{t}'_\kappa = \boldsymbol{R}_\kappa \boldsymbol{s}_\alpha + \boldsymbol{t}_\kappa$, and hence irrespective of the projection model, equations (3.3) and (3.10) must be ambiguous to changes in coordinates. A global rotation and translation is thus a gauge transformation, and hence this defines a set of orbits in the parameter space.

Whether the coordinate system indeterminacy is considered a real physical indeterminacy or else purely a mathematical indeterminacy depends on our model interpretation, $\mathcal{M}(\boldsymbol{\theta})$. If global translation and rotation are not part of the model, then this indeterminacy is purely a mathematical over-parametrization, and real properties of the model are not affected by it. The problem with this is that 3D features can no longer be considered as absolute points in space. Instead the basic entities are distances and relative positions of points, which provide more limited descriptions of the model, although adequate for some uses. So, typically we do consider our shape model to be a set of fixed points in space, and accept this coordinate indeterminacy as a loss of information about the object.

Projection Transformations

Due to the nature of camera projection, many different geometric solutions project onto the same points in all the images, as shown in Figure 1.2. Perspective projection has a scale ambiguity such that if we transform the shape and translation by a scale factor a:

$$\boldsymbol{s}'_\alpha = a\boldsymbol{s}_\alpha, \quad \text{and} \quad \boldsymbol{t}'_\kappa = a\boldsymbol{t}_\kappa, \tag{3.17}$$

we obtain identical projections: $\boldsymbol{\Pi}^P_\kappa[\boldsymbol{K}_\kappa(\boldsymbol{R}_\kappa\boldsymbol{s}'_\alpha + \boldsymbol{t}'_\kappa)] = \boldsymbol{\Pi}^P_\kappa[\boldsymbol{K}_\kappa(\boldsymbol{R}_\kappa\boldsymbol{s}_\alpha + \boldsymbol{t}_\kappa)]$. In orthography the depth or Z component does not affect the image, and hence the projection is invariant to the transformation:

$$\boldsymbol{t}'_\kappa = \boldsymbol{t}_\kappa + d_\kappa\boldsymbol{k} \tag{3.18}$$

for any value d_κ, and vector $\boldsymbol{k} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^\top$. We can remove these degrees of freedom by dropping the depth component of the translation, and so working with $\boldsymbol{t}'_{\kappa xy} = \begin{pmatrix} \boldsymbol{t}_x & \boldsymbol{t}_y \end{pmatrix}$. Orthography has a discrete reflection ambiguity, but since it is not continuous it will not affect our perturbation analysis and we do not consider it. Finally, weak perspective acts as a combination of orthography and perspective. It contains all of the degrees of freedom of the orthography, plus an extra scale freedom due to the μ_κ parameters. We find that if we rescale the parameters:

$$\boldsymbol{s}'_\alpha = a\boldsymbol{s}_\alpha, \quad \boldsymbol{t}'_\kappa = a\boldsymbol{t}_\kappa \quad \text{and} \quad \mu'_\kappa = \frac{\mu_\kappa}{a}, \tag{3.19}$$

the image projections are identical, and so we see that, just as in perspective projection, a scale degree of freedom is included.

In summary, the perspective camera model will give a solution up to a similarity transformation; that is, up to a global rotation, translation, and scale. Orthography defines the solution up to a rotation, translation and depth in each image. We note that the lack of scale freedom in orthography is due to the implicit assumption in orthography that the image plane projection is the same size as the real object. If the μ_κ parameters in weak perspective are interpreted as inverse depths, $1/\boldsymbol{t}_{\kappa z}$, then weak perspective, like perspective, obtains the solution up to a similarity transformation. These indeterminacies partition the parameter space into a set of gauge orbits.

3.4.2 Camera Projection Gauge Freedoms

At a point in the parameter space, $\boldsymbol{\theta} \in \mathcal{T}$, the gauge freedoms span the tangent plane to the gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, containing point $\boldsymbol{\theta}$. These freedoms can be found through a first

order expansion of the gauge transformations around $\boldsymbol{\theta}$. If $\boldsymbol{\theta}$ is a point on the gauge orbit $\mathcal{G}_{\boldsymbol{\theta}}$, and $\boldsymbol{\theta}'$ is a point very close to $\boldsymbol{\theta}$ and still in $\mathcal{G}_{\boldsymbol{\theta}}$, we can express $\boldsymbol{\theta}'$ as an expansion around $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \boldsymbol{D}(\boldsymbol{\theta}) + \dots \tag{3.20}$$

where $\boldsymbol{D}(\cdot)$ is an infinitesimal gauge generator acting in the tangent plane of $\mathcal{G}_{\boldsymbol{\theta}}$. In the following we will derive expressions for these generators under perspective, orthographic and weak perspective camera models.

Perspective Camera

Let us assume the perturbations in the gauge orbit for the perspective case, given by equations (3.16) and (3.17), are small. Let $\boldsymbol{R}'$, $\boldsymbol{t}'$, and a' be small perturbations from the identity transformation in each of rotation, translation and scale. Then we can write their expansions in terms of their infinitesimal gauge generators as:

$$\begin{aligned} \boldsymbol{R}' &= \boldsymbol{I} + \Delta\boldsymbol{\Omega} \times \boldsymbol{I}, \\ \boldsymbol{t}' &= \Delta\boldsymbol{t}, \\ a' &= 1 + \Delta a\,. \end{aligned} \tag{3.21}$$

We will use these to derive the gauge generator, $\boldsymbol{D}(\boldsymbol{\theta})$ for the complete set of parameters, $\boldsymbol{\theta}$. First we substitute these, and the first order expansions: $\boldsymbol{s}'_\alpha = \boldsymbol{s}_\alpha + \boldsymbol{D}(\boldsymbol{s}_\alpha)$, $\boldsymbol{R}'_\kappa = \boldsymbol{R}_\kappa + \boldsymbol{D}(\boldsymbol{R}_\kappa) = \boldsymbol{R}_\kappa + \boldsymbol{D}(\boldsymbol{\Omega}_\kappa) \times \boldsymbol{R}_\kappa$, and $\boldsymbol{t}'_\kappa = \boldsymbol{t}_\kappa + \boldsymbol{D}(\boldsymbol{t}_\kappa)$, into the gauge orbit equations (3.16) to obtain:

$$\begin{aligned} \boldsymbol{s}_\alpha + \boldsymbol{D}(\boldsymbol{s}_\alpha) &= (\boldsymbol{I} + \Delta\boldsymbol{\Omega} \times \boldsymbol{I})\ ((1 + \Delta a)\boldsymbol{s}_\alpha - \Delta\boldsymbol{t}) \\ \boldsymbol{R}_\kappa + \boldsymbol{D}(\boldsymbol{\Omega}_\kappa) \times \boldsymbol{R}_\kappa &= \boldsymbol{R}_\kappa(\boldsymbol{I} + \Delta\boldsymbol{\Omega} \times \boldsymbol{I}) \\ \boldsymbol{t}_\kappa + \boldsymbol{D}(\boldsymbol{t}_\kappa) &= \boldsymbol{R}_\kappa \Delta\boldsymbol{t} + (1 + \Delta a)\boldsymbol{t}_\kappa\,. \end{aligned} \tag{3.22}$$

The middle of these equations is simplified to become $\boldsymbol{D}(\boldsymbol{\Omega}_\kappa) \times \boldsymbol{R}_\kappa = \boldsymbol{R}_\kappa(\Delta\boldsymbol{\Omega}_\kappa \times \boldsymbol{I})$ and then right multiplying by $\boldsymbol{R}_\kappa^\top$ we obtain:

$$\boldsymbol{D}(\boldsymbol{\Omega}_\kappa) \times \boldsymbol{I} = \boldsymbol{R}_\kappa(\Delta\boldsymbol{\Omega} \times \boldsymbol{I})\boldsymbol{R}_\kappa^\top = (\boldsymbol{R}_\kappa \Delta\boldsymbol{\Omega}) \times \boldsymbol{I}\,, \tag{3.23}$$

where we have used identities from section 3.2.1. Finally we obtain $\boldsymbol{D}(\boldsymbol{\Omega}_\kappa) = \boldsymbol{R}_\kappa \Delta\boldsymbol{\Omega}$. In a similar way we simplify the rest of these equations to get the following infinitesimal gauge generators for the perspective camera:

$$
\begin{aligned}
\boldsymbol{D}(\boldsymbol{s}_\alpha) &= (\boldsymbol{s}_\alpha \times \boldsymbol{I})\Delta\boldsymbol{\Omega} - \Delta\boldsymbol{t} + \boldsymbol{s}_\alpha \Delta a, \\
\boldsymbol{D}(\boldsymbol{\Omega}_\kappa) &= \boldsymbol{R}_\kappa \Delta\boldsymbol{\Omega}, \\
\boldsymbol{D}(\boldsymbol{t}_\kappa) &= \boldsymbol{R}_\kappa \Delta\boldsymbol{t} + \boldsymbol{t}_\kappa \Delta a\,.
\end{aligned}
\tag{3.24}
$$

This tangent plane to the gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, has three rotational degrees of freedom, three translational degrees of freedom and one scale degree of freedom, and hence has dimension $r = 7$.

To find a matrix that spans the tangent plane, let us express the infinitesimal gauge generator for all the parameters at point $\boldsymbol{\theta}$ as follows:

$$
\boldsymbol{D}(\boldsymbol{\theta}) \equiv \begin{pmatrix} \boldsymbol{D}(\boldsymbol{s}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{s}_N) \\ \hline \boldsymbol{D}(\boldsymbol{\Omega}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{\Omega}_M) \\ \hline \boldsymbol{D}(\boldsymbol{t}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{t}_M) \end{pmatrix}_{1\times 3N+6M} = U_{\boldsymbol{\theta}} \begin{pmatrix} \Delta\boldsymbol{\Omega} \\ \Delta\boldsymbol{t} \\ \Delta a \end{pmatrix}_{1\times 7}
\tag{3.25}
$$

where the seven columns of matrix $U_{\boldsymbol{\theta}}$ span the tangent space to the gauge orbit. From equations (3.24 – 3.25), the rows of $U_{\boldsymbol{\theta}}$ are given by:

$$
U_{\boldsymbol{\theta}} = \begin{pmatrix} \boldsymbol{s}_1 \times \boldsymbol{I} & -\boldsymbol{I} & \boldsymbol{s}_1 \\ \vdots & \vdots & \vdots \\ \boldsymbol{s}_N \times \boldsymbol{I} & -\boldsymbol{I} & \boldsymbol{s}_N \\ \hline \boldsymbol{R}_1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ \boldsymbol{R}_M & 0 & 0 \\ \hline 0 & \boldsymbol{R}_1 & \boldsymbol{t}_1 \\ \vdots & \vdots & \vdots \\ 0 & \boldsymbol{R}_M & \boldsymbol{t}_M \end{pmatrix}_{7\times 3N+6M} .
\tag{3.26}
$$

This will be used along with the constraints to obtain matrix $\boldsymbol{Q}_{\boldsymbol{\theta}}$ that obliquely projects perturbations and covariances along the tangent space to the gauge orbit.

Orthographic Camera

The orthographic camera case is similar to the perspective case, except that there is no scale component, and the depth component of translation does not affect the projection. Following a similar derivation to the perspective case, and using equations (3.16) and (3.18), we obtain the following gauge generators:

$$\begin{aligned} \boldsymbol{D}(\boldsymbol{s}_\alpha) &= (\boldsymbol{s}_\alpha \times \boldsymbol{I})\Delta\boldsymbol{\Omega} - \Delta\boldsymbol{t}, \\ \boldsymbol{D}(\boldsymbol{\Omega}_\kappa) &= \boldsymbol{R}_\kappa \Delta\boldsymbol{\Omega}, \\ \boldsymbol{D}(\boldsymbol{t}_\kappa) &= \boldsymbol{R}_\kappa \Delta\boldsymbol{t} + \Delta d_\kappa \boldsymbol{k}\,. \end{aligned} \tag{3.27}$$

The tangent space, and hence gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, has dimension $r = M+6$. To simplify implementation issues, we drop the depth or Z component from translation in each image, and so replace the translation generator with:

$$\boldsymbol{D}(\boldsymbol{t}_{\kappa xy}) = \boldsymbol{R}_{\kappa xy}\Delta\boldsymbol{t}\,, \tag{3.28}$$

where $\boldsymbol{t}_{\kappa xy}$ and $\boldsymbol{R}_{\kappa xy}$ refer to the two top rows of $\boldsymbol{t}_\kappa$ and $\boldsymbol{R}_\kappa$ respectively. We thus have M less parameters to calculate, and M less degrees of freedom, and so $r = 6$.

The gauge generator, $\boldsymbol{D}(\boldsymbol{\theta})$, and the $6 \times 3N + 5M$ matrix, $U_{\boldsymbol{\theta}}$, spanning the gauge freedoms are encoded as:

$$\boldsymbol{D}(\boldsymbol{\theta}) \equiv \begin{pmatrix} \boldsymbol{D}(\boldsymbol{s}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{s}_N) \\ \hline \boldsymbol{D}(\boldsymbol{\Omega}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{\Omega}_M) \\ \hline \boldsymbol{D}(\boldsymbol{t}_{1xy}) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{t}_{Mxy}) \end{pmatrix} = U_{\boldsymbol{\theta}} \begin{pmatrix} \Delta\boldsymbol{\Omega} \\ \Delta\boldsymbol{t} \end{pmatrix}, \quad U_{\boldsymbol{\theta}} = \begin{pmatrix} \boldsymbol{s}_1 \times \boldsymbol{I} & -\boldsymbol{I} \\ \vdots & \vdots \\ \boldsymbol{s}_N \times \boldsymbol{I} & -\boldsymbol{I} \\ \hline \boldsymbol{R}_1 & 0 \\ \vdots & \vdots \\ \boldsymbol{R}_M & 0 \\ \hline 0 & \boldsymbol{R}_{1xy} \\ \vdots & \vdots \\ 0 & \boldsymbol{R}_{Mxy} \end{pmatrix}. \tag{3.29}$$

Weak Perspective Camera

Under weak perspective we take our orthographic results and add the μ_κ parameters which gives us a scale freedom. Thus from equations (3.16) and (3.19), we obtain:

$$\begin{aligned} \boldsymbol{D}(\boldsymbol{s}_\alpha) &= (\boldsymbol{s}_\alpha \times \boldsymbol{I})\Delta\boldsymbol{\Omega} - \Delta\boldsymbol{t} + \boldsymbol{s}_\alpha \Delta a, \\ \boldsymbol{D}(\boldsymbol{\Omega}_\kappa) &= \boldsymbol{R}_\kappa \Delta\boldsymbol{\Omega}, \end{aligned}$$

$$\begin{aligned} \boldsymbol{D}(\boldsymbol{t}_{\kappa xy}) &= \boldsymbol{R}_{\kappa xy}\Delta\boldsymbol{t} + \boldsymbol{t}_{\kappa xy}\Delta a, \\ \boldsymbol{D}(\mu_\kappa) &= -\mu_\kappa \Delta a\,. \end{aligned} \tag{3.30}$$

The gauge generator, $\boldsymbol{D}(\boldsymbol{\theta})$, and matrix, $U_{\boldsymbol{\theta}}$, spanning the gauge freedoms are encoded:

$$\boldsymbol{D}(\boldsymbol{\theta}) \equiv \begin{pmatrix} \boldsymbol{D}(\boldsymbol{s}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{s}_N) \\ \hline \boldsymbol{D}(\boldsymbol{\Omega}_1) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{\Omega}_M) \\ \hline \boldsymbol{D}(\boldsymbol{t}_{1xy}) \\ \vdots \\ \boldsymbol{D}(\boldsymbol{t}_{Mxy}) \\ \hline \boldsymbol{D}(\mu_1) \\ \vdots \\ \boldsymbol{D}(\mu_M) \end{pmatrix} = U_{\boldsymbol{\theta}} \begin{pmatrix} \Delta\boldsymbol{\Omega} \\ \Delta\boldsymbol{t} \\ \Delta a \end{pmatrix}, \quad U_{\boldsymbol{\theta}} = \begin{pmatrix} \boldsymbol{s}_1 \times \boldsymbol{I} & -\boldsymbol{I} & \boldsymbol{s}_1 \\ \vdots & \vdots & \vdots \\ \boldsymbol{s}_N \times \boldsymbol{I} & -\boldsymbol{I} & \boldsymbol{s}_N \\ \hline \boldsymbol{R}_1 & 0 & 0 \\ \vdots & \vdots & \vdots \\ \boldsymbol{R}_M & 0 & 0 \\ \hline 0 & \boldsymbol{R}_{1xy} & \boldsymbol{t}_{1xy} \\ \vdots & \vdots & \vdots \\ 0 & \boldsymbol{R}_{Mxy} & \boldsymbol{t}_{Mxy} \\ \hline 0 & 0 & -\mu_1 \\ \vdots & \vdots & \vdots \\ 0 & 0 & -\mu_M \end{pmatrix}. \tag{3.31}$$

Summary

In this section we have derived all of the gauge freedoms for perspective, orthographic, and weak perspective camera models. These are the instantaneous degrees of freedom spanning the tangent space to the gauge orbit. We have encapsulated them in the matrix $U_{\boldsymbol{\theta}}$. To obtain the gauge freedoms at another point, say $\boldsymbol{\theta}_{\mathcal{C}}$, we simply substitute the parameter values of $\boldsymbol{\theta}_{\mathcal{C}}$ into $U_{\boldsymbol{\theta}}$ and obtain $U_{\boldsymbol{\theta}_{\mathcal{C}}}$. We will use to obtain the oblique projection operator, $\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}$, which transforms covariances between different gauges.

When more parameters, such as focal lengths, are included in a SFM algorithm, their corresponding rows in the $U_{\boldsymbol{\theta}}$ matrix will be zero, and hence they do not change the gauge freedoms.

3.5 Gauge Constraints

We may wish to express our solution and its uncertainty in a particular gauge. To do this we must fix the gauge by defining r constraints as described in section 2.4.1. There are many possible sets of constraints we could use, and we analyze some of them in this

section. We will need not only the constraints, but their gradients in order to transform the covariances. We will work with the perspective camera as this is the most general, and because orthographic and weak perspective solutions can be easily obtained from the perspective results.

3.5.1 Satisfying Gauge Constraints

Let us assume that we have found an optimal solution point, $\boldsymbol{\theta}$, on a gauge orbit, $\mathcal{G}_{\boldsymbol{\theta}}$, and having covariance $V_{\boldsymbol{\theta}}$. Our goal is to express our solution and its covariance in gauge $\mathcal{C}$. The first step is to find the intersection of the gauge orbit and the gauge manifold as this gives us our transformed solution point: $\boldsymbol{\theta}_{\mathcal{C}} = \mathcal{G}_{\boldsymbol{\theta}} \cap \mathcal{C}$. From this we derive the gauge transformation, g, such that $\boldsymbol{\theta}_{\mathcal{C}} = g\boldsymbol{\theta}$. This will consist of a rotation, $\boldsymbol{R}$, translation $\boldsymbol{t}$, and scale a, according to the equations:

$$\begin{aligned} \boldsymbol{s}'_{\alpha} &= a\boldsymbol{R}^{\top}(\boldsymbol{s}_{\alpha} - \boldsymbol{t}), \\ \boldsymbol{R}'_{\kappa} &= \boldsymbol{R}_{\kappa}\boldsymbol{R}, \\ \boldsymbol{t}'_{\kappa} &= a(\boldsymbol{R}_{\kappa}\boldsymbol{t} + \boldsymbol{t}_{\kappa}) \,. \end{aligned} \tag{3.32}$$

Here the initial point, $\boldsymbol{\theta}$, contains the parameters $\boldsymbol{R}_{\kappa}$, $\boldsymbol{t}_{\kappa}$, and $\boldsymbol{s}_{\alpha}$ for all features, α, and images, κ. The transformed point, $\boldsymbol{\theta}_{\mathcal{C}}$, contains the corresponding parameters: $\boldsymbol{R}'_{\kappa}$, $\boldsymbol{t}'_{\kappa}$, and $\boldsymbol{s}'_{\alpha}$. The transformed covariance is obtained in an analogous way to that in equation (2.65) and so is given by:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}_{\boldsymbol{\theta}_{\mathcal{C}}}}{\partial \boldsymbol{\theta}_{\boldsymbol{\theta}}} V_{\boldsymbol{\theta}} \frac{\partial \boldsymbol{\theta}_{\boldsymbol{\theta}_{\mathcal{C}}}}{\partial \boldsymbol{\theta}_{\boldsymbol{\theta}}}^{\top} \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}{}^{\top} \,. \tag{3.33}$$

To obtain this we need both the Jacobian matrix, $\partial\boldsymbol{\theta}_{\mathcal{C}}/\partial\boldsymbol{\theta}$, and the oblique projection operator $\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}$. We will derive these in the following sections.

3.5.2 Jacobian Matrix

The Jacobian matrix, $\partial\boldsymbol{\theta}_{\mathcal{C}}/\partial\boldsymbol{\theta}$, maps a perturbation at $\boldsymbol{\theta}$ to one $\boldsymbol{\theta}_{\mathcal{C}}$:

$$\Delta\boldsymbol{\theta}_{\mathcal{C}} = \frac{\partial \boldsymbol{\theta}_{\mathcal{C}}}{\partial \boldsymbol{\theta}} \Delta\boldsymbol{\theta} \,. \tag{3.34}$$

We can derive this in terms of our gauge transformation quantities, $\boldsymbol{R}$, $\boldsymbol{t}$, and a, by a first order expansion of equations (3.32). Doing this and then simplifying we obtain:

$$\begin{aligned} \Delta\boldsymbol{s}'_{\alpha} &= a\boldsymbol{R}^{\top}\,\Delta\boldsymbol{s}_{\alpha} \\ \Delta\boldsymbol{\Omega}'_{\kappa} &= \Delta\boldsymbol{\Omega}_{\kappa} \end{aligned}$$

$$\begin{aligned}\Delta \boldsymbol{t}'_\kappa &= a(\Delta \boldsymbol{R}_\kappa\, \boldsymbol{t} + \Delta \boldsymbol{t}_\kappa) \\ &= a((\Delta \boldsymbol{\Omega}_\kappa \times \boldsymbol{R}_\kappa)\, \boldsymbol{t} + \Delta \boldsymbol{t}_\kappa) \\ &= a(-(\boldsymbol{R}_\kappa \boldsymbol{t} \times \boldsymbol{I})\Delta \boldsymbol{\Omega}_\kappa + \Delta \boldsymbol{t}_\kappa)\end{aligned} \tag{3.35}$$

In matrix form, we thus obtain the following expression for our Jacobian matrix:

$$\frac{\partial \boldsymbol{\theta}_{\mathcal{C}}}{\partial \boldsymbol{\theta}} = \left(\begin{array}{ccc|ccc|ccc} a\boldsymbol{R}^\top & & & & & & & & \\ & \ddots & & & & & & & \\ & & a\boldsymbol{R}^\top & & & & & & \\ \hline & & & \boldsymbol{I} & & & & & \\ & & & & \ddots & & & & \\ & & & & & \boldsymbol{I} & & & \\ \hline & & & -a\boldsymbol{R}_1 \boldsymbol{t} \times \boldsymbol{I} & & & a\boldsymbol{I} & & \\ & & & & \ddots & & & \ddots & \\ & & & & & -a\boldsymbol{R}_M \boldsymbol{t} \times \boldsymbol{I} & & & a\boldsymbol{I} \end{array}\right)_{(3N+6M)\times(3N+6M)} \tag{3.36}$$

where empty elements are zero, and where the elements of $\Delta\boldsymbol{\theta}$ are arranged as in equation (3.25).

3.5.3 Constraints and Oblique Projection

There are numerous ways by which we could define a gauge manifold, $\mathcal{C}$, to uniquely specify the solution point $\boldsymbol{\theta}_{\mathcal{C}}$. In the next chapter we will see that depending on what object information we have, we may wish to use various different constraints on our gauge orbit. In this section we will consider a few possible constraint sets and derive the corresponding matrix $V_{\boldsymbol{\theta}_{\mathcal{C}}}$ that spans the space orthogonal to the tangent space of the gauge manifold $\mathcal{C}$. Using this, and the gauge freedoms spanned by $U_{\boldsymbol{\theta}}$ from equation (3.26), we will be able to calculate the oblique projection operator $\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}}$.

Centroid Constraints

A popular way to constrain shape and motion is by fixing the origin of the coordinate system to the centroid, letting the first camera define the orientation, and fixing the scale using the average distance to the feature points. This gives us the following set of equations:

$$\sum_{\alpha=1}^{N} \boldsymbol{s}_\alpha = \boldsymbol{0}, \qquad \boldsymbol{R}_1 = \boldsymbol{I}, \qquad \sum_{\alpha=1}^{N} \|\boldsymbol{s}_\alpha\|^2 = N. \tag{3.37}$$

We are interested in infinitesimal perturbations in the tangent space to the gauge manifold defined by these constraints. The matrix $V_{\boldsymbol{\theta}_C}$ that spans the space orthogonal to this is obtained as the gradient of these constraints (see equation (2.46)), and can be expressed in a 7-column matrix as:

$$
\begin{aligned}
V_{\boldsymbol{\theta}_C} &= \left(\nabla_{\boldsymbol{\theta}} \sum_{\alpha=1}^{N} \boldsymbol{s}_\alpha \quad \nabla_{\boldsymbol{\theta}} \left(\boldsymbol{R}_1 - \boldsymbol{I} \right) \quad \nabla_{\boldsymbol{\theta}} \left(\sum_{\alpha=1}^{N} \|\boldsymbol{s}_\alpha\|^2 - N \right) \right) \\
&= \left(\begin{array}{ccc}
\boldsymbol{I} & 0 & \boldsymbol{s}_1 \\
\vdots & \vdots & \vdots \\
\boldsymbol{I} & 0 & \boldsymbol{s}_N \\
\hline
0 & \boldsymbol{I} & 0 \\
0 & 0 & 0 \\
\vdots & \vdots & \vdots \\
\hline
0 & 0 & 0 \\
\vdots & \vdots & \vdots
\end{array} \right) .
\end{aligned}
\tag{3.38}
$$

In orthography we can use the same gauge constraint equations, except that we do not need the scale constraint, and hence we simply drop the last column of $V_{\boldsymbol{\theta}_C}$ in equation (3.38). The perspective equations work under weak perspective, and so $V_{\boldsymbol{\theta}_C}$ is the same, except that the zero rows corresponding to $\Delta \boldsymbol{t}_{\kappa z}$ now correspond to $\Delta \mu_\kappa$.

Camera-based Constraints

Another useful way to fix the gauge is to constrain only the camera motion. For instance we could write:

$$
\boldsymbol{R}_1 = \boldsymbol{I}, \qquad \boldsymbol{t}_1 = 0, \qquad \|\boldsymbol{t}_2\| = k \, . \tag{3.39}
$$

This makes the first camera position the origin of our coordinate system and the fixes the distance to the second camera position to some value denoted as k. This is a common constraint for stereo, and so is useful primarily for the perspective camera model, although it can also be used in weak perspective. The matrix $V_{\boldsymbol{\theta}_C}$ is given by:

$$V_{\boldsymbol{\theta}_{\mathcal{C}}} = \begin{pmatrix} 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ \hline \boldsymbol{I} & 0 & 0 \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ \hline 0 & \boldsymbol{I} & 0 \\ 0 & 0 & \boldsymbol{t}_2/2 \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \end{pmatrix} . \tag{3.40}$$

Of course many more possible constraints can be defined and their gradients easily found. Using any of these, our oblique projection operator, from equation (2.51), is then simply:

$$\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}} = I - U_{\boldsymbol{\theta}_{\mathcal{C}}}(V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} U_{\boldsymbol{\theta}_{\mathcal{C}}})^{-1} V_{\boldsymbol{\theta}_{\mathcal{C}}}^{\top} , \tag{3.41}$$

and the covariance in our gauge $\mathcal{C}$ is easily found using equation (2.52).

3.6 Fast Covariance Estimation

So far we have determined how to transform covariances onto and between gauges, but have not discussed how to initially estimate a covariance. SFM solution techniques generally avoid full calculation of the covariance [1, 22], and so we may achieve an optimal solution without a full covariance estimate. One possible, and simple, technique is to simply calculate the normal covariance from equation (2.64). As we have seen this is geometrically equivalent to the covariance in any gauge, and so can be transformed to whichever gauge we want to work with. However, the calculation of the generalized inverse involves use of SVD, or equivalent method, taking $O(N^3 + M^3)$ operations, which for many feature points or images, is slow.

The Hessian, though, has sparse structure and this is used by photogrammetrists to obtain efficient solution methods. In this section we propose a fast covariance estimation method that takes advantage of this sparse structure.

Let us assume that our parameter vector is divided into shape and motion components, $\boldsymbol{\theta}_s$ and $\boldsymbol{\theta}_m$, respectively, such that $\boldsymbol{\theta} = (\boldsymbol{\theta}_s^{\top}, \boldsymbol{\theta}_m^{\top})^{\top}$. The Hessian is then split into its shape and motion components:

$$\nabla^2_{\boldsymbol{\theta}} J = \left(\begin{array}{c|c} \nabla^2_{\boldsymbol{\theta}_s} J & \nabla_{\boldsymbol{\theta}_{sm}} J \\ \hline \nabla_{\boldsymbol{\theta}_{ms}} J & \nabla^2_{\boldsymbol{\theta}_m} J \end{array}\right) = \begin{pmatrix} U & W \\ W^\top & V \end{pmatrix}. \tag{3.42}$$

When noise in each feature point specified by the data covariance, $\boldsymbol{V_p}$, is non-zero and independent of others, matrices U and V are full rank and sparse with $O(N)$ and $O(M)$ non-zero elements respectively, where N is the number of features and M is the number of images [22, 61]. The cross-term matrix, W, is not sparse however, and so applying a standard sparse techniques will not reduce the complexity of determining the generalized inverse. We thus propose the following fast inversion method.

In the case where the number of images is smaller than the number of features, we define the full rank matrix, T, as follows:

$$T = \left(\begin{array}{c|c} \boldsymbol{I} & 0 \\ \hline -W^\top U^{-1} & \boldsymbol{I} \end{array}\right) \tag{3.43}$$

and obtain the block diagonal matrix:

$$T \nabla^2_{\boldsymbol{\theta}} J T^\top = \left(\begin{array}{c|c} U & 0 \\ \hline 0 & V - W^\top U^{-1} W \end{array}\right). \tag{3.44}$$

Then we define the covariance $\boldsymbol{V_T}$ by:

$$\begin{aligned} \boldsymbol{V_T} &= T^\top (T \nabla^2_{\boldsymbol{\theta}} J T^\top)^-_{3N+6M-r} T \\ &= T^\top \left(\begin{array}{c|c} U^{-1} & 0 \\ \hline 0 & (V - W^\top U^{-1} W)^-_{6M-r} \end{array}\right) T. \end{aligned} \tag{3.45}$$

This can be obtained in $O(N^2 M + M^3)$ operations which, when the number of images is small (i.e. $M \ll N$), is much faster than the original SVD which is $O(N^3 + M^3)$. When there are fewer features than images, an analogous transformation can be created to take advantage of this.

To illustrate the potential speed-up achievable by this method, we created an example in which we calculated the normal covariance using SVD and compared this to our fast method for calculating $\boldsymbol{V_T}$. In our example we had 500 features and 10 images. From the analysis above we would expect between one and two orders of magnitude increase in speed, and indeed our results reflected this. Calculating $\boldsymbol{V}_{\perp\mathcal{G}}$ took 37 minutes, whereas calculating $\boldsymbol{V_T}$ took only 24 seconds.

In order for this new covariance expression, $\boldsymbol{V_T}$, to be a valid description of the uncertainty, we must show that it is geometrically equivalent to the normal covariance, $\boldsymbol{V}_{\perp\mathcal{G}}$. To do this we rely on the following theorem:

Theorem 3.6.1. *Let A be a square matrix whose nullspace corresponds to the tangent space of the gauge orbit $\mathcal{G}_{\boldsymbol{\theta}}$, and let T be a full rank square matrix of the same size as A. Then A^- is geometrically equivalent to $T^\top(TAT^\top)^- T$, which we write this as:*

$$A \equiv T^\top(TAT^\top)^- T \mod \mathcal{G}_{\boldsymbol{\theta}} . \tag{3.46}$$

Proof. Consider the equation:

$$A\boldsymbol{x} = \boldsymbol{u} \tag{3.47}$$

where $\boldsymbol{u}$ is in the column space of A. The general solution is a combination of a unique particular solution, $\boldsymbol{x}_p = A^-\boldsymbol{u}$, in the column space of A, and a homogeneous solution, $\boldsymbol{x}_h$, which is any vector in the nullspace of A, i.e. $A\boldsymbol{x}_h = 0$. We left multiply equation (3.47) by T and rearrange to obtain:

$$(TAT^\top)T^{-\top}\boldsymbol{x} = T\boldsymbol{u}. \tag{3.48}$$

Then changing variables: $\boldsymbol{y} = T^{-\top}\boldsymbol{x}$, and solving for $\boldsymbol{y}$ we obtain: $\boldsymbol{y} = (TAT^\top)^- T\boldsymbol{u} + \boldsymbol{y}_h$ where $(TAT^\top)\boldsymbol{y}_h = 0$. Now transforming back to $\boldsymbol{x}$ we can decompose the solution into the particular and homogeneous parts:

$$\boldsymbol{x} = T^\top(TAT^\top)^- T\boldsymbol{u} + T^\top\boldsymbol{y}_h = \boldsymbol{x}_p + \boldsymbol{x}_h, \tag{3.49}$$

where $\boldsymbol{x}_p = A^-\boldsymbol{u}$ is the particular solution obtained in equation (3.47). It is easy to see that $T^\top\boldsymbol{y}_h$ is in the nullspace of A, and hence $T^\top(TAT^\top)^- T\boldsymbol{u} = \boldsymbol{x}_p + \boldsymbol{x}'_h$ for some vector $\boldsymbol{x}'_h$ in the nullspace of A.

We now apply the geometric equivalence test to $\boldsymbol{V}_{\perp\mathcal{G}} = A^-$ and $\boldsymbol{V}_{\boldsymbol{T}} = T^\top(TAT^\top)^- T$. The Jacobian matrix is the identity, and the orthogonal component to the tangent space to the gauge orbit $\mathcal{G}_{\boldsymbol{\theta}}$ is spanned by the columns of A and so $\boldsymbol{u}$ is any vector in the column space of A. Applying the equivalence relationship we obtain:

$$\boldsymbol{u}^\top(A^- - T^\top(TAT^\top)^- T)\boldsymbol{u} = \boldsymbol{u}^\top(\boldsymbol{x}_p - \boldsymbol{x}_p - \boldsymbol{x}'_h) = \boldsymbol{u}^\top(-\boldsymbol{x}'_h) = 0, \tag{3.50}$$

for all $\boldsymbol{u}$ in the column and row space of A, since $\boldsymbol{x}'_h$ is in the nullspace. □

This theorem thus permits us to use a fast method for estimating the covariance of the recovered shape and motion in equation (3.45).

3.7 Invariants

Absolute feature positions, camera rotations and translations are not invariant to gauge transformations. The values of these properties and their uncertainties will depend on the choice of gauge. To evaluate these properties and their uncertainties we need to work in a gauge. However, we can avoid the special choice of a gauge by instead considering only gauge invariants. In SFM, the basic quantities that are invariant to similarity transformations are ratios of lengths and angles. It is easy to confirm that changing the scale, rotation, and translation of two pairs of points will not affect the ratio of their separation lengths. Similarly the angle between two lines intersecting at a point, or two planes intersecting at a line will not change under similarity transformations. Other gauge invariants can be defined in a similar fashion or by adding together different gauge invariants. The value of a gauge invariant will remain constant along all points on a gauge orbit. Furthermore its covariance will be the same irrespective of the gauge in which it is calculated, or if the normal covariance is used or if our fast covariance method is used.

3.8 Experiments

We present a number of synthetic and real experiments illustrating our uncertainty modeling. In our first experiment we consider a simulated object viewed from 11 positions by a weak perspective camera as illustrated in Figure 3.2. A sample set of images are shown with noise added according to a known covariance. Finally the reconstruction of the shape from these images, along with its uncertainty ellipsoids, is shown.

We obtain our uncertainty ellipsoids for this shape from the normal covariance which is shown in Figure 3.3. To confirm that this predicted uncertainty corresponds to the actual covariance of the shape reconstruction given our data covariance, we performed a Monte Carlo simulation. First we chose a gauge, given by equations (3.37), in which to express our reconstructions and their uncertainties. Transforming our normal covariance into this gauge we obtain the predicted covariance in Figure 3.4. Then we performed a set of 400 SFM reconstructions. The covariance of the resulting reconstruction parameters is also shown in Figure 3.4. The predicted and the actual covariances conform very well, as can also be seen in Figure 3.5.

The problem with the shape and motion covariance plots is their dependence on choice of gauge, as is vividly illustrated by the difference between the normal covariance in Figure 3.3 and the gauge-constrained covariance in Figure 3.4. Gauge invariants,

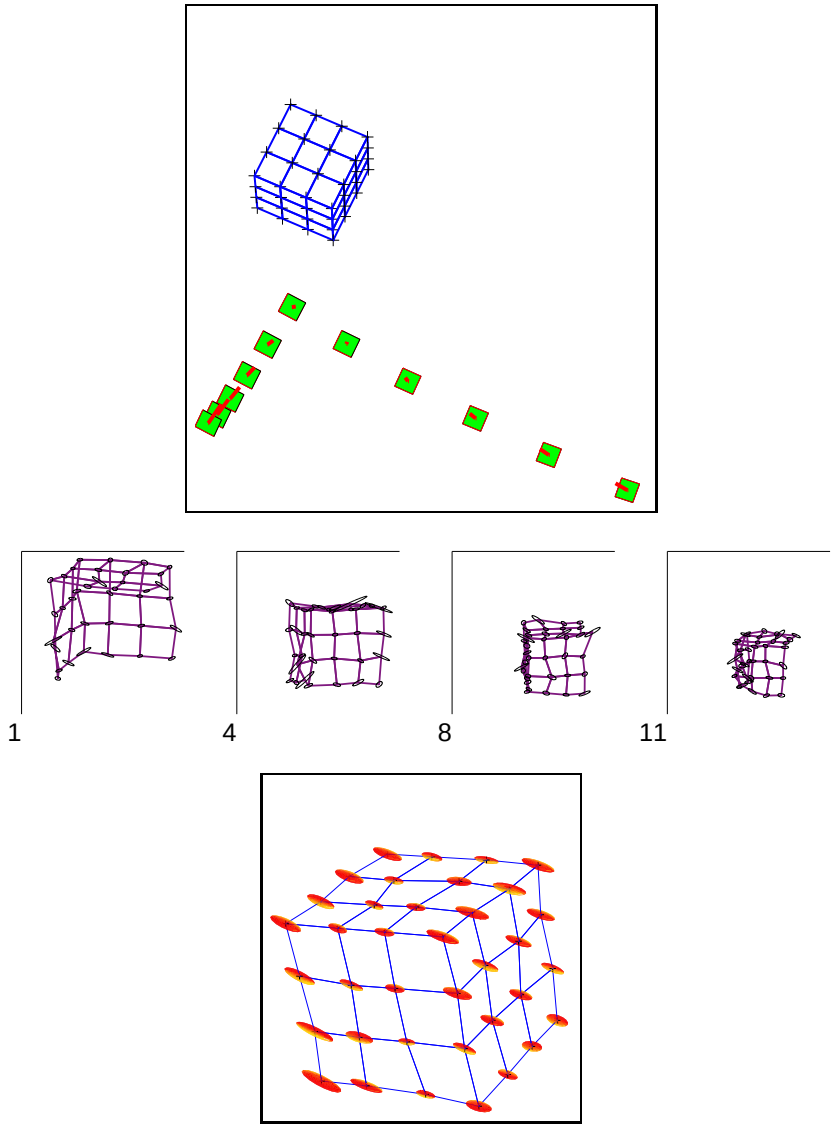


Fig. 3.2. (a) Our simulated experimental setup with 11 cameras viewing our synthetic object is shown. (The lines connecting points are only present for viewing.) (b) Four images of the 11 image sequence with added Gaussian are shown. (c) Finally an optimal reconstruction, given the noise estimates, is shown with uncertainty ellipsoids representing three times the standard deviation of each feature point. The full covariance matrix is shown in Figure 3.4.

however, will give us unambiguous measures for the uncertainty of the results. We chose two invariants on our synthetic object: an angle between two lines and the ratio of two lengths. Their statistics are shown in Table 3.1, confirming very good matching between

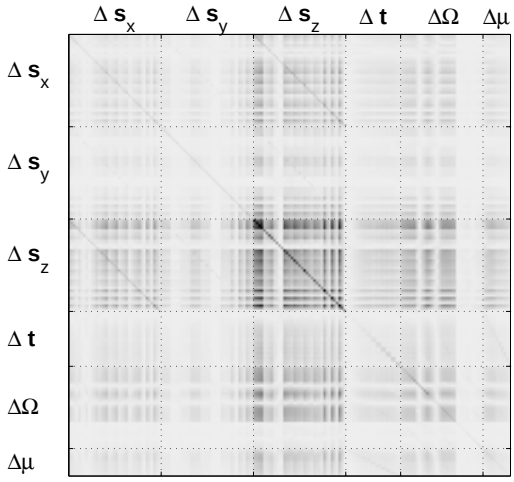


Fig. 3.3. The predicted normal covariance matrix giving us the geometric uncertainty of the reconstructed synthetic object. The scaled absolute value is shown by the darkness of the shading. Here weak perspective was used and μ is the recovered scale for each image. We note that it can be altered by adding components in the tangent plane to gauge orbit without changing the underlying uncertainty, as we see in Figure 3.4.

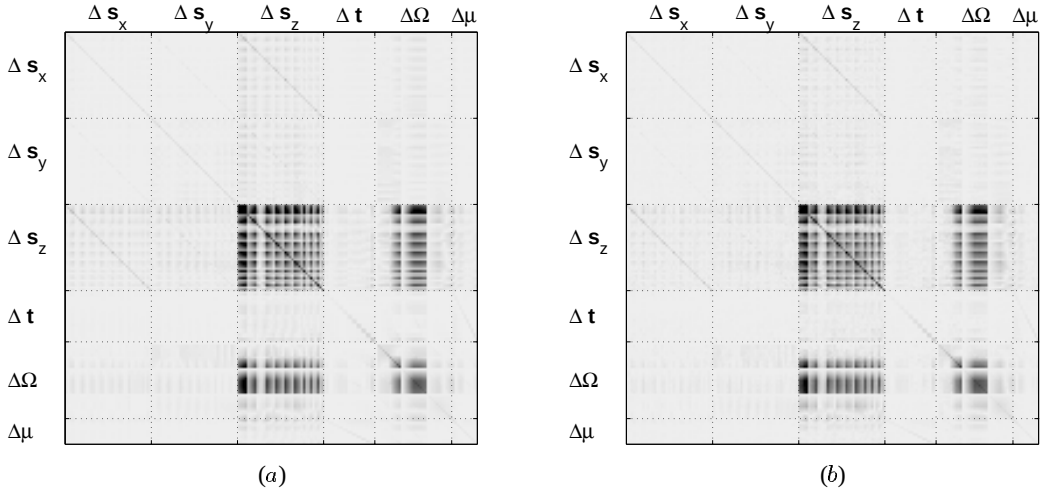


Fig. 3.4. (a) The predicted covariance in an arbitrary gauge, see equation (2.34). We note that the values and correlations are significantly different from the normal covariance in Figure 3.3, and yet it still contains the same geometric uncertainty. (b) The actual Monte Carlo covariance in this gauge. It shows close similarity to the predicted covariance in this gauge as can also be seen in Figure 3.5.

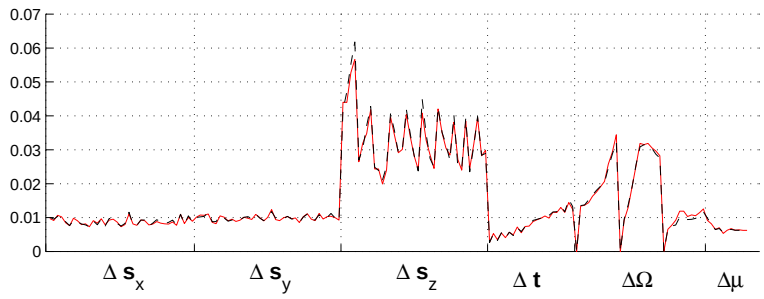


Fig. 3.5. The square root of the diagonal elements of the covariances in Figure 3.4 are shown here. This gives the net standard deviation of each parameter in the experimental gauge (3.37) obtained from the diagonal of the covariance. The solid line is the experimentally measured value and the dashed line is our prediction from the projected normal covariance.

predicted uncertainty and actual uncertainty. Since these properties are invariants, their uncertainties are exact, and not subject to an unknown transformation like scaling. Thus the true errors are revealed, rather than possibly being hidden in the choice of coordinate system. We believe reporting this uncertainty measure is essential for most quantitative analyses of the shape, and must be done for gauge invariant properties.

Table 3.1. Predicted and measured values, along with their uncertainties in standard deviations, of two gauge independent properties of the synthetic object in Figure 3.2: (left) the angle between two lines, and (right) the ratio of two lengths.

Angle	*Mean*	*Uncertainty*	**Ratio**	*Mean*	*Uncertainty*
Predicted:	90.11°	$\pm 2.10^\circ$	Predicted:	0.9990	$\pm$0.0332
Recovered:	90.02°	$\pm 2.10^\circ$	Recovered:	1.0005	$\pm$0.0345

Next we illustrate 3D reconstruction in different gauges for another synthetic shape under perspective projection. Figure 3.6 shows three images from a six image sequence, and Figure 3.7 shows the optimal reconstruction together with the uncertainty ellipsoids in four different gauges. Evidently the choice of gauge significantly affects the uncertainties. Nevertheless if we consider invariants, all of these covariances will give the same uncertainties. Some sample invariants are illustrated in Table 3.2.

Finally we show results for a real image sequence of a chapel in Figures 3.9 and 3.10 along with the reconstructed shape from SFM. The feature correspondences were determined manually. Not only can we obtain a texture-mapped reconstruction, we can also obtain measurements of similarity invariant properties such as angles with their

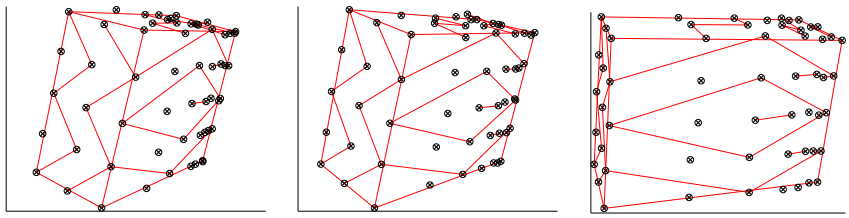


Fig. 3.6. Three images from a 6 image sequence of a synthetic object with uniform Gaussian independent noise added. The circles plotted are at twice the standard deviation of the noise.

Table 3.2. A set of invariants created from the lines marked in Figure 3.8. These include the angles between two pairs of lines, and the ratio of the lengths of two pairs of lines. The uncertainties are given in standard deviations.

Invariant	*True value*	*Estimated value*	*Uncertainty*
Angle α	31.9°	32.3	$\pm 0.6^\circ$
Angle β	90°	87.6	$\pm 6.3^\circ$
Ratio $\boldsymbol{a} : \boldsymbol{b}$	1	0.92	± 0.2
Ratio $\boldsymbol{c} : \boldsymbol{d}$	1	0.96	± 0.04

uncertainties. We found the angle and its uncertainty (in standard deviations) between two walls separated by a buttress: $117^\circ \pm 3.2^\circ$, as well as two other angles on the chapel: $46.2^\circ \pm 2.1^\circ$ and $93.2^\circ \pm 2.6^\circ$ as described in the Figure caption.

3.9 Discussion

In SFM there are basic indeterminacies in the parameters. Under perspective and weak perspective projection these correspond to a similarity transformation, and for orthography they correspond to a Euclidean transformation. In this chapter we derived explicit expressions for the gauge freedoms of these models, which permits us to transform the covariance of the shape and motion parameters between any gauge constraints we choose. We showed, through a number of examples, how choice of gauge can significantly affect the covariances of the parameters.

An important use of covariance estimates is to evaluate a method for estimating parametrized quantities, such as 3D position of points. If a method is unbiased and errors are small, the smaller the covariance the better. However, the choice of gauge may be critical in this evaluation. In some gauges the parameters may have very small covariance and in other gauges they have may have large covariances. If there is no prior justification of a particular gauge, then one can be tempted to choose a favorable gauge,

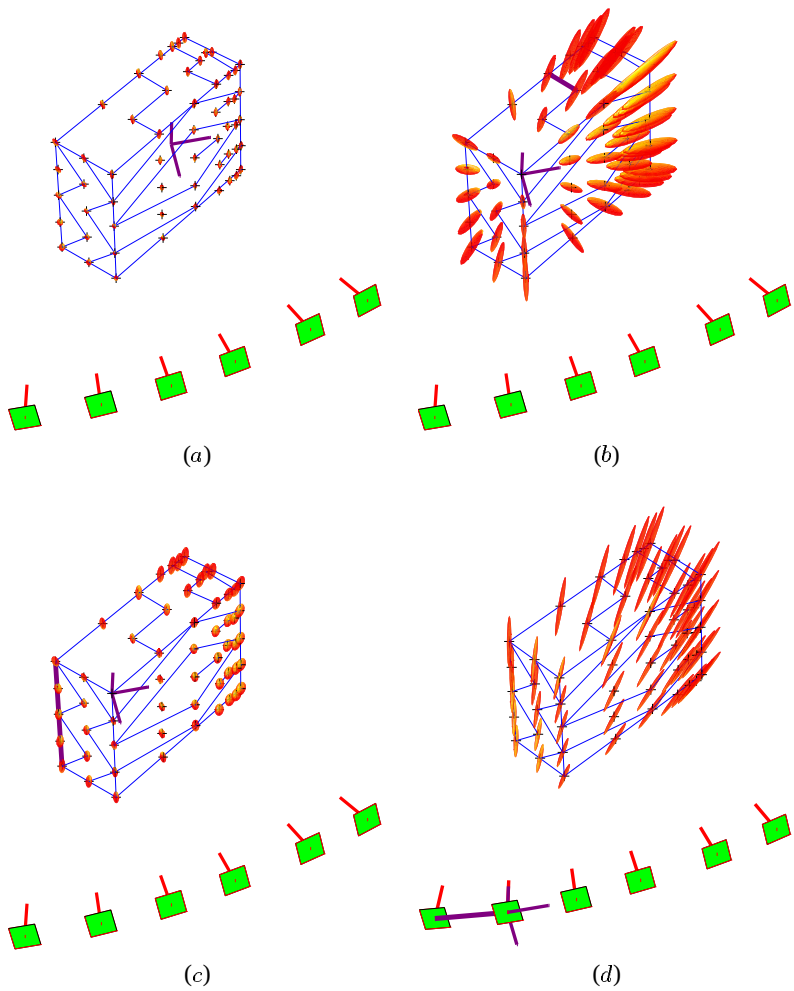


Fig. 3.7. The reconstruction of the synthetic object from the sequence in Figure 3.6 in four different gauges. The coordinate system is shown at the origin of each, and a thick solid line indicating which distance was used to fix the scale. The ellipsoids give the probability surfaces at three times the standard deviation of the position of the feature points. (a) A coordinate system at the centroid and scale fixed using the average distance to the features. (b) The coordinate system is fixed to the corner point, and a short length is used to fix the scale. (c) The coordinate system is fixed to the corner point, and a long length is used to fix the scale. (d) The coordinate system is on the second camera, and the distance between two cameras is used to fix the scale.

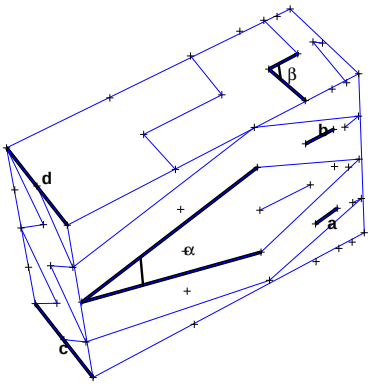


Fig. 3.8. Some angles and lengths are marked on the object. Invariants created from these and their uncertainties are calculated and shown in Table 3.2.

Fig. 3.9. Three images from a 6 image sequence of a chapel with features registered by hand.

and in doing this a significant portion of the noise can be "normalized" away. This is bad if real applications do not give us knowledge of this gauge. The safest approach to comparing uncertainties from different algorithms is to compare uncertainties of invariant properties. These do not depend on choice of gauge and so will not be increased or decreased by arbitrary choices in the experiment. Alternatively to working with invariants one could choose a gauge and justify this choice by use of real measurements of the object. This leads to the question of what is the best way to choose a gauge, and that is the subject of the next two chapters.

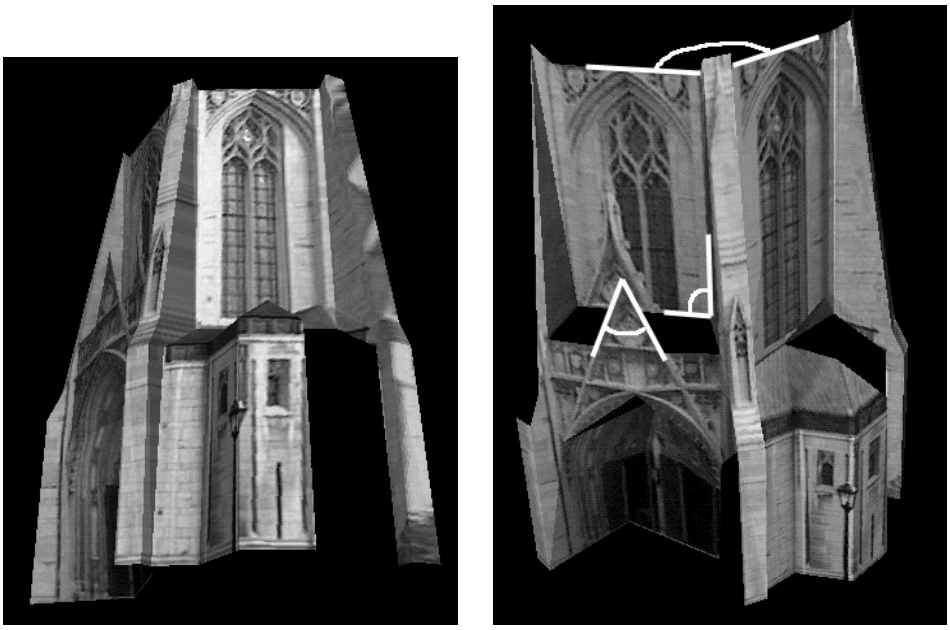


Fig. 3.10. The shape reconstruction from the image sequence in Figure 3.9. We can obtain quantitative measures and uncertainties of invariants from this reconstruction. In this case we estimated the angle between to walls separated by a buttress, and two other angles as illustrated on the far right. The values (anti-clockwise from the top) are: $117^\circ \pm 3.2^\circ$, $46.2^\circ \pm 2.1^\circ$ and $93.2^\circ \pm 2.6^\circ$.

Finally, calculating the inverse of the Fisher information matrix to obtain the covariance can be very computationally expensive. To address this we defined a new inverse of the Fisher information matrix that can be computed rapidly by taking advantage of the sparse nature of this matrix. Our geometric equivalence relationship enabled us to prove that this inverse is geometrically equivalent to the normal covariance, and hence gives us a valid covariance estimate once projected into a gauge. By reducing the computation from cubic to quadratic in feature points for many cases, this dramatically reduces the cost of uncertainty analysis.

4. Determining Scale

We have shown that we can estimate gauge invariant quantities exactly, and obtain a unique covariance for these irrespective of gauge choice. However, many quantities that we would like to estimate are not gauge invariant. For example the size of an object is not an invariant as it depends on scale. In order to estimate non-invariant quantities we must use additional information to constrain our solution. But using additional information to constrain the solution, which we call gauge fixing, will affect the accuracy of the solution. It is this effect on accuracy that we would like to investigate. In particular we would like to know how to achieve the greatest accuracy with the least information.

The non-invariant property that we are particularly interested in is the length of an object, or the distance between two points on an object. Typically length is one of the key properties that we want to know about an object whose shape we have estimated. It is more useful than the ratio of two lengths, which is an invariant. For example, in analyzing the shape of a door, we could obtain a full invariant that specifies the ratio of the height to the width of the door, but evidently this is less useful than knowing what the height and the width actually are.

Length has a special property that is useful for applications and is easier to analyze than other non-invariant quantities such as 3D position. Length is invariant to rotations and translations. We thus call it a partial invariant. If we can fix scale, then length becomes an invariant. And more generally partial invariants can be made into full invariants of a reduced parameter space by fixing only a partial subset of the gauge freedoms. For length estimates there still remains the problem of how best to constrain the scale of the parameter space, but this is an easier problem than trying to fix scale, translation and rotation.

In this chapter we will apply our gauge constraint and gauge freedom theory to the estimation of partial invariants. We will consider length measurements as our partial invariants in the SFM domain. Our aim will be to see how fixing scale affects the accu-

racy of length estimates, and how we may be able to maximize the accuracy of length estimates.

4.1 Fixing Scale

In order to obtain length estimates on a model known up to a similarity transformation, we need to determine the scale factor. Traditionally this unknown scale factor in 3D estimation has caused no great concern in the computer vision field. It is typically thought that we can measure a length of the real object, find the unknown scale factor, and then use this as a change-of-variables transformation to rescale both the model and the covariance. We will show in this section that this widely held assumption is false. Then in subsequent sections we will use gauge theory we have developed to derive the correct transformation of a covariance matrix when the model scale is determined.

4.1.1 Problem Statement

Let us represent the shape of our object with a vector, $\boldsymbol{s}$, that contains all of the x, y, and z coordinates of all the individual feature points:

$$\boldsymbol{s} = \begin{pmatrix} \boldsymbol{s}_1 \\ \vdots \\ \boldsymbol{s}_N \end{pmatrix}, \quad \text{where} \quad \boldsymbol{s}_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}. \tag{4.1}$$

Here, $\boldsymbol{s}_i$ denotes the 3D position of the ith feature point. Let us assume further that we only know $\boldsymbol{s}'$, the total shape up to an unknown scale factor, a, where $\boldsymbol{s} = a\boldsymbol{s}'$. We also know that the uncertainty of the shape, $\Delta\boldsymbol{s}'$, has zero mean and a covariance matrix, $\boldsymbol{V}'_{\boldsymbol{s}} = E[\Delta\boldsymbol{s}'\,\Delta\boldsymbol{s}'^{\top}]$.

In this chapter we ignore rotational and translational degrees of freedom, since our goal is to obtain distance estimates which are invariant to these degrees of freedom. A set of 3D points implicitly contains a fixed rotation and translation, and so we choose to work in this default gauge. The lengths we predict will be invariant to changes in this gauge.

Since our model is known only up to a scale factor, we must perform an additional measurement in order to specify the model exactly. Say we measure the distance between feature points, i and j, on the real object and find that its value is d. We conclude that for our model, $\boldsymbol{s}'$, to correspond to the true object, it must be rescaled by a scale factor given by:

$$a = \frac{d}{\|\boldsymbol{s}'_i - \boldsymbol{s}'_j\|} \,. \tag{4.2}$$

The new model is then:

$$\boldsymbol{s} = a\boldsymbol{s}' \,. \tag{4.3}$$

Our question is: Now that we know the scale, what is the covariance, $\boldsymbol{V_s}$, of the rescaled model?

4.1.2 Naive Solution

A direct answer may be obtained as follows. Perturbations of the new model are given by:

$$\Delta\boldsymbol{s} = a\Delta\boldsymbol{s}' \,, \tag{4.4}$$

with fixed scale a, and so the Jacobian matrix for a change-of-variables is:

$$\frac{\partial \boldsymbol{s}}{\partial \boldsymbol{s}'} = a\boldsymbol{I} \,. \tag{4.5}$$

The covariance for the new model would then be:

$$\begin{aligned} \boldsymbol{V_s} &= E[\Delta\boldsymbol{s}\,\Delta\boldsymbol{s}^\top] \\ \boldsymbol{V_s} &= \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{s}'} E[\Delta\boldsymbol{s}'\,\Delta\boldsymbol{s}'^\top] \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{s}'}^\top \\ &= a^2 \boldsymbol{V_{s'}} \,. \end{aligned} \tag{4.6}$$

Thus we would conclude that we can rescale the covariance with the square of the scale factor. This is exactly a change-of-variables transformation.

4.1.3 Contradiction

It is easy to see that there is something wrong with equation (4.6). We have measured the distance, d, between points i and j on our rescaled model, $\boldsymbol{s}$. Assume there are no errors in the measurement of d.[1] Then this distance has no uncertainty and so must have zero variance. However in our new model, $\boldsymbol{s}$, we have: $d = \|\boldsymbol{s}_i - \boldsymbol{s}_j\|$, and so perturbations of d are expressed:

$$\Delta d = \nabla_{\boldsymbol{s}} d^\top \Delta\boldsymbol{s} \,. \tag{4.7}$$

[1] We show in section 4.2.3 how to incorporate measurement uncertainties, but for simplicity we do not consider that here.

Then, using the covariance in equation (4.6), the variance of d would be:

$$\sigma_d^2 = \nabla_{\boldsymbol{s}} d^\top \boldsymbol{V}_{\boldsymbol{s}} \nabla_{\boldsymbol{s}} d \,, \tag{4.8}$$

where we have defined the gradient of a scalar to be:

$$\nabla_{\boldsymbol{s}} d = \begin{pmatrix} \partial d / \partial s_{x1} \\ \vdots \\ \partial d / \partial s_{zN} \end{pmatrix} . \tag{4.9}$$

This calculated variance, σ_d^2, will not be zero in general, except for a special covariance, $\boldsymbol{V}_{\boldsymbol{s}'}$, and choice of features. By assumption, however, it is zero, and so there is a contradiction. We conclude that even though we have the correct scale factor a, unless the variance, σ_d^2, between the points we chose is zero, simply scaling the covariance by a^2, as in equation (4.6), is the wrong transformation.

4.2 Gauge Fixing

The contradiction between the properties we expected our covariance to have and the properties of the rescaled covariance was caused by ignoring the gauge freedoms and gauge constraints of the problem. The 3D points are known up to a scale factor, and hence have a gauge freedom. While the rescaling of the covariance was correct, we ought to have regarded our measurement as a gauge constraint, and so applied our oblique projection operator along the gauge freedom onto this constraint. In this section we seek to reinterpret the problem with gauge freedoms in mind and derive the correct solution.

4.2.1 Problem Reformulation

We started with a shape, $\boldsymbol{s}'$, and covariance, $\boldsymbol{V}_{\boldsymbol{s}'}$, in an unknown gauge, $\mathcal{C}'$. We made a measurement, d, and will use this as a constraint on our model:

$$d - \|\boldsymbol{s}_i - \boldsymbol{s}_j\| = 0 \,, \tag{4.10}$$

and as such it defines a new gauge, $\mathcal{C}$. We obtained a rescaled shape, $\boldsymbol{s} = a\boldsymbol{s}'$, that belongs to the same gauge orbit, $\mathcal{G}_{\boldsymbol{s}}$, and also satisfies the constraint of this new gauge, $\mathcal{C}$. Now we want the covariance, $\boldsymbol{V}_{\boldsymbol{s}}$, at $\boldsymbol{s}$ that lies in the tangent space to $\mathcal{C}$.

4.2.2 Oblique Projection

The transformed covariance, $\boldsymbol{V_s}$, besides being in the tangent space to $\mathcal{C}$, must be geometrically equivalent to the original covariance, $\boldsymbol{V_{s'}}$. The solution to this problem is to rescale and project the original covariance into the new gauge as given by equation (2.52) and expressed in our parameters as:

$$\boldsymbol{V_s} = \boldsymbol{Q_s^{\mathcal{C}}} \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{s'}} \boldsymbol{V_{s'}} \frac{\partial \boldsymbol{s}}{\partial \boldsymbol{s'}}^{\top} \boldsymbol{Q_s^{\mathcal{C}}}^{\top} . \tag{4.11}$$

Here $\partial \boldsymbol{s} / \partial \boldsymbol{s'}$ is the rescaling Jacobian from equation (4.5), and $\boldsymbol{Q_s^{\mathcal{C}}}$ obliquely projects onto the tangent space to $\mathcal{C}$ while maintaining geometric equivalence. This transformed covariance thus gives the true uncertainty of the shape after we make measurement d. Now we simply need to derive what $\boldsymbol{Q_s^{\mathcal{C}}}$ is.

The oblique projection, $\boldsymbol{Q_s^{\mathcal{C}}}$, will be of the form derived in section 2.4.4:

$$\boldsymbol{Q_s^{\mathcal{C}}} = I - U_{\boldsymbol{s}} (V_{\boldsymbol{s}}^{\top} U_{\boldsymbol{s}})^{-1} V_{\boldsymbol{s}}^{\top} . \tag{4.12}$$

Here $U_{\boldsymbol{s}}$ spans the tangent space to the gauge orbit, and $V_{\boldsymbol{s}}$ spans the orthogonal complement of the gauge $\mathcal{C}$. We can derive these from the infinitesimal gauge generators as done for SFM in section 3.4.2.

Perturbations in the tangent space of the gauge orbit, $\mathcal{G}_{\boldsymbol{s}}$, are given by:

$$\boldsymbol{s} + \boldsymbol{D}(\boldsymbol{s}) = (1 + \Delta a)\boldsymbol{s} \tag{4.13}$$

which implies

$$\boldsymbol{D}(\boldsymbol{s}) = \boldsymbol{s} \Delta a \tag{4.14}$$

and so at $\boldsymbol{s}$, $U_{\boldsymbol{s}} = \boldsymbol{s}$.

Let $V_{\boldsymbol{s}} = \boldsymbol{v}$ be a vector orthogonal to the gauge tangent space, $\boldsymbol{v} \in T[\mathcal{C}]^{\perp}$. One such vector is given by the gradient of the constraint:

$$\boldsymbol{v} = \nabla_{\boldsymbol{s}} c(\boldsymbol{s}) = \frac{\partial \sqrt{\|\boldsymbol{s}_i - \boldsymbol{s}_j\|}}{\partial \boldsymbol{s}} = \begin{pmatrix} 0 \\ \vdots \\ (\boldsymbol{s}_i - \boldsymbol{s}_j)/d \\ 0 \\ \vdots \\ (\boldsymbol{s}_j - \boldsymbol{s}_i)/d \\ 0 \\ \vdots \end{pmatrix} . \tag{4.15}$$

The constraint $c(\boldsymbol{s}) = 0$ was defined in equation (4.10). Finally we obtain an expression for our oblique projection operator:

$$\boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}} = \boldsymbol{I} - \frac{\boldsymbol{s}\boldsymbol{v}^\top}{\boldsymbol{v}^\top \boldsymbol{s}} \,. \tag{4.16}$$

This oblique projection operator is illustrated in Figure 2.7. It takes any perturbation $\Delta \boldsymbol{s}'$ to a geometrically equivalent perturbation $\Delta \boldsymbol{s}$ in a new gauge $\mathcal{C}$. It can thus be used to transform our covariance in equation (4.11) to a covariance in our new gauge $\mathcal{C}$.

4.2.3 Measurement Uncertainty

We can now ask what happens when there is uncertainty in the measurement of length d on the object, which is often the case. We expand our basic scale equation (4.3) including a perturbation term, Δa, on scale itself:

$$\Delta \boldsymbol{s} = a \boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}} \Delta \boldsymbol{s}' + \boldsymbol{s}' \Delta a \,. \tag{4.17}$$

The second term is simplified as follows. The scale is $a = d/d'$ where $d' = \|\boldsymbol{s}_i' - \boldsymbol{s}_j'\|$, and so $\Delta a = \Delta d/d'$. If d is measured with standard deviation σ_m, then substituting for Δa into equation (4.17), and finding the covariance we obtain:

$$\boldsymbol{V}_{\boldsymbol{s}} = a^2 \boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}} \boldsymbol{V}_{\boldsymbol{s}'} \boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}\top} + \sigma_m^2 \frac{\boldsymbol{s}'\boldsymbol{s}'^\top}{d'^2} = a^2 \boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}} \boldsymbol{V}_{\boldsymbol{s}'} \boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}\top} + \sigma_m^2 \frac{\boldsymbol{s}\boldsymbol{s}^\top}{d^2} \,. \tag{4.18}$$

We see that the measurement component to the variance is inversely weighted by the length on the object that is measured. Thus measuring longer lengths in the shape often reduces this component of the error. This is quite intuitive.

4.3 Finding the Best Gauge

The choice of gauge affects the final accuracy of our results. It is natural to want to choose a gauge that maximizes the accuracy of the model. In this section we consider two measures of final accuracy and how gauges should be chosen to optimize these.

4.3.1 Optimizing Overall Accuracy

One measure for accuracy is the trace of $\boldsymbol{V}_{\boldsymbol{s}}$, which is the sum of the individual 3D feature-point variances. Reducing this corresponds to improving the overall accuracy, but ignoring cross-correlation effects. If $\boldsymbol{V}_{\boldsymbol{s}'}$ includes off-diagonal elements, the analysis becomes complicated, and so we approximate it with uniform and uncorrelated noise:

$\boldsymbol{V}_{\boldsymbol{s}'} = \sigma_0^2 \boldsymbol{I}$. Now we ask: If we can measure the distance, $d = \|\boldsymbol{s}_i - \boldsymbol{s}_j\|$, between any two points i and j, which two points should be used to minimize the trace: Tr[$\boldsymbol{V}_{\boldsymbol{s}}$]?

The vector $\boldsymbol{v}$, from equation (4.15), is zero except for the ith and jth elements which are: $(\boldsymbol{s}_i - \boldsymbol{s}_j)/d$ and $(\boldsymbol{s}_j - \boldsymbol{s}_i)/d$ respectively. From this we get: $\boldsymbol{v}^\top \boldsymbol{s} = d$ and $\boldsymbol{v}^\top \boldsymbol{v} = 2$, and $\boldsymbol{s}\boldsymbol{v}^\top$ is a square matrix whose only two, non-zero, diagonal blocks are: $\boldsymbol{s}_i(\boldsymbol{s}_i - \boldsymbol{s}_j)^\top$ and $\boldsymbol{s}_j(\boldsymbol{s}_j - \boldsymbol{s}_i)^\top$. Using these, and substituting for $\boldsymbol{Q}_{\boldsymbol{s}}^{\mathcal{C}}$ and $\boldsymbol{V}_{\boldsymbol{s}'}$ in equation (4.18), we obtain:

$$\mathrm{Tr}[\boldsymbol{V}_{\boldsymbol{s}}] = (3N-2)a^2\sigma_0^2 + (2a^2\sigma_0^2 + \sigma_m^2)\frac{\|\boldsymbol{s}\|^2}{d^2}\,. \tag{4.19}$$

We see that given constant measurement error, σ_m, the longer the length on the model we choose, d, the smaller the total uncertainty.

4.3.2 Optimizing One-Length Accuracy

For real models, there is typically strong correlation between features and so the approximation that $\boldsymbol{V}_{\boldsymbol{s}'}$ is uncorrelated may be poor. Also, we may be interested in the accuracy of only part of the model, and in some cases just one length. Let us say that our goal is to estimate a certain length, $e = \|\boldsymbol{s}_k - \boldsymbol{s}_l\|$, with the greatest accuracy, and that we can set the scale factor by measuring another length, d. What qualities should d have to minimize the variance of e?

Let $\boldsymbol{g}' = (e' \;\; d')^\top$ and $\boldsymbol{g} = a\boldsymbol{g}'$, and so we can write:

$$\begin{aligned}\boldsymbol{V}_{\boldsymbol{g}'} &= \nabla_{\boldsymbol{s}'}\boldsymbol{g}'^\top \boldsymbol{V}_{\boldsymbol{s}'}\nabla_{\boldsymbol{s}'}\boldsymbol{g}' \\ &\equiv \begin{pmatrix} \sigma_{e'}^2 & \sigma_{e'd'} \\ \sigma_{d'e'} & \sigma_{d'}^2 \end{pmatrix}\,.\end{aligned} \tag{4.20}$$

If we measure d with variance σ_m^2, we obtain from equation (4.18):

$$\boldsymbol{V}_{\boldsymbol{g}} = a^2\boldsymbol{Q}_{\boldsymbol{g}}^{\mathcal{C}}\boldsymbol{V}_{\boldsymbol{g}'}\boldsymbol{Q}_{\boldsymbol{g}}^{\mathcal{C}\top} + \sigma_m^2\frac{\boldsymbol{g}\boldsymbol{g}^\top}{d^2}. \tag{4.21}$$

Here our scale factor gives us a gauge freedom: $\partial\boldsymbol{g}/\partial a = \boldsymbol{g}'$, and our constraint that $\|d\|$ is a constant gives us a local constraint: $\boldsymbol{v} = (0 \;\; 1)^\top$, and so

$$\boldsymbol{Q}_{\boldsymbol{g}}^{\mathcal{C}} = \begin{pmatrix} 1 & -e/d \\ 0 & 0 \end{pmatrix}\,. \tag{4.22}$$

We want to minimize the variance of e, which can be obtained algebraically from equation (4.21) as:

$$\sigma_e^2 = a^2(\sigma_{e'}{}^2 - 2\frac{e}{d}\sigma_{e'd'} + \left(\frac{e}{d}\right)^2\sigma_{d'}^2) + \left(\frac{e}{d}\right)^2\sigma_m^2\,. \tag{4.23}$$

This is a quadratic in the ratio e/d, and its minimum has two cases. The first is when the noise is uncorrelated or anti-correlated, ($\sigma_{e'd'} \leq 0$). The variance, σ_e^2, is reduced when the ratio, e/d, is reduced. Thus given constant variances, the longer the length we measure, d, the more accurate our estimate for e is. The second case is when the noise is positively correlated, $\sigma_{e'd'} > 0$. The ratio that minimizes σ_e^2 is then:

$$\frac{e}{d} = \frac{a^2\sigma_{e'd'}}{a^2\sigma_{d'}^2 + \sigma_m^2} \,. \tag{4.24}$$

If the noise is perfectly correlated, namely $\sigma_{e'd'} = \sigma_{e'}\sigma_{d'}$, and $\sigma_m^2 = 0$, and the ratio, e/d, is given by equation (4.24), then the length e will be perfectly estimated with zero covariance.

But we can also understand equation (4.21) geometrically using our oblique projection interpretation. Let us represent $\boldsymbol{V}_{\boldsymbol{g}'}$ with a 2D ellipse giving the standard deviation of d and e on two axes. The oblique projection operator, $\boldsymbol{Q}_{\boldsymbol{g}}^{\mathcal{C}}$, projects this along the gauge freedom onto the e axis. The gauge freedom is a line whose slope is e/d. Figure 4.1 shows how the magnitude of the resulting variance, σ_e^2, is obtained by the projecting the ellipse, $\boldsymbol{V}_{\boldsymbol{g}'}$, onto the vertical axis. The steeper the slope, or the smaller d is compared to e, the larger the resulting variance, σ_e^2, is. The relationship between ratio $d : e$ and σ_e is plotted for this example in Figure 4.2. We note that while it appears that the rank of the variance is being reduced from two to one, this is because we are considering only two dimensions of a higher dimensional space, and if we looked at the full covariance of all the parameters its rank would not change by the oblique projection.

A further interesting consequence of the projection shown in Figure 4.1(a) is that if the ratio, $e/d = 1$, then the gauge freedom will be at $45°$, and the projection of the ellipse onto the coordinate axes will be the same if either d or e is fixed. This means that if the lines are the same length, then holding one line fixed and predicting the other, or holding the other fixed and predicting the first will give exactly the same uncertainty. This symmetry property, depending only on the length, gives intuition on the effects of gauge fixing, and will simplify gauge fixing calculations for lines of equal length.

4.4 Feature Localization Error

In doing some real experiments we realized that there is a systematic error in 3D reconstruction that is often overlooked, but that can create significant bias errors in the prediction of 3D shape. We will refer to it as the *feature localization error*, and will distinguish it from correspondence error.

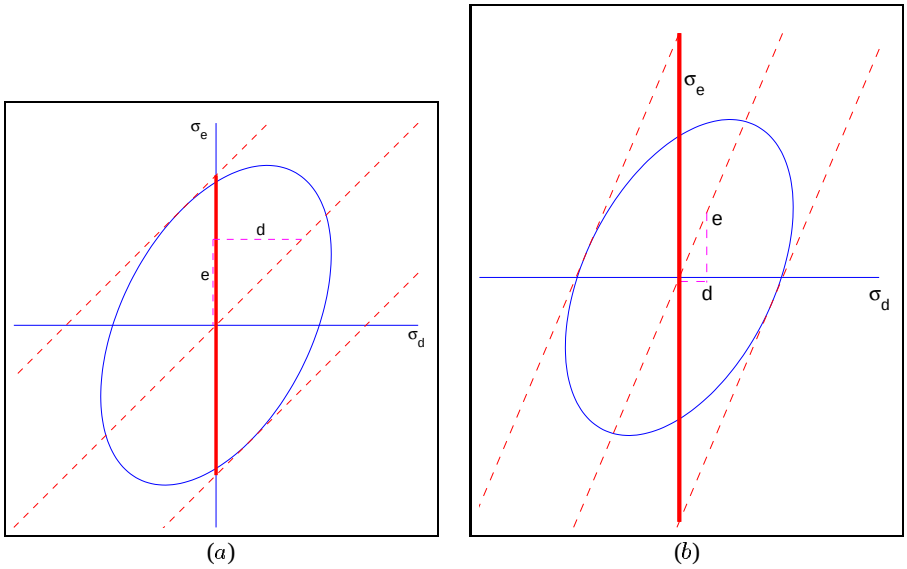


Fig. 4.1. Illustrated are the projections of a covariance $\boldsymbol{V}_{\boldsymbol{g}'}$ onto a single axis to give us σ_e^2, as described in equation (4.21) with σ_m=0. In (a) the gauge freedom direction is given by e/d = 1, whereas in (b) $e/d > 1$. Notice that the ratio of lengths e/d, and hence gauge freedom direction, significantly affects the magnitude of the projection onto the e axis. In general the larger d is compared to e the smaller the projection, although when there is positive correlation, there is a minimum projection when this ratio is given by equation (4.24).

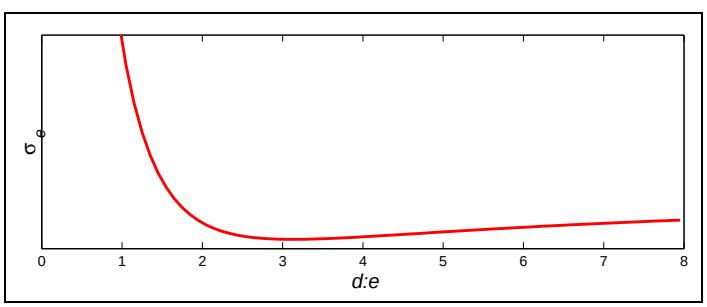


Fig. 4.2. The plot shows the standard deviation, σ_e, of the predicted line length, e, as a function of the ratio of the measured line length to the predicted line length, $d : e$. In general the larger d the better the estimate, except beyond a certain value after which the uncertainty increases moderately to an asymptote.

In the feature correspondence problem, we are given the projection of a 3D point in one image, and seek for the corresponding point in the rest of the images. It is this problem that is usually dealt with when feature tracking errors are modeled.

The feature localization problem, however, corresponds to identifying the actual projection point of a 3D feature point. If images were of infinite resolution, this could be obtained, but due to the finite resolution of pixels, we cannot locate the exact position in an image corresponding to a 3D point. This is illustrated in Figure 4.3. Here we see that, if we look closely at the image around the corner point, the corner is fuzzy and it is not possible to give its precise coordinates. The best we can do is to give an estimate with some uncertainty of a few pixels.

From Figure 4.3 we see that feature localization error is not the same as the correspondence error. To find correspondences we look at another image of the corner, and find the region in it that most closely resembles the region identified in the first image. This correspondence will have some error since the intensities between images change and the regions warp due to affine or perspective effects. This error is generally modeled as an unbiased perturbation, and when we perform SFM we can hope to reduce it as much as possible, and obtain an unbiased estimate along with a covariance estimate depending on the amount of this noise. However, if we have an error in feature localization, as shown in Figure 4.3 and also in Figure 4.4, this will propagate throughout the correspondences, and all the corresponding image points will incorporate this localization component. As we will see, this will result in a biased final estimate for the 3D point.

We will model the localization error as follows. Let, $\boldsymbol{s}$, be the 3D point that we wish to estimate, and $\boldsymbol{p}$ be its projection coordinates in the image in which we initially selected the features. Let us assume that we have a localization error in which we measure a point $\boldsymbol{p}_L$ close to $\boldsymbol{p}$. Further let us assume that when we find corresponding features in the other images, these all correspond to a 3D point, $\boldsymbol{s}_L$, close to $\boldsymbol{s}$. When we do SFM we will obtain an estimate for $\boldsymbol{s}_L$, not for $\boldsymbol{s}$, and so will obtain a 3D bias equal to the difference between these points: $\boldsymbol{s}_L - \boldsymbol{s}$.

There are a few points to note about this localization error. First it only affects the shape parameters and does not affect the motion parameters. Actually the localization error in each 2D point only affects the 3D estimate of that feature, and no other 3D feature, although every 3D point may have its own localization error. In applications that simply need any 3D points on or near the surface of an object, this localization error is unimportant. However, when we want to estimate an actual 3D point on the object, or

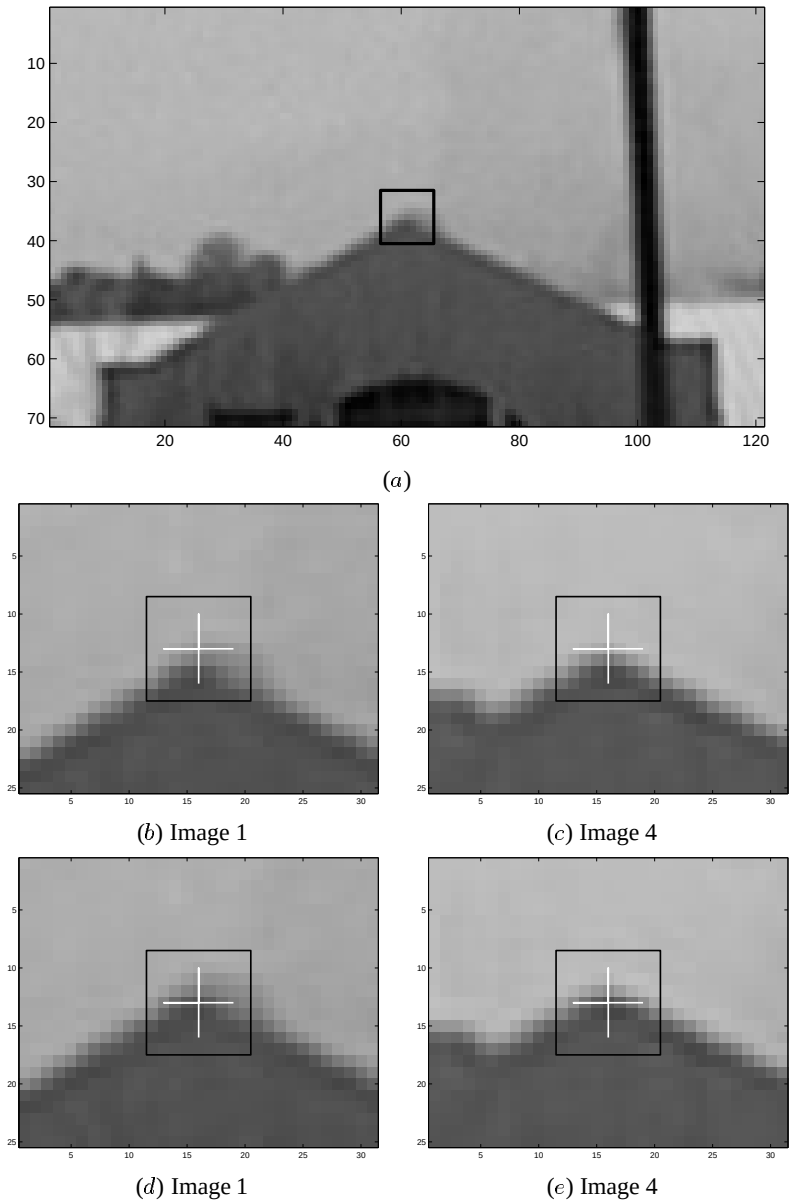


Fig. 4.3. We would like to find the image position corresponding to the corner of the roof-top shown in the top image, and the corresponding positions in all the images. We see from the close-up views that it is not easy to identify the exact coordinates of the corner. We could assume it is the point in (b) in the first image. Then it is not too hard to identify the corresponding point in the fourth image as shown in (c). But alternatively we could estimate the position of the corner to be that shown in (d), and then the corresponding point in the fourth image would be that in (e). The matching errors between (b) and (c), and between (d) and (e), are correspondence errors, and we assume these are unbiased. The difference between (b) and (d), and also (c) and (e), is the result of image-localization error. This has a 3D cause and will result in biased 3D estimates.

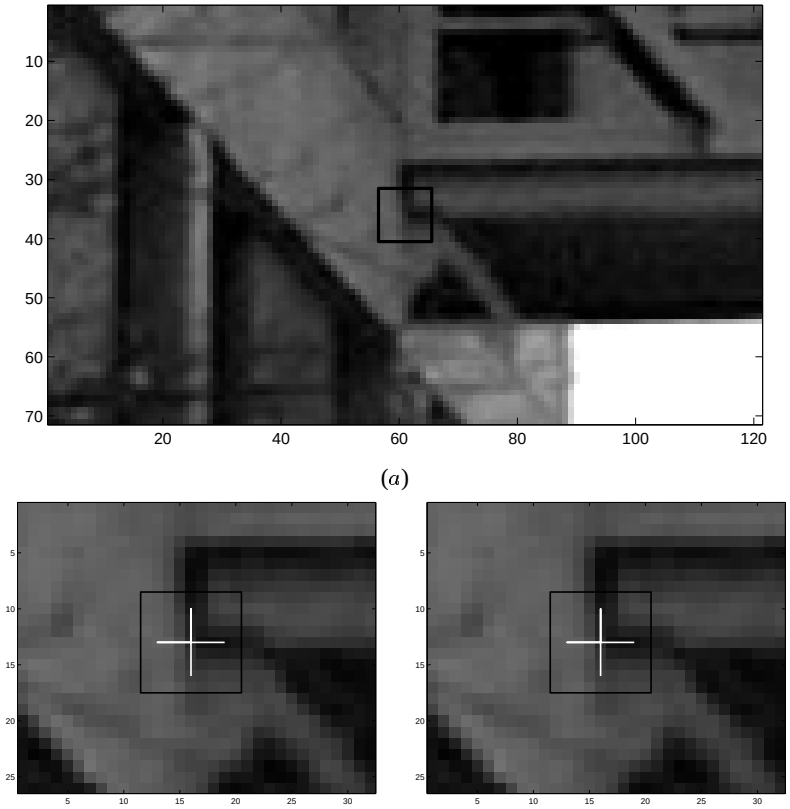


Fig. 4.4. Shown above is another example of the difficulty in feature localization. We want our feature to correspond to the projection of the 3D point where the bottom of the bridge connects to the building. But, even though we may do sub-pixel tracking between images, we cannot tell where this true point is in image 1 to more than a couple pixel accuracy.

actual line lengths on the object, this localization error will result in unmodeled error, namely, the error will be larger than our expected error.

We have two options: either minimize the localization error, or incorporate a model of it in our uncertainty formulation. There are various approaches taken in minimizing localization error. One approach is to take images of a feature from all around it. But this creates a much harder task for the image taker and feature matcher, and besides many features are not visible from a large set of angles. Furthermore, this should not be just a planar circling of the feature, but a full 3D encircling. Evidently this is not feasible for most situations. Other approaches to reducing this error include, instead of using templates to find corners, to look for the intersection of lines. This may reduce the error,

but even edges are fuzzy as can be seen in Figure 4.3, and it is not clear if the middle or one side of the image edge corresponds to the projection of the actual edge in 3D. Hence edges themselves will have this localization error.

Since we cannot remove localization error, we will explicitly create a model of it and use this to augment our uncertainty model and remove the bias from the gauge fixing. One approach to incorporating localization error is to increase the image-based feature uncertainties, but this will not correctly incorporate this error into our uncertainty model as it will affect the camera motion uncertainty which is not affected by localization error. Alternatively we could treat this effect as an inter-frame correlation between corresponding features. While this approach may work, it is not clear what the appropriate amount of correlation to add would be.

Our approach for incorporating localization error in the uncertainty model is as follows. First we choose one image in which to express the image-localization errors. A 3D point, $\boldsymbol{s}_i = (X_i, Y_i, Z_i)$, is projected onto an image point, $\boldsymbol{p}_i$, under perspective projection in the form:

$$\boldsymbol{p}_i = \frac{f}{Z_i} \begin{pmatrix} X_i \\ Y_i \end{pmatrix} , \tag{4.25}$$

where f is the focal length in pixels. Let us assume that our image-localization error, $\boldsymbol{p}_{Li} - \boldsymbol{p}_i$, has a uniform 2D Gaussian distribution centered at the measured point $\boldsymbol{p}_{L_i}$ and with known variance σ_L^2. Let us also assume that the 3D localization error, $\boldsymbol{s}_L - \boldsymbol{s}$, also has a Gaussian distribution with uniform, but unknown variance, $\sigma_{\boldsymbol{s}L}^2$ in all directions. It is easy to see that these variances are related by the formula:

$$\sigma_{\boldsymbol{s}L}^2 = \frac{Z_i^2}{f^2} \sigma_L^2 \tag{4.26}$$

Thus given an image distribution of our bias, we can estimate the 3D distribution of the bias. We see from this equation that the standard deviation of a point, due to localization error, is simply proportional to the depth of the point.

Our goal is to predict the variance of an estimated line length, $d = \|\boldsymbol{s}_i - \boldsymbol{s}_j\|$, due to localization error. Assuming that all image points have equal localization error variance, σ_L^2, we find that this 3D length will have a variance of:

$$\sigma_{dL}^2 = \frac{\sigma_L^2}{f^2} (Z_i^2 + Z_j^2), \tag{4.27}$$

where Z_i and Z_j are the depths of its end-points.

The localization error in line length given by equation (4.27) can be incorporated into our uncertainty model the same way as the measurement error, σ_m, was added in

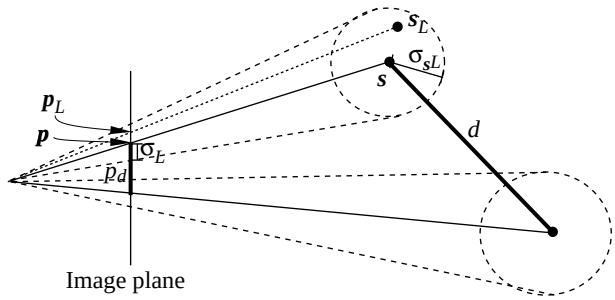


Fig. 4.5. A 3D feature $\boldsymbol{s}_L$ will have a reconstruction bias which we assume has a Gaussian distribution represented by the sphere around the true 3D point $\boldsymbol{s}$. We obtain the radius of this sphere, and hence the standard deviation of this point by projecting a known image-based bias likelihood onto this point using equation 4.26. Points further from the image will have a larger uncertainty. The variance of a length d is obtained by adding the variances of its endpoints.

equations (4.18) and (4.21). Assume we are trying to predict the length $e = ae'$ between points $\boldsymbol{s}_k$ and $\boldsymbol{s}_l$, by measuring the length d between points $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ in the scene. Let $\boldsymbol{g}' = (e' \;\; d')^\top$. Then if we incorporate the localization error into equation (4.21), the new estimate for the covariance of $\boldsymbol{g} = a\boldsymbol{g}'$ becomes:

$$\boldsymbol{V}_{\boldsymbol{g}} = a^2 \boldsymbol{Q}_{\boldsymbol{g}}^{\mathcal{C}} \boldsymbol{V}_{\boldsymbol{g}'} \boldsymbol{Q}_{\boldsymbol{g}}^{\mathcal{C}\top} + (\sigma_m^2 + \sigma_{dL}^2)\frac{\boldsymbol{g}\boldsymbol{g}^\top}{d^2} + \begin{pmatrix} \sigma_{eL}^2 & 0 \\ 0 & 0 \end{pmatrix} . \tag{4.28}$$

The variance of the localization error, σ_{dL}^2, on length d is added to the measurement variance on this length, σ_m^2, and the last term simply adds the variance of the localization error of length e to σ_e^2. In adding these components to the variance we are implicitly assuming that they are independent of the other components, and this seems to be a reasonable assumption. The final covariance of our predicted length, e, is obtained from this as:

$$\begin{aligned} \sigma_e^2 = a^2 &(\sigma_{e'}^2 - 2\frac{e}{d}\sigma_{e'd'} + \left(\frac{e}{d}\right)^2 \sigma_{d'}^2) \\ &+ \left(\frac{e}{d}\right)^2 \sigma_m^2 \\ &+ \left(\frac{e}{d}\right)^2 \frac{\sigma_L^2}{f^2}(Z_i^2 + Z_j^2) + \frac{\sigma_L^2}{f^2}(Z_k^2 + Z_l^2) . \end{aligned} \tag{4.29}$$

The terms on the right in the first line are due to the gauge freedom projection. The second line contains the measurement error of d component. The third line is the localization error of the end-points of lines d and e. One thing to notice is that when the end-points of d have the same depth, that is when $Z_i = Z_j$, then $Z_i/f = d/p_d$ where p_d

is the projected length of d in the image. If line e is also parallel to the image, then the third line of equation (4.29) simplifies to:

$$2e^2(\frac{\sigma_L^2}{p_d^2} + \frac{\sigma_L^2}{p_e^2}). \tag{4.30}$$

The magnitude of localization error of a point depends on its depth; the further away the point is from the cameras the larger the expected localization error. When a line is used for gauge fixing, we see from equation (4.29) that the localization error is magnified by the ratio e/d, and so it will cause shorter lines to generate more error.

4.5 Experiments

Our goals for this section are to demonstrate empirically that our uncertainty modeling using gauge fixing gives quantitatively correct results, and also to obtain a qualitative assessment of the effects of gauge fixing on uncertainties. In real 3D estimation tasks we may have to insert a measuring rod or other device into a scene to fix the scale before taking the images and doing reconstruction. This means we cannot use a computer to optimize its position based on the 3D reconstruction before inserting it into the scene, but rather must rely on qualitative knowledge of how its position and orientation will affect 3D accuracy to guide its placement. Hence we perform a number of synthetic experiments from which we derive a qualitative understanding of how the position of a line affects its usefulness in fixing scale. We validate these results and those derived in this chapter on three real sequences.

4.5.1 TV Reconstruction

First we illustrate, on a real 3D object, how model accuracy changes when different object measurements are used to fix the scale. We start with a set of registered features in an image sequence as shown in Figure 4.6. A batch Structure from Motion algorithm, using our free-gauge optimization procedure from Chapter 2.5.1, was used to obtain the 3D shape, $\boldsymbol{s}'$ and camera motion. The covariance of all the parameters was calculated using our fast inversion method and then the shape components were taken from this to give us, $\boldsymbol{V}_{\boldsymbol{s}}'$, the shape covariance of rank $3N - 1$ due to an unknown scale factor. Implicitly there is also an unknown rotation and translation, but this does not matter as we will only calculate distance measures which are invariant to rotation and translation. The shape and covariance are illustrated in Figure 4.7.

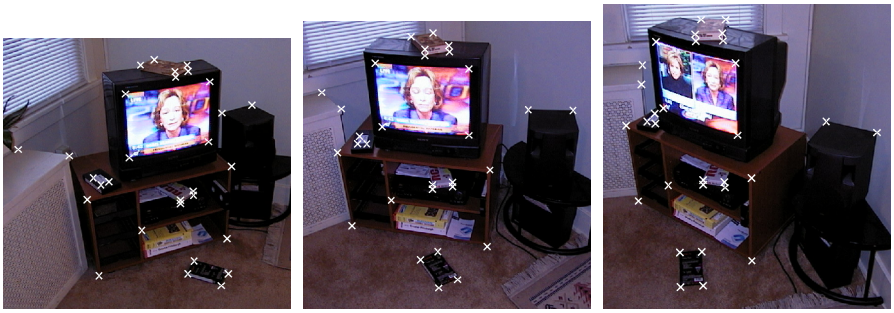


Fig. 4.6. Three images from a seven-image sequence with hand-registered features.

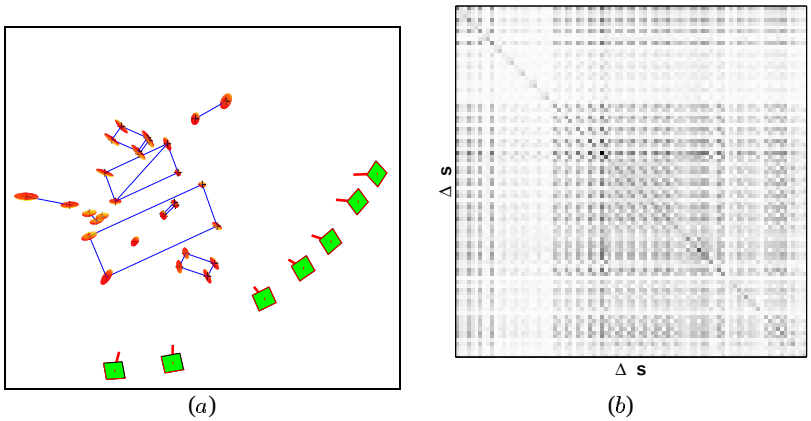


Fig. 4.7. (a) The 3D reconstruction of point features, $\boldsymbol{s}'$, from the image sequence in Figure 4.6 using a Structure from Motion algorithm. Lines are drawn for clarity only. Ellipsoids represent the 3×3 blocks of the covariance matrix $\boldsymbol{V}_s'$, but scaled by a factor of 20 to aid viewing. The recovered camera positions are also shown. (b) This is a schematic of the covariance, $\boldsymbol{V}_s'$, which corresponds to the normal covariance of this shape. In calculating this, image features were assumed to have uniform, identical noise. This plot shows that there is a strong correlation between the reconstructed features.

Suppose we want to estimate the diagonal length of the television screen. How accurately can we know this length? Since we can recover 3D shape only up to a scale factor, the shape alone will not determine this length. We first have to find the overall scale of the recovered shape. Now it may be that we know the length of another object in the scene, in which case this could be used to obtain the scale. For the purpose of this experiment, we will look at a number of different objects in the scene whose lengths we know, and find out which gives us the best estimate of the TV diagonal length.

Consider, then, the 13 lines, measured in 3D, and shown in Figure 4.8. We made 13 separate estimates of the TV diagonal length, in each case using one of these lengths to

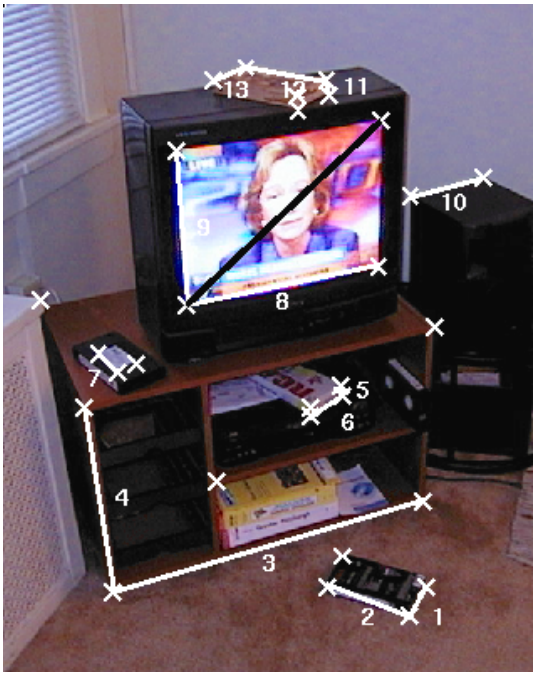


Fig. 4.8. We want to estimate the diagonal length of the TV shown by the black line. To do this we need to know the length of one of the white lines.

fix our gauge, $\mathcal{C}$, and obtain the scale. The effects of fixing the scale using these lines are shown for some sample lines in Figure 4.9. The absolute uncertainties of positions, as shown by the ellipsoids in this figure are not our goal since these depend on choice of coordinate system. Rather we want to estimate lengths on the object and their uncertainties. Figure 4.10 shows these estimates, along with their predicted standard deviations, σ_e, obtained using equation (4.18). We assumed no error in the measurement of the 3D line lengths, $\sigma_m = 0$, and no localization error, $\sigma_L = 0$. We note that the actual error corresponds well with the uncertainty given by the predicted error.

We notice, from Figure 4.11, that there is a large variation in uncertainty of the TV diagonal, depending on which line is used to fix the scale. Figure 4.11 shows that in general longer lines lead to better estimates, and the shortest lines, 5 and 11, give the greatest uncertainty. The puzzle is lines 8 and 9, which give more accurate estimates than line 3, which is longer. This is explained by Figures 4.12 and 4.13 which show that line 8 is more strongly correlated with the TV diagonal length, and that the ratio with the TV diagonal length, given by equation (4.24), is closer to its optimal value for line 8 than for line 3. The main reason for this strong correlation between line 8 and the diagonal

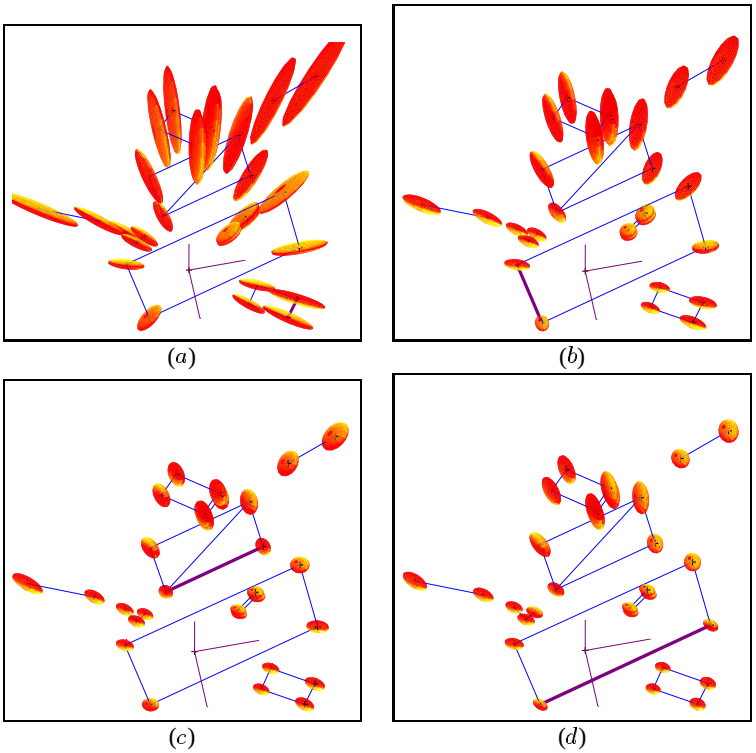


Fig. 4.9. We illustrate the effect of fixing the scale on the uncertainties using several gauges. In each case the ellipsoids represent 20 times the standard deviation of the features. (a) Line 1 is fixed, (b) line 4 is fixed, (c) line 8 is fixed, and (d) line 3 is fixed.

is that they share one of their end-points. So, in a sense, it is unfair to use line 8. In the rest of the experiments we will avoid using lines that share end-points with the line they predict. We will also seek for a better understanding of why, besides length, some lines provide better measures than others for fixing the scale.

4.5.2 Sphere Reconstruction

We have seen that line length is important in the accuracy of gauge fixing. We would also like to gain an intuition on how relative orientation of lines, both with respect to the cameras and with respect to other lines, affects the uncertainties of the lines and their relative correlation. To do this, we created a number of synthetic sequences, and investigated the correlation effects. For simplicity we eliminated measurement error σ_m and localization error σ_L from these experiments.

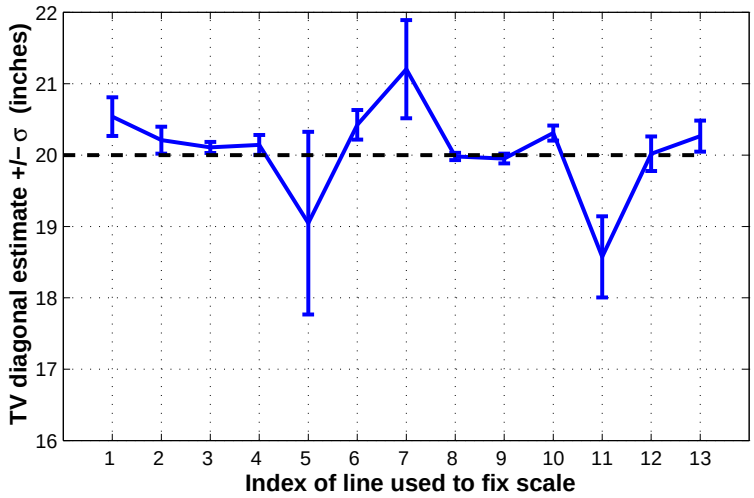


Fig. 4.10. The true length of the TV diagonal is 20 inches. This plot shows the estimate we obtained of this length using each of the different lines to fix the scale. The number on the abscissa corresponds to the line in Figure 4.8. The uncertainty, σ_e, given by the error bars, varies greatly depending on which line was used; the largest uncertainty with standard deviation of 1.3 inches is obtained using line 5, and the smallest with standard deviation of 0.05 inches is obtained using line 8.

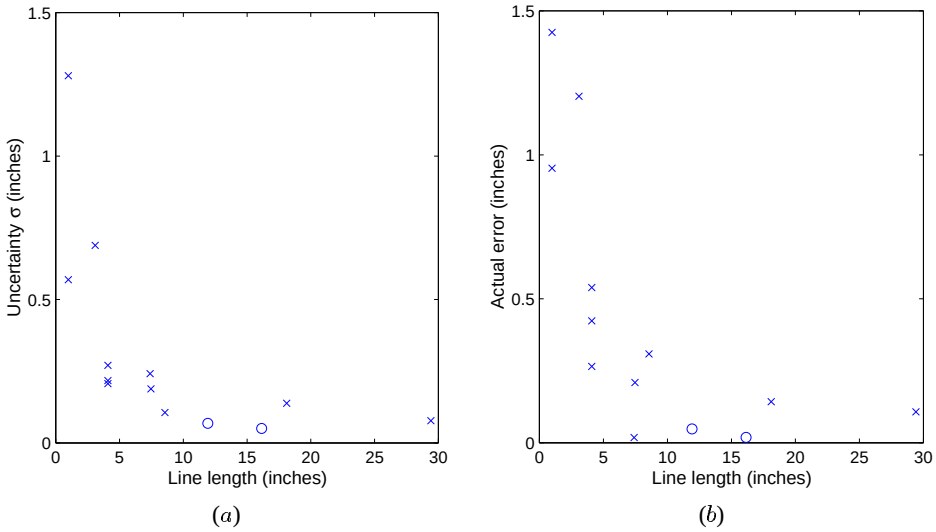


Fig. 4.11. (a) Plot of uncertainty vs. line length used to fix the scale. Lines 8 and 9 are marked with an "o" symbol as they share an end-point with the TV diagonal which we are estimating, and hence have an unfair advantage. (b) Actual errors in prediction vs. line length.

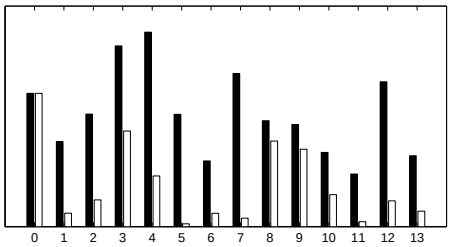


Fig. 4.12. The black bars show the standard deviation, σ_i, for each line length in Figure 4.8, calculated from the unscaled shape covariance, $\boldsymbol{V_s}$. Bar 0 corresponds to the TV diagonal. The scale is arbitrary and so left unmarked. The white bars indicate the proportion of cross-correlation, and are calculated as: $r_{0i}\sigma_i$, where $r_{0i} = \sigma_{0i}/\sigma_0\,\sigma_i$ is the correlation coefficient ($-1 \leq r_{0i} \leq 1$). When perfectly correlated with the TV diagonal, the white bar will equal the black bar. Hence, apart from line 0 itself, we see that lines 8 and 9 are most strongly correlated with the TV diagonal.

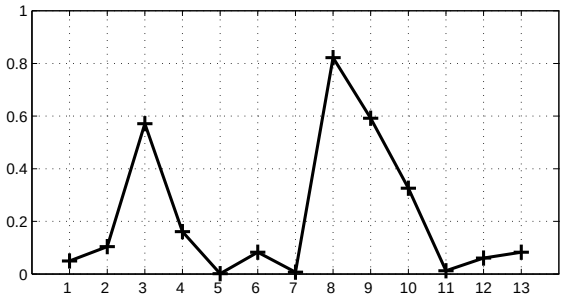


Fig. 4.13. Given the corresponding variance and cross-correlation plotted in Figure 4.12, we can use equation (4.24) to calculate what the optimal length would be for each line. By "optimal" we mean it would minimize the variance of the TV diagonal estimate. Here we plot the ratio of the actual line length to the optimal line length for each line. A value of 1 means the line has optimal length. Line 8 is closest to its optimal length. This is a better measure to use than length in selecting a line to fix the scale. While all of these lines are shorter than their optimal lengths, we found in other experiments that lines can also be longer than their optimal lengths.

Our first sequence is of a sphere created by orthogonally intersecting 3 circles. Lines are formed by connecting antipodal points, and so are all of the same length. The camera faces the sphere and translates a distance equal to the diameter of the sphere perpendicular to the viewing direction along which five images are taken, as illustrated in Figure 4.14. Uniform uncorrelated Gaussian noise is assumed for each point. Our goal is to most accurately predict the four of the diameters labeled 1, 4, 7 and 18 in the figure. For each case we will test to see which line provides the best gauge constraint for maximum prediction accuracy.

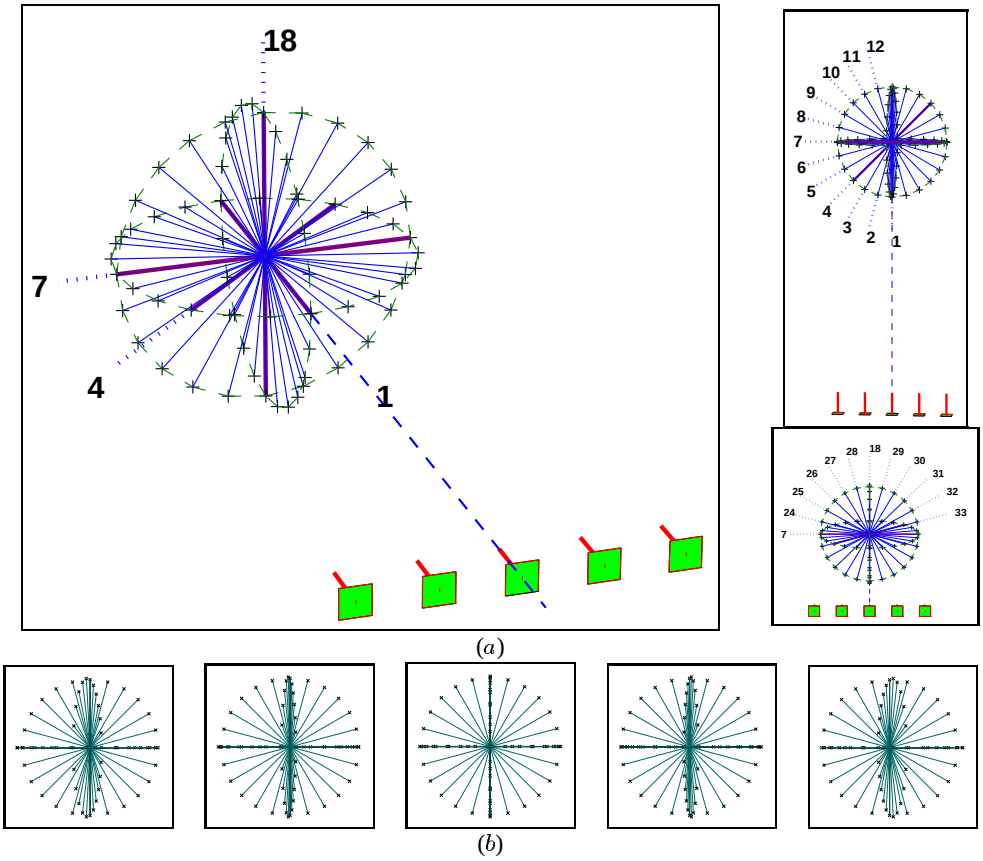


Fig. 4.14. (a) Views of a sphere created by intersecting 3 circles orthogonally, and five cameras. We are interested in predicting four diameters labeled 1, 4, 7 and 18, where diameter 1 is along the viewing direction, 4 is at $45°$ to this, and 7 and 18 are in the plane orthogonal to the viewing direction. To predict these lengths we will individually fix all of the other diameters marked with solid lines. The dashed circles are only to aid in viewing. (b) Images seen by the cameras.

As a first step in predicting line 1, the normal covariance is calculated with the origin at the center of the sphere. The standard deviation of each line calculated directly from this is plotted in Figure 4.15. Only the relative magnitudes of these values are relevant as the scale has not been fixed yet. In this figure the goal is to predict line 1, and so we also show the correlation between each of these lines and line 1. We find the standard deviation of the error in predicting line 1 using each of these lines individually to fix the scale as in equation (4.23). Since all the lines are the same length, this is the same as the uncertainty of using line 1 to predict each of the other lines. Figure 4.16 shows the analogous uncertainties for predicting lines 4, 7 and 18.

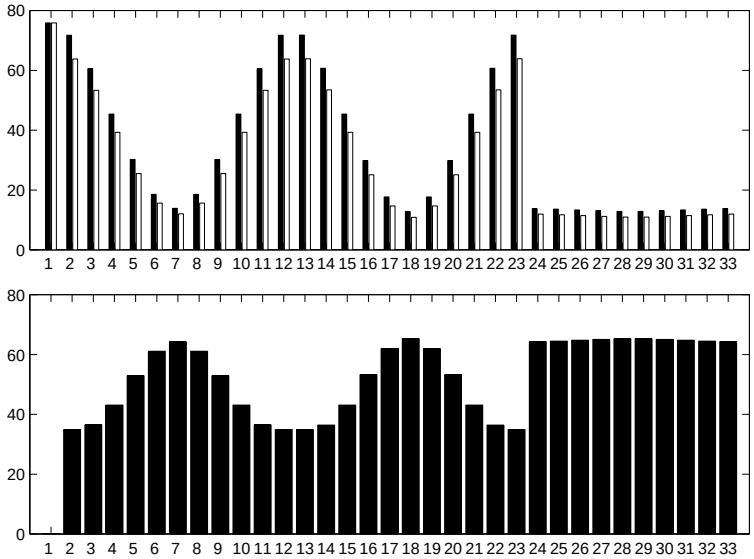


Fig. 4.15. We want to predict line 1 with minimum variance by fixing one of the other lines. The black bars in the top plot show the standard deviation of each line calculated from the normal covariance. The light bars show the correlation factor between each of these lines and line 1, as described in Figure 4.12. We can use these correlations to give the uncertainty in predicting line 1, and this is shown in the bottom plot for each line. Lines 1–12 are diameters of the circle with line 18 as its axis. Lines 1, 13–23 are diameters of the circle with line 7 as its axis, and lines 7, 18, 24–33 are diameters of the circle with line 1 as its axis. The four lines with minimum error are: 2, 12, 13, and 23, and these correspond to the four lines closest to being parallel to line 1.

There are a number of qualitative patterns we can see in these plots. First a general comment. We note from Figure 4.14 that all the cameras are relatively close together and have the same viewing direction, and so we will approximate this as our single viewing line from center camera to the mid-point of all the lines at the center of the sphere. Then, as we would expect, lines that are close to, and approximately parallel to, the viewing line have much greater uncertainty, as calculated from the normal covariance, than lines that are in the plane orthogonal to this line.

Line 1 is along this viewing line and so has a large uncertainty in the normal covariance. Lines close to being parallel to line 1 have larger uncertainties than perpendicular lines, but they also have larger correlation with line 1. From the uncertainty plot in Figure 4.15, we see that this correlation effect overwhelms the additional uncertainty of these lines, and results in lines that are close to being parallel to the viewing direction being better measures for gauge fixing than orthogonal lines, when we want to estimate lines along the viewing direction. This is the main surprise from this experiment.

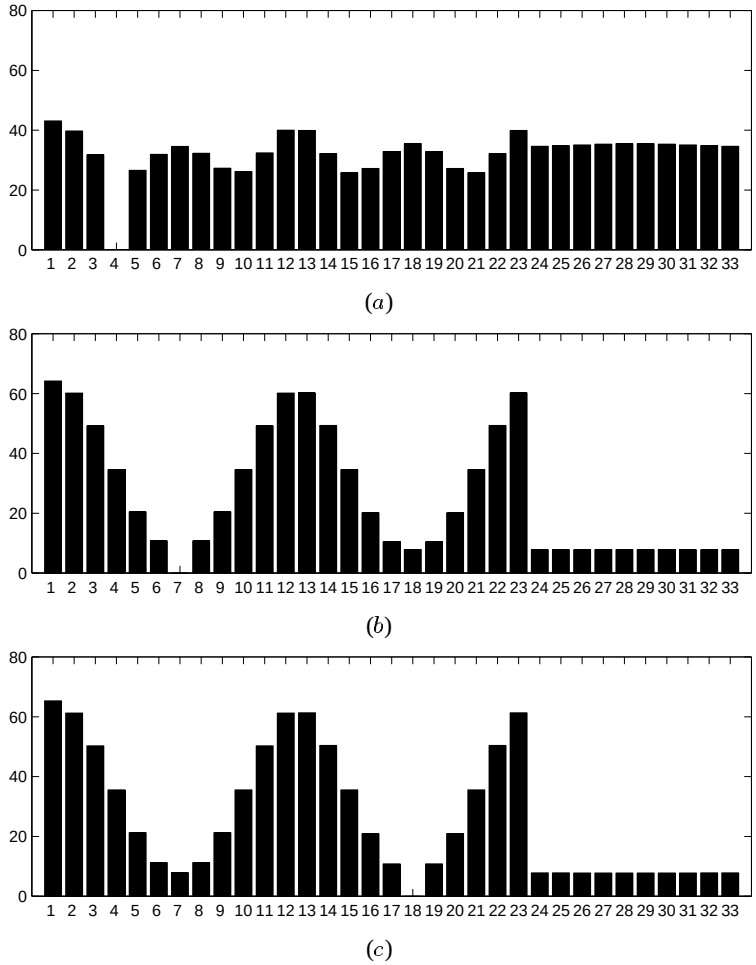


Fig. 4.16. Shown are the errors obtained when using each of the lines on the sphere (see Figures 4.14 and 4.15) are used to predict lines (a) 4, (b) 7, and (c) 18. In predicting line 4, the three lines with the minimum error (apart from line 4 itself) are the three other lines on the the same $45°$ cone. In predicting lines 7 and 18, we find that all lines on the plane orthogonal to the viewing direction are equally good, irrespective of direction.

From Figure 4.16 we see that line 4 is best predicted by fixing another line with the same angle to the viewing direction as itself. This is because these lines are more strongly correlated with line 4 than other lines.

Lines 7 and 18 are in the plane orthogonal to the viewing line, and, as we might expect, other lines in this plane are better measures for gauge fixing than lines out of this plane and closer to the viewing direction. An interesting fact is that the orientation of the line in this plane does not affect its accuracy in predicting other lines in this plane.

A model that links all of the above results is to conclude that the prediction accuracy depends on the angle between the line and the viewing direction. The larger this angle and closer to being orthogonal, the more accurately a line can be predicted. Moreover the best gauge constraint to select in each case is a line that has the same angle with respect to the viewing direction.

4.5.3 Reconstruction of Sphere with Additional Features

Now in this example our sphere is the whole object. We would like to know how our results might change when the lines we are interested in are only part of the object that we are reconstructing. Adding more features on or close to the sphere will not qualitatively change the results. This is shown in Appendix B. But if our object is large compared to the sphere, the results might change. In the next set of experiments we investigate how changing the object dimensions will affect the relative accuracies of the lines in our sphere.

We added one hundred randomly positioned features, within a rectangular parallelpided region, that are assumed to be rigidly attached to the sphere. Figures 4.17 through 4.21 show cases with five different regions of added points. These additional points form the rest of the synthetic object in each case. Our goal is to see if the uncertainty patterns we elucidated for the single sphere still apply when the full object is much larger.

In each case the prediction of lines in the plane orthogonal to the viewing direction is the same, namely, these lines are always most accurately predicted by other lines in the same plane, and also they most accurately predict other lines in the plane. Hence we do not show plots of these.

The main difference between the plots is in the accuracy of predicting line 1; the line along the viewing direction. We see in Figures 4.17, 4.18 and 4.19 that line 1 is most accurately predicted by measuring a line close to parallel to it. On the other hand, in Figures 4.20 and 4.21 we find that lines in the plane orthogonal to the viewing direction become the best predictors. In these last two figures the uncertainty pattern is inverted, and lines in the perpendicular plane are always better for gauge fixing than lines along the viewing direction, irrespective of the line that is being predicted. We would like to understand this phenomenon, and be able to predict when it will happen.

We can understand the differences in the uncertainties in predicting line 1 as follows. In Figures 4.17, 4.18 and 4.17, there is a strong correlation between lines close to the viewing direction, whereas in Figures 4.20 and 4.21 this correlation disappears.

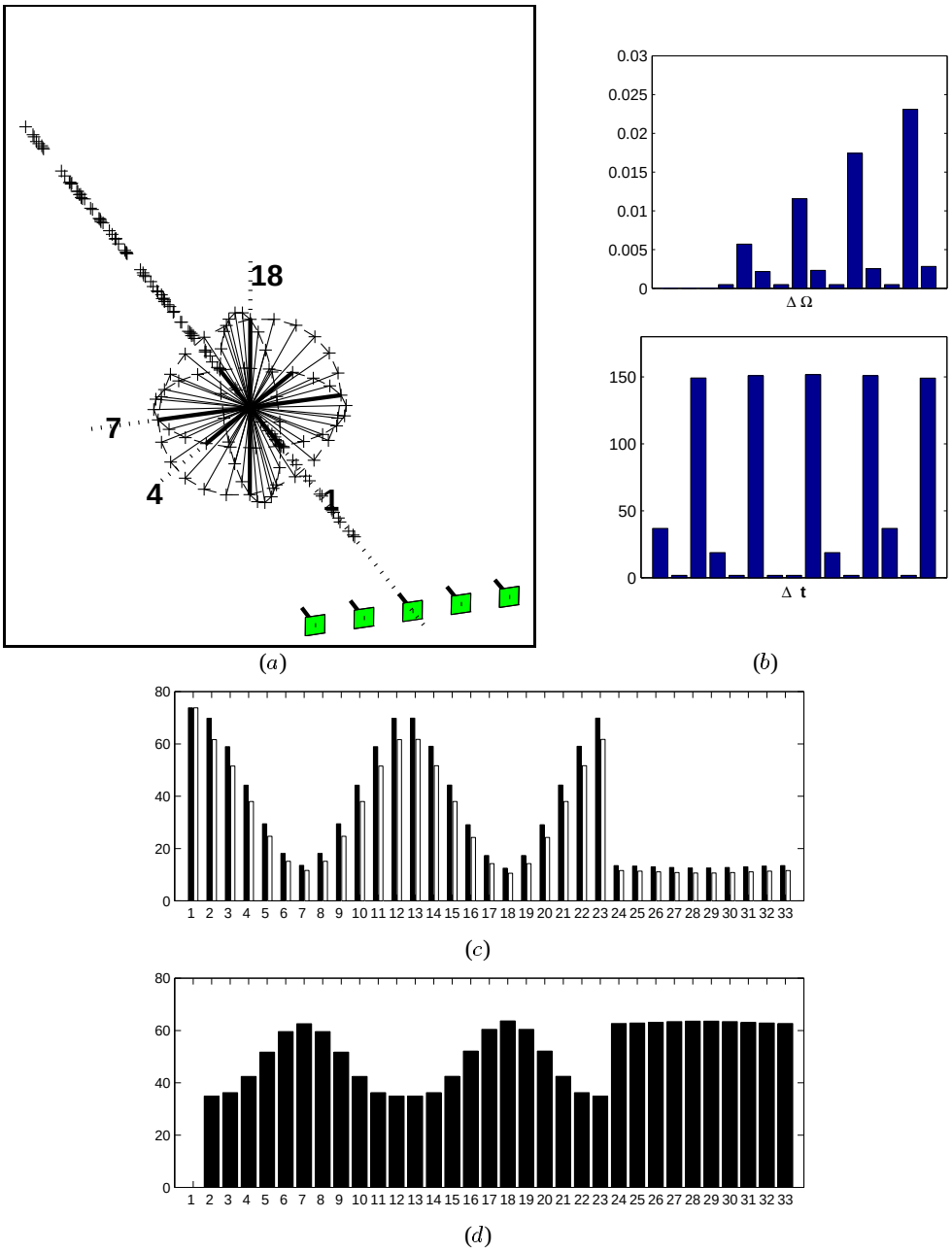


Fig. 4.17. (a) A random set of 3D points are added to the 3D sphere. (b) The rotation uncertainty, $\Delta\boldsymbol{\Omega}$, and translation uncertainty, $\Delta\boldsymbol{t}$, for all five cameras, with the rotation in the first camera being fixed. (c) The normal covariance for each line (dark) and the correlation between each line and line 1 (white). (d) The resulting uncertainty in predicting line 1 when each of the other lines is fixed.

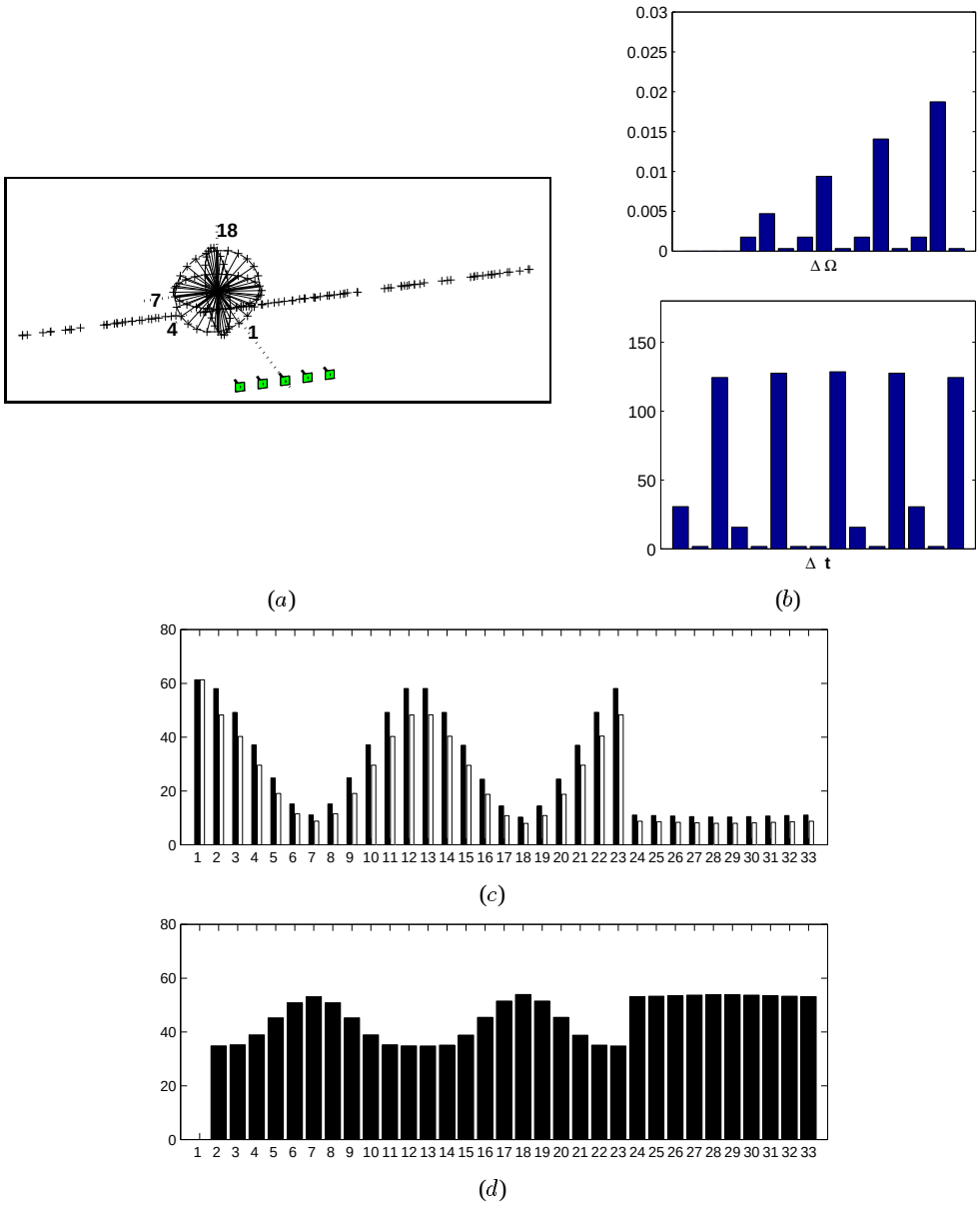


Fig. 4.18. (a) A random set of 3D points are added to the 3D sphere. (b) The rotation uncertainty, $\Delta\boldsymbol{\Omega}$, and translation uncertainty, $\Delta\boldsymbol{t}$, for all five cameras, with the rotation in the first camera being fixed. (c) The normal covariance for each line (dark) and the correlation between each line and line 1 (white). (d) The resulting uncertainty in predicting line 1 when each of the other lines is fixed.

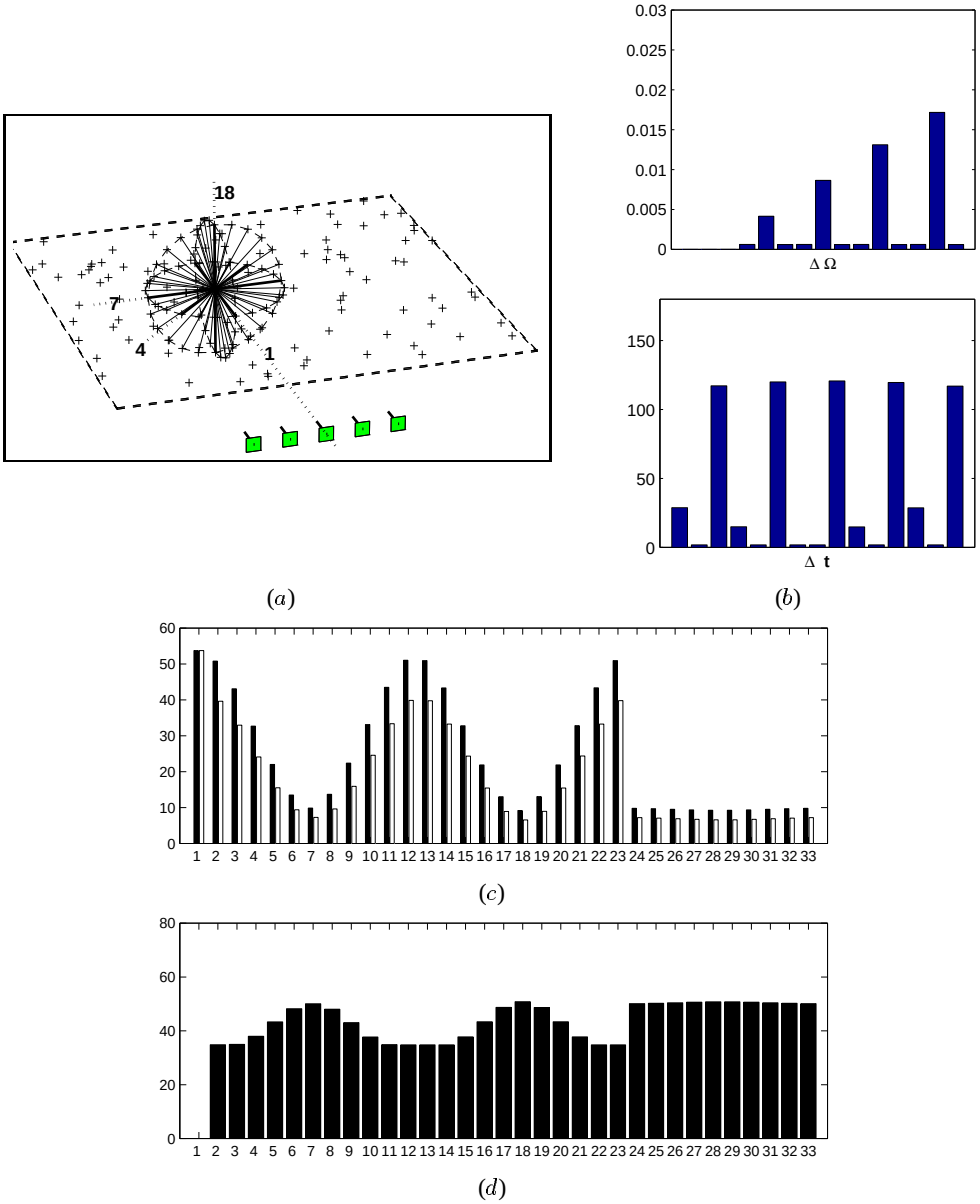


Fig. 4.19. (a) A random set of 3D points in a rectangular parallelpided region are added to the 3D sphere. (b) The rotation uncertainty, $\Delta\Omega$, and translation uncertainty, Δt, for all five cameras, with the rotation in the first camera being fixed. (c) The normal covariance for each line (dark) and the correlation between each line and line 1 (white). (d) The resulting uncertainty in predicting line 1 when each of the other lines is fixed.

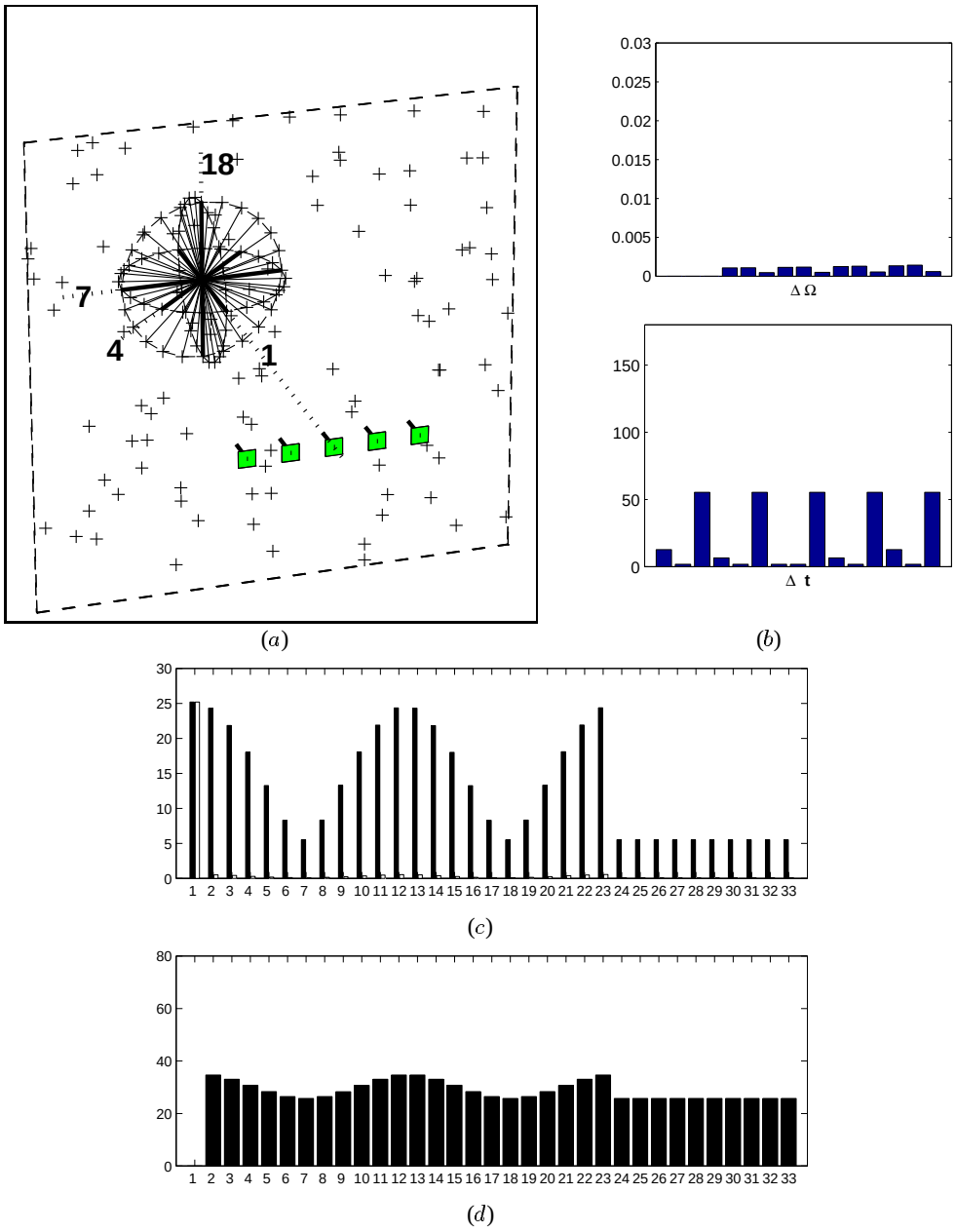


Fig. 4.20. (a) A random set of 3D points in a rectangular parallelpided region are added to the 3D sphere. (b) The rotation uncertainty, $\Delta\Omega$, and translation uncertainty, Δt, for all five cameras, with the rotation in the first camera being fixed. (c) The normal covariance for each line (dark) and the correlation between each line and line 1 (white). (d) The resulting uncertainty in predicting line 1 when each of the other lines is fixed.

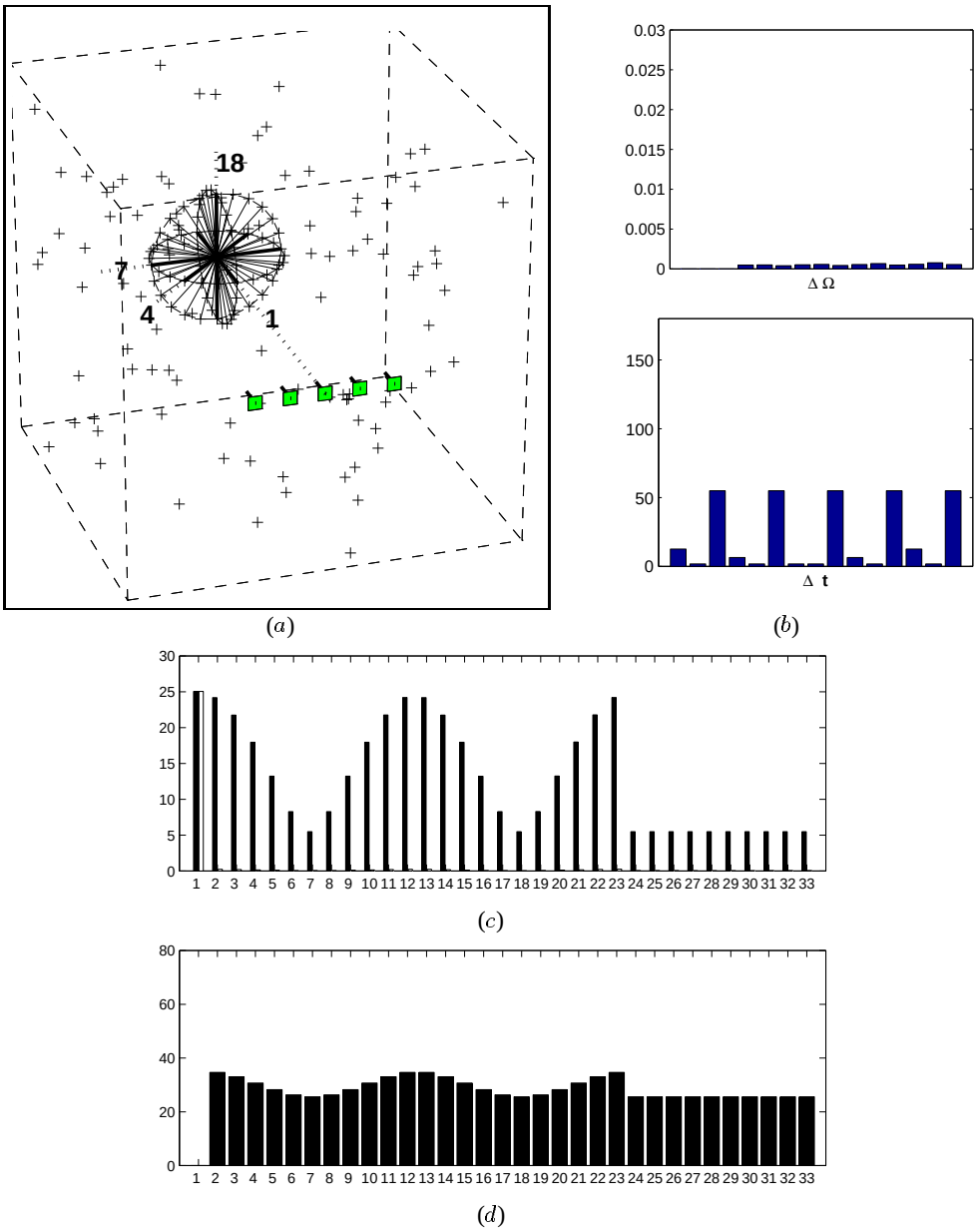


Fig. 4.21. (a) A random set of 3D points in a rectangular parallelpided region are added to the 3D sphere. (b) The rotation uncertainty, $\Delta\Omega$, and translation uncertainty, Δt, for all five cameras, with the rotation in the first camera being fixed. (c) The normal covariance for each line (dark) and the correlation between each line and line 1 (white). (d) The resulting uncertainty in predicting line 1 when each of the other lines is fixed.

We see from equation (4.23) that without the correlation between lines, fixing lines in the plane perpendicular to the viewing direction, which are more accurately located than other lines, will result in a more accurate estimate for line 1. Now if we knew the camera motion exactly there would be no correlation between the 3D feature points[2]. It is precisely because of the uncertainties in camera motion that there is correlation between 3D points. And indeed, Figures 4.17, 4.18 and 4.17 have large camera motion uncertainty and a large correlation between line lengths, whereas Figures 4.20 and 4.21 have small camera motion uncertainty and also very little line length correlation. We propose that the reason for the accurate motion estimates of Figures 4.20 and 4.21 is the large span of additional feature points in height and width. Or equivalently we can say that when the object has a large field of view compared to the lines we are interested in, the accuracy in motion estimation is sufficient that it does not significantly contribute to correlation between lines.

We are now in a position to explain why in some gauges and for some objects we can predict lines along the viewing direction more accurately than lines perpendicular to it. Figure 4.22 illustrates that when we estimate features from a set of cameras with known positions and orientations, the uncertainty is greater along the viewing direction than perpendicular to it, (see Matthies and Shafer [46]). Line lengths, which are the distance between two features, can thus be more accurately estimated perpendicular to the viewing direction than along it. This is the case for known motion and no indeterminacies. When the camera motion is unknown and there is a scale indeterminacy the individual reconstructed features will have a qualitatively similar uncertainty, but in addition the motion will have an uncertainty which causes the feature uncertainties to be correlated. Line lengths in turn will be correlated, and parallel lines along the viewing direction can become quite strongly correlated, depending on the object size. When two lines are strongly correlated, equation (4.23) shows that fixing or measuring one will lead to an accurate estimate of the other.

We conclude from this set of experiments that when our object is large in width and height, compared to the set of lines we want to estimate, then irrespective of the orientation of the line we are estimating, it is best to measure lines orthogonal to the viewing direction. On the other hand, for objects with lines all clustered around a single point, lines in the plane orthogonal to the viewing direction are best predicted by lines also in this plane, but lines along the viewing direction are best predicted by other lines along the viewing direction.

[2] This is shown for the stereo case in Chapter 5

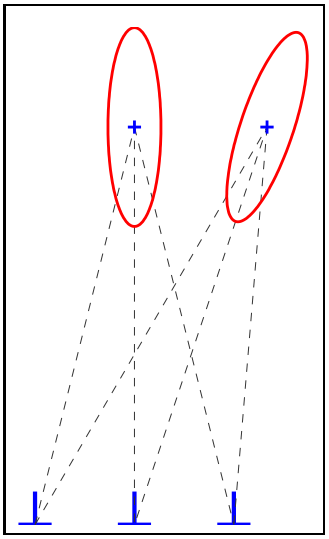


Fig. 4.22. Two features are estimated from a set of images with known camera positions and orientation. The uncertainty, illustrated with ellipses, has a characteristic form shown here with its long axis along the viewing direction; see Matthies and Shafer [46]. When there is no uncertainty in camera motion, and when multiple features are estimated, they are uncorrolated. Hence lines, which are the distance between two points, are estimated more accurately perpendicular to the viewing direction than along the viewing direction. When the camera motion is also estimated, feature points become correlated and this deduction for lines no longer holds. We show in this section how correlation affects line length estimation.

There are more factors that may affect these correlations. In Appendix B we investigate changing the number of images, changing the distance to the object and adding more features around the object. We find that varying the number of images taken of the object may increase or decrease the total accuracy but does not affect the uncertainty pattern. Similarly, adding more features right around the object increases the accuracy, but does not change the uncertainty pattern. Hence we expect our results to generalize over sequences with different number of images and different number of features.

We did find a change in pattern due to varying the distance to the object. Increasing the distance to the object tends to accentuate the difference between the uncertainties of the lines along the viewing direction and the lines orthogonal to the viewing direction, and decreasing the distance reduced the difference. This is not too surprising since the further the object the less accurately the camera motion can be determined and hence the more correlation we expect between line lengths.

A question remains on how to interpret the viewing direction when the object is not directly in front of the camera. Figure 4.23 illustrates an example with a sphere shifted so

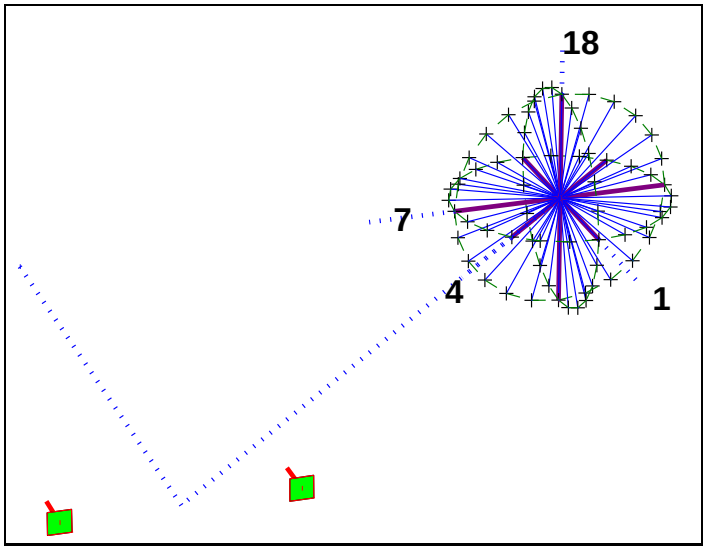


Fig. 4.23. Our sphere is shifted so that the new viewing direction is 45° from the old one. We are interested in predicting the diameters labeled 4, 7 and 18.

that it is at 45 degrees from the forward direction. The uncertainties in predicting three of the lines are shown in Figure 4.24. From this we see that if we define the viewing direction to be the direction from the center camera to the center of the sphere, then we obtain the same pattern of results as for the single sphere. That is, the lines orthogonal to this new direction are most accurately estimated and the line along it least accurately estimated. Since it is a small object the line along the viewing direction are most accurately predicted by other lines along the viewing direction. Hence, in general, we will define the *viewing direction* for any line on the object to be the direction from the center camera to the mid-point of the line.

In these examples we have only considered lines that pass through the same point. We cannot say what happens if lines are at different depths or along different viewing directions.

4.5.4 3D Multi-planar Shape Reconstruction

The next experiment we performed involved a 3D multi-planar shape containing three lines in five parallel planes that we wished to predict as shown in Figure 4.25. All lines have the same length in 3D, and uniform, uncorrelated Gaussian noise in the projected images is assumed. The parallel planes are at different distances from the cameras, and

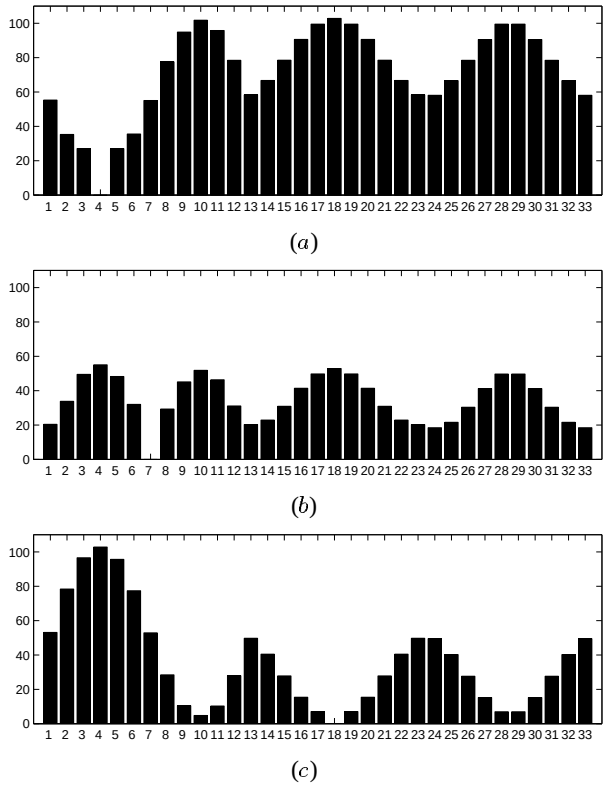


Fig. 4.24. Shown are the errors obtained when using each of the lines on the sphere (see Figure 4.23) are used to predict lines (a) 4, (b) 7, and (c) 18. In predicting line 4, the three lines with the minimum error (apart from line 4 itself) are the three other lines on the the same 45° cone. In predicting lines 7 and 18, we find that all lines on the plane orthogonal to the viewing direction are equally good, irrespective of direction.

the goal is to see how distance from the camera affects accuracy of reconstruction. In addition, the lines in each plane do not have mid-points directly in front of the cameras, and we would like to see if this affects the accuracy of reconstruction.

Figure 4.26 shows the results in predicting lines 1, 7 and 13 using each of the other lines. Since all line lengths are the same, from our symmetry relationship in Figure 4.1, we deduce that these uncertainties are the same as using lines 1, 7 and 13 respectively to predict each of the other lines.

We can make a number of general statements about the uncertainties resulting from fixing different lines. Firstly, our object is small, that is, all the lines are clustered around a single point. Thus we expect significant correlation between lines.

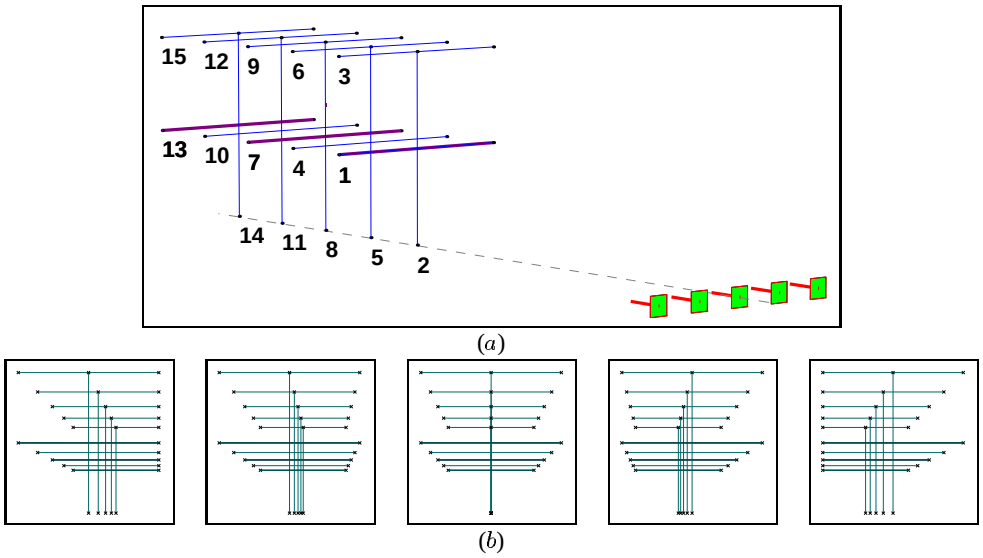


Fig. 4.25. Five views of 3D "∓" shape are shown with line numbers labeled. We are interested in predicting lines 1, 7 and 13.

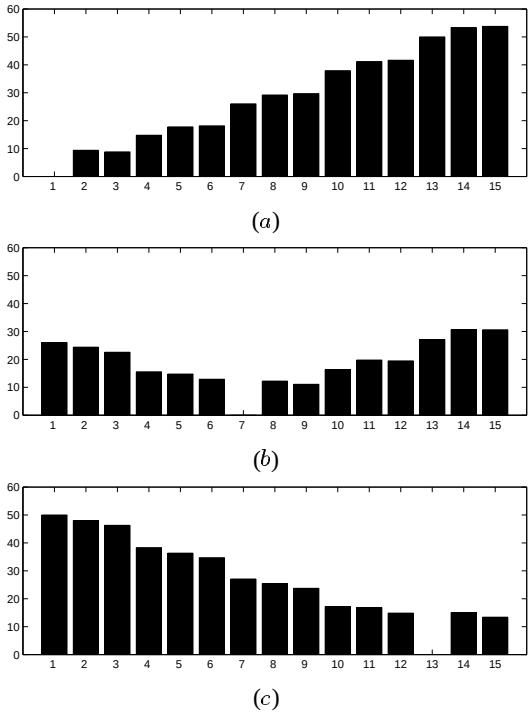


Fig. 4.26. The uncertainty of line prediction is shown in (a) for line 1, in (b) for line 7, and in (c) for line 13.

From the results we find that in general, to predict a line in one of the planes, it is best to fix another line in the same plane. Fixing lines close to the camera gives a poor estimate for lines far from the camera and vice versa. Figure 4.27 shows that if our goal is to minimize the total uncertainty of all the lines, then one should use a line in the middle of the object to fix the gauge.

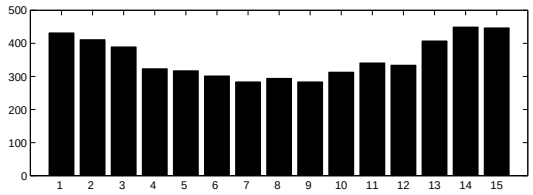


Fig. 4.27. The total error in using a line to predict all the other lines is shown for each line. This indicates that if overall accuracy is our goal, fixing a line near the center of the object is best.

To see how the results change when the object is large, we added a plane of random 3D points to the object as shown in Figure 4.28, and Figure 4.29 shows the accuracies in predicting lines 1 and 13. As in the small field-of-view case, lines close to the cameras are best predicted by fixing other lines close to the camera. But, reversing the small field-of-view pattern, lines far from the camera are slightly better predicted by fixing lines close to the camera. Thus we deduce that overall accuracy is maximized when a line closest to the camera is fixed, rather than a middle line as in the small field-of-view case. However, we see that depth now has a relatively small effect on accuracy. Rather, position in the plane seems to be a more dominant effect. In our next experiments we will investigate this.

4.5.5 Position in the Plane

Given that we have a set of lines in a plane orthogonal to the viewing direction, we would like to know which line to choose to fix the scale. In this experiment we created a synthetic object shown in Figure 4.30. It has two planes to give it a 3D shape, but we only looked at lines in the first plane. It takes up a wide field of view compared to the lines.

Figure 4.31 shows the resulting uncertainties when a selection of the lines are used to fix the scale. We notice first that irrespective of which line is used to fix the scale, the accuracy order of the lines stays the same (of course excluding the line that is fixed in

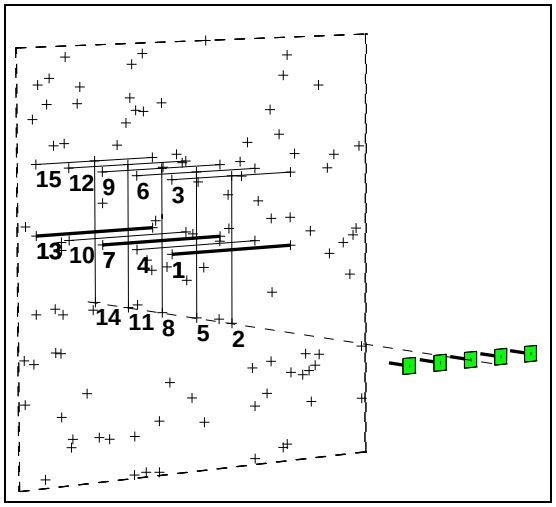


Fig. 4.28. Five views are taken of an enlarged shape in width and depth compared to Figure 4.25.

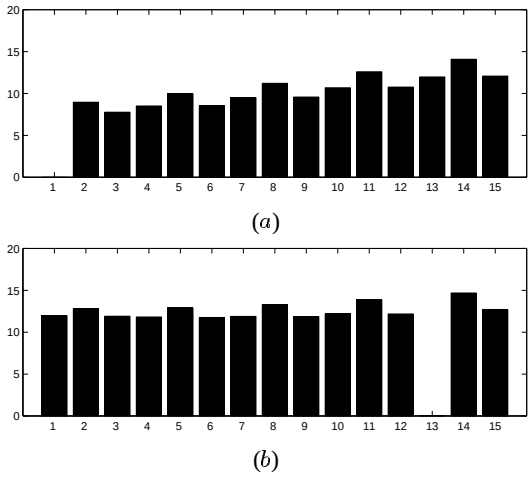


Fig. 4.29. The uncertainty of line prediction is shown in (a) for line 1, and in (b) for line 13.

each case). That is, lines 1, 4, 7, 10, 13 and 16 are always most accurately predicted, lines 2, 5, 8, 11, 14 and 17 are next, and lines 3, 6, 9, 12, 15 and 18 are least accurately predicted. This result applies in reverse too, that is, when one of the first set of lines is used to fix the scale, the results are more accurate than the results for the second set of lines which in turn are more accurate than the results for the third set of lines.

A first guess at a factor that could explain or at least predict which lines are better for gauge fixing, is the projected line length in the images. However, this cannot be the

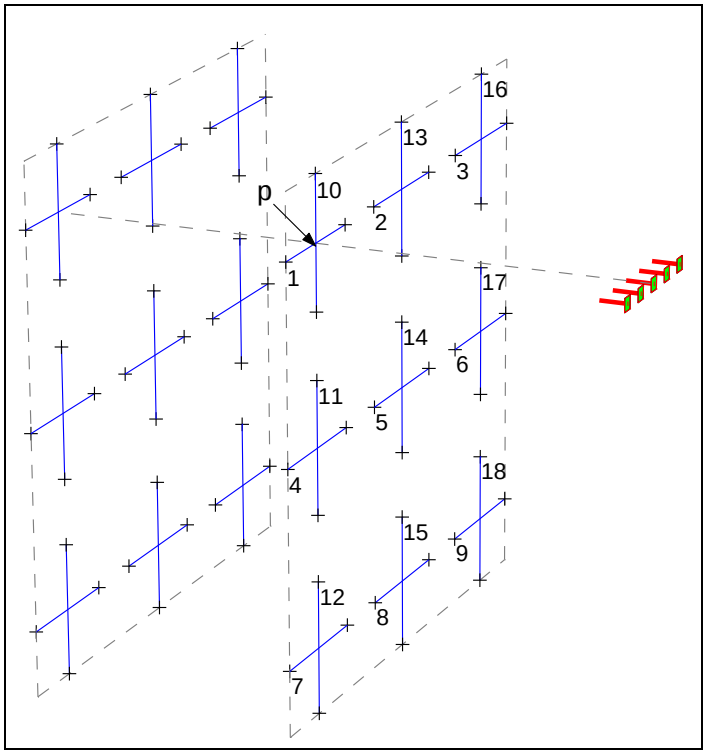


Fig. 4.30. Eighteen lines of equal length in a plane are labeled. Our goal is to discover which line or lines provide the best way to fix the scale.

case since all the lines in the plane have the same projected line length in all the images. This can be easily derived from the perspective camera equations. Thus we must look elsewhere for a predictive property for the line uncertainties.

The characteristic that is common among each set of lines, and that also distinguishes each set of lines, is the following. Let point p denote the point on the plane closest to the center camera, so that a line from it to the camera is orthogonal to the plane, as shown in Figure 4.30. For each line, l, we draw its perpendicular bisector in the plane. Then we measure the shortest distance from point p to line l and label it d_{lp}, as shown in Figure 4.32. Now all of the first set of lines have $d_{lp} = 0$, all of the second set of lines have a equal but larger d_{lp} and all of the third set of lines have an equal and even larger d_{lp}. We propose that the smaller d_{lp} is for a line, the more accurately it can be predicted, and the better it is as a tool for fixing the scale. Further evidence that uncertainty monotonically increases with d_{lp} is shown in Figure 4.33. So far we have not provided a geometric explanation for why d_{lp} of a line is indicative of its uncertainty, but even without an

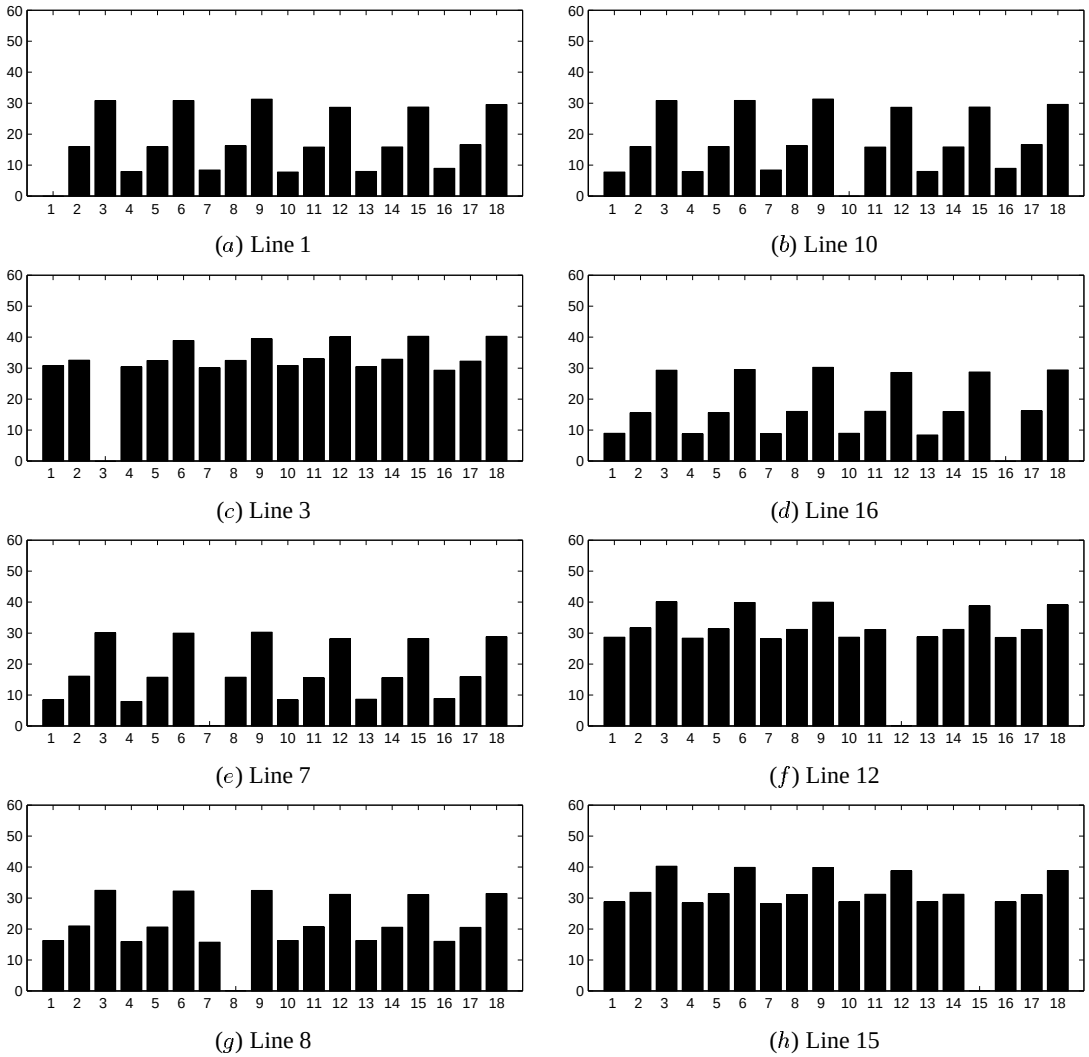


Fig. 4.31. The uncertainty of line prediction is shown for each of the labeled lines.

explanation this is a useful property for deciding which line in a scene to measure or where to place a measuring rod to maximize accuracy: namely the line's d_{lp} value should be as small as possible. Our next experiment will seek an explanation for this factor.

4.5.6 Viewing Directions and Perpendicular Lines

Consider the row consisting of lines 1, 2, 3, 10, 13, and 16 in Figure 4.30. We find that fixing any one of the lines 10, 13 or 16, generates equal accuracy in predicting other

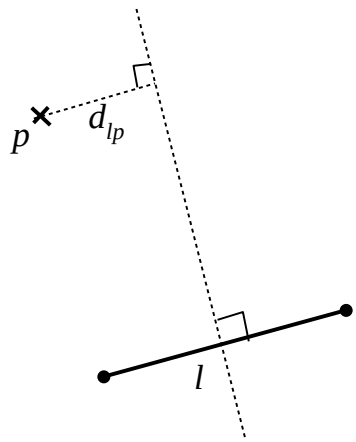
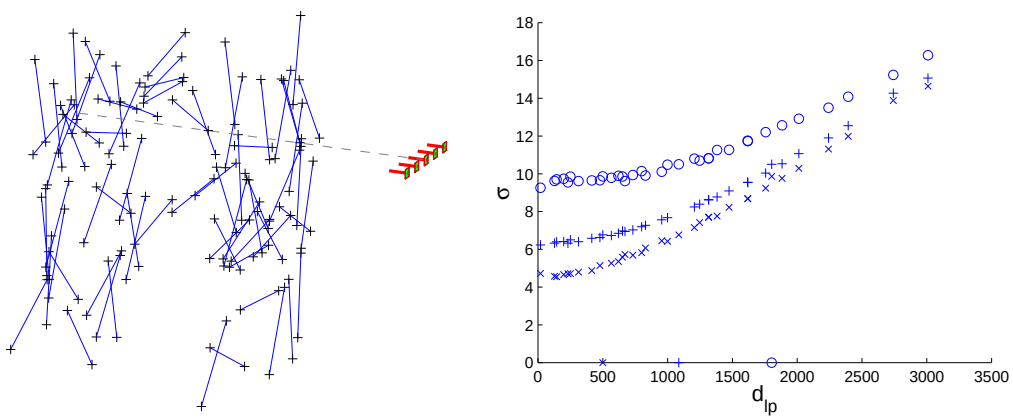


Fig. 4.32. The accuracy of predicting a line, or using it to predict other lines, depends on the distance, d_{lp}, from its perpendicular bisector to the point p, where the viewing direction line intersects the plane.



(a) (b)

Fig. 4.33. (a) To test our hypothesis that the length d_{lp} in Figure 4.32 is an indicator of how accurately a line can be predicted, we created a synthetic object with two planes containing a set of equal length lines. (b) In three separate experiments we fixed a line in the front plane and obtained an accuracy estimate for each of the lines. Plotted for each experiment is the uncertainty for each line against its d_{lp} value.

lines, but that fixing one of lines 1, 2 or 3, gives increasingly poor accuracy. This is modeled by our d_{lp} measure, but we would like to understand the cause of this effect from a geometric standpoint. Hence we created the following experiment.

We took a row of lines from the parallel planes example, shown in Figure 4.30, and, keeping their mid-points fixed, we swivelled the lines so that each one is orthogonal to its viewing direction. Thus we obtained the shape shown in Figure 4.34. The results of predicting lines 1 and 7 are shown in Figure 4.35. We see from this plot that lines that share a viewing direction, and are orthogonal to it and are the same distance to the cameras, have approximately the same uncertainty. We note that while this example has a large width compared to the line lengths, its height is small, and hence the line lengths are strongly correlated with each other.

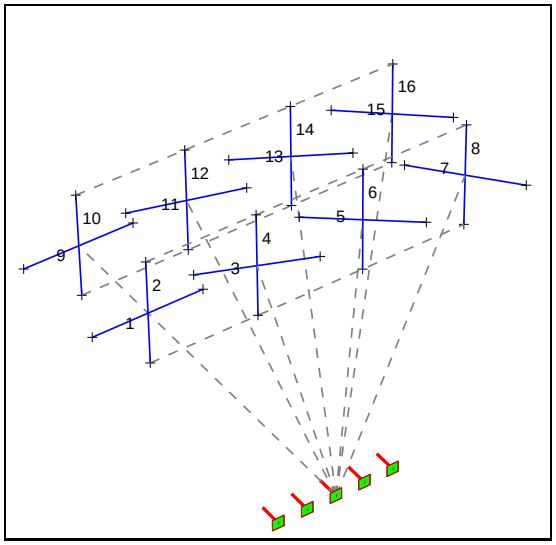


Fig. 4.34. Five views of a shape made from two planes. The mid-points of lines 1 through 8 are on one plane and the rest on the plane behind this. Lines 2, 4, 6 and 8 are fully on the front plane, and lines 1, 3, 5 and 7 are of the same length but oriented so that they are orthogonal to their viewing directions.

In order to remove this correlation, and so make our setup analogous to the previous parallel plane example, we added a set of random points to our shape and obtained the new object shown in Figure 4.36. The resulting uncertainties are shown in Figure 4.37. Since the object now has a large field of view compared to the lines, there is little correlation between lines, and all lines with mid-points on the front plane have the same uncertainty. The lines on the plane behind this have slightly larger uncertainty. This

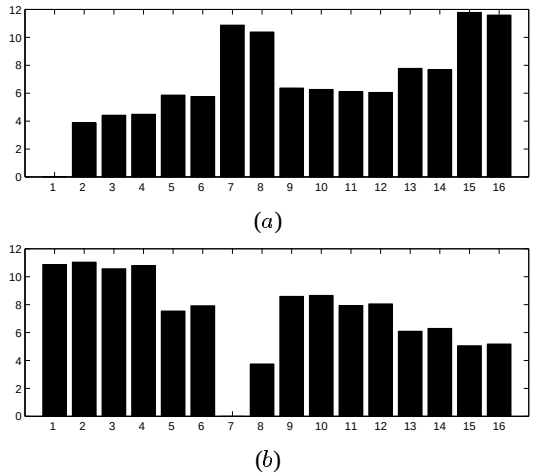


Fig. 4.35. The uncertainty of the line prediction is shown in (a) for line 1, and in (b) for line 7.

confirms our result in sections 4.5.4 and 4.5.2 that distance along the viewing direction is not such a significant factor for line accuracy, but it is rather line orientation with respect to the viewing direction plays a determining role. In this case all lines are orthogonal to their viewing direction, and hence they have very similar accuracy.

This leads us to our explanation for the significance of the d_{lp} measure for lines in a plane. Notice that lines in which d_{lp} is zero are orthogonal to their viewing direction. Furthermore the larger the value of d_{lp} the smaller the angle between the line and its viewing direction. Hence d_{lp} is really a measure of how orthogonal a line is to its viewing direction. We thus expect that if all of the lines in Figure 4.30 with non-zero d_{lp} are swivelled around their mid-points, either out of the plane or in the plane, so that they become orthogonal to the viewing direction, their uncertainties will be reduced to the same uncertainties as those lines having $d_{lp} = 0$.

4.5.7 Bridge Reconstruction

Our next experiment involved the reconstruction of a bridge between two buildings. Seven images were taken of it using a digital camera and features were registered in the sequence as shown in Figure 4.38. A 3D reconstruction was then performed and the result is in Figure 4.39.

Now our goal is to estimate the length of the bridge between the buildings. A line representing this is labeled 1 the figures. We were able to measure a number of lengths

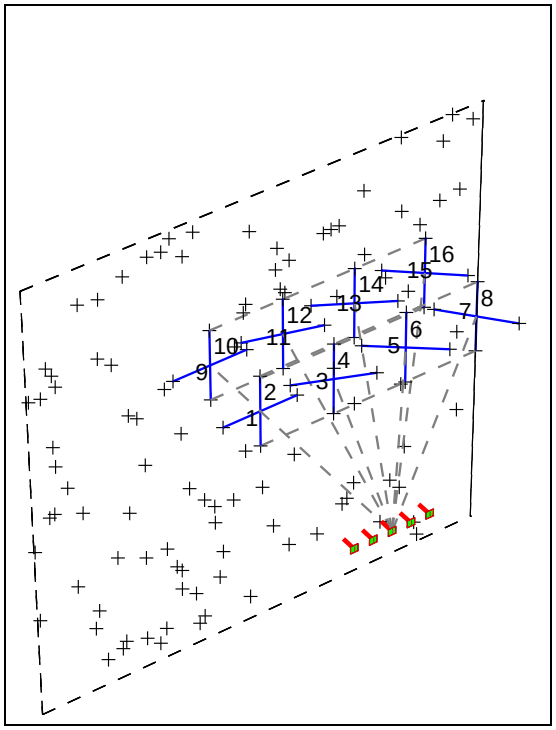
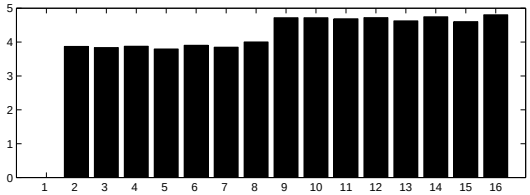
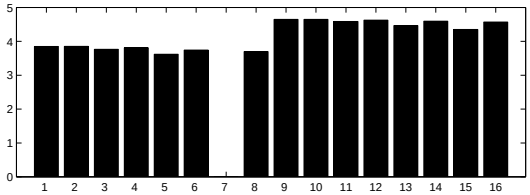


Fig. 4.36. Five views of a shape made from two planes. The mid-points of lines 1 through 8 are on one plane and the rest on the plane behind this. Lines 2, 4, 6 and 8 are fully on the front plane, and lines 1, 3, 5 and 7 are of the same length but oriented so that they are orthogonal to their viewing directions.



(a)



(b)

Fig. 4.37. The uncertainty of the line prediction is shown in (a) for line 1, and in (b) for line 7.

in the scene, and these are marked in the images. Table 4.1 gives their lengths and rough uncertainties.

In this example it is difficult to judge which of lines 2 through 5 would give the best accuracy for predicting line 1. None of the measured lines are in the plane of line 1, although they are all in parallel planes. The two most likely options are line 4 and line 3. Line 4 is by far the longest of these, but it is also significantly further from the cameras and has greater measurement uncertainty. Line 3 is the next longest, and while closest to the cameras, it is not as close to line 1 as lines 2 and 5.

The accuracy results in this case are given in Figure 4.40 for the case when we ignore localization error and when we include it. For the first case, plot (a) shows the standard deviation generated by gauge fixing. Here gauge fixing with line 4 gives the smallest uncertainty, and hence line length is the factor that dominates. However, notice that the standard deviations are too small to account for the large actual variations, with the true result being 4 or more standard deviations away from the predictions. This we believe is due to the localization error. If we assume an image-based localization error of 1 pixel standard deviation, then we obtain the uncertainties in plot (b). This results in a significant increase in standard deviations, and now line 3 has the smallest standard deviation, due to its proximity to the cameras.

Fig. 4.38. The two end-images from a seven image sequence of a bridge. Our goal is to estimate the length of the bridge, marked with line 1, and to do this we measured the lengths of lines 2 through 5.

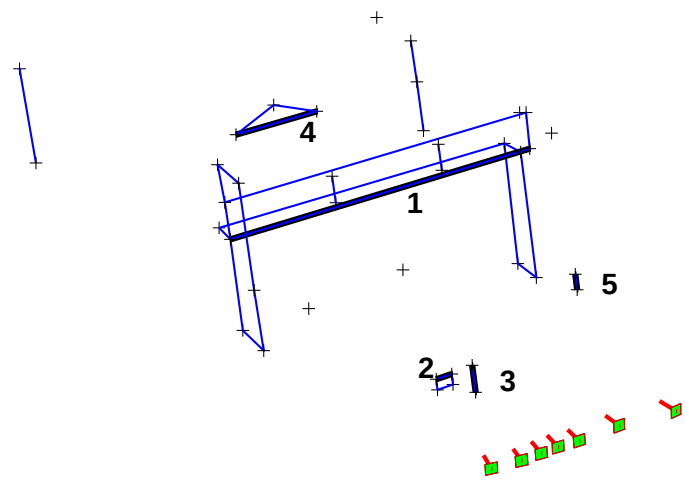


Fig. 4.39. The 3D reconstruction of the bridge is shown along with the camera positions.

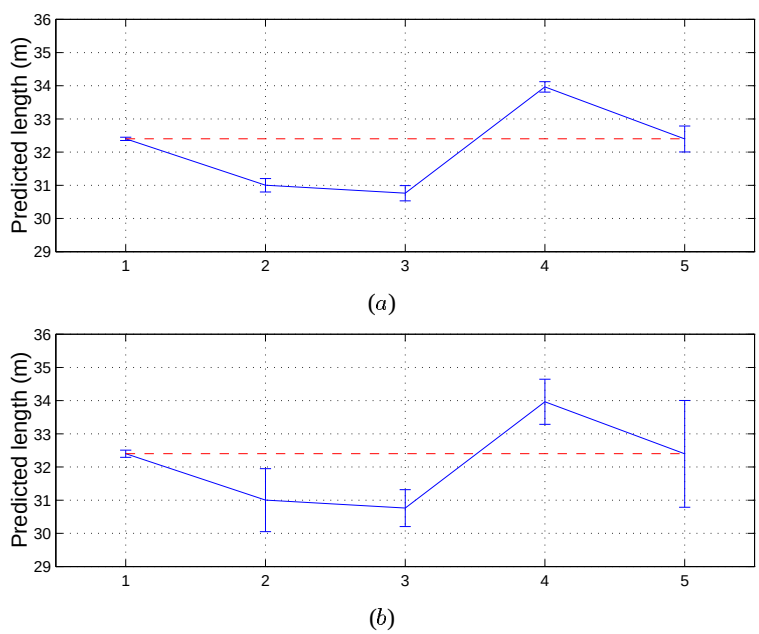


Fig. 4.40. The predicted bridge length is shown, along with its uncertainty in standard deviations for each of the five lines, calculated using equation (4.29). The uncertainties predicted in (a) used the measurement uncertainties from Table 4.1 but assumed the localization error σ_L was zero. In plot (b) we also included the uncertainty of localization error, assuming $\sigma_L = 1$ pixel for all features.

Table 4.1. The measured length of lines in the bridge sequence of Figure 4.38, and their uncertainties which were estimated while making the measurements.

Line No.	*Length (m)*	*Uncertainty (m)*
1	32.40	0.05
2	1.60	0.01
3	2.77	0.02
4	9.77	0.04
5	1.68	0.02

4.5.8 Desk Reconstruction

Our next experiment involved the reconstruction of a computer and desk. Seven images were taken of it using a digital camera and features were registered in the sequence as shown in Figure 4.41. A 3D reconstruction was then performed and the result is shown in Figure 4.42.

The goal for this sequence was to estimate the width of the computer monitor labeled as line 1 in the images. We measured a number of lengths in the scene so see how each of these would do in fixing the scale for predicting line 1. These lengths are given in Table 4.2. All were assumed to have the same uncertainty.

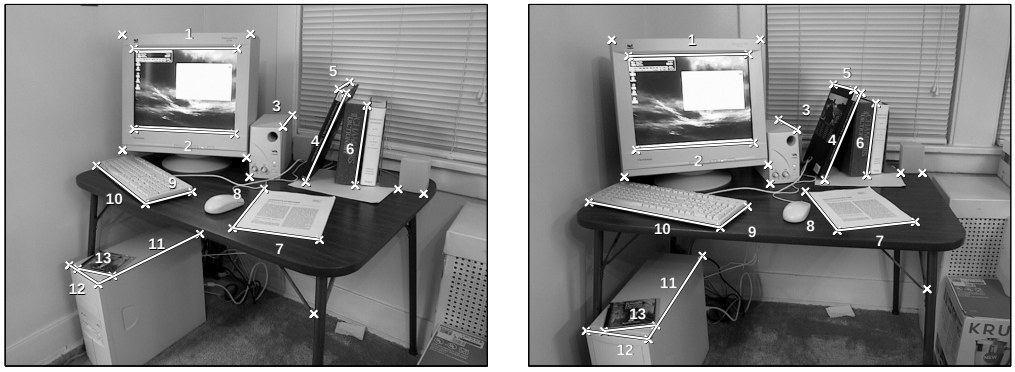


Fig. 4.41. The two end-images from a seven image sequence of a computer and desk. Our goal is to estimate the width of the monitor, marked with line 1, and to do this we measured the lengths of lines 2 through 13.

The most promising lines for optimizing the accuracy of predicting line 1 through gauge fixing are 2, 4 and 6 since they are all long and in the same plane as line 1. Next is line 7 in a parallel plane is also long. We expect 3 and 5 to be poor since they are

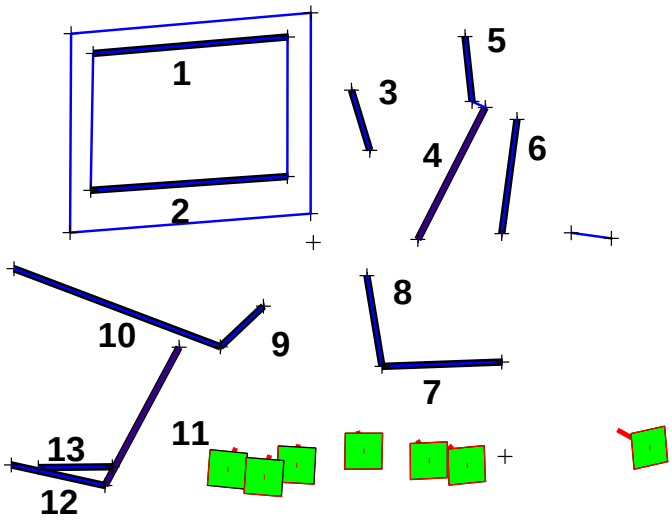


Fig. 4.42. The 3D reconstruction of the desk is shown along with the camera positions.

Table 4.2. The measured length of lines in the desk sequence of Figure 4.41, and their uncertainties. (We assumed all measurements had the same uncertainty).

Line No.	*Length (cm)*	*Uncertainty (cm)*	**Line No.**	*Length (cm)*	*Uncertainty (cm)*
1	36.51	0.05	8	27.94	0.05
2	36.51	0.05	9	15.56	0.05
3	21.11	0.05	10	45.72	0.05
4	31.27	0.05	11	46.99	0.05
5	23.65	0.05	12	19.05	0.05
6	25.24	0.05	13	14.21	0.05
7	21.59	0.05			

oriented along the viewing direction and 9, 12 and 13 also to be poor since they are short. In terms of localization error, the lines are all about the same distance from the cameras, and so the main distinguishing effects will be that the short lines, 9, 12 and 13, become even worse. Figure 4.43 sustains all of these predictions. We also see, as in the bridge example, that if the localization error is ignored, the standard deviations do not account for the variations in prediction, indicating a source of bias. When we include 1 pixel localization error, the variation in predictions agree quite well with the standard deviations.

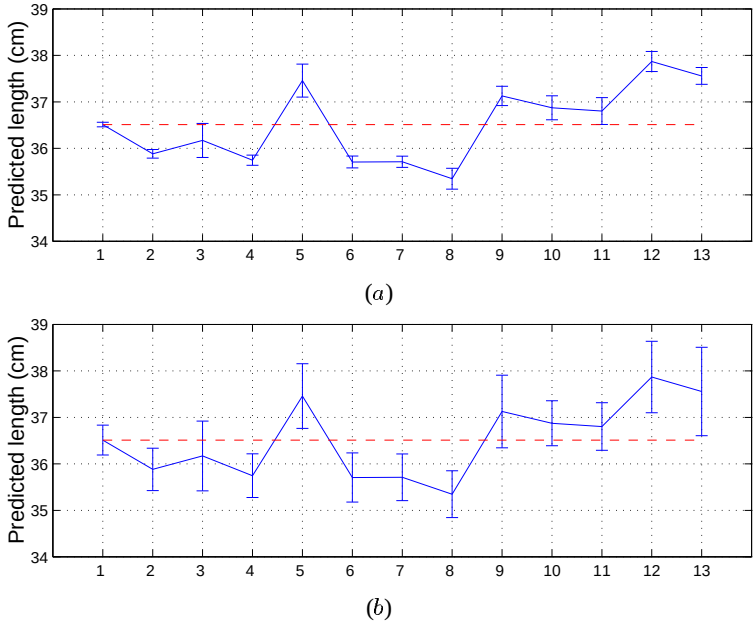


Fig. 4.43. The predicted monitor width is shown, along with its uncertainty in standard deviations for each of the thirteen lines, calculated using equation (4.29). The uncertainties predicted in (a) used the measurement uncertainties from Table 4.1 but assumed the localization error σ_L was zero. In plot (b) we included the uncertainty of localization error assuming $\sigma_L = 1$ pixel for all features.

Now imagine that instead of trying to estimate the monitor width, we wanted to know the length of the sub-woofer, indicated as line 3 in the image. The key question here is whether it would be better to know line 5, which is parallel to line 3, or else one of the lines orthogonal to the viewing direction. Since the desk has a large field of view in both height and width compared to the line lengths, we expect that the lines are relatively uncorrelated and hence that lines orthogonal to the viewing direction would be better than line 5. This is confirmed in Figure 4.44.

But let us consider another situation in which the only features that we tracked are those belonging to lines 2, 3, 4, 5 and 6. We can do shape estimation with just this data, and then estimate the sub-woofer length by measuring one of the other lengths. With points from just these lines, the object has a relatively small field of view compared to the line lengths. Hence we might expect that measuring line 5 would give us the greatest accuracy. We see in Figure 4.45 that this is indeed the case.

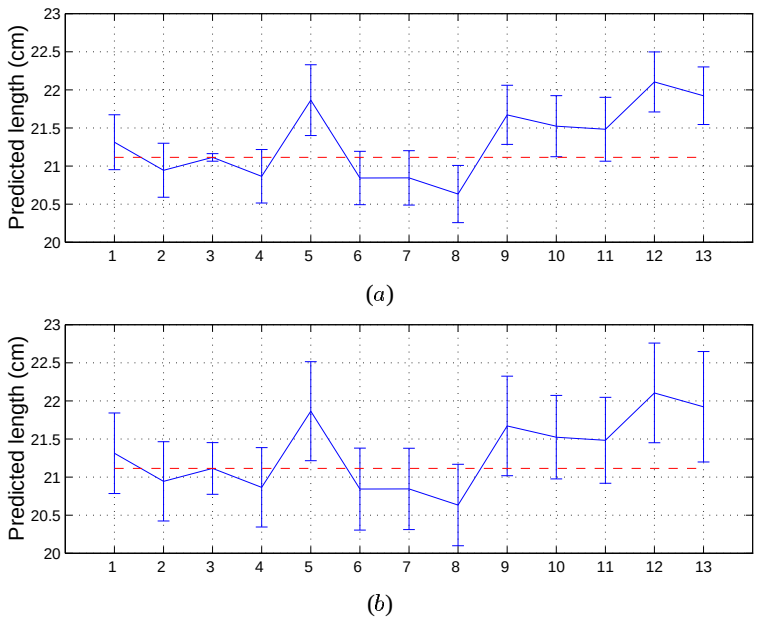


Fig. 4.44. The predicted length of the sub-woofer is shown, along with its uncertainty in standard deviations for each of the thirteen lines, calculated using equation (4.29). The uncertainties predicted in (a) used the measurement uncertainties from Table 4.1 but assumed the localization error σ_L was zero. In plot (b) we included the uncertainty of localization error assuming $\sigma_L = 1$ pixel for all features.

4.6 Discussion

We have derived and demonstrated an unexpected consequence of fixing the scale of an object known only up to a scale factor. The covariance of the resulting shape is significantly affected by how the scale is determined. We assume the scale is determined by the distance between two points on the object. Simply rescaling the covariance matrix with the square of the scale factor leaves the indeterminacy in the matrix and does not account for the measurement. Instead, the covariance must be transformed so that the model uncertainty between the measured points is correctly distributed over the rest of the points. We used gauge theory to derive the covariance transformation, when the scale is fixed by making a measurement, and showed how choosing different gauges will affect the final accuracy.

Our focus was to develop a qualitative understanding of the effects of gauge fixing on accuracy. This is important for use in real 3D modeling situations where one may not have time or opportunity to measure many lengths in the scene. With a good qualita-

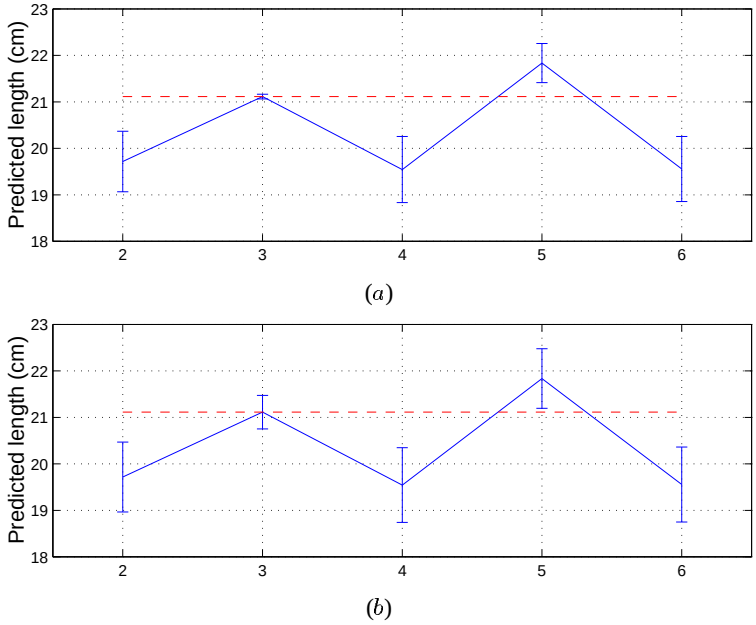


Fig. 4.45. The predicted length of the sub-woofer is shown, along with its uncertainty in standard deviations for just lines 2, 3, 4, 5 and 6, calculated using equation (4.29). The uncertainties predicted in (a) used the measurement uncertainties from Table 4.1 but assumed the localization error σ_L was zero. In plot (b) we included the uncertainty of localization error assuming $\sigma_L = 1$ pixel for all features.

tive understanding of the accuracy effects, a close-to-optimal length can be selected for measurement *before* 3D reconstruction is done.

A qualitative analysis of our results leads to a number of general conclusions.

- When there is little or no correlation between feature points and lengths, accuracy is best when the gauge is fixed by measuring the largest distance on the object.
- The effect of measurement error on the 3D lengths is reduced for longer lengths.
- When there is significant positive correlation between lengths in the model, then measuring a longer line is not necessarily better. Instead, the closer the ratio, of measured to estimated length, is to the ratio in equation (4.24), the better the final accuracy.

We also looked at correlation effects, and factored out line length effects by using lines of all the same length. We considered cases in which there was a predominant viewing direction, and found a number of factors affecting uncertainties and correlations of lines. We divided reconstruction into two classes. The first class consists of objects that subtend a large field of view compared to the line lengths, and the second class

consists of objects in which the lines are clustered around each other and hence the object subtends a relatively small field of view compared to line lengths.

Objects subtending a large field of view compared to the line lengths resulted in accurate motion estimation and hence low correlation between line lengths. In choosing a gauge to fix we found the following effects:

- The property that dominates accuracy is the angle between the line and its viewing direction. Lines that are orthogonal to their viewing direction can be better estimated and also provide better gauge constraints than lines at smaller angles to the viewing direction. Hence even for lines along the viewing direction, the best accuracy is obtained by measuring lines orthogonal to the viewing direction.
- Accuracy is less sensitive to depth in objects with a large field of view than for objects with small field of view.
- Given a set of lines in a plane, we created a measure, d_{lp}, that can be used to predict the relative accuracy of the lines. This value is effective in indicating relative accuracy because it is related to the relative orientation of the line and its viewing direction.

Objects consisting of a cluster of lines whose total field of view is not much larger than that subtended by line have strong correlation in their line lengths. This leads to a number of properties.

- The angle between the viewing direction and the line is key, and unlike the wide field-of-view case, lines with similar angles are more strongly correlated than lines with differing angles, and so provide better constraints. This means that lines along the viewing direction are best predicted by other lines along the viewing direction.
- The distance of the lines from the camera plays an important role. Lines of similar distance tend to be more strongly correlated and hence provide better gauge constraints for each other.
- If overall accuracy is desired, the best option is a line near the middle of the scene and in the plane perpendicular to the viewing direction.

All of the results for line lengths are invariant to choice of coordinate origin and orientation.

More questions remain in uncertainty understanding. Given a 3D scene, we would like to be able to select the best camera views for reconstructing it. If our views are constrained to be along a single direction, then our results may apply directly. But if camera direction can vary, then it remains an open question for how one should best position and orient the camera.

In addition to characterizing uncertainties due to object shape, we also proposed that a significant source of error in 3D estimation comes from what we termed localization error. We created a model for it and included it in our accuracy estimates. This resulted in a larger covariance for our line estimates, and so better predicted the real uncertainties of our results. Our estimate of one pixel standard deviation for image localization error was a rough approximation and we would like to obtain a more justified estimate of this. We showed that localization error affects further points from the camera more strongly, and that shorter line lengths amplify this effect more. Moreover, while the effect of tracking error decreases as more images are taken, this is not the case with localization error which remains constant despite tracking features over more images. Hence we expect localization error to have a greater relative effect the more images are taken.

5. Stereo with Gauge Freedoms

Stereo vision is a well-studied field with numerous algorithms in use. Real time and video-frame-rate depth maps can be generated on a standard PC. Moreover, stereo often generates more accurate and denser depth maps than are achieved by SFM. This is largely due to algorithms taking advantage of the fixed epipolar geometry to simplify and improve feature matching and depth estimation.

However, using just the epipolar geometry and feature matches, depth can only be estimated up to a scale factor. In order to obtain absolute depth maps, the scale must somehow be determined. Typically this is achieved by knowledge of the baseline length which itself is obtained through a calibration process. In some cases, however, it may not be easy to obtain this metric information. When calibration is done using self-calibration routines, as is often the case in active-vision applications, there may be no metric information, and so while the epipolar geometry can be obtained, the baseline length cannot be estimated.

The question we ask in this chapter is: Is it always best to have the baseline length, or could we achieve as accurate or even more accurate 3D estimation by using metric information in the scene? Another way to ask this is whether or not the baseline provides the best gauge constraint for determining the scale. The alternative is to use some measurement in the scene as was done in chapter 4. We will address this question for two cases: when the epipolar geometry is known, and when it is unknown.

5.1 Calibration Assumptions

It is important to first identify our camera calibration assumptions. In general we will work with a pin-hole perspective model as given by equation (3.4). The options we are faced with depend on what we know about the internal camera parameters, equation (3.6), and the extrinsic parameters, namely the relative position and rotation of the cameras.

We will identify a range of possible calibration "modes" for stereo, based on the prior parameter knowledge. One classification division is between projective stereo methods and Euclidean stereo methods. The other division is between known epipolar geometry and unknown epipolar geometry. This leaves us with four classes of algorithms, listed as follows:

- *Projective stereo without epipolar geometry*. This is the most general situation with no assumed knowledge of intrinsic or extrinsic camera parameters. Projective shape can be obtained. It is also possible to use self-calibration to estimate the epipolar geometry.
- *Projective stereo with known epipolar geometry*. The fundamental matrix defines the epipolar geometry of the camera pair, and enables more efficient feature registration. The projective shape of the feature points can be estimated.
- *Euclidean stereo without epipolar geometry*. The intrinsic camera parameters are assumed to be pre-calibrated,[1] but the relative position and orientation of the cameras is unknown. This is analagous to the two-frame SFM case. Shape up to a scale factor can be estimated. It is also possible to estimate the epipolar geometry.
- *Euclidean stereo with known epipolar geometry*. The essential matrix defines the epipolar geometry of the camera pair. Feature depths, and hence 3D positions, are obtained from disparities using a triangulation method [25]. Without knowledge of the baseline, 3D points can be estimated only up to a scale factor, but with a known baseline the absolute positions can be determined.

In this thesis we are interested in uncertainties of physical quantities, and hence we will only consider the latter two classes describing Euclidean stereo[2]. These will be denoted as stereo with known epipolar geometry or known essential matrix E, and stereo with unknown epipolar geometry or unknown E. With these we can obtain shape and depth up to a scale factor, and if we fix the scale somehow, we can obtain absolute distance measurements and their uncertainties. Stereo algorithms typically use the length of the baseline, between the centers of projection of the two cameras, to fix the scale. This is difficult to measure physically, and so calibration methods using objects of known dimensions are employed to estimate both the baseline and the epipolar geometry.

The type of calibration in which the baseline is found typically requires careful manual design and measurement of a calibration object, as well as an interactive process

[1] The focal lengths need not be known as these can be solved for [38].

[2] While it is possible to estimate up to two internal camera parameters from the epipolar equations, for simplicity we only consider cases in which we already know all the internal parameters.

in which this object is viewed by the stereo pair from possibly multiple directions and correspondances are identified. From these correspondances and the known calibration object geometry, both the epipolar geometry and baseline are estimated. The question we ask in this case is, given perfect knowledge of the epipolar geometry, is scale best determined by knowldege of the baseline or by knowledge of some length in the scene?

While pre-calibrating a stereo pair is useful if it will remain firmly fixed in place, it does not help for applications in which the cameras may change relative orientation between calibration in the laboratory and actual use. This includes, for example, active vision applications where tracking an object may entail zooming or relative swivelling of the cameras as the object moves towards the stereo pair, just as our eyes start to cross when we focus on a nearby object. In these cases, an alternative approach that is more automated and closer to self calibration is preferable. Self-calibration uses only point matches in the images to estimate the epipolar geometry. The baseline, however, generally cannot be attained this way since, while an automated system may find point correspondances using robust methods, it will not know lengths in the scene. It thus remains for a later stage, when the 3D data is analyzed, to impose metric constraints. We ask a similar question in this case. While measuring the baseline may be physically difficult in this case, nevertheless ought we to measure it to maximize accuracy, or could we achieve equal or better accuracy by measuring a line in the scene to determine scale?

5.2 Epipolar Geometry

The effects of knowing internal and external parameters can be brought out more fully by defining the epipolar geometry. Initially we will work in homogeneous coordinates. Let $\boldsymbol{X}$ be a 4-vector specifying the 3D position of a feature point, and $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ be 3-vectors giving the image positions of the feature in each image. Homogeneous perspective camera projection gives us:

$$\begin{aligned} \boldsymbol{x}_1 &= K_1 \left(\boldsymbol{R}_1 \mid \boldsymbol{t}_1 \right) \boldsymbol{X} \\ \boldsymbol{x}_2 &= K_2 \left(\boldsymbol{R}_2 \mid \boldsymbol{t}_2 \right) \boldsymbol{X} \, . \end{aligned} \tag{5.1}$$

In the remaining portion of this chapter we will consider only distances relative to the stereo head, and are free to choose an arbitrary gauge for rotation and translation. We let $\boldsymbol{R}_1 = \boldsymbol{I}$ and $\boldsymbol{t}_1 = \boldsymbol{0}$ be our motion constraints. Then we let $\boldsymbol{R} = \boldsymbol{R}_2$ and $\boldsymbol{t} = \boldsymbol{t}_2$, and can rewrite our equations as:

$$\boldsymbol{x}_1 = K_1 \left(\boldsymbol{I} \mid \boldsymbol{0} \right) \boldsymbol{X}$$
$$\boldsymbol{x}_2 = K_2 \left(\boldsymbol{R} \mid \boldsymbol{t} \right) \boldsymbol{X} \,. \tag{5.2}$$

From these equations, it is standard [27] to derive the following epipolar relationship:

$$\boldsymbol{x}_2^\top F \boldsymbol{x}_1 = 0 \,, \tag{5.3}$$

where F is the 3×3 fundamental matrix. Now F can be written as a function of the motion and calibration parameters: $\boldsymbol{R}$, $\boldsymbol{t}$, K_1, K_2 as follows:

$$F = K_2^{-\top} \boldsymbol{t} \times \boldsymbol{R} K_1^{-1} \,. \tag{5.4}$$

where the tensor product is defined in section 3.2.1.

When the intrinsic camera parameters, K_1 and K_2, are known, we can work in normalized feature coordinates, $\hat{\boldsymbol{x}}_1 = \begin{pmatrix} x_1 & y_1 & 1 \end{pmatrix}^\top$ and $\hat{\boldsymbol{x}}_2 = \begin{pmatrix} x_2 & y_2 & 1 \end{pmatrix}^\top$, given by: $\hat{\boldsymbol{x}}_1 = K_1^{-1} \boldsymbol{x}_1$ and $\hat{\boldsymbol{x}}_2 = K_2^{-1} \boldsymbol{x}_2$. The epipolar geometry is then defined by the equation

$$\hat{\boldsymbol{x}}_2^\top E \hat{\boldsymbol{x}}_1 = 0 \,, \tag{5.5}$$

where E is the essential matrix. This calibrated version of the fundamental matrix equation (5.3) was derived by Longet-Higgens in 1981 [45]. The essential matrix is related to the fundamental matrix via the internal camera parameter matrices:

$$E = K_2^\top F K_1 \,, \tag{5.6}$$

and so must be given by:

$$E = \boldsymbol{t} \times \boldsymbol{R} \,. \tag{5.7}$$

While the essential matrix is a 3×3 matrix with nine parameters, it has only five degrees of freedom. As we can see from equation (5.7), it encodes the relative rotation between the cameras and the relative translation. However, from equation (5.5) we see that it actually constrains points only up to a scale factor, and hence only defines translation up to a scale factor.

When these equations are applied to multiple points, we can put them into the same format and notation as previous chapters. We write the vector equation encoding the feature projections as:

$$\boldsymbol{p} = \boldsymbol{\Pi}_s[\boldsymbol{\theta}] \,. \tag{5.8}$$

Here $\boldsymbol{p}$ is a vector containing the individual (x, y) coordinates of all the features in both images, $\boldsymbol{\Pi}_s[\cdot]$ encodes the camera projection equations (5.2) for all features in both images and $\boldsymbol{\theta}$ is a vector of the unknown parameters. In the unknown E case, the vector

$\boldsymbol{\theta}$ contains the unknown 3D feature points, $\boldsymbol{s}_\alpha$, $\alpha = 1, \ldots, N$, and the essential matrix, E, or equivalently, the rotation, $\boldsymbol{R}$, and translation $\boldsymbol{t}$. In the known E case, the 3D feature points are included, but the essential matrix is known and so is not included, while the baseline, b, which may be unknown is included in $\boldsymbol{\theta}$.

The result is that we have the same set of equations as we had for SFM in Chapter 3, except that now some of the previously unknown parameters are known. We would like to know how uncertainties are affected by gauge fixing in this more restricted domain.

5.3 Stereo with Known Epipolar Geometry

When the essential matrix is known, the only unknowns are the baseline and the 3D feature point positions, and these can be estimated up to a scale factor. The question we want to address is what is the best way to fix the scale? This will be done as in Chapter 4 by measuring a length. But is it best to measure the baseline or to measure a length on the object? We will use the accuracy of shape resonstruction as a measure in comparing these options and determining which is best.

5.3.1 Parametrization

We start with our known essential matrix, E, or equivalently the rotation, $\boldsymbol{R}$, and unit translation, $\boldsymbol{t}$, where $\|\boldsymbol{t}\| = 1$, since these can be obtained from E as explained in [27, 45]. These parameters are included in our projection equations, and the remaining unknown parameters for N features and baseline are encoded as:

$$\boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{s}_1 \\ \vdots \\ \boldsymbol{s}_N \\ b \end{pmatrix}, \tag{5.9}$$

where $b = \|\boldsymbol{t}\|$ is the baseline.

5.3.2 Optimal Estimate and Accuracy

To solve for these parameters we define a cost as in equation (2.8):

$$J(\boldsymbol{\theta}) = \|\boldsymbol{p} - \boldsymbol{\Pi}_s[\boldsymbol{\theta}]\|^2_{\boldsymbol{V}_{\boldsymbol{p}}} . \tag{5.10}$$

In this chapter we will assume that, while 2D feature measurements may have their individual covariances, all of the feature measurements are independent. Thus $\boldsymbol{V}_{\boldsymbol{p}}$ can

be encoded as a 2×2 block diagonal matrix. Minimizing the cost in equation (5.10) will give us an optimal estimate, $\boldsymbol{\theta}$, which will lie on a gauge orbit determined by the scale factor.

The accuracy of our estimate depends on the covariance. To first order, the Fisher information matrix is equal to the Hessian at the optimal solution point, as derived in equation (2.19):

$$\boldsymbol{J}_{\boldsymbol{\theta}} = \nabla^2_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \,, \tag{5.11}$$

and the normal covariance is obtained as the Moore-Penrose generalized inverse of this:

$$\boldsymbol{V}_{\perp\mathcal{G}} = (\boldsymbol{J}_{\boldsymbol{\theta}})^{-}_{3N} \,. \tag{5.12}$$

5.3.3 Accuracy within Gauges

The covariance within a gauge, $\mathcal{C}$, is given by the oblique projection of the normal covariance into the tangent space to the gauge:

$$\boldsymbol{V}_{\mathcal{C}} = \boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}} \boldsymbol{V}_{\perp\mathcal{G}} \boldsymbol{Q}^{\mathcal{C}\,\top}_{\boldsymbol{\theta}} \,, \tag{5.13}$$

where we have assumed that our estimate $\boldsymbol{\theta}$ has already been transformed to the correct scale, $\boldsymbol{\theta} \leftarrow g\boldsymbol{\theta}$, so that the Jacobian matrix is the identity. What remains to calculate is the projection operator, $\boldsymbol{Q}^{\mathcal{C}}_{\boldsymbol{\theta}} = \boldsymbol{I} - U_{\boldsymbol{\theta}_{\mathcal{C}}}(V^{\top}_{\boldsymbol{\theta}_{\mathcal{C}}} U_{\boldsymbol{\theta}_{\mathcal{C}}})^{-1} V^{\top}_{\boldsymbol{\theta}_{\mathcal{C}}}$, which depends on the gauge freedom matrix $U_{\boldsymbol{\theta}_{\mathcal{C}}}$ and the constraint gradient matrix $V_{\boldsymbol{\theta}_{\mathcal{C}}}$.

Since we only need to consider one gauge freedom, that of scale, let $\boldsymbol{u} = U_{\boldsymbol{\theta}_{\mathcal{C}}}$ be a vector encoding this gauge freedom, and $\boldsymbol{v} = V_{\boldsymbol{\theta}_{\mathcal{C}}}$ be a vector encoding the constraints. In an analogous manner to that in Chapter 4, we derive the gauge freedom as:

$$\boldsymbol{u} = \begin{pmatrix} \boldsymbol{s}_1 \\ \vdots \\ \boldsymbol{s}_N \\ b \end{pmatrix} . \tag{5.14}$$

We will consider two types of gauges or constraints. Let $\mathcal{C}^{ij}$ be the gauge defined when we find that the distance between two points, $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$, is d_{ij}. We write the constraint as $||\boldsymbol{s}_i - \boldsymbol{s}_j|| - d_{ij} = 0$. Let $\mathcal{C}^b$ be the gauge defined when we measure the baseline as d. We write this constraint as: $b - d = 0$. Let the gradients of these be denoted as $\boldsymbol{v}_{ij}$ and $\boldsymbol{v}_b$ respectively. They are obtained explicitly as:

$$\boldsymbol{v}_{ij} = \begin{pmatrix} 0 \\ \vdots \\ (\boldsymbol{s}_i - \boldsymbol{s}_j)/d_{ij} \\ 0 \\ \vdots \\ (\boldsymbol{s}_j - \boldsymbol{s}_i)/d_{ij} \\ 0 \\ \vdots \end{pmatrix}, \qquad \boldsymbol{v}_b = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}, \tag{5.15}$$

where only the rows corresponding to the ith and jth points are non-zero in $\boldsymbol{v}_{ij}$. Using the appropriate constraint vector, we can calculate the projection operator, $\boldsymbol{Q}_{\boldsymbol{\theta}}^{\mathcal{C}}$, and hence the covariance in either of these gauges.

We would like to compare the covariances in between gauges $\mathcal{C}^{ij}$ and $\mathcal{C}^{b}$ to see which leads to more accurate estimates. To do this we will examine component-wise the structure of the covariances. Our first observation is that the Fisher information matrix has the following sparse structure:

$$\boldsymbol{J}_{\boldsymbol{\theta}} = \left(\begin{array}{ccc|c} \nabla^2_{\boldsymbol{s}_1} J & & 0 & \nabla_{\boldsymbol{s}_1 b} J \\ & \ddots & & \vdots \\ 0 & & \nabla^2_{\boldsymbol{s}_N} J & \nabla_{\boldsymbol{s}_N b} J \\ \hline \nabla_{b\boldsymbol{s}_1} J & \cdots & \nabla_{b\boldsymbol{s}_N} J & \nabla^2_b J \end{array}\right). \tag{5.16}$$

It is 3×3 block diagonal except for the last row and column corresponding to the baseline. The normal covariance given by the generalized inverse of $\boldsymbol{J}_{\boldsymbol{\theta}}$ will not, however, be sparse. The correlation terms between the baseline and the shape points (in the last row and column) conspire to fill the elements of the normal covariance. Nevertheless there is structure hidden below the surface. We see this if we calculate the covariance in the gauge obtained by fixing the baseline:

$$\boldsymbol{V}_b = \boldsymbol{Q}_{\boldsymbol{\theta}}^{b} \boldsymbol{V}_{\perp\mathcal{G}} {\boldsymbol{Q}_{\boldsymbol{\theta}}^{b}}^{\top}, \tag{5.17}$$

where $\boldsymbol{Q}_{\boldsymbol{\theta}}^{b} = \boldsymbol{I} - \boldsymbol{u}\boldsymbol{v}_b^{\top}/\boldsymbol{v}_b^{\top}\boldsymbol{u}$. This is exactly the same as solving for all the shape components with a known baseline. Its Fisher information matrix will be the same as $\boldsymbol{J}_{\boldsymbol{\theta}}$ in equation (5.16) except with the last row and column zeroed. Hence the covariance, formed by inverting all but the last column and row of $\boldsymbol{J}_{\boldsymbol{\theta}}$ must have the 3×3 block diagonal form:

$$\boldsymbol{V}_b = \left(\begin{array}{ccc|c} \boldsymbol{V}_{\boldsymbol{s}_1} & & 0 & 0 \\ & \ddots & & \vdots \\ 0 & & \boldsymbol{V}_{\boldsymbol{s}_N} & 0 \\ \hline 0 & \cdots & 0 & 0 \end{array}\right) , \tag{5.18}$$

Where the 3×3 blocks are given by: $\boldsymbol{V}_{\boldsymbol{s}_i} = (\nabla^2_{\boldsymbol{s}_i} J)^{-1}$, and correspond to the individual feature covariances. We see that since the baseline is known, all elements containing its correlation terms are zero, and moreover, unlike in the normal covariance, all of the individual shape covariances are now uncorrelated with each other.

Since $\boldsymbol{V}_b$ can be related to the normal covariance, $\boldsymbol{V}_{\perp\mathcal{G}}$, by equation (5.17), it must be geometrically equivalent to $\boldsymbol{V}_{\perp\mathcal{G}}$. Hence we can obtain the covariance in any other gauge by obliquely projecting $\boldsymbol{V}_b$, instead of $\boldsymbol{V}_{\perp\mathcal{G}}$, as in equation (5.13). Let $\boldsymbol{V}_{d_{ij}}$ be the covariance in the gauge defined by measuring distance d_{ij}. Again assuming an identity Jacobian matrix, it can be obtained as:

$$\boldsymbol{V}_{d_{ij}} = \boldsymbol{Q}_{\boldsymbol{\theta}}^{d_{ij}} \boldsymbol{V}_b \boldsymbol{Q}_{\boldsymbol{\theta}}^{d_{ij}\top} . \tag{5.19}$$

We can expand this in components of $\boldsymbol{Q}_{\boldsymbol{\theta}}^{d_{ij}}$ as follows:

$$\boldsymbol{V}_{d_{ij}} = \boldsymbol{V}_b - \frac{1}{b}\boldsymbol{u}\boldsymbol{v}_{ij}^\top \boldsymbol{V}_b - \frac{1}{b}\boldsymbol{V}_b \boldsymbol{v}_{ij}\boldsymbol{u}^\top + \frac{1}{b^2}\boldsymbol{v}_{ij}^\top \boldsymbol{V}_b \boldsymbol{v}_{ij}\boldsymbol{u}\boldsymbol{u}^\top . \tag{5.20}$$

The second term on the right is a zero matrix except for the ith and jth columns. The third term is the transpose of this and so is zero except the ith and jth rows. The final term is just a constant times the positive semi-definite matrix $\boldsymbol{u}\boldsymbol{u}^\top$.

Say our goal is to accurately estimate length $d_{kl} = \|\boldsymbol{s}_k - \boldsymbol{s}_l\|$. Can we say in which gauge it would be best to work? Let ${\sigma_{kl}^b}^2$ be the variance of this in gauge $\mathcal{C}^b$ with measured baseline, and ${\sigma_{kl}^{ij}}^2$ be the variance in gauge $\mathcal{C}^{ij}$ with measured distance d_{ij}. We first obtain for the baseline gauge:

$${\sigma_{kl}^b}^2 = \nabla_{\boldsymbol{\theta}} d_{kl}^\top \, \boldsymbol{V}_b \, \nabla_{\boldsymbol{\theta}} d_{kl} . \tag{5.21}$$

Now in gauge $\mathcal{C}^{ij}$ it is clear that if we want to predict the same distance that we measured, the uncertainty will be zero. So let us assume that the length we measure, d_{ij}, is not the length we are predicting, d_{kl}, and further assume that it does not share endpoints with the line we are predicting, that is $\boldsymbol{s}_i$, $\boldsymbol{s}_j$, $\boldsymbol{s}_k$, and $\boldsymbol{s}_l$ are all distinct features denoted as: $i, j \neq k, l$. Then we obtain:

$$\begin{aligned} {\sigma_{kl}^{ij}}^2 &= \nabla_{\boldsymbol{\theta}} d_{kl}^\top \, \boldsymbol{V}_{d_{ij}} \, \nabla_{\boldsymbol{\theta}} d_{kl} \\ &= \nabla_{\boldsymbol{\theta}} d_{kl}^\top \left(\boldsymbol{V}_b - \frac{1}{b}\boldsymbol{u}\boldsymbol{v}_{ij}^\top \boldsymbol{V}_b - \frac{1}{b}\boldsymbol{V}_b \boldsymbol{v}_{ij}\boldsymbol{u}^\top + \frac{1}{b^2}\boldsymbol{v}_{ij}^\top \boldsymbol{V}_b \boldsymbol{v}_{ij}\boldsymbol{u}\boldsymbol{u}^\top \right) \nabla_{\boldsymbol{\theta}} d_{kl} \end{aligned}$$

$$
\begin{aligned}
&= {\sigma_{kl}^{b}}^2 + \frac{1}{b^2}(\boldsymbol{v}_{ij}^\top \boldsymbol{V}_b \boldsymbol{v}_{ij})(\nabla_{\boldsymbol{\theta}} d_{kl}^\top \boldsymbol{u})(\boldsymbol{u}^\top \nabla_{\boldsymbol{\theta}} d_{kl}) \\
&= {\sigma_{kl}^{b}}^2 + \frac{1}{b^2}(\boldsymbol{v}_{ij}^\top \boldsymbol{V}_b \boldsymbol{v}_{ij})(\nabla_{\boldsymbol{\theta}} d_{kl}^\top \boldsymbol{u})^2 \qquad (5.22)
\end{aligned}
$$

The second and third terms of the expansion of $\boldsymbol{V}_{d_{ij}}$ drop out since their non-zero rows and columns do not overlap the non-zero elements of $\nabla_{\boldsymbol{\theta}} d_{kl}$. The last term of equation (5.22) is strictly greater than zero so long as both distances, d_{ij} and d_{kl}, are non-zero, and $\boldsymbol{V}_b$ is rank $3N$. Hence we can deduce the following theorem.

Theorem 5.3.1. *Consider an object model reconstructed from a perfectly calibrated stereo pair. Let $\mathcal{C}^{ij}$ be the gauge defined by fixing distance d_{ij}, and $\mathcal{C}^b$ be the gauge defined by fixing the baseline b so that the object scale is the same in each gauge. Further let d_{kl} be the length we wish to estimate not sharing feature points with d_{ij}, and ${\sigma_{kl}^{ij}}^2$ and ${\sigma_{kl}^{b}}^2$ be its variances in gauges $\mathcal{C}^{ij}$ and $\mathcal{C}^b$ respectively. Then if the normal covariance (or any geometrically equivalent covariance) of the object has a nullspace of rank 1, we can state:*

$$
{\sigma_{kl}^{ij}}^2 > {\sigma_{kl}^{b}}^2 . \qquad (5.23)
$$

That is, for a stereo pair with known E, if we can measure the baseline perfectly or any other length on the object perfectly, we will obtain a more accurate estimate of length d_{kl} using the baseline to fix the gauge.

This result does not apply if the measured and predicted lengths, d_{ij} and d_{kl}, share a feature point. Intuitively this is understandable from Chapter 4 since then these lengths, depending on their orientations, may be strongly correlated and fixing one may strongly reduce the uncertainty on the other.

An intuitive reason for this theorem is as follows. From the structure of the Hessian or the Fisher information matrix, we see that the cause of the correlation between feature points is only due to their respective correlation with the baseline. When the baseline is fixed they are uncorrelated. Hence we can think of fixing a point as first affecting the baseline, and only through this effect it then affects other points. Directly fixing the baseline thus has a more pronounced affect on the feature points.

5.3.4 Sensitivity to Measurement Error

The use of theorem 5.3.1 in real applications depends on the gauge fixing measurements having perfect accuracy. If our measurement of length d_{ij} or of the baseline b has error

in it, then final estimate of d_{kl} will have an additional uncertainty term. This uncertainty term corresponds precisely to that we derived in Chapter 4, sections 4.2.3 and 4.3.2.

Let σ_b^2 and σ_{ij}^2 be the variances of the measurement of the baseline and the length d_{ij} respectively. Then it follows, from an analogous derivation to that in Section 4.3.2, that in gauge $\mathcal{C}^b$ the variance of our estimated length d_{kl} becomes:

$$\sigma_{kl}^{b\ 2} = \nabla_{\boldsymbol{\theta}} d_{kl}^\top \boldsymbol{V}_b \nabla_{\boldsymbol{\theta}} d_{kl} + \sigma_b^2 \frac{d_{kl}^2}{b^2} \,. \tag{5.24}$$

In gauge $\mathcal{C}^{ij}$, the variance is:

$$\sigma_{kl}^{ij\,2} = \nabla_{\boldsymbol{\theta}} d_{kl}^\top \boldsymbol{V}_{d_{ij}} \nabla_{\boldsymbol{\theta}} d_{kl} + \sigma_{ij}^2 \frac{d_{kl}^2}{d_{ij}^2} \,. \tag{5.25}$$

We see that the additional error term is scaled by the square of the ratio of the length we are estimating to the length we measured. The baseline is treated identically to any other length. Thus the error for longer lengths is reduced more than for shorter lengths. Hence if $\sigma_b^2 = \sigma_{ij}^2$ and if we measure a length on the object that is longer than the baseline, the error in its measurement will be less significant than the error of the baseline in predicting another length.

5.4 Stereo with Unknown Epipolar Geometry

Our previous result that compares the accuracy of knowing the baseline versus another lengths in the scene applies only in the case of known epipolar geometry. We now want to derive a similar comparison for the case when the epipolar geometry is unknown, and itself must also be estimated along with the shape from matched feature points.

5.4.1 Parametrization

We assume now that the positions of the N 3D points, the baseline and the essential matrix are all unknown. Since the essential matrix contains only five independent parameters, we choose to encode it using the camera rotation and translation as in equation (5.7). The translation will only be recovered up to a scale factor. The baseline is conveniently encoded as the the magnitude of the translation: $b = ||\boldsymbol{t}||$. Perturbations of our parameters are thus encoded in the $3N + 6$ element vector:

$$\Delta\boldsymbol{\theta} = \begin{pmatrix} \Delta\boldsymbol{s}_1 \\ \vdots \\ \Delta\boldsymbol{s}_N \\ \Delta\boldsymbol{t} \\ \Delta\boldsymbol{\Omega} \end{pmatrix} . \tag{5.26}$$

The optimal solution and its accuracy are calculated analogously to section 5.3.2, but the rank of the normal covariance is $3N + 5$.

5.4.2 Accuracy within Gauges

In order to calculate the covariance within a gauge, we will use the gauge freedoms and constraints. Since overall translation and rotation are fixed, there is only a scale freedom. This gauge freedom is encoded in vector form as:

$$\boldsymbol{u} = \begin{pmatrix} \boldsymbol{s}_1 \\ \vdots \\ \boldsymbol{s}_N \\ \boldsymbol{t} \\ 0 \end{pmatrix} . \tag{5.27}$$

Again we consider two kinds of constraints. The first is when the distance between points $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ is found to be d_{ij}, and the second when the baseline, given by $b = \|\boldsymbol{t}\|$ is of length d. In the same form as equation (5.15), the gradients of these constraints are respectively:

$$\boldsymbol{v}_{ij} = \begin{pmatrix} 0 \\ \vdots \\ (\boldsymbol{s}_i - \boldsymbol{s}_j)/d_{ij} \\ 0 \\ \vdots \\ (\boldsymbol{s}_j - \boldsymbol{s}_i)/d_{ij} \\ 0 \\ \vdots \end{pmatrix} , \qquad \boldsymbol{v}_b = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \boldsymbol{t}/d \\ 0 \end{pmatrix} , \tag{5.28}$$

In this case we cannot prove the same result as when the essential matrix is known. The Fisher information matrix now has the form:

$$\boldsymbol{J_\theta} = \left(\begin{array}{ccc|c|c} \nabla^2_{\boldsymbol{s}_1} J & & 0 & \nabla_{\boldsymbol{s}_1 \boldsymbol{t}} J & \nabla_{\boldsymbol{s}_1 \boldsymbol{\Omega}} J \\ & \ddots & & \vdots & \vdots \\ 0 & & \nabla^2_{\boldsymbol{s}_N} J & \nabla_{\boldsymbol{s}_N \boldsymbol{t}} J & \nabla_{\boldsymbol{s}_N \boldsymbol{\Omega}} J \\ \hline \nabla_{\boldsymbol{t}\boldsymbol{s}_1} J & \cdots & \nabla_{\boldsymbol{\Omega}\boldsymbol{s}_N} J & \nabla^2_{\boldsymbol{t}} J & \nabla_{\boldsymbol{t}\boldsymbol{\Omega}} J \\ \hline \nabla_{\boldsymbol{\Omega}\boldsymbol{s}_1} J & \cdots & \nabla_{\boldsymbol{\Omega}\boldsymbol{s}_N} J & \nabla_{\boldsymbol{\Omega}\boldsymbol{t}} J & \nabla^2_{\boldsymbol{\Omega}} J \end{array}\right) . \tag{5.29}$$

We see that fixing the baseline still leaves correlation terms between shape and the other translation and rotation components. Thus we use the normal covariance in our uncertainty modeling:

$$\boldsymbol{V}_{\perp\mathcal{G}} = (\boldsymbol{J_\theta})^{-}_{3N+5} . \tag{5.30}$$

This is transformed into the gauge corresponding to our measurement, d_{ij} or b, and we obtain the predicted variance, ${\sigma^{ij}_{kl}}^2$ or ${\sigma^{b}_{kl}}^2$, of our estimated length, e_{kl} as in equations (5.24) and (5.25):

To compare knowledge of the baseline with knowledge of lengths on the object we will instead rely on numerical analysis. This mode of stereo, in which we solve for the epipolar geometry, is precisely the 2-image SFM problem with known internal parameters. Thus our results from Chapter 4 apply and can be used to determine which length in the scene it is best to measure. Here we would also like to know how fixing the baseling affects accuracy in comparison to lengths on the object.

5.5 Experiments

We illustrate our error estimation method on stereo views of a real object with ground truth, and a synthetic sequence.

5.5.1 Stereo Pairs of a Cube

Figure 5.1 shows a stereo pair of a $30 \times 30 \times 30$cm cube with a set of registered features. The images have been rectified and are obtained using a pre-calibrated Digiclops stereo head. The features are hand-registered features and we assume they have independent and uniform error distribution, but with unknown overall variance magnitude. The 3D positions of these features on the cube were measured to ± 0.5mm accuracy. We selected two lines on the cube, a long one $\boldsymbol{L}$, and a short one $\boldsymbol{S}$, which we will use to illustrate the effect of gauge fixing. An even shorter line, $\boldsymbol{T}$, is used in some of our examples.

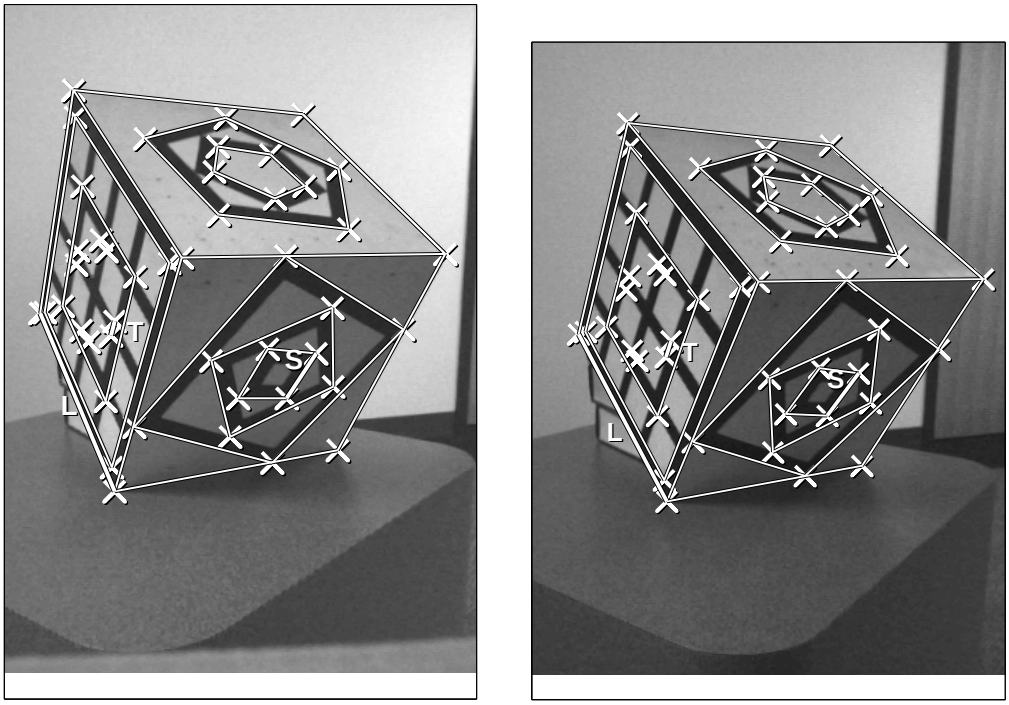
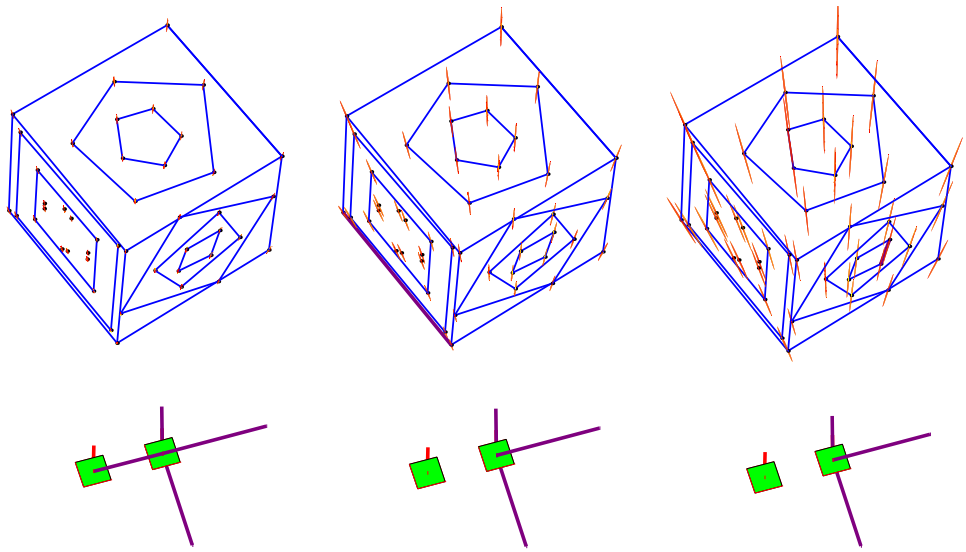


Fig. 5.1. A stereo pair of a $30 \times 30 \times 30$cm cube with 45 features matched by hand. We measured the 3D ground-truth position of each of these features on the cube to ± 0.5mm accuracy. Three of the lines on the cube will be used for gauge fixing, and these are labeled $\boldsymbol{L}$, $\boldsymbol{S}$ and $\boldsymbol{T}$.

Now the 3D shape can be estimated either assuming the stereo head has known essential matrix, E, or unknown E. The 3D reconstruction along with uncertainty ellipsoids is shown for the known E case in Figure 5.2 and the unknown E case in Figure 5.3. Not surprisingly the error is significantly larger when the relative rotation and translation are unknown. In both of these Figures the uncertainties are shown for the three gauges corresponding to (a) fixing the baseline, (b) fixing line $\boldsymbol{L}$, and (c) fixing line $\boldsymbol{S}$. The variation in the ellipses between the gauges shows the significant effect of the gauge. Large ellipses may sometimes hide significant correlation between features. So while Figure 5.3 (b) appears to have significant error, this is mostly due to uncertainty in object position, and when the coordinate system is placed on the object, as in (d), we see that the inter-feature error is small. This illustrates our point that we can only properly talk of estimates that are invariant to the unknowns, namely overall rotation and translation, and so we cannot meaningfully define absolute feature position and uncertainty, but can give estimates of lengths and their uncertainties.



(a) (b) (c)

Fig. 5.2. The estimated 3D shape with ellipsoids representing standard deviations is shown, along with the two camera positions for the known E stereo case. A different gauge is chosen for each of these plots. In all of them the coordinate origin and orientation is fixed with the right camera. In (a) the baseline is used to fix the scale, in (b) line $\boldsymbol{L}$ is used to fix the scale, and in (c) line $\boldsymbol{S}$ is used to fix the scale.

On top of the 3D reconstructions in Figures 5.2 and 5.3 we include a set of 40 lines connecting features to aid in viewing. If we know the scale of the cube, we can predict the length of each of these lines, as well as our uncertainties in these predictions, using equations (5.24) and (5.25). The lengths and their uncertainties are invariant to the translation and orientation of the coordinate system of the stereo pair. Figure 5.4 shows the uncertainties in the predictions of each of these lines for four gauges: known baseline, known $\boldsymbol{L}$, known $\boldsymbol{S}$, and known $\boldsymbol{T}$. We see that when E is already known, knowing the baseline is better than knowing $\boldsymbol{L}$, $\boldsymbol{S}$ or $\boldsymbol{T}$, except in predicting a few lines which correspond to those that share endpoints with $\boldsymbol{L}$ or $\boldsymbol{S}$. But in the example when E is unknown, the opposite is true; knowing the baseline is generally the least helpful gauge, always giving more uncertainty than $\boldsymbol{L}$ or $\boldsymbol{S}$, although better than $\boldsymbol{T}$.

To confirm that the uncertainty measures shown in Figure 5.4 do actually correspond to the real standard deviations of the lines given our noise model, we performed a set of Monte Carlo experiments. The results shown in Figure 5.5 confirm that these uncertainty measures are correct indicators of the standard deviation.

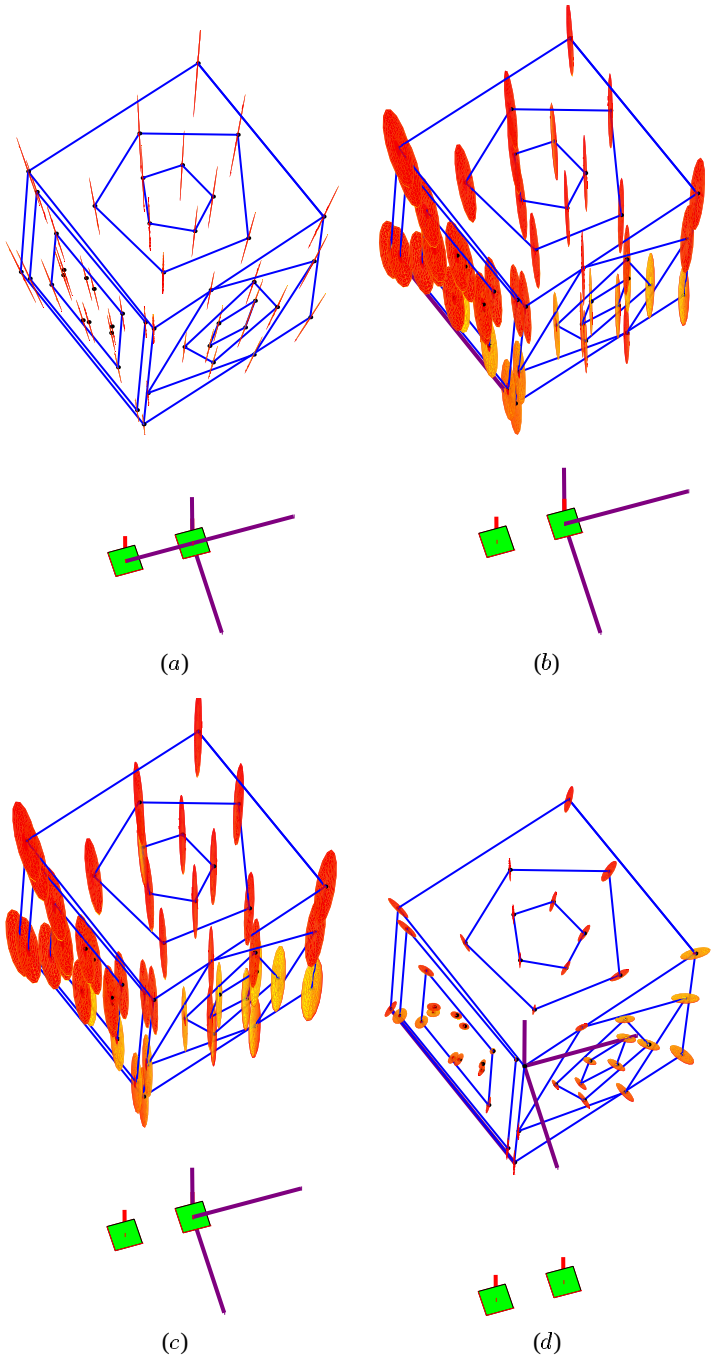


Fig. 5.3. As in Figure 5.2, the estimated 3D shape with ellipsoids representing standard deviations is shown, along with the two camera positions, but here for the unknown E stereo case. A different gauge is chosen for each of these plots. In (a–c) the coordinate origin and orientation is fixed with the right camera. In (a) the baseline is used to fix the scale, in (b) line $\boldsymbol{L}$ is used to fix the scale, and in (c) line $\boldsymbol{S}$ is used to fix the scale. The plot in (d) is similar to (b) in that line $\boldsymbol{L}$ is used to fix the gauge, but the origin of the coordinate system is fixed to the middle corner. The resulting decrease in covariances when an object–centered coordinate system is used shows that the covariances in (a–c) are strongly correlated due to the uncertainty of the overall object position. We conclude that, at least in this example with unknown E, stereo provides better shape estimation than depth estimation.

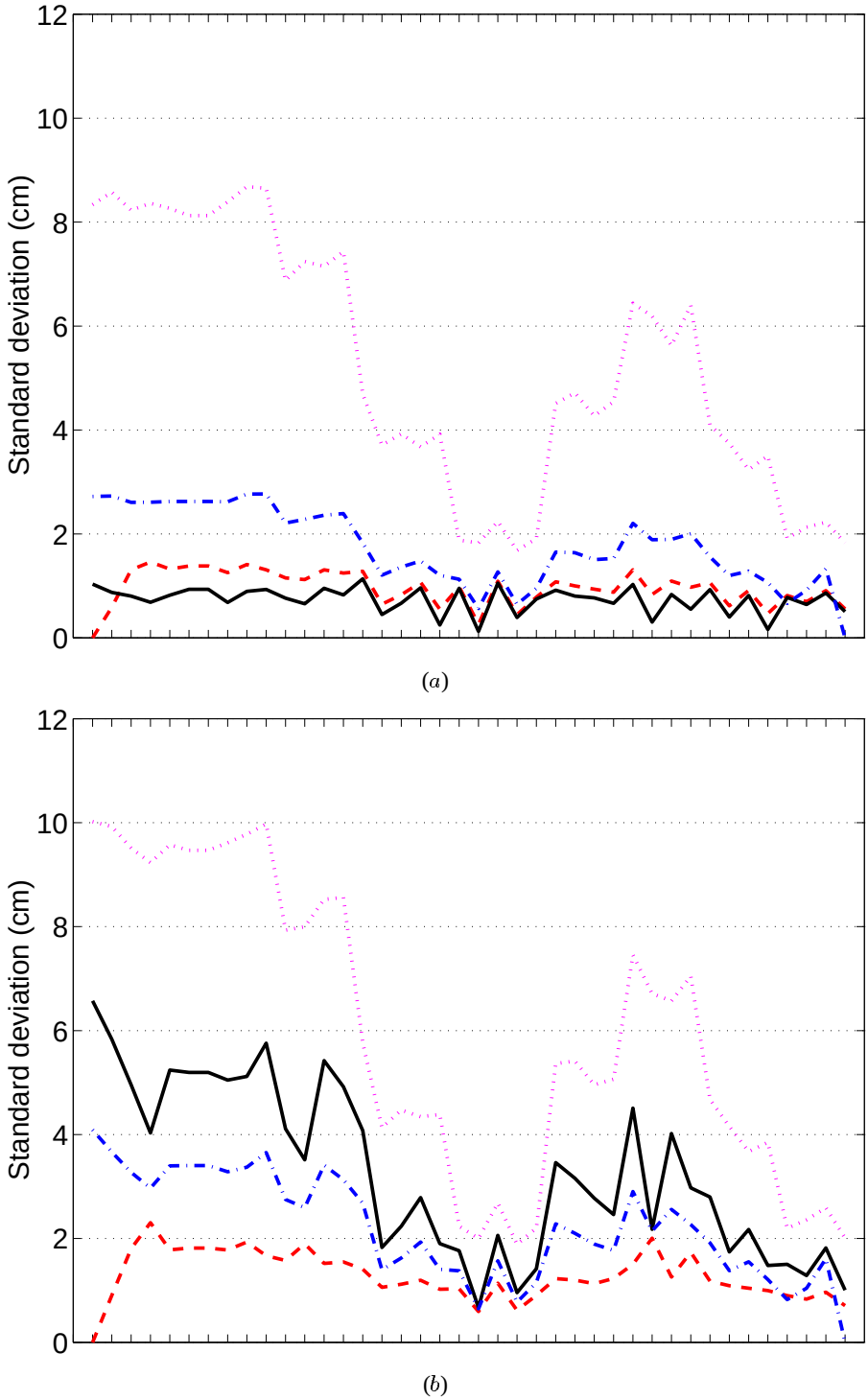


Fig. 5.4. The line prediction uncertainties due to gauge fixing are shown for (a) known E stereo, and (b) unknown E stereo. These plots show the standard deviation of each line when one of the line lengths is assumed to be known. The solid line corresponds to the estimate obtained when the baseline is fixed, the dashed line when line $\boldsymbol{L}$ is fixed, the dot-dashed line when line $\boldsymbol{S}$ is fixed, and the dotted line when $\boldsymbol{T}$ is fixed.

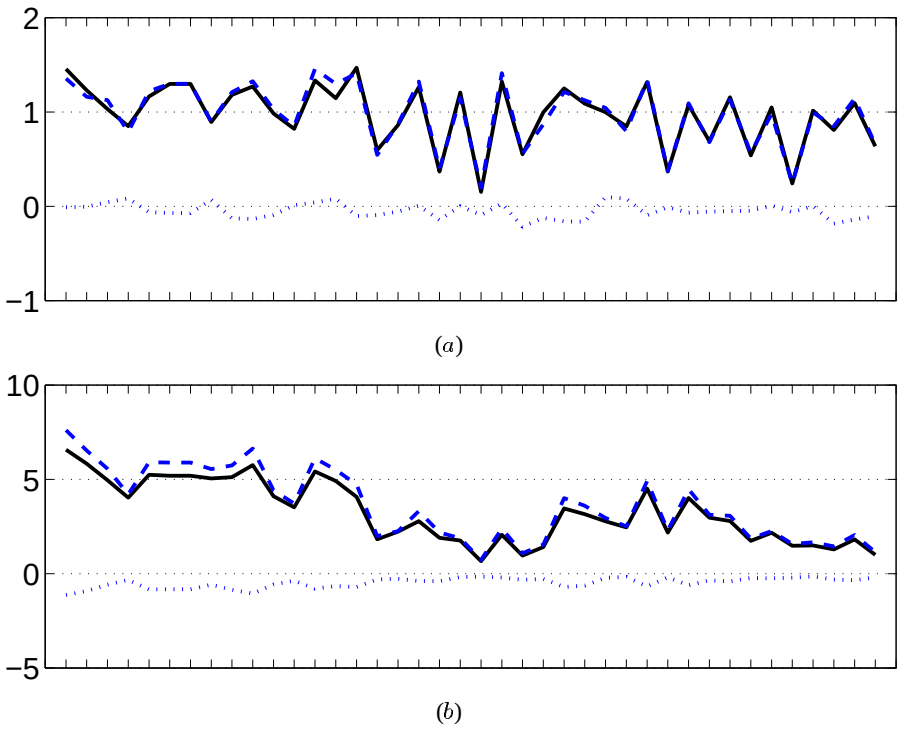


Fig. 5.5. In order to confirm that the uncertainties plotted in Figure 5.4 actually do correspond to the standard deviations of a model with Gaussian noise assumption, we performed a Monte Carlo simulation of the stereo algorithm. We used the model originally estimated as our "truth", and added uniform Gaussian noise. We performed 200 simulations and recorded the estimated line lengths for each. Plot (a) is the known E case and plot (b) the unknown E case, and in both cases we assumed the baseline was known. The solid line shows the predicted standard deviation in the estimate of each line, the dashed line is the empirically measured standard deviation for 200 runs, and the dotted line is the mean error. As the number of simulations increases to infinity, we expect the measured standard deviation to equal the predicted standard deviation, and the mean error to be zero. Units are in centimeters.

We can also use the ground truth of each line as a way to fix the scale, and then to estimate our lengths. Figure 5.6 shows this result for predicting line $\boldsymbol{L}$. The true length is 30cm, and the error in the predictions correspond well with the uncertainties. We also show the predictions and uncertainty when the baseline is known. We see that in the known E case this provides a lower bound on uncertainty, and in the unknown E case it is worse than all the lines on the cube.

We have shown results at a single baseline, but would like to see how the results vary when the baseline is changed. Figures 5.7, 5.8, and 5.9 give results of prediting line lengths in different gauges and four different baselines, using equations (5.24) and

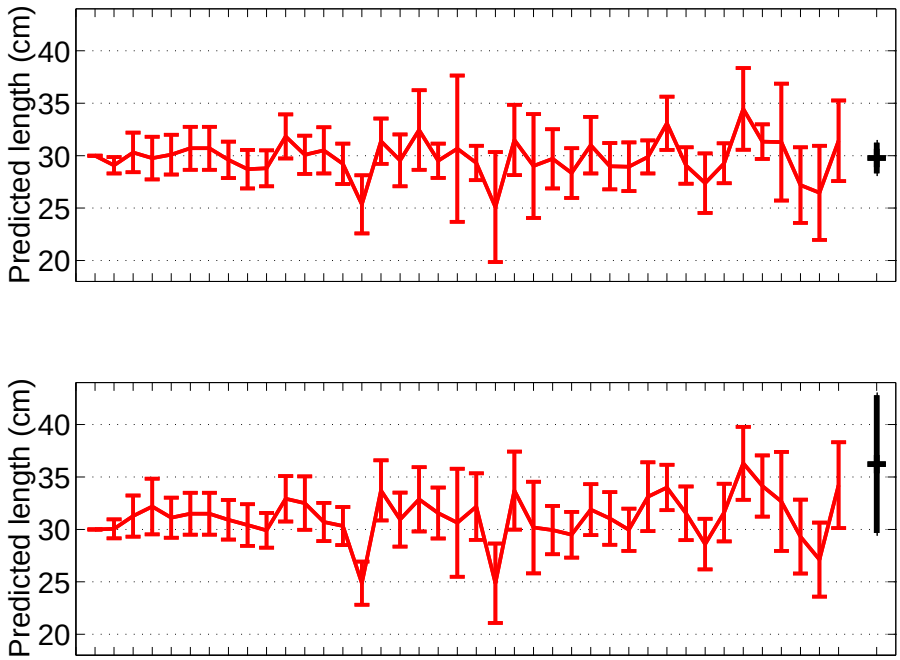


Fig. 5.6. We show the predicted value of line $\boldsymbol{L}$, using each of the other lines to fix the scale, along with the uncertainty in standard deviations for (a) stereo with known E and (b) stereo with unknown E. The right-most error bar is the prediction and uncertainty when the baseline is used to fix the scale. The true length is 30cm.

(5.25). As we would expect for smaller baselines the uncertainty increases, and larger baselines give more accurate estimates. Throughout the range of these the trend is maintained of fixing the baseline being the best option for stereo known E and a poor option for stereo with unknown E. Only for the very small baseline does this break down and all gauges are more or less equivalently poor. We expect this is because the error is so large at that point that our first order approximation of our recovered solution being at the true solution breaks down.

5.5.2 Stereo Pairs of a Synthetic Sphere

In theory we have solved the baseline problem for stereo with known epipolar geometry, and confirmed empirically that knowing the baseline is better than lengths on the object. However, in the case in which we simultaneously solve for epipolar geometry and object shape, our results show that compared to some lines fixing the baseline is better,

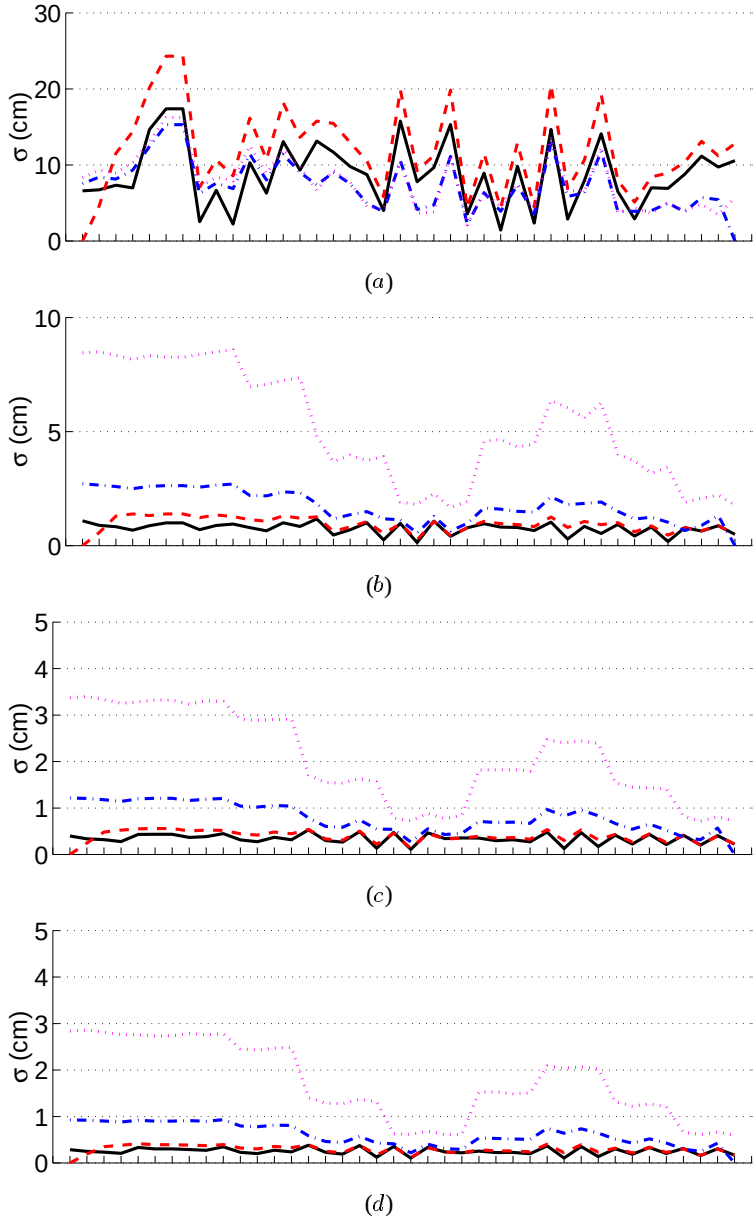


Fig. 5.7. Analogously to Figure 5.9, we show the uncertainties in predicting each of the line lengths using the baselines from Figure 5.8, but with a camera pair having known E. The length of the baseline for each plot is: (a) 1.2cm, (b) 10cm, (c) 32.5cm, and (d) 41.9cm. In each case the solid line indicates the uncertainty when the baseline is fixed, the dashed line when $\boldsymbol{L}$ is fixed, and the dot-dashed line when $\boldsymbol{S}$ is fixed. (See Figure 5.1 for the images in the 10cm baseline case.) Plot (a) shows the case for a very small baseline. Here the uncertainties are very large and our claim that fixing the baseline is always best fails. For larger baselines, though, we see that indeed fixing the baseline is always better than fixing one of the other two lengths on the object, except when that line includes one or two of the endpoints of the line we want to predict.

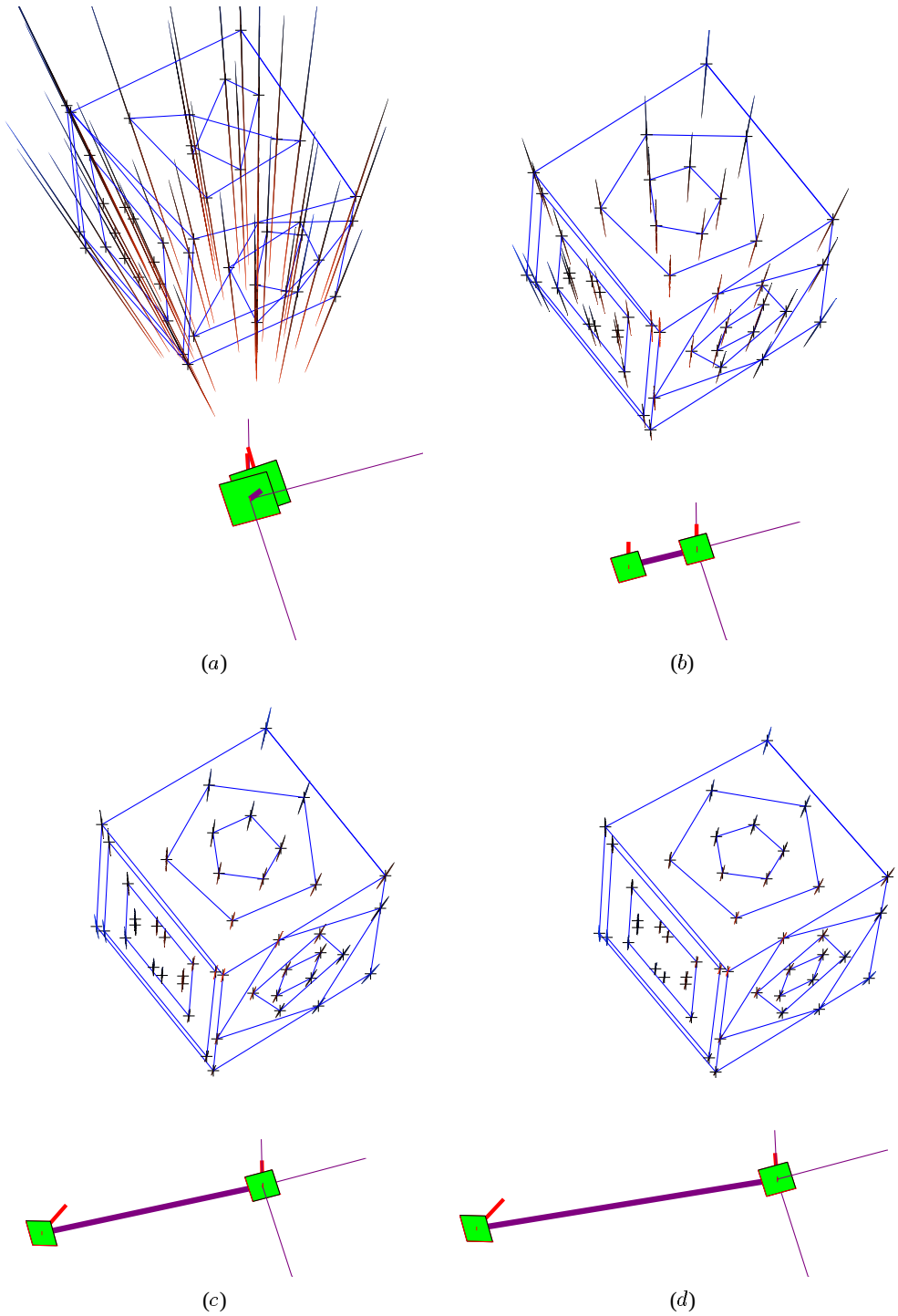


Fig. 5.8. Whichever gauge is used, the uncertainty will depend on the baseline of the camera pair. Here we show the 3D uncertainties for different baselines of the unknown E camera pair and with the baseline used to fix the scale. The length of the baseline in each case is: (a) 1.2cm, (b) 10cm, (c) 32.5cm, (d) 41.9cm. Figure 5.9 shows the uncertainties in predicting lengths for each of these.

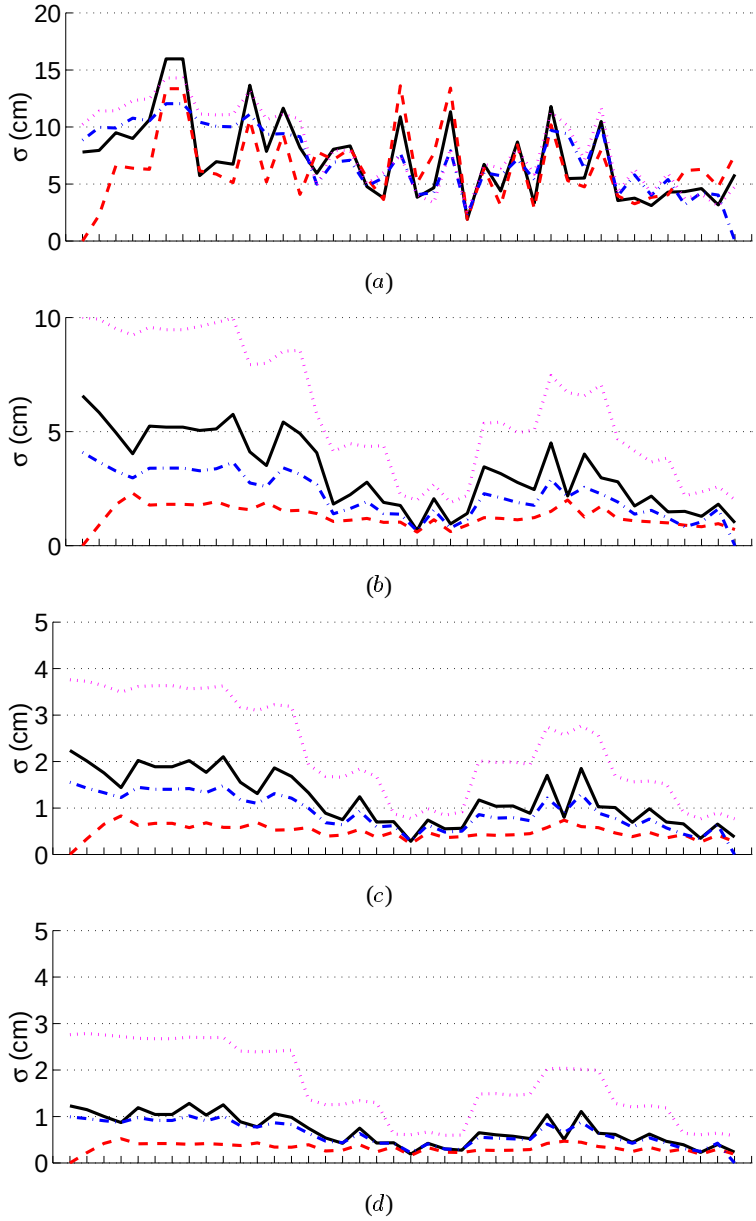


Fig. 5.9. We show the uncertainties in predicting each of the line lengths using the baselines from 5.8, and an unknown E camera pair. In each case the solid line indicates the uncertainty when the baseline is fixed, the dashed line when lenght $\boldsymbol{L}$ is fixed, and the dot-dashed line when $\boldsymbol{S}$ is fixed. (See Figure 5.1 for the images in the 10cm baseline case.) Plot (a) illustrates that when the baseline is very small, the errors become very large, and in this case too large to make the results meaningful. We notice that as the baseline is increased in plots (b–d), the uncertainties decrease, but the uncertainties using the baseline to fix scale always remain larger than the uncertainties using the other object-based gauges.

but compared to other lines fixing the baseline is worse. Thus we would like a better characterization of when knowing the baseline is better and when it is worse.

We give some empirical results for stereo pairs on the synthetic sphere described in section 4.5.2. The two furthest-apart cameras are used for stereo, and these have a baseline exactly equal to the diameter of the sphere, as illustrated in Figure 5.10. The

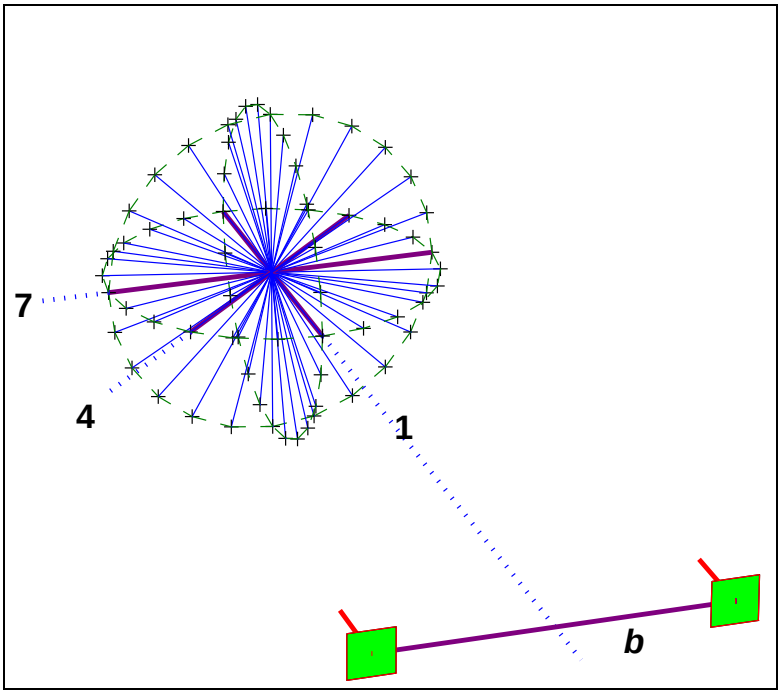


Fig. 5.10. A sphere created by intersecting 3 circles orthogonally. We are interested in predicting three diameters labeled 1, 4 and 7, as well as the baseline b whose length is one diameter. To predict these lengths we will individually fix all of the other diameters marked with solid lines. The dashed circles are only to aid in viewing. We compare this to SFM results with five cameras in Figure 4.14.

uncertainties of predicting each of these, or equivalently, due to our symmetry relationship in Figure 4.1, of the uncertainties in using each of these lines to predict the other lines, are shown in Figure 5.11. We notice, first of all, that the uncertainties generated by gauge fixing with lines 1, 4 and 7 have the same pattern as in the multi-image case of SFM shown in Chapter 4, Figures 4.15 and 4.16. The only significant difference is that with two cameras the uncertainties are larger. As we anticipated from the equations, stereo with unknown E is a special case of SFM, and so the results we derived in Chap-

ter 4 and our numerical analysis for the SFM case with a small range of camera motion, will apply to this type of stereo.

Figure 5.11 also permits us to compare fixing the baseline with fixing other lines on the object that have the same length. We see that in almost all cases fixing the baseline gives greater uncertainty than fixing the other lines. The only exceptions are in comparison to line 1, along the viewing direction. Fixing line 1 gives slightly greater uncertainty in predicting the lines in the plane perpendicular to the viewing direction than the uncertainties obtained for these lines by fixing the baseline. This result if for a compact object with small field of view. With a large field of view we would expect more accurate estimation of the camera motion, and hence measuring the baseline will not be as poor a method for fixing the scale.

5.6 Discussion

In this chapter we have examined the effects of gauge fixing on length estimation for stereo. Two different stereo configurations were considered. In the first we assume that the epipolar geometry is perfectly known, but the baseline is unknown. In the second we include estimation of the epipolar geometry along with the shape estimation for feature correspondances. The main question we asked was whether it is better to measure and fix the baseline or to measure and fix a length in the scene for optimal accuracy.

Our result for the case of known epipolar geometry proved that it is always best to know the baseline perfectly rather than a line on the object. If the baseline and other lengths can only be known up to a limited accuracy, then the longer the length we fix the less this error will propagate. Hence it is feasible that if the baseline is known up to a limited accuracy that it may be better to measure a much longer line in the scene. In general, though, this result justifies the standard approach of pre-calibrating the baseline along with the rest of the stereo parameters.

Unlike the previous case, we found numerically that when the epipolar geometry is unknown and estimated along with shape, fixing the baseline is one of the worst measures in terms of final shape accuracy. This is a useful fact for active vision systems, since it means that it is probably not a good idea to try and carefully measure the baseline of cameras that swivel with respect to each other. Rather it is better to obtain metric information by using a known dimension of an object in the scene.

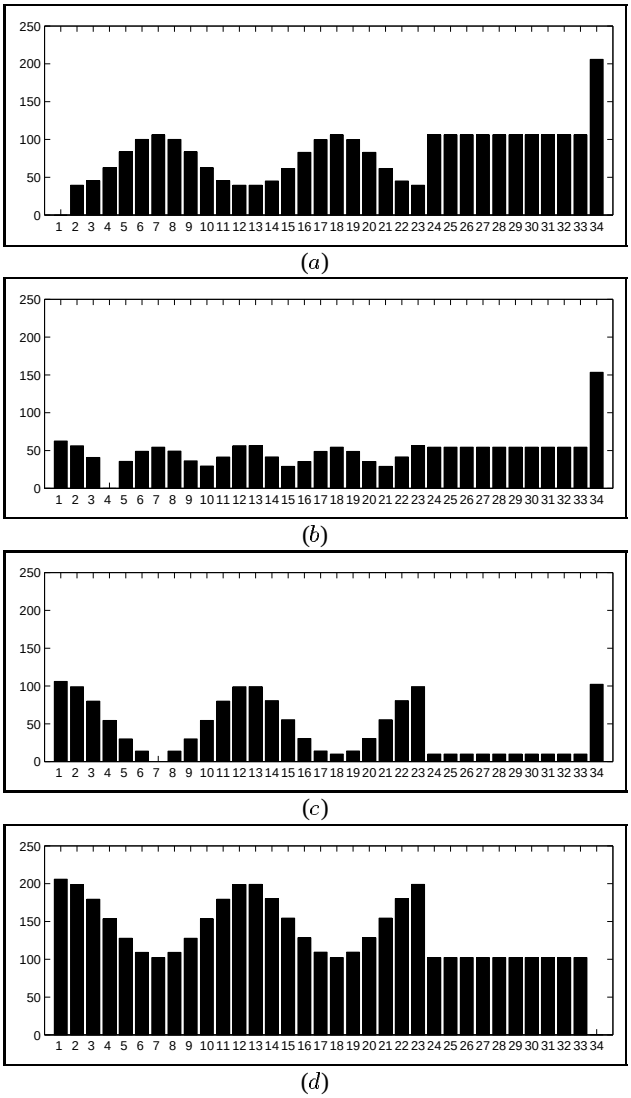


Fig. 5.11. The uncertainties in line lengths obtained by fixing: (a) line 1, (b) line 4, (c) line 7, and (d) the baseline. Column 34 corresponds to the baseline uncertainty. Fixing the baseline gives the worst results except for some of the lines predicted by fixing line 1. The epipolar geometry is assumed to be unknown.

6. Discussion

We have investigated the problem of parametrized model fitting when there are indeterminacies in the fitted parameters. The problem we focussed on was 3D computer vision. Our first goal was to see what consequences these indeterminacies have on perturbations of the solution, and hence how the uncertainty of the solution is impacted. This was followed by our analysis of the gauge fixing problem and how to best choose gauges to optimize accuracy. This investigation has proceeded from a theoretical to an empirical analysis and sought to discover the implications for 3D estimation from images both by SFM and by stereo. We now wish to summarize the key concepts that we elucidated in the theory and in the empirical analysis. We will also identify some of the limitations of our analysis, and propose ideas for continued research.

6.1 Summary and Contributions

We used the idea of conceptualizing indeterminacies as gauge orbits in a high dimensional parameter space, as described in Chapter 2. All the points on a gauge orbit are geometrically equivalent given only our measurements. Solving for the "true solution" becomes finding the correct gauge orbit in this space. Using this as our framework, the next step was to analyze the uncertainty of the solution.

6.1.1 Uncertainty and Invariants

While all points on a gauge orbit are geometrically equivalent given our measurements, their parameter values may vary significantly. This means that the parameters that vary along the gauge orbit are not true geometric properties of the solution. Rather it must be gauge invariants that describe true geometric entities, as these remain constant on a gauge orbit. So we could consider using only gauge invariants of our solution and their uncertainties as valid descriptions of our model. This would eliminate the problem of indeterminacies.

Furthermore, invariants are particularly useful in evaluating the quality of a method for estimating parametrized quantities, such as 3D position of points. If an estimator is unbiased and errors are small, generally the smaller the covariance the better. But if non-invariant properties are predicted by the estimator, the choice of gauge may be critical in this evaluation. In some gauges the parameters may have very small covariances and in other gauges they may have large covariances. If there is no prior justification for a particular gauge, one can be tempted to choose a favorable gauge, and in doing this "normalize" away a significant portion of the noise. In comparisons it is thus safer to report invariant properties. These do not depend on choice of gauge and so will not be increased or decreased by arbitrary choices in the experiment.

However, for the actual operation of an algorithm and in the interpretation of the results, it is generally not feasible to restrict our model parameter set to include only those parameters that are gauge invariant. An object description of pure length ratios and angles is not easy to create or to interpret, whereas it is easy to interpret and manipulate a model based on 3D point positions and camera rotations. But 3D point positions and camera rotations are not gauge invariant. Thus we would like our methods to work using non-invariants, but to operate on them in a way such that when they are evaluated or compared only the essential geometric information that is gauge invariant is considered.

6.1.2 Uncertainty and Geometric Equivalence

Our first goal, starting in Chapter 2, was to characterize the uncertainty of 3D model estimation using non-invariants. We assumed that outliers have been removed and that the noise is small, and focused on first order perturbation analysis of the solution.

Gauge Freedoms. A first step of perturbation analysis is to locally linearize the parameter space around the point of interest. If the parameters are invariant, and there are sufficient measurements, the solution is locally unique and the analysis becomes standard covariance propagation. When the solution lies on a gauge orbit, as will happen when the parameters are not invariant, we identify the gauge freedoms as consisting of the tangent space to the gauge orbit at that point. The factor that makes our analysis interesting and different from standard perturbation analysis is precisely the effects of these gauge freedoms on perturbations.

A perturbation along a gauge freedom is not detectable by our measurements and hence must be eliminated. We eliminate these gauge freedoms by applying a set of gauge constraints that define a locally unique solution. These constraints effectively define lo-

cally a new parameter space with no gauge freedoms. Standard perturbation analysis can be performed in this parameter space, but now the problem becomes how to relate this analysis to our original problem parameters that have gauge freedoms.

Geometric Equivalence. We derived a geometric equivalence test with which to compare perturbations of parameters in different gauges. This links the uncertainty description with any one set of constraints to the uncertainty description that uses another set of constraints. Only the essential geometric properties that affect the invariants are used in this comparison.

Oblique Projection. We also derived an operator to transform perturbations or covariances to different points on the gauge orbit and between different gauge manifolds. This operator involves a change of variables step, and then an oblique projection along the tangent space to the gauge orbit and onto the gauge manifold. By this process the geometric equivalence of the uncertainty models is maintained between different sets of gauge constraints.

Normal Covariance. If we wish to obtain a covariance at a point on the gauge orbit without specifying a set of gauge constraints, then calculating normal covariance is a convenient method. It is defined independent of gauge constraints and gives us a lower bound on the total covariance at that point. It can subsequently be transformed into any particular gauge. The normal covariance can be thought of as being in the gauge manifold perpendicular to the gauge orbit.

Gauge Freedoms in SFM and Stereo. Now in 3D estimation from images the basic indeterminacy is a similarity transformation. We argued in Chapter 3 that there is no "correct" coordinate system, and no overall scale, as these quantities are lost in the projection process, except in orthography which ideally keeps the correct scale. In order to meaningfully express the uncertainty of the estimated shape and motion parameters requires that we choose a gauge to express the covariance in. We derived the gauge freedoms for perspective, weak perspective and orthographic camera models, and then using these freedoms, along with whichever set of constraints we like, we showed how to transform any covariance into any gauge we choose.

Fast Covariance Estimation in SFM. As an estimate for the covariance in an arbitrary gauge we could use the normal covariance as this can be obtained by simply taking the Moore-Penrose inverse of the Hessian at the solution. Alternatively, since this may be computationally expensive, if either the number of images or features is small, we could

use the fast inversion method that we derived in Chapter 3 and proved was geometrically equivalent to the normal covariance.

6.1.3 Uncertainty Within Gauges

Sometimes we may obtain information from an additional measurement to actually determine the gauge. If we are given freedom in selecting this measurement from a set of possibilities, we would like to know which option will give us the greatest accuracy. Of course this depends on which quantities one wants to maximize the accuracy of. We chose to look at line lengths in a scene. These have the property that they are invariant to overall translation and rotation, and hence choice of gauge involves only fixing the scale.

Gauge Fixing with Lines. In order to calculate the accuracy of line lengths, in Chapter 4 we derived how the variance of each other line will be obtained when we fix one line. We then investigated how the geometric properties of a line, in particular its orientation with regard to the cameras, its position and its length, affect its effectiveness in predicting other line lengths. We divided objects into two classes: those having a large field of view compared to the line lengths we are comparing, and those having a small field of view. Objects with a large field of view resulted in there being little correlation between lines, and hence lines that are closer to the cameras, longer, and oriented orthogonal to the viewing direction are most accurately estimated and give the best gauge constraints. On the other hand, objects with a small field of view tend to have large correlation between line lengths, and so choosing the best length and orientation of a line to fix depends on which line one wants to estimate. For example, unlike large field of view objects, we showed that lines parallel to the viewing direction can be more accurately estimated by measuring other lines also parallel to the viewing direction.

Localization Error. We identified feature localization error as an additional source of error in 3D estimation from SFM and stereo. Standard methods do not model this source of error. We created a model of it and incorporated this in our uncertainty analysis, thus overcoming the tendency of methods that ignore this effect to underestimate error. The more images there are in a sequence, the greater the role played by the localization error.

Stereo. The stereo problem was also considered. We examined stereo with known epipolar geometry and stereo with unknown epipolar geometry, and asked what the best method for fixing scale is. In the case with known epipolar geometry we proved that

if we can measure the baseline with perfect accuracy, that this results in the most accurate 3D estimation procedure. In the case where the epipolar geometry is solved for along with the shape, this result does not hold, and we showed numerically that measuring the baseline is among the worst measures with which to fix the scale.

6.2 Limitations and Future Work

There are a number of limitations to our approach, and in our analyzes. We will identify a number of these limitations and for some propose directions for future work.

6.2.1 Large Errors

We have not dealt with the occurrence of large noise or gross errors. Gross measurement errors will bias the results, and the covariance will then only give a precision measure of our estimate and not a true uncertainty estimate. We dealt with this problem by manually removing bad matches and correcting some point coordinates that had severe errors. For real vision systems it would be much preferable if these could be removed through automatic outlier detection. We would like to see how accurate a model of a house a purely automatic system could create. Part of this should include failure evaluation. If the noise is too large then the linear approximations, to rotations for example, fail, and our uncertainties are not meaningful. But it may not be easy to decide what is simply a large error and what is a meaningless result because the linear approximation has failed.

6.2.2 Over-fitting and Under-fitting

A potentially difficult problem to diagnose is over-fitting or under-fitting our model. Both of these problems could result in severely biased results. For example, using an orthographic camera to estimate a scene having large fore-shortening will give skewed results due to under-fitting. But similarly using a perspective camera to model a scene that is close to orthographic may over fit and also produce skewed results. This is harder to detect since the actual error will be small. Sometimes this is revealed by the Hessian having more than the expected number of close-to-zero singular values. But this is not a reliable test when there is noise. One approach to solving this is to use geometric AIC described by Kanatani [34]. We can estimate the shape using all possible camera models and compare their information criterion numbers. The one with the smallest number has fit the data best with the least over-fitting. In actual practice we roughly emulated this

approach since we fit multiple camera models to the data and chose the one that gave us the “best looking” result. But having a quantifiable measure of this would be preferable.

6.2.3 Critical Configurations

Critical configurations create a difficulties for our uncertainty modeling. We can think of these as places in which gauge orbits intersect or split. They may actually introduce additional indeterminacies, and if the uncertainty modeling is to be correct we will need to model these additional indeterminacies. It may be fruitful to look at additional possible indeterminacies and characterize their gauge freedoms. Then one also has to detect that one has entered a critical configuration, or else is near a critical configuration. Being near these configurations could make our linear approximations poor. Hence there is much work to do for uncertainty modeling near critical configurations.

6.2.4 Local Minima?

In order for our covariance model to correspond to the true uncertainty, our solution must be at the global minimum of the cost. Hence our analysis assumed that we always achieved this point. It was easy to identify many of the local minima by examining the errors after convergence. For manual tracking these errors had a standard deviation of under a pixel, and if convergence lead to a significantly larger error then we concluded that it was trapped by a local minimum. Local minima were circumvented by changing the starting point or first applying a different camera model. All of this involved significant manual intervention, and there was no ultimate guarantee that the global solution had been achieved. This, however, is a problem common to many nonlinear optimization tasks, and one approach is to use something like Simulated Annealing.

6.2.5 Sources of Bias

We identified feature localization as a source of bias in the standard SFM and stereo algorithms. Our method in dealing with this made some rough approximations, and assumed that we knew the variance of image-based feature localization. We then guessed a number for image localization error, but we would like to investigate this further. It should be possible to characterize this error for a given feature detector using synthetic images. Then we could obtain a less arbitrary variance estimate for our localization modeling scheme.

There may be other sources of bias. For example the camera model may not be precise. For each possible source we should estimate the magnitude of its effect on our solution and from this decide if it is worth modeling.

6.2.6 Line Fixing Correlations

We have numerically analyzed and compared a few cases of gauge fixing using different lines. We identified a number of patterns in which some lines provide better gauge constraints than other lines. We expect there are more patterns to be identified. The only SFM cases we considered were those where there is a single approximate viewing direction. If, however, the camera motion is large, then the considerations for determining which line provides the best gauge may change.

6.2.7 Gauge Fixing and Stereo

We only looked at two cases of stereo: known epipolar geometry and unknown epipolar geometry. There are a number of intermediate cases to consider. If only some of the extrinsic parameters are unknown, for example the cameras can translate but not rotate, it would be nice to know what the effects of gauge fixing for these systems are, and how fixing the baseline compares to fixing line lengths on the object.

A. Exponential Mapping for Perturbations

We need to make small perturbations of our parameter vector $\boldsymbol{\theta}$ that stay in the parameter space $\mathcal{T}$. This is not a purely linear operation since some of the parameters are highly constrained. In particular the nine parameters in a rotation matrix have only three degrees of freedom. This is achieved through use of an exponential map. If $\boldsymbol{\theta}$ is a point in $\mathcal{T}$, then a perturbed point, $\boldsymbol{\theta}'$, is also in $\mathcal{T}$ if it is obtained as:

$$\boldsymbol{\theta}' = \exp(\Delta\boldsymbol{\theta})\boldsymbol{\theta}, \tag{A.1}$$

The exponential maps a perturbation, $\Delta\boldsymbol{\theta}$, in the tangent space $T_\theta[\mathcal{T}]$ onto the space $\mathcal{T}$ itself [40]. This is necessary when the perturbation $\Delta\boldsymbol{\theta}$ is not in the space $\mathcal{T}$.

A.1 Translation Perturbations

Perturbations of translations are obtained as:

$$\boldsymbol{t}' = \exp(\Delta\boldsymbol{t})\boldsymbol{t} = \boldsymbol{t} + \Delta\boldsymbol{t}, \tag{A.2}$$

with higher order terms being zero.

A.2 Rotation Perturbations

Perturbations of rotations can be expressed as:

$$\boldsymbol{R}' = \exp(\Delta\boldsymbol{R})\boldsymbol{R} = \Big(\cos\Delta\Omega\,\boldsymbol{I} + (1-\cos\Omega)\boldsymbol{l}\boldsymbol{l}^\top + \sin\Omega\,\boldsymbol{l}\times\boldsymbol{I}\Big)\,\boldsymbol{R} \tag{A.3}$$

where $\boldsymbol{\Omega}$ is a vector with three rotation parameters, $\Omega = \|\boldsymbol{\Omega}\|$ and $\boldsymbol{l} = \boldsymbol{\Omega}/\|\boldsymbol{\Omega}\|$. This is an exact expansion and is known as Rodrigues formula. It ensures that our rotation perturbation, given by matrix $\exp(\boldsymbol{\Omega})$, is a valid rotation. It rotates an angle $\|\boldsymbol{\Omega}\|$ around axis $\boldsymbol{\Omega}$.

To first order the perturbation can be written as:

$$\Delta\boldsymbol{R} = \Delta\boldsymbol{\Omega}\times\boldsymbol{R}. \tag{A.4}$$

or alternatively in the form:

$$\Delta \boldsymbol{r} = \boldsymbol{H}^{\top} \Delta \boldsymbol{\Omega} \tag{A.5}$$

where

$$\boldsymbol{R} = \left(\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3 \right), \;\; \Delta \boldsymbol{r} = \begin{pmatrix} \Delta \boldsymbol{r}_1 \\ \Delta \boldsymbol{r}_2 \end{pmatrix} \quad \text{and} \quad \boldsymbol{H}^{\top} = - \begin{pmatrix} \boldsymbol{r}_1 \times \boldsymbol{I} \\ \boldsymbol{r}_2 \times \boldsymbol{I} \end{pmatrix}. \tag{A.6}$$

The third column of the rotation matrix $\boldsymbol{R}$ can be obtained by the cross product: $\boldsymbol{r}_3 = \boldsymbol{r}_1 \times \boldsymbol{r}_2$, and so it suffices to use just $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ to encode the global rotation. Matrix $\boldsymbol{H}$ is a change of variables matrix and so can be expressed as:

$$\boldsymbol{H} = \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\Omega}}. \tag{A.7}$$

This can be used in the chain rule when we have a function $\mathcal{F}(\boldsymbol{r})$ and want to obtain its gradient with respect to $\boldsymbol{\Omega}$ parameters:

$$\nabla_{\boldsymbol{\Omega}} \mathcal{F}(\boldsymbol{r}) = \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\Omega}} \frac{\partial \mathcal{F}(\boldsymbol{r})}{\partial \boldsymbol{r}} = \boldsymbol{H} \nabla_{\boldsymbol{r}} \mathcal{F}(\boldsymbol{r}). \tag{A.8}$$

B. More Stereo Uncertainty Results

In the case of unknown epipolar geometry, stereo estimation is just the two-image case of SFM. It turns out that we obtain exactly the same qualtitative uncertainty analysis as that investigated for SFM with multiple images taken from a dominant viewing direction as done in Chapter 5. Thus all the results derived in that chapter apply. In this appendix we analyze some more factors that we have not considered and that might alter the patterns. These include changing the distance of the stereo pair to the object, and adding more feature points to the object.

B.1 Varying the Distance to the Stereo Pair

We will use the synthetic sphere illustrated in Figures 4.14 and 5.10, and use a stereo camera pair as in the latter figure. Our first test is on the effect of the distance from the stereo pair to the object. We would like to know if varying this might make knowledge of the baseline more or less effective in fixing the gauge. We will compare our results here with those reported in Figures 4.15 and 5.10.

First we illustrate the stereo pair at half and double the original distance to the object in Figure B.1. Then in Figures B.3 and B.4 we show uncertainties for both of these cases. Comparing these to Figures 4.15 and 5.10 we see that the patterns of relative accuracies in predictions using the baseline and predictions using lines 1, 4 and 7 all remain about the same, although being further from the object tends to accentuate the errors for lines along the viewing direction more than for errors in the perpendicular plane. On the other hand the closer the object, the larger an angle it subtends, and the more similar the results become to wide field of view objects described in section 4.5.2. The main difference, though, is that the further the object the greater the uncertainties in predicted lengths.

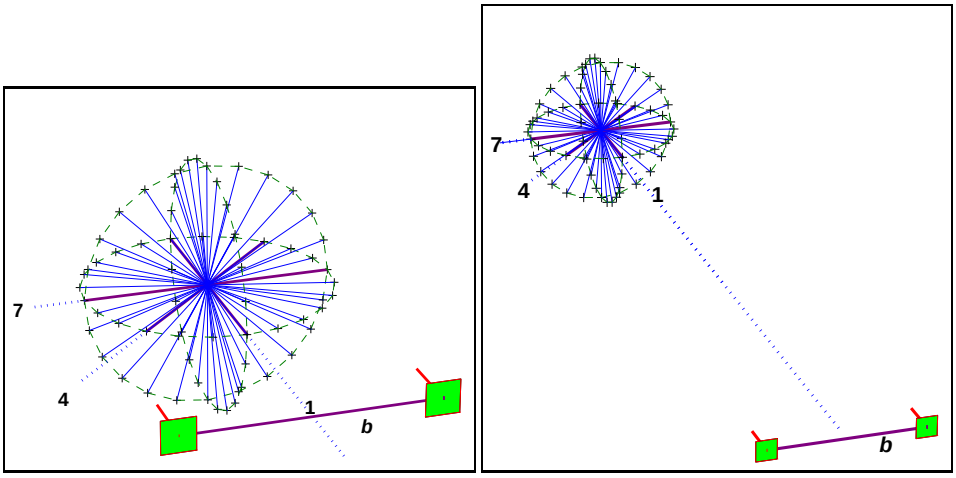


Fig. B.1. Sphere viewed at half the distance and double the distance as in Figure 5.10.

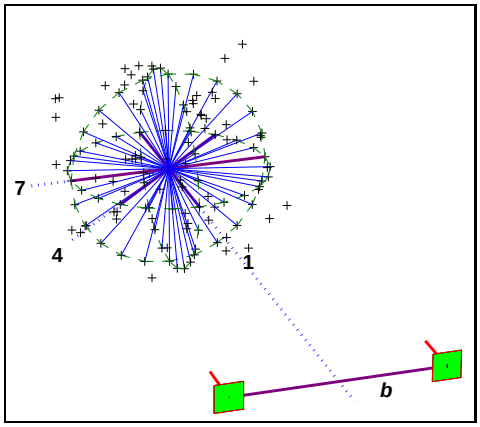


Fig. B.2. Additional random features are added around the sphere in Figure 5.10.

B.2 Adding More Features to the Shape

Another concern is that if we add more features to our shape this will change the patterns we found. Adding more features will increase the accuracy, but we want to know if it will change the relative accuracies, when the baseline and other lines are fixed.

To test this we created a sequence with our synthetic sphere and doubled the number of feature points by adding random points around the sphere, as shown in Figure B.2. We then tested the prediction accuracies obtained by fixing the baseline and lines 1, 4, and 7, and these are shown in Figure B.5. While the overall accuracy of the results

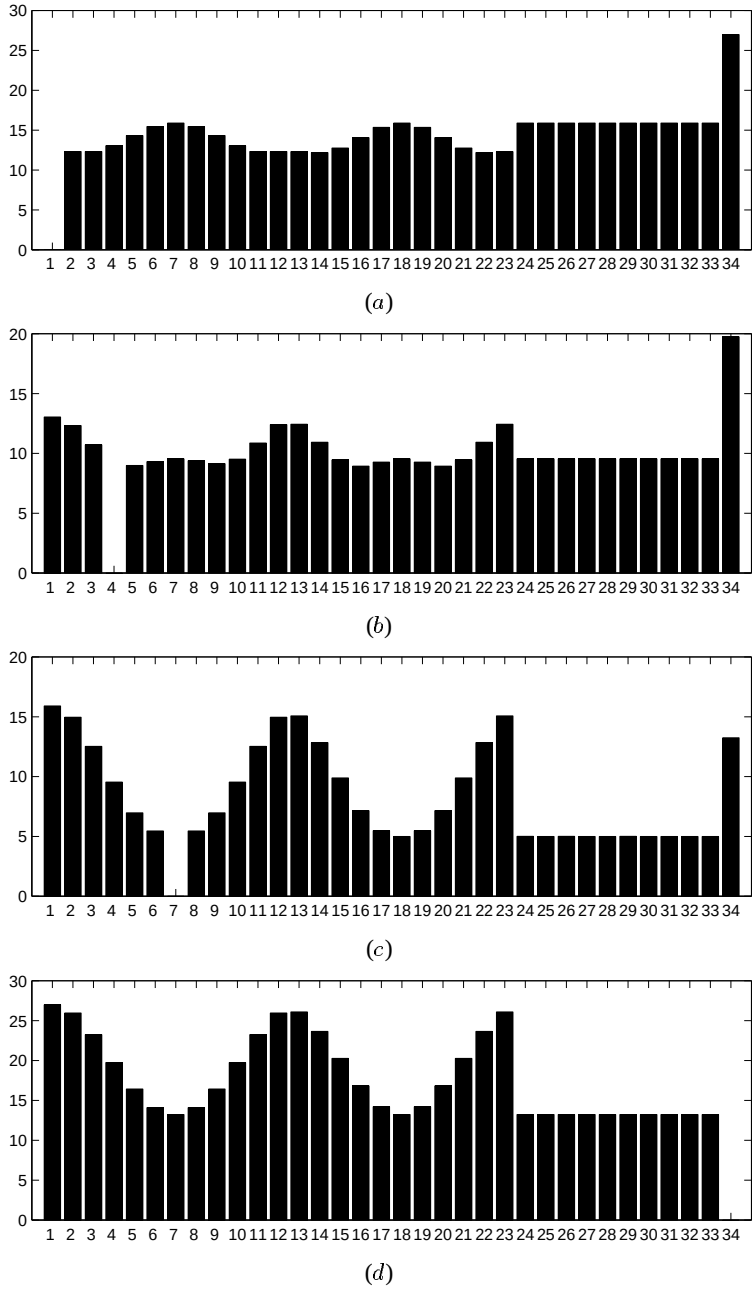


Fig. B.3. The stereo pair close to the object. The uncertainties in predicting the three lines plus the baseline are shown above using each of the other lines to fix the scale: (a) line 1, (b) line 4, (c) line 7, and (d) the baseline.

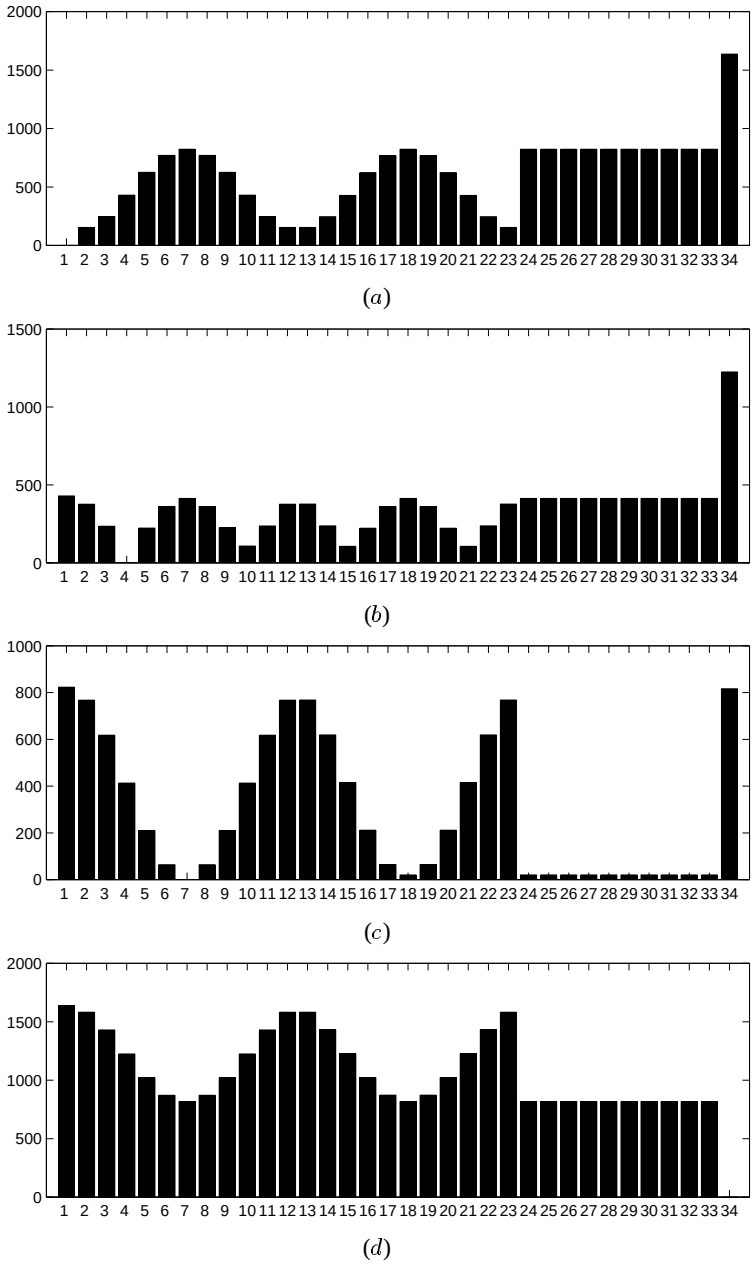


Fig. B.4. The stereo pair far from the object. The uncertainties in predicting the three lines plus the baseline are shown above using each of the other lines to fix the scale: (a) line 1, (b) line 4, (c) line 7, and (d) the baseline.

improves compared to those in Figure 4.15, the patterns remain almost identical. Thus we conclude that our results are not significantly changed when additional features are included in the reconstruction.

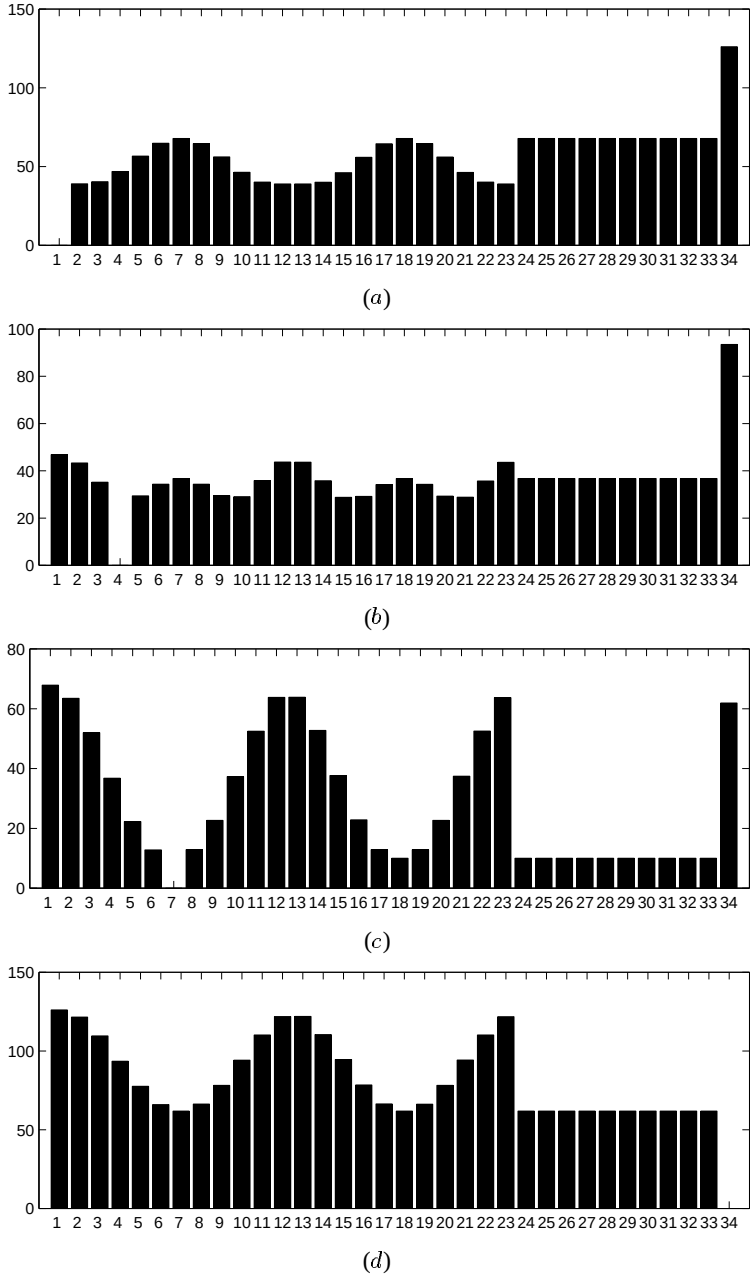


Fig. B.5. Stereo with extra features. The uncertainties in predicting the three lines plus the baseline are shown above using each of the other lines to fix the scale: (a) line 1, (b) line 4, (c) line 7, and (d) the baseline.

Bibliography

1. K. B. Atkinson, *Close Range Photogrammetry and Machine Vision*, Whittles Publishing, Caithness, Scotland, (1996).
2. W. Baarda, *S-Transformations and Criterion Matrices*, vol. Band 5 der Reihe 1, Netherlands Geodetic Commission, (1973).
3. A. Bjerhammar, Några synpunkter påmatriskalkylens användning inom utjämningsräkningen. (Some considerations regarding the application of the calculus of matrices to the method of least squares), *Sven. Lantmäteritidskr*, 6, pp. 460–489, (1948).
4. A. Bjerhammar, *Theory of Errors and Generalized Matrix Inverses*, Elsevier, (1973).
5. D. C. Brown, A solution to the general problem of multiple station analytical stereo-triangulation, *Tech. Rep. AFMTC TR 58-8*, RCA-MTP Data Reduction Technical Report No. 43, Patrick Airforce Base, Florida, (1958).
6. D. C. Brown, The bundle adjustment—progress and prospects, *Int. Archives Photogrammetry*, 21 (3), pp. 637–663, paper number 3–03 (33 pages), (1976).
7. M. A. R. Cooper & P. A. Cross, Statistical concepts and their application in photogrammetry and surveying, *Photogrammetric Record*, 12 (71), pp. 637–663, (1988).
8. M. A. R. Cooper & P. A. Cross, Statistical concepts and their application in photogrammetry and surveying (continued), *Photogrammetric Record*, 13 (77), pp. 645–678, (1991).
9. G. Csurka, C. Zeller, Z. Zhang & O. D. Faugeras, Characterizing the uncertainty of the fundamental matrix, *Computer Vision and Image Understanding: CVIU*, 68 (1), pp. 18–36, (Oct. 1997).
10. K. Daniilidis & M. E. Spetsakis, *Visual Navigation*, Lawrence Erlbaum Associates, Hillsdale, NJ, chapter in book edited by Y. Aloimonos, (1996).

11. A. Dermanis, Free network solutions with the DLT method, *ISPRS Journal of Photogrammetry and Remote Sensing*, 49 (2), pp. 2–12, (1994).
12. A. Dermanis, The photogrammetric inner constraints, *ISPRS Journal of Photogrammetry and Remote Sensing*, 49 (1), pp. 25–39, (1994).
13. H. Ebner, Analysis of covariance matrices, in *Proc. Symp. Int. Soc. of Photogrammetry, Commission III*, Deutsche Geodätische Kommission, Stuttgart, vol. Reihe B, Heft Nr. 214, pp. 111–121, (Sep. 1974).
14. O. Faugeras, Q. Luong & S. Maybank, Camera self-calibration: Theory and experiments, in *Proc. Euro. Conf. Comp. Vision*, Springer-Verlag, pp. 321–334, (1992).
15. O. Faugeras & S. Maybank, What can be seen in three dimensions with an uncalibrated stereo rig?, in *Proc. Euro. Conf. Comp. Vision*, Springer-Verlag, pp. 563–578, (1992).
16. C. F. Gauss, Werke, in *Königlichen Gesellschaft der Wissenschaften zu Göttingen*, (1870–1928).
17. C. F. Gauss, *Theoria Combinationis Observationum Errorbus Minimis Obnoxiae (Theory of the Combination of Observations Least Subject to Errors)*, SIAM Press, Philadelphia, PA, originally published in *Commentatines Societas Regiae Scientarium Gottingensis Recentiores*, 1823, 1828. Translation and commentary by G. W. Stewart, (1995).
18. S. I. Granshaw, Bundle adjustment methods in engineering photogrammetry, *Photogrammetric Record*, 10 (56), pp. 637–663, (1980).
19. R. M. Haralick, Propagating covariance in computer vision, *Int. J. of Patt. Recog. and AI.*, 10 (5), pp. 561–572, (1996).
20. R. I. Hartley, Estimation of relative camera positions for uncalibrated cameras, in *Proc. Euro. Conf. Comp. Vision*, Springer-Verlag, pp. 579–587, (1992).
21. R. I. Hartley, Euclidean reconstruction from multiple views, in *2nd Europe-U.S. Workshop on Invariance*, Azores, Portugal, pp. 237–56, (Oct. 1993).
22. R. I. Hartley, Euclidean reconstruction from uncalibrated views, in *Proc. DARPA–ESPRIT Workshop App. Invariants in Comp. Vision*, Azores, Portugal, pp. 187–202, (Oct. 1993).
23. R. I. Hartley, In defence of the 8-point algorithm, in *Proc. Fifth Int. Conf. Comp. Vision*, Cambridge, MA, USA, pp. 1064–70, (Jun. 1995).
24. R. I. Hartley, An object-oriented approach to scene reconstruction, in *IEEE Conf. Systems, Man & Cybernetics*, Beijing, pp. 2475–2480, (Oct. 1996).

25. R. I. Hartley, Triangulation, *Computer Vision and Image Understanding*, 68 (2), pp. 146–157, (1997).
26. R. I. Hartley, R. Gupta & T. Chang, Stereo from uncalibrated cameras, in *Proc. Comp. Vision Patt. Recog.*, pp. 761–764, (1992).
27. R. I. Hartley & A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, (2000).
28. J. Heikkila & O. Silven, A four-step camera calibration procedure with implicit image correction, in *Proc. Comp. Vision Patt. Recog.*, Peurto Rico, pp. 1106–1112, (Jun. 1997).
29. A. Heyden & K. Astrom, Euclidean reconstruction from image sequences with varying and unknown focal length and principal point, in *Proc. Comp. Vision Patt. Recog.*, Peurto Rico, pp. 438–443, (Jun. 1997).
30. T. Kanade & D. D. Morris, Factorization methods for structure from motion, in *Phil. Trans. R. Soc. Long. A*, pp. 1153–1173, (1998).
31. K. Kanatani, *Group-Theoretical Methods in Image Understanding*, Springer, Berlin, (1990).
32. K. Kanatani, *Geometric Computation for Machine Vision*, Oxford University Press, Oxford, (1993).
33. K. Kanatani, Unbiased estimation and statistical analysis of 3-D rigid motion from two views, *ieeepami*, 15 (1), pp. 37–50, (1993).
34. K. Kanatani, *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier, Amsterdam, (1996).
35. K. Kanatani, Statistical optimization and geometric inference in computer vision, in *Phil. Trans. R. Soc. Long. A*, pp. 1303–1320, (1998).
36. K. Kanatani, Gauge-based reliability analysis of 3-D reconstruction from two uncalibrated perspective views, in *Proc. 15th Int. Conf. Pattern Recog. (ICPR 2000)*, Barcelona, Spain, vol. 1, pp. 76–79, (Sep. 2000).
37. K. Kanatani, Optimal fundamental matrix computation: algorithm and reliability analysis, in *Proceedings of the Sixth Symposium on Sensing via Imaging Information (SSII 2000)*, pp. 291–296, (2000).
38. K. Kanatani & C. Matsunaga, Closed-form expression for focal lengths from the fundamental matrix, in *Proc. Fourth Asian Conf. Comp. Vision*, Taiwan, pp. 128–133, (2000).

39. K. Kanatani & D. D. Morris, Gauges and gauge transformations in 3-D reconstruction from a sequence of images, in *Proc. Fourth Asian Conf. Comp. Vision*, Taiwan, vol. 2, pp. 1046–1051, (2000).
40. K. Kanatani & D. D. Morris, Gauge and gauge transformations for uncertainty description of geometric structure with indeterminacy, *IEEE Transactions on Information Theory*, to appear, (2001).
41. K. Kanatani & N. Ohta, Accuracy bounds and optimal computation of homography for image mosaicing applications, in *Proc. Seventh Int. Conf. Comp. Vision*, (1999).
42. K.-R. Koch, *Parameter Estimation and Hypothesis Testing in Linear Models*, Springer-Verlag, (1987).
43. L. L. Kontsevich, M. L. Kontsevich & A. K. Shen, Two algorithms for reconstructing shapes, *Avtometriya*, (5), pp. 76–81, (Transl. Optoelec., Instrum. Data Proc., no. 5. 76–81, 1987), (1987).
44. A. M. Legendre, *Nouvelles méthodes pour la détermination des orbites des comètes*, Coucier, Paris, appendix on least squares, (1805).
45. H. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections, *Nature*, 293, pp. 133–35, (1981).
46. L. Matties & S. A. Shafer, Error modeling in stereo navigation, *IEEE Journal of Robotics and Automation*, RA-3 (3), pp. 239–48, (Jun. 1987).
47. P. F. McLauchlan, Gauge independence in optimization algorithms for 3D vision, in *Vision Algorithms: Theory and Practice*, Springer, Berlin, pp. 183–199, (2000).
48. P. Meissl, Über die innere genauigkeit dreidimensionaler punkthaufen, *Z. Vermessungswesen*, 90 (4), pp. 109–118, (1965).
49. E. H. Moore, Abstract, *Bull. Amer. Math. Soc*, 26, pp. 394–395, (1920).
50. E. H. Moore, Generalized analysis, *Mem. Amer. Phil. Soc*, 1, (1955).
51. D. D. Morris & T. Kanade, A unified factorization algorithm for points, line segments and planes with uncertainty models, in *Proc. Sixth Int. Conf. Comp. Vision*, Bombay, India, pp. 696–702, (Jan. 1998).
52. D. D. Morris, K. Kanatani & T. Kanade, 3D model accuracy and gauge fixing, *Tech. Rep. CMU-RI-TR-00-32*, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, (Dec. 2000).
53. D. D. Morris, K. Kanatani & T. Kanade, Uncertainty modeling for optimal structure from motion, in *Vision Algorithms: Theory and Practice*, Springer, Berlin, pp. 200–217, (2000).

54. H. B. Papo, Bases of null-space in analytical photogrammetry, *Photogrammetria (PRS)*, 41, pp. 233–245, (1987).

55. R. A. Penrose, A generalized inverse for matrices, in *Proc. Camb. Phil. Soc.*, vol. 51, pp. 406–413, (1955).

56. R. A. Penrose, On best approximate solutions of linear matrix equations, in *Proc. Camb. Phil. Soc.*, vol. 52, pp. 17–19, (1956).

57. C. Poelman & T. Kanade, A paraperspective factorization method for shape and motion recovery, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 19 (3), pp. 206–18, (Mar. 1997).

58. M. Pollefeys, R. Koch & L. Van Gool, Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters, *Int. J. Comp. Vision*, 32 (1), pp. 7–26, (1999).

59. W. Press, S. Teukolsky, W. Vettering & B. Flannery, *Numerical Recipes in C*, Cambridge University Press, (1992).

60. L. Quan & T. Kanade, Affine structure from line correspondances with uncalibrated affine cameras, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 19 (8), pp. 834–45, (1997).

61. C. C. Slama (ed.), *Manual of Photogrammetry*, American Society of Photogrammetry and Remote Sensing, Falls Church, Virginia, USA, (1980).

62. P. Sturm & B. Triggs, A factorization based algorithm for multi-image projective structure and motion, in *Proc. Euro. Conf. Comp. Vision*, Cambridge, UK, pp. 709–20, (1996).

63. Z. Sun, V. Ramesh & A. M. Tekalp, Error characterization of the factorization approach to shape and motion recovery, in *Vision Algorithms: Theory and Practice*, Springer, Berlin, pp. 218–235, (2000).

64. R. Szeliski & S. B. Kang, Shape ambiguities in structure from motion, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 19 (5), pp. 506–512, (May 1997).

65. C. J. Taylor, Structure and motion from line segments in multiple images, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 17 (11), pp. 1021–32, (Nov. 1995).

66. P. Teunissen, Zero order design: Generalized inverses, adjustment, the datum problem and s-transformations, in *Optimization and Design of Geodetic Networks*, Springer-Verlag, Berlin, pp. 11–55, chap. 1 in book edited by E. W. Garafarend and F. Sanso, (1985).

67. J. I. Thomas, A. Hanson & J. Oliensis, Refining 3D reconstructions: A theoretical and experimental study of the effect of cross-correlations, *CVGIP*, 60 (3), pp. 359–370, (Nov. 1994).
68. C. Tomasi & T. Kanade, Shape and motion from image streams under orthography: a factorization method, *Int. J. Comp. Vision*, 9 (2), pp. 137–54, (Nov. 1992).
69. P. Torr & A. Zisserman, Robust computation and parametrization of multiple view relations, in *Proc. Sixth Int. Conf. Comp. Vision*, pp. 727–732, (1998).
70. B. Triggs, Optimal estimation of matching constraint, in *3D Structure from Multiple Images of Large-Scale Environments*, Lecture Notes in Computer Science, 1506, Springer, pp. 63–77, (1998).
71. B. Triggs, P. F. McLauchlan, R. I. Hartley & A. W. Fitzibbon, Bundle adjustment – a modern synthesis, in *Vision Algorithms: Theory and Practice*, Springer, Berlin, pp. 298–375, (2000).
72. R. Y. Tsai, An efficient and accurate camera calibration technique for 3D machine vision, in *Proc. Comp. Vision Patt. Recog.*, pp. 364–374, (1986).
73. J. Weng, N. Ahuja & T. Huang, Optimal motion and structure estimation, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 15 (9), pp. 864–884, (1993).
74. J. Weng, T. Huang & N. Ahuja, Motion and structure from two perspective views: Algorithms, error analysis, and error estimation, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 11 (2), pp. 451–476, (1989).
75. G.-S. J. Young & R. Chellappa, Statistical analysis of inherent ambiguities in recovering 3-D motion from a noisy flow field, *IEEE Trans. Patt. Anal. Mach. Intelligence*, 14 (10), pp. 995–1013, (1992).
76. Z. Zhang, Parameter estimation techniques: A tutorial with application to conic fitting, *International Journal of Image and Vision Computing*, 15 (1), pp. 59–76, (1997).