International Journal of Computer Vision 44(3), 219–249, 2001

Optimization Criteria and Geometric Algorithms for Motion and Structure Estimation*

YI MA

Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 1406 West Green Street, Urbana, IL 61801, USA

yima@uiuc.edu

JANA KOŠECKÁ

Department of Computer Science, George Mason University, 4400 University Drive #MS4A5, Fairfax, VA 22030, USA

kosecka@cs.gmu.edu

SHANKAR SASTRY

Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720,USA

sastry@eecs.berkeley.edu

Received May 21, 1999; Revised June 8, 2001; Accepted June 11, 2001

Abstract. Prevailing efforts to study the standard formulation of motion and structure recovery have recently been focused on issues of sensitivity and robustness of existing techniques. While many cogent observations have been made and verified experimentally, many statements do not hold in general settings and make a comparison of existing techniques difficult. With an ultimate goal of clarifying these issues, we study the main aspects of motion and structure recovery: the choice of objective function, optimization techniques and sensitivity and robustness issues in the presence of noise.

We clearly reveal the relationship among different objective functions, such as "(normalized) epipolar constraints," "reprojection error" or "triangulation," all of which can be unified in a new "optimal triangulation" procedure. Regardless of various choices of the objective function, the optimization problems all inherit the same unknown parameter space, the so-called "essential manifold." Based on recent developments of optimization techniques on Riemannian manifolds, in particular on Stiefel or Grassmann manifolds, we propose a Riemannian Newton algorithm to solve the motion and structure recovery problem, making use of the natural differential geometric structure of the essential manifold.

We provide a clear account of sensitivity and robustness of the proposed linear and nonlinear optimization techniques and study the analytical and practical equivalence of different objective functions. The geometric characterization of critical points and the simulation results clarify the difference between the effect of bas-relief ambiguity, rotation and translation confounding and other types of local minima. This leads to consistent interpretations of simulation results over a large range of signal-to-noise ratio and variety of configurations.

*This work is supported by ARO under the MURI grant DAAH04-96-1-0341 at University of California Berkeley.

Keywords: motion and structure recovery, optimal triangulation, essential manifold, Riemannian Newton's algorithm, Stiefel manifold

1. Introduction

The problem of recovering structure and motion from a sequence of images has been one of the central problems in Computer Vision over the past decade and has been studied extensively from various perspectives. The proposed techniques have varied depending on the type of features they used, the types of assumptions they made about the environment and projection models and the type of algorithms. Based on image measurements the techniques can be viewed either as discrete: using point or line features, or differential: using measurements of optical flow. While the geometric relationships governing the motion and structure recovery problem have been long understood, the robust solutions are still sought. New studies of the sensitivity of different algorithms, search for intrinsic local minima and new algorithms are still subject of great interest. Algebraic manipulation of intrinsic geometric relationships typically gives rise to different objective functions, making the comparison of the performance of different techniques difficult and often obstructing issues intrinsic to the problem. In this paper, we provide new algorithms and insights by giving answers to the following three questions, which we believe are the main aspects of the motion and structure recovery problem (in the simplified two-view, point-feature scenario):

(i) What is the correct choice of the objective function and its associated statistical and geometric meaning? What are the fundamental relationships among different existing objective functions from an estimation theoretic viewpoint?

(ii) What is the core optimization problem which is common to all objective functions associated with motion and structure estimation? We propose a new intrinsic (i.e., independent of any particular parameterization of the search space) optimization scheme which goes along with this problem.

(iii) Using extensive simulations, we show how the choice of the objective functions and configurations affects the sensitivity and robustness of the estimates. We also reveal the effect of the bas-relief ambiguity and other ambiguities on the sensitivity and robustness of the proposed algorithms. The nonlinear algorithms are initialized using linear algorithms.

The seminal work of Longuet-Higgins (1981) on the characterization of the so-called *epipolar constraint*, enabled the decoupling of the structure and motion problems and led to the development of numerous linear and nonlinear algorithms for motion estimation (see Maybank, 1993; Faugeras, 1993; Kanatani, 1993; Weng et al., 1993a for overviews). The epipolar constraint has been formulated both in a discrete and a differential setting and our recent work (Ma et al., 2000) has demonstrated the possibility of a parallel development of linear algorithms for both cases: namely using point feature correspondences and optical flow. The original 8-point algorithm proposed by Longuet-Higgins is easily generalizable to the uncalibrated camera case, where the epipolar constraint is captured by the so-called fundamental matrix. Detailed analysis of linear and nonlinear techniques for estimation of fundamental matrix exploring the use of different objective functions can be found in Luong and Faugeras (1996).

While the (analytic) geometrical aspects of the linear approach have been understood, the proposed solutions to the problem have been shown to be sensitive to noise and have often failed in practical applications. These experiences have motivated further studies which focus on the use of a statistical analysis of existing techniques and understanding of various assumptions which affect the performance of existing algorithms. These studies have been done both in an analytical (Danilidis, 1997; Spetsakis, 1994) and experimental setting (Tian et al., 1996). The appeal of linear algorithms which use the epipolar constraint (in the discrete case (Weng, et al., 1993a; Kanatani, 1993; Longuet-Higgins, 1981; Maybank, 1993) and in the differential case (Jepson and Heeger, 1993; Ma et al., 2000; Thomas and Simoncelli, 1995)) is the closed form solution to the problem which, in the absence of noise, provides a true estimate of the motion. However, a deeper analysis of linear techniques reveals an inherent bias in the translation estimates (Jepson and Heeger, 1993). Attempts made to compensate for the bias slightly improve the performance of the linear techniques (Kanatani, 1993).

The attempts to remove bias have led to different choices of nonlinear objective functions. The performance of numerical optimization techniques which minimize nonlinear objective functions has been shown superior to linear ones. The objective functions used are either (normalized) versions of the epipolar constraint or distances between measured and reconstructed image points (the so-called reprojection error) (Weng et al., 1993b; Luong and Faugeras, 1996; Zhang, 1998; Horn, 1990). These techniques either require iterative numerical optimization (Weng et al., 1993a; Soatto and Brockett, 1998) or use Monte-Carlo simulations (Jepson and Heeger, 1993) to sample the space of the unknown parameters. Extensive experiments reveal problems with convergence when initialized far away from the true solution (Tian et al., 1996). Since nonlinear objective functions have been obtained from quite different approaches, it is necessary to understand the relationship among the existing objective functions. Although a preliminary comparison has been made in Zhang (1998), in this paper, we provide a more detailed and rigorous account of this relationship and how it affects the complexity of the optimization. In this paper, we will show, by answering the question (i), that "minimizing epipolar constraint," "minimizing (geometrically or statistically[1]) normalized epipolar constraint" (Weng et al., 1993b; Luong and Faugeras, 1996; Zhang, 1998), "minimizing reprojection error" (Weng et al., 1993b), and "triangulation" (Hartley and Sturm, 1997) can all be unified in a single geometric optimization procedure, the so-called "optimal triangulation." As a by-product of this approach, a simpler triangulation method than (Hartley and Sturm, 1997) is given along with the proposed algorithm. A highlight of our method is an optimization scheme which iterates between motion and structure estimates without introducing any 3D scale (or depth).

Different objective functions have been used in different optimization techniques (Horn, 1990; Weng et al., 1993b; Taylor and Kriegman, 1995). Horn (1990) first proposed an iterative procedure where the update of the estimate takes into account the orthonormal constraint of the unknown rotation. This algorithm and the algorithm proposed in Taylor and Kriegman (1995) are examples of the few which explicitly consider the differential geometric properties of the rotation group $SO(3)$. In most cases, the underlying search space has been parameterized for computational convenience instead of being loyal to its intrinsic geometric structure. Consequently, in these algorithms, solving for optimal updating direction typically involves using Lagrangian multipliers to deal with the constraints on the search space. "Walking" on such a space is done approximately by an *update-then-project* procedure rather than exploiting geometric properties of the entire space of essential matrices as characterized in our recent paper (Ma et al., 2000) or in Soatto and Brockett (1998). As an answer to the question (ii), we will show that optimizing existing objective functions can all be reduced to optimization problems on the essential manifold. Due to recent developments of optimization techniques on Riemannian manifolds (especially on Lie groups and homogeneous spaces) (Smith, 1993; Edelman et al., to appear), we are able to explicitly compute all the necessary ingredients, such as *gradient*, *Hessian and geodesics*, for carrying out intrinsic nonlinear search schemes. In this paper, we will first give a review of the nonlinear optimization problem associated with the motion and structure recovery. Using a generalized Newton's algorithm as a prototype example, we will apply our methods to solve the optimal motion and structure estimation problem by exploiting the intrinsic Riemannian structure of the essential manifold. The rate of convergence of the algorithm is also studied in some detail. We believe the proposed geometric algorithm will provide us with an analytic framework for design of (Kalman) filters on the essential manifold for dynamic motion estimation (see Soatto and Perona, 1996). The algorithm also provides new perspectives for design of algorithms for multiple views.

In this paper, only the discrete case will be studied, since in the differential case the search space is essentially Euclidean and good optimization schemes already exist and have been well studied (see Soatto and Brockett, 1998; Zhang and Tomasi, 1999). For the differential case, recent studies (Soatto and Brockett, 1998) have clarified the source of some of the difficulties (for example, rotation and translation confounding) from the point of view of noise and explored the source and presence of local extrema which are intrinsic to the structure from motion problem. The most sensitive direction in which the rotation and translation estimates are prone to be confound with each other is demonstrated as a bas-relief ambiguity (for additional details see Adiv, 1989; Weng et al., 1993b; Soatto and Brockett, 1998). Here we apply the same line of thought to the discrete case. In addition to the bas-relief effect which is evident only when the field of view and the depth variation of the scene are small, we will also characterize other intrinsic extrema which occur at a high noise level even for a general configuration, where a base line, field of view and depth variation are all large. As an answer to the question (iii), we will show both analytically and experimentally that some ambiguities are introduced at a high noise level by bifurcations of local minima of the objective function and usually result in a sudden 90° flip in the translation estimate. Understanding such ambiguities is crucial for properly evaluating the performance (especially the robustness) of the algorithms when applied to general configurations. Based on analytical and experimental results, we will give a clear profile of the performance of different algorithms over a large range of signal-to-noise ratio, or under various motion and structure configurations.

Paper outline: Section 2 focuses on motion recovery from epipolar constraint and introduces Newton's algorithm for optimizing various objective functions associated with the epipolar constraint. Section 3 introduces a new optimal triangulation method, which is a single optimization procedure designed for estimating optimal motion and structure together. The objective function and optimization procedure proposed here unifies existing objective functions previously proposed in the literature and gives clear answers to both questions (i) and (ii). Section 4 gives a geometric characterization of extrema of any function on the essential manifold and demonstrates bifurcations of some local minima if the algorithm is initialized using the linear techniques. Sensitivity study and experimental comparison between different objective functions are given in Section 5. Sections 4 and 5 give a detailed account of the question (iii). Although this paper introduces the concept of optimization on Riemannian manifolds to the structure and motion recovery problem, background in Riemannian geometry is not truly required. Some familiarity with Edelman et al.'s work on optimization on Stiefel manifolds (Edelman et al., to appear) and some background in Riemannian geometry (Spivak, 1979; Kobayashi and Nomizu, 1996) may improve the understanding of the material. For interested readers, Appendix A and B provide more detailed discussions on these subjects.

2. Motion from Epipolar Constraint

The purpose of this section is to introduce the optimization problem of recovery of camera motion and 3D structure from image correspondences. We first emphasize the importance of proper characterization of the underlying parameter space for this problem and in a simplified setting outline a new Riemannian optimization scheme for solving the nonlinear optimization.

Newton's and conjugate gradient methods are classical nonlinear optimization techniques to minimize a function $f(x)$, where x belongs to an open subset of Euclidean space $\mathbb{R}^n$. Recent developments in optimization algorithms on Riemannian manifolds have provided geometric insights for generalizing Newton's and conjugate gradient methods to certain classes of Riemannian manifolds. Smith (1993) gave a detailed treatment of a theory of optimization on general Riemannian manifolds; Edelman et al. (to appear) further studied the case of Stiefel and Grassmann manifolds,[2] and presented a unified geometric framework for applying Newton and conjugate gradient algorithms on these manifolds. These new mathematical schemes solve the more general optimization problem of minimizing a function $f(x)$, where x belongs to some Riemannian manifold (M, g), where $g : TM \times TM \rightarrow C^\infty(M)$ is the Riemannian metric on M (and TM denotes the tangent space of M). An intuitive comparison between the Euclidean and Riemannian nonlinear optimization schemes is illustrated in Fig. 1.

Conventional approaches for solving such an optimization problem are usually application-dependent (or parameterization-dependent). The manifold M is first embedded as a submanifold into a higher dimensional Euclidean space $\mathbb{R}^N$ by choosing certain (global or local) *parameterization* of M. *Lagrangian multipliers* are often used to incorporate additional constraints that these parameters should satisfy. In order for x to always stay on the manifold, after each update it needs to be *projected* back onto the manifold M. However,

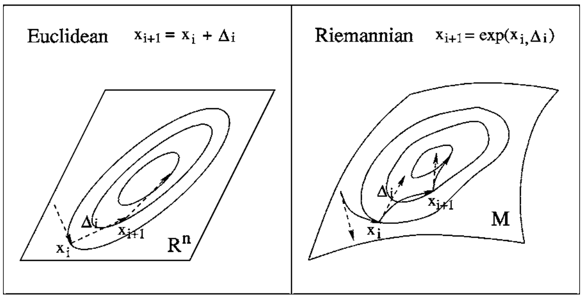

Figure 1. Comparison between the Euclidean and Riemannian nonlinear optimization schemes. At each step, an (optimal) updating vector $\Delta_i \in T_{x_i}M$ is computed using the Riemannian metric at x_i. Then the state variable is updated by following the geodesic from x_i in the direction Δ_i by a distance of $\sqrt{g(\Delta_i, \Delta_i)}$ (the Riemannian norm of Δ_i). This geodesic is denoted in Riemannian geometry by the *exponential map* $\exp(x_i, \Delta_i)$.

the new analysis of Edelman et al. (to appear) shows that, for "nice" manifolds, for example Lie groups, or homogeneous spaces such as Stiefel and Grassmann manifolds, one can make use of the *canonical* Riemannian structure of these manifolds and systematically develop a Riemannian version of the Newton's algorithm or conjugate gradient methods for optimizing a function defined on them. Since the parameterization and metrics are canonical and the state is updated using geodesics (therefore always staying on the manifold), the performance of these algorithms is no longer parameterization dependent, and in addition they typically have polynomial complexity and super-linear (quadratic) rate of convergence (Smith, 1993). An intuitive comparison between the conventional update-then-project approach and the Riemannian method is demonstrated in Fig. 2 (where M is illustrated as the standard 2D sphere $\mathbb{S}^2 = \{x \in \mathbb{R}^3 \mid \| x \|^2 = 1\}$).

As we will soon see, the underlying Riemannian manifold for this problem, the so-called essential manifold, is a *product* of Stiefel manifolds. Appendix A demonstrates how to extend such optimization schemes to a product of Riemannian manifolds in general.

2.1. Riemannian Structure of the Essential Manifold

The key towards characterization of the Riemannian structure of the underlying parameter space of structure and motion estimation problem is the concept of epipolar geometry, in particular the so-called *essential manifold* (for details see Ma et al., 2000). Camera motion is modeled as rigid body motion in $\mathbb{R}^3$. The displacement of the camera belongs to the special Euclidean group $SE(3)$:

$$SE(3) = \{(R, T) : T \in \mathbb{R}^3, R \in SO(3)\} \tag{1}$$

where $SO(3) \in \mathbb{R}^{3\times 3}$ is the space of rotation matrices (orthogonal matrices with determinant $+1$). An element $g = (R, T)$ in this group is used to represent the coordinate transformation of a point in $\mathbb{R}^3$. Denote the coordinates of the point before and after the transformation as $\mathbf{X}_1 = [X_1, Y_1, Z_1]^T \in \mathbb{R}^3$ and $\mathbf{X}_2 = [X_2, Y_2, Z_2]^T \in \mathbb{R}^3$ respectively. Then, $\mathbf{X}_1$ and $\mathbf{X}_2$ are associated by:

$$\mathbf{X}_2 = R\mathbf{X}_1 + T. \tag{2}$$

The image coordinates of $\mathbf{X}_1$ and $\mathbf{X}_2$ are given by $\mathbf{x}_1 = [\frac{X_1}{Z_1}, \frac{Y_1}{Z_1}, 1]^T \in \mathbb{R}^3$ and $\mathbf{x}_2 = [\frac{X_2}{Z_2}, \frac{Y_2}{Z_2}, 1]^T \in \mathbb{R}^3$ respectively.[3] The rigid body motion described in terms of image coordinates then becomes:

$$\lambda_2\mathbf{x}_2 = R\lambda_1\mathbf{x}_1 + T. \tag{3}$$

where λ_1 and λ_2 are the unknown scales (depths) of points $\mathbf{x}_1$ and $\mathbf{x}_2$.

The main purpose of this paper is to revisit the following classic problem of structure and motion recovery:

> **Motion and Structure Recovery Problem:** *For a given set of corresponding image points* $\{(\mathbf{x}_1^i, \mathbf{x}_2^i)\}_{i=1}^N$, *recover the camera motion* (R, T) *and the* 3*D coordinates* (3*D structure*) *of the points that these image points correspond to.*

It is well known in Computer Vision literature that two corresponding image points $\mathbf{x}_1$ and $\mathbf{x}_2$ satisfy the so-called *epipolar constraint* (Longuet-Higgins, 1981):

$$\mathbf{x}_2^T \hat{T} R \mathbf{x}_1 = 0. \tag{4}$$

This intrinsic constraint is independent of depth information and hence decouples the problem of motion recovery from 3D structure.[4] The following section is devoted to recovery of motion (R, T) using directly using this constraint and its variations. In Section 3, we will see how this constraint has to be modified when we consider recovering (optimal) motion and structure simultaneously.

The matrix $\hat{T}R$ in the epipolar constraint is the so-called *essential matrix*, and the *essential manifold* is

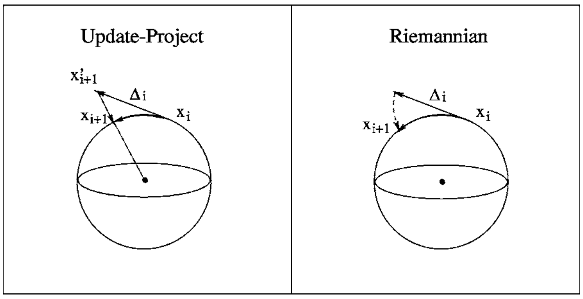

Figure 2. Comparison between the conventional update-then-project approach and the Riemannian scheme. For the conventional method, the state x_i is first updated to x'_{i+1} according to the updating vector Δ_i and then x'_{i+1} is projected back to the manifold at x_{i+1}. For the Riemannian scheme, the new state x_{i+1} is obtained by following the geodesic, i.e., $x_{i+1} = \exp(x_i, \Delta_i)$.

defined to be the space of all such matrices, denoted by:

$$\mathcal{E} = \{\hat{T}R \mid R \in SO(3), \hat{T} \in so(3)\}.$$

$SO(3)$ is a Lie group of 3×3 rotation matrices, and $so(3)$ is the Lie algebra of $SO(3)$, i.e., the tangent plane of $SO(3)$ at the identity. $so(3)$ then consists of all 3×3 skew-symmetric matrices and $\hat{T} \in so(3)$. As we will show later in this paper, for the problem of recovering camera motion (R, T) from the corresponding image points $\mathbf{x}_1$ and $\mathbf{x}_2$, the associated objective functions are usually functions of the epipolar constraint. Hence they are of the form $f(E) \in \mathbb{R}$ with $E \in \mathcal{E}$. Furthermore such functions in general are homogeneous in E. Thus the problem of motion recovery is formulated as optimization of functions defined on the so-called *normalized essential manifold*:

$$\mathcal{E}_1 = \left\{\hat{T}R \mid R \in SO(3), \hat{T} \in so(3), \frac{1}{2}tr(\hat{T}^T\hat{T}) = 1\right\}.$$

Note that $\frac{1}{2}tr(\hat{T}^T\hat{T}) = T^T T$. The exact forms of $f(E)$ will be derived from statistical and geometric considerations in Sections 2.3 and 3 of this paper.

In order to study the optimization problem on the normalized essential manifold it is crucial to understand its Riemannian structure. We start with the Riemannian structure on the tangent bundle of the Lie group $SO(3)$, i.e., $T(SO(3))$. The tangent space of $SO(3)$ at the identity e is simply its Lie algebra $so(3)$:

$$T_e(SO(3)) = SO(3).$$

Since $SO(3)$ is a compact Lie group, it has an intrinsic bi-invariant metric (Boothby, 1986) (such metric is unique up to a constant scale). In matrix form, this metric is given explicitly by:

$$g_0(\hat{T}_1, \hat{T}_2) = \frac{1}{2}tr(\hat{T}_1^T\hat{T}_2), \quad \hat{T}_1, \hat{T}_2 \in SO(3).$$

where $tr(A)$ refers to the trace of the matrix A. Notice that this metric is induced from the Euclidean metric on $SO(3)$ as a Stiefel submanifold embedded in $\mathbb{R}^{3\times 3}$. The tangent space at any other point $R \in SO(3)$ is given by the push-forward map R_*:

$$T_R(SO(3)) = R_*(so(3)) = \{\hat{T}R \mid \hat{T} \in so(3)\}.$$

Thus the tangent bundle of $SO(3)$is:

$$T(SO(3)) = \bigcup_{R \in SO(3)} T_R(SO(3))$$

The tangent bundle of a Lie group is trivial (Spivak, 1979) that is, $T(SO(3))$ is equivalent to the product $SO(3) \times so(3)$. $T(SO(3))$ can then be expressed as:

$$\begin{aligned} T(SO(3)) &= \{(R, \hat{T}R) \mid R \in SO(3), \hat{T} \in so(3)\} \\ &\cong SO(3) \times so(3). \end{aligned}$$

The tangent space of $SO(3)$ is $\mathbb{R}^3$ and $SO(3)$ itself is parameterized by $\mathbb{R}^3$. Hence we will use the same notation for $SO(3)$ and its tangent space. Consequently the metric g_0 of $SO(3)$ induces a canonical metric on the tangent bundle $T(SO(3))$:

$$\begin{aligned} \tilde{g}(X, Y) &= g_0(X_1, X_2) \\ &\quad + g_0(Y_1, Y_2), \; X, Y \in so(3) \times so(3). \end{aligned}$$

Note that the metric defined on the fiber $SO(3)$ of $T(SO(3))$ is the same as the Euclidean metric if we identify $so(3)$ with $\mathbb{R}^3$. Such an induced metric on $T(SO(3))$ is left-invariant under the action of $SO(3)$. Then the metric $\tilde{g}$ on the whole tangent bundle $T(SO(3))$ induces a canonical metric g on the unit tangent bundle of $T(SO(3))$:

$$\begin{aligned} T_1(SO(3)) \cong \Bigg\{&(R, \hat{T}R) \mid R \in SO(3), \hat{T} \\ &\in so(3), \frac{1}{2}tr(\hat{T}^T\hat{T}) = 1\Bigg\}. \end{aligned}$$

It is direct to check that with the identification of $SO(3)$ with $\mathbb{R}^3$, the unit tangent bundle is simply the product $SO(3) \times \mathbb{S}^2$ where $\mathbb{S}^2$ is the standard 2-sphere embedded in $\mathbb{R}^3$. According to Edelman et al. (to appear), $SO(3)$ and $\mathbb{S}^2$ both are Stiefel manifolds $V(n, k)$ of the type $n = k = 3$ and $n = 3, k = 1$, respectively. As Stiefel manifolds, they both possess canonical metrics by viewing them as quotients between orthogonal groups. Here $SO(3) = O(3)/O(0)$ and $\mathbb{S}^2 = O(3)/O(2)$. Fortunately, for Stiefel manifolds of the special type $k = n$ or $k = 1$, the canonical metrics are the same as the Euclidean metrics induced as submanifold embedded in $\mathbb{R}^{n\times k}$. From the above discussion, we have

Theorem 1. *The unit tangent bundle $T_1(SO(3))$ is equivalent to $SO(3) \times \mathbb{S}^2$. Its Riemannian metric g*

induced from the bi-invariant metric on $SO(3)$ is the same as that induced from the Euclidean metric with $T_1(SO(3))$ naturally embedded in $\mathbb{R}^{3\times 4}$. Further, $(T_1(SO(3)), g)$ is the product Riemannian manifold of $(SO(3), g_1)$ and $(\mathbb{S}^2, g_2)$ with g_1 and g_2 canonical metrics for $SO(3)$ and $\mathbb{S}^2$ as Stiefel manifolds.

It is well known that there are two pairs of rotation and translation, which correspond to the same essential matrix. Hence the unit tangent bundle $T_1(SO(3))$ is not exactly the normalized essential manifold $\mathcal{E}_1$. It is a double covering of the normalized essential space $\mathcal{E}_1$, i.e., $\mathcal{E}_1 = T_1(SO(3))/\mathbb{Z}^2$ (for details see Ma et al., 2000). The natural covering map from $T_1(SO(3))$ to $\mathcal{E}_1$ is:

$$\begin{aligned} h : T_1(SO(3)) &\rightarrow \mathcal{E}_1 \\ (R, \hat{T}R) \in T_1(SO(3)) &\mapsto \hat{T}R \in \mathcal{E}_1. \end{aligned}$$

The inverse of this map is given by:

$$h^{-1}(\hat{T}R) = \{(R, \hat{T}R),\ (\exp(-\hat{T}\pi)R, \hat{T}R)\}.$$

This double covering h is equivalent to identifying a left-invariant vector field on $SO(3)$ with the one obtained by flowing it along the corresponding geodesic by distance π, the so-called time-π map of the geodesic flow on $SO(3)$.[5]

If we take for $\mathcal{E}_1$ the Riemannian structure induced from the covering map h, the original optimization problem of optimizing $f(E)$ on $\mathcal{E}_1$ is converted to optimizing $f(R, T)$ on $T_1(SO(3))$.[6] Generalizing Edelman et al.'s methods to the product Riemannian manifolds, we may obtain intrinsic Riemannian Newton's or conjugate gradient algorithms for solving such an optimization problem. Appendix B summarizes the Riemannian Newton's algorithm for minimizing a general function defined on the essential manifold. Due to Theorem 1, we can simply choose the induced Euclidean metric on $T_1(SO(3))$ and explicitly implement these intrinsic algorithms in terms of the matrix representation of $T_1(SO(3))$. Since this Euclidean metric is the same as the intrinsic one, the apparently extrinsic representation preserves all intrinsic geometric properties of the given optimization problem. In this sense, the algorithms we are about to develop for the motion recovery are different from other existing algorithms which make use of particular parameterizations of the underlying search manifold $T_1(SO(3))$.

2.2. *Minimizing Epipolar Constraint*

In this section, we demonstrate in a *simplified* case how to derive the Riemannian Newton's algorithm for solving the the motion recovery problem from a given set of image correspondences $\mathbf{x}_1^i, \mathbf{x}_2^i \in \mathbb{R}^3, i = 1, \ldots, N$. We here consider a *naive* objective function associated directly with the epipolar constraint:

$$F(R, T) = \sum_{i=1}^{N} \left(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2, \qquad \mathbf{x}_1^i, \mathbf{x}_2^i \in \mathbb{R}^3,\ (R, T) \in SO(3) \times \mathbb{S}^2. \quad (5)$$

We will give explicit formulae for calculating all the ingredients needed for the Newton's algorithm: geodesics, gradient G, hessian Hess F and the optimal updating vector $\Delta = -\text{Hess}^{-1}G$. It is well known that such an explicit formula for the Hessian is also important for sensitivity analysis of the algorithm (Danilidis, 1997). Furthermore, using these formulae, we will be able to show that, under certain conditions, the Hessian is guaranteed non-degenerate, hence the Newton's algorithm is guaranteed to have a *quadratic* rate of convergence.

One must notice that the objective function $F(R, T)$ given in (5), although simple, is not yet well justified. That is, it may not be related to a meaningful geometric or statistical error measure. In the next Section 2.3, we will then discuss how to modify the function $F(R, T)$ to ones which do take into account proper geometric and statistical error measures. However, this by no means diminishes the significance of the study of the simplified objective function $F(R, T)$. As we will see, the epipolar term "$(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i)^2$" is the basic module component of all the objective functions to be introduced. Hence the computation of gradient and Hessian of those functions essentially can be reduced to this basic case.[7]

Here we present the computation of the gradient and Hessian by using explicit formulae of geodesics. The general formulae are given in the Appendix B. On $SO(3)$, the formula for the geodesic at R in the direction $\Delta_1 \in T_R(SO(3)) = R_*(SO(3))$ is:

$$\begin{aligned} R(t) &= \exp(R, \Delta_1 t) \\ &= R \exp \hat{\omega} t \\ &= R(I + \hat{\omega} \sin t + \hat{\omega}^2(1 - \cos t)) \end{aligned} \quad (6)$$

where $t \in \mathbb{R}$, $\hat{\omega} = \Delta_1 R^T \in so(3)$. The last equation is called the *Rodrigues' formula* (Murray et al., 1994). $\mathbb{S}^2$ (as a Stiefel manifold) also has a very simple expression for geodesics. At the point T along the direction $\Delta_2 \in T_T(\mathbb{S}^2)$ the geodesic is given by:

$$T(t) = \exp(T, \Delta_2 t) = T \cos\sigma t + U \sin\sigma t \tag{7}$$

where $\sigma = ||\Delta_2||$ and $U = \Delta_2/\sigma$, then $T^T U = 0$ since $T^T\Delta_2 = 0$.

Using the formulae (6) and (7) for geodesics , we can calculate the first and second derivatives of $F(R, T)$ in the direction $\Delta = (\Delta_1, \Delta_2) \in T_R(SO(3)) \times T_T(\mathbb{S}^2)$:

$$\begin{aligned} dF(\Delta) &= \left.\frac{dF(R(t), T(t))}{dt}\right|_{t=0} \\ &= \sum_{i=1}^{N} \mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\big(\mathbf{x}_2^{iT}\hat{T}\Delta_1\mathbf{x}_1^i + \mathbf{x}_2^{iT}\hat{\Delta}_2 R\mathbf{x}_1^i\big), \end{aligned} \tag{8}$$

$$\begin{aligned} \text{Hess}\, F(\Delta, \Delta) &= \left.\frac{d^2F(R(t), T(t))}{dt^2}\right|_{t=0} \\ &= \sum_{i=1}^{N} \big[\mathbf{x}_2^{iT}(\hat{T}\Delta_1 + \hat{\Delta}_2 R)\mathbf{x}_1^i\big]^2 \\ &\quad + \mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\big[\mathbf{x}_2^{iT}(-\hat{T}R\Delta_1^T\Delta_1 \\ &\quad - \hat{T}R\Delta_2^T\Delta_2 + 2\hat{\Delta}_2\Delta_1)\mathbf{x}_1^i\big]. \end{aligned} \tag{9}$$

From the first order derivative, the gradient $G = (G_1, G_2) \in T_R(SO(3)) \times T_T(\mathbb{S}^2)$ of $F(R, T)$ is:

$$\begin{aligned} G = \sum_{i=1}^{N} &\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\big(\hat{T}^T\mathbf{x}_2^i\mathbf{x}_1^{iT} - R\mathbf{x}_1^i\mathbf{x}_2^{iT}\hat{T}R, \\ &- \hat{\mathbf{x}}_2^i R\mathbf{x}_1^i - T\mathbf{x}_1^{iT}R^T\hat{\mathbf{x}}_2^i T\big) \end{aligned} \tag{10}$$

It is direct to check that $G_1R^T \in so(3)$ and $T^TG_2 = 0$, so that the G given by the above expression is a vector in $T_R(SO(3)) \times T_T(\mathbb{S}^2)$.

For any pair of vectors $X, Y \in T_R(SO(3)) \times T_T(\mathbb{S}^2)$, we may polarize[8] Hess $F(\Delta, \Delta)$ to get the expression for Hess $F(X, Y)$:

$$\text{Hess}\, F(X, Y) \tag{11}$$

$$\begin{aligned} &= \frac{1}{4}[\text{Hess}\, F(X+Y, X+Y) - \text{Hess}\, F(X-Y, X-Y)] \\ &= \sum_{i=1}^{N} \mathbf{x}_2^{iT}(\hat{T}X_1 + \hat{X}_2R)\mathbf{x}_1^i\mathbf{x}_2^{iT}(\hat{T}Y_1 + \hat{Y}_2R)\mathbf{x}_1^i \\ &\quad + \mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\left[\mathbf{x}_2^{iT}\left(-\frac{1}{2}\hat{T}R\big(X_1^TY_1 + Y_1^TX_1\big)\right.\right. \\ &\quad \left.\left. - \hat{T}RX_2^TY_2 + (\hat{Y}_2X_1 + \hat{X}_2Y_1)\right)\mathbf{x}_1^i\right]. \end{aligned} \tag{12}$$

To make sure this expression is correct, if we let $X = Y = \Delta$, then we get the same expression for Hess$F(\Delta, \Delta)$ as that obtained directly from the second order derivative. The following theorem shows that this Hessian is non-degenerate in a neighborhood of the optimal solution, therefore the Newton's algorithm will have a locally quadratic rate of convergence by Theorem 3.4 of Smith (1993).

Theorem 2. *Consider the objective function $F(R, T)$ in the Eq.* (5). *Its Hessian is non-degenerate in a neighborhood of the optimal solution if there is a unique* (*up to a scale*) *solution to the system of linear equations*:

$$\mathbf{x}_2^{iT}E\mathbf{x}_1^i = 0, \quad E \in \mathbb{R}^{3\times 3}, \quad i = 1, \ldots, N.$$

If so, *the Riemannian Newton's algorithm has locally quadratic rate of convergence.*

Proof: It suffices to prove for any $\Delta \neq 0$, Hess$F(\Delta, \Delta) > 0$. According to the epipolar constraint, at the optimal solution, we have $\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i \equiv 0$. The Hessian is then simplified to:

$$\text{Hess}F(\Delta, \Delta) = \sum_{i=1}^{N}\big[\mathbf{x}_2^{iT}(\hat{T}\Delta_1 + \hat{\Delta}_2R)\mathbf{x}_1^i\big]^2.$$

Thus Hess $F(\Delta, \Delta) = 0$ if and only if:

$$\mathbf{x}_2^{iT}(\hat{T}\Delta_1 + \hat{\Delta}_2R)\,\mathbf{x}_1^i = 0, \quad i = 1, \ldots, N.$$

Since we also have:

$$\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i = 0, \quad i = 1, \ldots, N.$$

Then both $\hat{T}\Delta_1 + \hat{\Delta}_2R$ and $\hat{T}R$ are solutions for the same system of linear equations which by assumption has a unique solution, hence Hess$F(\Delta, \Delta) = 0$ if and only if:

$$\begin{aligned} &\hat{T}\Delta_1 + \hat{\Delta}_2R = \lambda\hat{T}R, \quad \text{for some } \lambda \in \mathbb{R} \\ &\quad \Leftrightarrow \hat{T}\hat{\omega} + \hat{\Delta}_2 = \lambda\hat{T} \quad \text{for } \omega = \Delta_1R^T \\ &\quad \Leftrightarrow \hat{T}\hat{\omega} = \lambda\hat{T} \quad \text{and } \Delta_2 = 0, \quad \text{since } T^T\Delta_2 = 0 \\ &\quad \Leftrightarrow \omega = 0, \text{ and } \Delta_2 = 0, \quad \text{since } T \neq 0 \\ &\quad \Leftrightarrow \Delta = 0. \end{aligned}$$

□

Comment 1 (*Non-degeneracy of Hessian*). *In the previous theorem*, *regarding the* 3×3 *matrix E in the* *equations* $\mathbf{x}_2^{iT} E \mathbf{x}_1^i = 0$ *as a vector in* $\mathbb{R}^9$, *one needs at least eight equations to uniquely solve E up to a scale. This implies that we need at least eight image correspondences* $\{(\mathbf{x}_1^i, \mathbf{x}_2^i)\}_{i=1}^N$, $N \geq 8$ *to guarantee the Hessian non-degenerate whence the iterative search algorithm locally converges in quadratic rate. If we study this problem more carefully using transversality theory*, *one may show that five image correspondences in general position is the minimal data to guarantee the Hessian non-degenerate* (*Maybank*, 1993). *However*, *the five point technique usually leads to many* (*up to twenty*) *ambiguous solutions*, *as pointed out by Horn* (1990). *Moreover*, *numerical errors usually make the algorithm not work exactly on the essential manifold and the extra solutions for the equations* $\mathbf{x}_2^{iT} E \mathbf{x}_1^i = 0$ *may cause the algorithm to converge very slowly in these directions. It is not just a coincidence that the conditions for the Hessian to be non-degenerate are exactly the same as that for the eight-point linear algorithm* (*see Maybank*, 1993; *Ma et al.*, 2000) *to have a unique solution. A heuristic explanation is that the objective function here is a quadratic form of the epipolar constraint on which the linear algorithm is directly based.*

Returning to the Newton's algorithm, assume that the Hessian is always non-degenerate and hence invertible. Then, at each point on the essential manifold, we can solve for the optimal updating vector Δ such that $\Delta = \text{Hess}^{-1} G$, or in other words we can find a unique Δ such that:

$$\begin{aligned} \text{Hess}\, F(Y, \Delta) &= g(-G, Y) \\ &= -dF(Y), \quad \text{for all vector fields } Y. \end{aligned}$$

Pick five linearly independent vectors E^k, $j = 1, \dots, 5$ forming a basis of $T_R(SO(3)) \times T_T(\mathbb{S}^2)$. One then obtains five linear equations:

$$\text{Hess}\, F(E^k, \Delta) = -dF(E^k), \quad k = 1, \dots 5 \qquad (13)$$

Since Hessian is invertible, these five linear equations uniquely determine Δ. In particular one can choose the simplest basis such that for $E^k = [\hat{e}_k R, 0]$ where e_k for $k = 1, 2, 3$ is the standard basis for $\mathbb{R}^3$. The vectors e_4, e_5 can be obtained using Gram-Schimdt process. Define a 5×5 matrix $A \in \mathbb{R}^{5\times 5}$ and a 5 dimensional vector $\mathbf{b} \in \mathbb{R}^5$ to be:

$$A_{kl} = \text{Hess}\, F(E^k, E^l), \quad \mathbf{b}_k = -dF(E^k), \quad k, l = 1, \dots, 5.$$

Then solve for the vector $\mathbf{a} = [a_1, a_2, a_3, a_4, a_5]^T \in \mathbb{R}^5$:

$$\mathbf{a} = A^{-1}\mathbf{b}.$$

Let $u = [a_1, a_2, a_3]^T \in \mathbb{R}^3$ and $v = a_4 e_4 + a_5 a_5 \in \mathbb{R}^3$. Then for the optimal updating vector $\Delta = (\Delta_1, \Delta_2)$, we have $\Delta_1 = \hat{u}R$ and $\Delta_2 = v$. We now summarize the Riemannian Newton's algorithm for minimizing the epipolar constraint which can be directly implemented.

Riemannian Newton's Algorithm for Motion Recovery from the Objective Function

$$F(R, T) = \sum_{i=1}^N \left(\mathbf{x}_2^{iT} \hat{T} R \mathbf{x}_1^i\right)^2, \quad \mathbf{x}_1^i, \mathbf{x}_2^i \in \mathbb{R}^3, \ (R, T) \in SO(3) \times \mathbb{S}^2.$$

- **Compute the Optimal Updating Vector:** *At the point* $(R, T) \in SO(3) \times \mathbb{S}^2$, *compute the optimal updating vector* $\Delta = -Hess^{-1}\, G$:

  1. *Compute vectors* e_4, e_5 *from* T *using Gram-Schimdt process and obtain five basis tangent vectors* $E^k \in T_R(SO(3)) \times T_T(\mathbb{S}^2)$, $k = 1, \dots, 5$.
  2. *Compute the* 5×5 *matrix* $(A)_{kl} = Hess\, F\, (E^k, E^l)$, $1 \leq k, l \leq 5$ *using the Hessian formula* (12).
  3. *Compute the 5 dimensional vector* $\mathbf{b}_k = -dF\, (E^k)$, $1 \leq k \leq 5$ *using the formula for the first derivative dF* (8).
  4. *Compute the vector* $\mathbf{a}=[a_1, a_2, a_3, a_4, a_5]^T \in \mathbb{R}^5$ *such that* $\mathbf{a} = A^{-1}\mathbf{b}$.
  5. *Define* $u = [a_1, a_2, a_3]^T \in \mathbb{R}^3$ *and* $v = a_4 e_4 + a_5 e_5 \in \mathbb{R}^3$. *Then the optimal updating vector* Δ *is given by*:

  $$\Delta = -\text{Hess}^{-1}\, G = (\hat{u}R, v).$$

- **Update the Search Spate:** *Move* (R, T) *in the direction* Δ *along the geodesic to* $(\exp(R, \Delta_1 t), \exp(T, \Delta_2 t))$, *using the formulae* (6) *and* (7) *for geodesics on* $SO(3)$ *and* $\mathbb{S}^2$ *respectively*:

$$\begin{aligned} R(t) &= \exp(R, \Delta_1 t) = R \exp \hat{\omega} t \\ &= R(I + \hat{\omega} \sin t + \hat{\omega}^2 (1 - \cos t)) \\ T(t) &= \exp(T, \Delta_2 t) = T \cos \sigma t + U \sin \sigma t \end{aligned}$$

where $t = \sqrt{\frac{1}{2} tr(\Delta_1^T \Delta_1)}$, $\omega = \Delta_1 R^T / t$, $\sigma = \sqrt{\frac{1}{2} tr(\Delta_2^T \Delta_2)}$, $U = \Delta_2 / \sigma$.

- **Return:** *to step* 1 *if* $\|\mathbf{b}\| \geq \epsilon$ *for some pre-determined error tolerance* $\epsilon > 0$.

2.3. Minimizing Normalized Epipolar Constraints

The epipolar constraint (5) gives the only necessary (depth independent) condition that image pairs have to satisfy. Thus the motion estimates obtained from minimizing the objective function (5) are not necessarily statistically or geometrically optimal for the commonly used noise model of image correspondences. We note here that this would continue to be a problem even if the terms associated with each image correspondence were given positive weights. It has been observed previously (Weng et al., 1993b) that in order to get less biased estimates the epipolar constraints need to properly *normalized.* In this section, we will give a brief account of these normalized versions of epipolar constraints. In the next section we demostrate that these normalizations can be unified by a single procedure for getting optimal estimates of motion and structure.

In the perspective projection case,[9] coordinates of image points $\mathbf{x}_1$ and $\mathbf{x}_2$ are of the form $\mathbf{x} = [x, y, 1]^T \in \mathbb{R}^3$. Suppose that the actual measured image coordinates of N pairs of image points are:

$$\mathbf{x}_1^i = \tilde{\mathbf{x}}_1^i + \alpha^i, \qquad \mathbf{x}_2^i = \tilde{\mathbf{x}}_2^i + \beta^i, \quad i = 1, \ldots, N \tag{14}$$

where $\tilde{\mathbf{x}}_2^i$ and $\tilde{\mathbf{x}}_1^i$ are ideal (noise free) image coordinates, $\alpha^i = [\alpha_1^i, \alpha_2^i, 0]^T \in \mathbb{R}^3$ and $\beta^i = [\beta_1^i, \beta_2^i, 0]^T \in \mathbb{R}^3$ and $\alpha_1^i, \alpha_2^i, \beta_1^i, \beta_2^i$ are independent Gaussian random variables of identical distribution $N(0, \sigma^2)$. Substituting $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$ into the epipolar constraint (5), we obtain:

$$\mathbf{x}_2^{iT} \hat{T} R \mathbf{x}_1^i = \beta^{iT} \hat{T} R \tilde{\mathbf{x}}_1^i + \tilde{\mathbf{x}}_2^{iT} \hat{T} R \alpha^i + \beta^{iT} \hat{T} R \alpha^i. \tag{15}$$

Since the image coordinates $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$ are usually magnitude larger than α^i and β^i, one can omit the last term in the equation above. Then $\mathbf{x}_2^{iT} \hat{T} R \mathbf{x}_1^i$ are independent random variables of *approximately* Gaussian distribution $N(0, \sigma^2(\|\hat{e}_3 \hat{T} R \mathbf{x}_1^i\|^2 + \|\mathbf{x}_2^{iT} \hat{T} R \hat{e}_3^T\|^2))$ where $e_3 = [0, 0, 1]^T \in \mathbb{R}^3$. If we assume the *a priori* distribution of the motion (R, T) is uniform, the maximum *a posteriori* (MAP) estimate of (R, T) is then the global minimum of the objective function:

$$F_s(R, T) = \sum_{i=1}^{N} \frac{\left(\mathbf{x}_2^{iT} \hat{T} R \mathbf{x}_1^i\right)^2}{\left\|\hat{e}_3 \hat{T} R \mathbf{x}_1^i\right\|^2 + \left\|\mathbf{x}_2^{iT} \hat{T} R \hat{e}_3^T\right\|^2}, \quad \mathbf{x}_1^i, \mathbf{x}_2^i \in \mathbb{R}^3, (R, T) \in SO(3) \times \mathbb{S}^2. \tag{16}$$

Here we use F_s to denote the *statistically normalized* objective function associated with the epipolar constraint. This objective function is also referred to in the literature as *gradient criteria* (Luong and Faugeras, 1996) or *epipolar improvement* (Weng et al., 1993a). Therefore, we have:

$$(R, T)_{MAP} \approx \arg\min F_s(R, T) \tag{17}$$

Note that in the noise free case, F_s achieves zeros just like the unnormalized objective function F of Eq. (5). Asymptotically, MAP estimates approach the unbiased minimum mean square estimates (MMSE). So, in general, the MAP estimator gives less biased estimates than the unnormalized objective function F. The reason for $\approx$ is that we have dropped one term in the expression (15).

Note that F_s is still a function defined on the manifold $SO(3) \times \mathbb{S}^2$. Moreover the numerator of each term of F_s is the same as that in F, and the denominator of each term in F_s is simply:

$$\begin{aligned} \left\|\hat{e}_3 \hat{T} R \mathbf{x}_1^i\right\|^2 + \left\|\mathbf{x}_2^{iT} \hat{T} R \hat{e}_3^T\right\|^2 &= \left(e_1^T \hat{T} R \mathbf{x}_1^i\right)^2 + \left(e_2^T \hat{T} R \mathbf{x}_1^i\right)^2 \\ &\quad + \left(\mathbf{x}_2^{iT} \hat{T} R e_1\right)^2 + \left(\mathbf{x}_2^{iT} \hat{T} R e_2\right)^2 \end{aligned} \tag{18}$$

where $e_1 = [1, 0, 0]^T \in \mathbb{R}^3$ and $e_2 = [0, 1, 0]^T \in \mathbb{R}^3$. That is, the components of each term of the normalized objective function F_s are essentially of the same form as that in the unnormalized one F. Therefore, we can exclusively use the formulae for the first and second order derivatives $dF(\Delta)$ and $\text{Hess}F(\Delta, \Delta)$ of the unnormalized objective function F to express those for the normalized objective function F_s by simply replacing $\mathbf{x}_1^i$ or $\mathbf{x}_2^i$ with e_1 or e_2 at proper places. This is one reason why the epipolar constraint is so important and studied first. Since for each term of F_s we now need to evaluate the derivatives of five similar components $(e_1^T \hat{T} R \mathbf{x}_1^i)^2$, $(e_2^T \hat{T} R \mathbf{x}_1^i)^2$, $(\mathbf{x}_2^{iT} \hat{T} R e_1)^2$, $(\mathbf{x}_2^{iT} \hat{T} R e_2)^2$ and $(\mathbf{x}_2^{iT} \hat{T} R \mathbf{x}_1^i)^2$, as compared to one in the unnormalized case, the Newton's algorithm for the normalized objective function is in general five times slower than that for the unnormalized objective function F. The normalized objective function gives statistically much better estimates, as we will demonstrate in the experimental section.

Another commonly used criterion to recover motion is to minimize the geometric distances between image points and corresponding epipolar lines. This objective function is given as:

$$F_g(R,T) = \sum_{i=1}^{N} \frac{\left(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2}{\|\hat{e}_3\hat{T}R\mathbf{x}_1^i\|^2} + \frac{\left(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2}{\|\mathbf{x}_2^{iT}\hat{T}R\hat{e}_3^T\|^2},$$
$$\mathbf{x}_1^i, \mathbf{x}_2^i \in \mathbb{R}^3,\ (R,T) \in SO(3)\times\mathbb{S}^2. \quad (19)$$

Here we use F_g to denote this *geometrically normalized* objective function. For a more detailed derivation and geometric meaning of this objective function see Luong and Faugerai (1996) and Zhang (1998). Notice that, similar to F and F_s, F_g is also a function defined on the essential manifold and can be minimized using the given Newton's algorithm. As we have demonstrated in Ma et al. (2000), in the differential case, the normalization has no effect when the translational motion is in the image plane, i.e., the unnormalized and normalized objective functions are in fact equivalent. For the discrete case, we have a similar claim. Suppose the camera motion is given by $(R,T) \in SE(3)$ with $T \in \mathbb{S}^2$ and $R = e^{\hat{\omega}\theta}$ for some $\omega \in \mathbb{S}^2$ and $\theta \in \mathbb{R}$. If $\omega = [0,0,1]^T$ and $T = [T_1,T_2,0]^T$, i.e., the translation direction is in the image plane, then, since R and $\hat{e}_3$ now commute, the expression $\|\hat{e}_3\hat{T}R\mathbf{x}_1^i\|^2 = \|\mathbf{x}_2^{iT}\hat{T}R\hat{e}_3^T\|^2 = \|T\|^2 = 1$. Hence, in this case, all the three objective functions F, F_s and F_g are very similar to each other around the actual (R,T).[10] Practically, when the translation is in the image plane and rotation is small (i.e., $R \approx I$), the normalization will have little effect on the motion estimates, as will be verified by the simulation.[11]

The relationship between the two objective functions F_s and F_g, each justified by its own reason will be revealed in the next section, where we study the problem of recovering motion and structure *simultaneously* as a following constrained optimization problem.

3. Motion and Structure from Optimal Triangulation

Note that, in the presence of noise, for the motion (R,T) recovered from minimizing the unnormalized or normalized objective functions F, F_s or F_g, the value of the objective functions is not necessarily zero. That is, in general:

$$\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i \neq 0, \quad i = 1,\ldots,N. \qquad (20)$$

Consequently, if one directly uses $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$ to recover the 3D location of the point to which the two image points $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$ correspond, the two rays corresponding to $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$ may not be coplanar, hence may not intersect at one 3D point. Also, when we derived the normalized epipolar constraint F_s, we ignored the second order terms. Therefore, rigorously speaking, it does not give the exact MAP estimates. Here we want to clarify the effect of such an approximation on the estimates both analytically and experimentally. Furthermore, since F_g also gives another reasonable approximation of the MAP estimates, can we relate both F_s and F_g to the MAP estimates in a unified way? This will be studied in this section. Experimental comparison will be given in the next section.

Under the assumption of Gaussian noise model (14), in order to obtain the optimal (MAP) estimates of camera motion and a consistent 3D structure, in principle we need to solve the following optimization problem:

Optimal Triangulation Problem. *Seek camera motion* (R,T) *and points* $\tilde{\mathbf{x}}_1^i \in \mathbb{R}^3$ *and* $\tilde{\mathbf{x}}_2^i \in \mathbb{R}^3$ *on the image plane such that they minimize the distance from* $\mathbf{x}_1^i$ *and* $\mathbf{x}_2^i$:

$$F_t(R,T,\tilde{\mathbf{x}}_1^i,\tilde{\mathbf{x}}_2^i) = \sum_{i=1}^{N} \|\tilde{\mathbf{x}}_1^i - \mathbf{x}_1^i\|^2 + \|\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\|^2 \qquad (21)$$

subject to the conditions:

$$\tilde{\mathbf{x}}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i = 0, \quad \tilde{\mathbf{x}}_1^{iT}e_3 = 1, \quad \tilde{\mathbf{x}}_2^{iT}e_3 = 1, \quad i = 1,\ldots,N. \qquad (22)$$

In the problem formulation above, we use F_t to denote the objective function for triangulation. This objective function is referred to in the literature as the reprojection error. Unlike Hartley and Sturm (1997), we do not assume a known essential matrix $\hat{T}R$. Instead we simultaneously seek $\tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i$ *and* (R,T) which minimize the objective function F_t given by (21). The objective function F_t then implicitly depends on the variables (R,T) through the constraints (22). The optimal solution to this problem is exactly equivalent to the optimal MAP estimates of both motion *and* structure. Using

Lagrangian multipliers, we can convert the minimization problem to an unconstrained one:

$$\min_{R,T,\tilde{\mathbf{x}}_1^i,\tilde{\mathbf{x}}_2^i} \sum_{i=1}^{N} \left\|\tilde{\mathbf{x}}_1^i - \mathbf{x}_1^i\right\|^2 + \left\|\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\right\|^2 + \lambda^i \tilde{\mathbf{x}}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i + \gamma^i\left(\tilde{\mathbf{x}}_1^{iT}e_3 - 1\right) + \eta^i\left(\tilde{\mathbf{x}}_2^{iT}e_3 - 1\right). \tag{23}$$

The necessary conditions for minima of this objective function are:

$$2\left(\tilde{\mathbf{x}}_1^i - \mathbf{x}_1^i\right) + \lambda^i R^T\hat{T}^T\tilde{\mathbf{x}}_2^i + \gamma^i e_3 = 0 \tag{24}$$

$$2\left(\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\right) + \lambda^i\hat{T}R\tilde{\mathbf{x}}_1^i + \eta^i e_3 = 0 \tag{25}$$

Under the necessary conditions, we obtain:

$$\begin{cases} \tilde{\mathbf{x}}_1^i & = \mathbf{x}_1^i - \frac{1}{2}\lambda^i\hat{e}_3^T\hat{e}_3R^T\hat{T}^T\tilde{\mathbf{x}}_2^i \\ \tilde{\mathbf{x}}_2^i & = \mathbf{x}_2^i - \frac{1}{2}\lambda^i\hat{e}_3^T\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i \\ \tilde{\mathbf{x}}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i & = 0 \end{cases} \tag{26}$$

where λ^i is given by:

$$\lambda^i = \frac{2\left(\mathbf{x}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i + \tilde{\mathbf{x}}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)}{\tilde{\mathbf{x}}_1^{iT}R^T\hat{T}^T\hat{e}_3^T\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i + \tilde{\mathbf{x}}_2^{iT}\hat{T}R\hat{e}_3^T\hat{e}_3R^T\hat{T}^T\tilde{\mathbf{x}}_2^i} \tag{27}$$

or

$$\lambda^i = \frac{2\mathbf{x}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i}{\tilde{\mathbf{x}}_1^{iT}R^T\hat{T}^T\hat{e}_3^T\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i} = \frac{2\tilde{\mathbf{x}}_2^{iT}\hat{T}R\mathbf{x}_1^i}{\tilde{\mathbf{x}}_2^{iT}\hat{T}R\hat{e}_3^T\hat{e}_3R^T\hat{T}^T\tilde{\mathbf{x}}_2^i}. \tag{28}$$

Substituting (26) and (27) into F_t, we obtain:

$$F_t\left(R, T, \tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i\right) = \sum_{i=1}^{N} \frac{\left(\mathbf{x}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i + \tilde{\mathbf{x}}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2}{\left\|\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i\right\|^2 + \left\|\tilde{\mathbf{x}}_2^{iT}\hat{T}R\hat{e}_3^T\right\|^2} \tag{29}$$

and using (26) and (28) instead, we get:

$$F_t\left(R, T, \tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i\right) = \sum_{i=1}^{N} \frac{\left(\mathbf{x}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i\right)^2}{\left\|\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i\right\|^2} + \frac{\left(\tilde{\mathbf{x}}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2}{\left\|\tilde{\mathbf{x}}_2^{iT}\hat{T}R\hat{e}_3^T\right\|^2}. \tag{30}$$

Geometrically, both expressions of F_t are the distances from the image points $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$ to the epipolar lines specified by $\tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i$ and (R, T). Equations (29) and (30) give explicit formulae of the residue of $\|\tilde{\mathbf{x}}_1^i - \mathbf{x}_1^i\|^2 + \|\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\|^2$ as $\mathbf{x}_1^i, \mathbf{x}_2^i$ being triangulated by $\tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i$. Note that the terms in F_t are normalized *crossed epipolar constraints* between $\mathbf{x}_1^i$ and $\tilde{\mathbf{x}}_2^i$ or between $\tilde{\mathbf{x}}_1^i$ and $\mathbf{x}_2^i$. These expressions of F_t can be further used to solve for (R, T) which minimizes F_t. This leads to the following iterative scheme for obtaining optimal estimates of both motion and structure, without explicitly introducing scale factors (or depths) of the 3D points.

Optimal Triangulation Algorithm Outline: *The procedure for minimizing F_t can be outlined as follows*:

1. **Initialization:** *Initialize* $\tilde{\mathbf{x}}_1^i(R, T), \tilde{\mathbf{x}}_2^i(R, T)$ as $\mathbf{x}_2^i, \mathbf{x}_1^i$.
2. **Motion:** *Update* (R, T) *by minimizing* $F_t^*(R, T) = F_t(R, T, \tilde{\mathbf{x}}_1^i(R, T), \tilde{\mathbf{x}}_2^i(R, T))$ *given by* (29) *or* (30) *as a function defined on the manifold* $SO(3) \times \mathbb{S}^2$.
3. **Structure (Triangulation):** *Solve for* $\tilde{\mathbf{x}}_1^i(R, T)$ *and* $\tilde{\mathbf{x}}_2^i(R, T)$ *which minimize the objective function* F_t (21) *with respect to* (R, T) *computed in the previous step.*
4. *Back to step 2 until updates are small enough.*

At step 2, $F_t^*(R, T)$:

$$\begin{aligned} F_t^*(R, T) &= \sum_{i=1}^{N} \frac{\left(\mathbf{x}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i + \tilde{\mathbf{x}}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2}{\left\|\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i\right\|^2 + \left\|\tilde{\mathbf{x}}_2^{iT}\hat{T}R\hat{e}_3^T\right\|^2} \\ &= \sum_{i=1}^{N} \frac{\left(\mathbf{x}_2^{iT}\hat{T}R\tilde{\mathbf{x}}_1^i\right)^2}{\left\|\hat{e}_3\hat{T}R\tilde{\mathbf{x}}_1^i\right\|^2} + \frac{\left(\tilde{\mathbf{x}}_2^{iT}\hat{T}R\mathbf{x}_1^i\right)^2}{\left\|\tilde{\mathbf{x}}_2^{iT}\hat{T}R\hat{e}_3^T\right\|^2} \end{aligned} \tag{31}$$

is a sum of normalized crossed epipolar constraints. It is a function defined on the manifold $SO(3) \times \mathbb{S}^2$, hence it can be minimized using the Riemannian Newton's algorithm, which is essentially the same as minimizing the normalized epipolar constraint (16) studied in the preceding section. The algorithm ends when (R, T) is already a minimum of F_t^*. It can be shown that if (R, T) is a critical point of F_t^*, then $(R, T, \tilde{\mathbf{x}}_1^i(R, T), \tilde{\mathbf{x}}_2^i(R, T))$ is necessarily a critical point of the original objective function F_t given by (21).

At step 3, for a fixed (R, T), $\tilde{\mathbf{x}}_1^i(R, T)$ and $\tilde{\mathbf{x}}_2^i(R, T)$ can be computed by minimizing the distance $\|\tilde{\mathbf{x}}_1^i - \mathbf{x}_1^i\|^2 + \|\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\|^2$ for each pair of image points. Let $t_2^i \in \mathbb{R}^3$ be the normal vector (of unit length) to the (epipolar) plane spanned by $(\tilde{\mathbf{x}}_2^i, T)$. Given such a t_2^i,

$\tilde{\mathbf{x}}_1^i$ and $\tilde{\mathbf{x}}_2^i$ are determined by:

$$\begin{aligned} \tilde{\mathbf{x}}_1^i\left(t_1^i\right) &= \frac{\hat{e}_3 t_1^i t_1^{iT} \hat{e}_3^T \mathbf{x}_1^i + \hat{t}_1^{iT} \hat{t}_1^i e_3}{e_3^T \hat{t}_1^{iT} \hat{t}_1^i e_3}, \\ \tilde{\mathbf{x}}_2^i\left(t_2^i\right) &= \frac{\hat{e}_3 t_2^i t_2^{iT} \hat{e}_3^T \mathbf{x}_2^i + \hat{t}_2^{iT} \hat{t}_2^i e_3}{e_3^T \hat{t}_2^{iT} \hat{t}_2^i e_3}, \end{aligned} \tag{32}$$

where $t_2^i = R^T t_1^i$. Then the distance can be explicitly expressed as:

$$\begin{aligned} \|\tilde{\mathbf{x}}_1^i - \mathbf{x}_1^i\|^2 + \|\tilde{\mathbf{x}}_2^i - \mathbf{x}_2^i\|^2 = \|\mathbf{x}_1^i\|^2 + \frac{t_2^{iT} A^i t_2^i}{t_2^{iT} B^i t_2^i} \\ + \|\mathbf{x}_2^i\|^2 + \frac{t_1^{iT} C^i t_1^i}{t_1^{iT} D^i t_1^i}, \end{aligned} \tag{33}$$

where

$$\begin{aligned} A^i &= I - \left(\hat{e}_3 \mathbf{x}_2^i \mathbf{x}_2^{iT} \hat{e}_3^T + \hat{\mathbf{x}}_2^i \hat{e}_3 + \hat{e}_3 \hat{\mathbf{x}}_2\right), \\ B^i &= \hat{e}_3^T \hat{e}_3 \\ C^i &= I - \left(\hat{e}_3 \mathbf{x}_1^i \mathbf{x}_1^{iT} \hat{e}_3^T + \hat{\mathbf{x}}_1^i \hat{e}_3 + \hat{e}_3 \hat{\mathbf{x}}_1^i\right), \\ D^i &= \hat{e}_3^T \hat{e}_3, \end{aligned} \tag{34}$$

The problem of finding $\tilde{\mathbf{x}}_1^i(R, T)$ and $\tilde{\mathbf{x}}_2^i(R, T)$ becomes one of finding t_2^i which minimizes the function of a sum of two *singular Rayleigh quotients*:

$$\min_{t_2^{iT} T = 0,\, t_2^{iT} t_2^i = 1} V\left(t_2^i\right) = \frac{t_2^{iT} A^i t_2^i}{t_2^{iT} B^i t_2^i} + \frac{t_2^{iT} R C^i R^T t_2^i}{t_2^{iT} R D^i R^T t_2^i}. \tag{35}$$

This is an optimization problem on a unit circle $\mathbb{S}^1$ in the plane orthogonal to the vector T (therefore, geometrically, motion and structure recovery from N pairs of image correspondences is an optimization problem on the space $SO(3) \times \mathbb{S}^2 \times \mathbb{T}^N$ where $\mathbb{T}^N$ is an N-torus, i.e., an N-fold product of $\mathbb{S}^1$). If $n_1, n_2 \in \mathbb{R}^3$ are vectors such that T, n_1, n_2 form an orthonormal basis of $\mathbb{R}^3$, then $t_2^i = \cos(\theta) n_1 + \sin(\theta) n_2$ with $\theta \in \mathbb{R}$. We only need to find θ^* which minimizes the function $V(t_2^i(\theta))$. From the geometric interpretation of the optimal solution, we also know that the global minimum θ^* should lie between two values: θ_1 and θ_2 such that $t_2^i(\theta_1)$ and $t_2^i(\theta_2)$ correspond to normal vectors of the two planes spanned by $(\mathbf{x}_2^i, T)$ and $(R\mathbf{x}_1^i, T)$ respectively. If $\mathbf{x}_1^i, \mathbf{x}_2^i$ are already triangulated, these two planes coincide. Therefore, in our approach the local minima is no longer an issue for triangulation, as oppose to the method proposed in Hartley and Sturm (1997). The problem now becomes a simple bounded minimization problem for a scalar function and can be efficiently solved using standard optimization routines such as "fmin" in Matlab or the Newton's algorithm. If one properly parameterizes $t_2^i(\theta)$, $t_2^i(\theta^*)$ can also be obtained by solving a 6-degree polynomial equation, as shown in Hartley and Sturm (1997) (and an approximate version results in solving a 4-degree polynomial equation Weng et al., 1993a). However, the method given in Hartley and Sturm (1997) involves coordinate transformation for each image pair and the given parameterization is by no means canonical. For example, if one chooses instead the commonly used parameterization of a circle $\mathbb{S}^1$:

$$\sin(2\theta) = \frac{2\lambda}{1 + \lambda^2}, \qquad \cos(2\theta) = \frac{1 - \lambda^2}{1 + \lambda^2}, \quad \lambda \in \mathbb{R}, \tag{36}$$

then it is straightforward to show from the Rayleigh quotient sum (35) that the necessary condition for minima of $V(t_2^i)$ is equivalent to a 6-degree polynomial equation in λ.[12] The triangulated pairs $(\tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i)$ and the camera motion (R, T) obtained from the minimization automatically give a consistent (optimal) 3D structure reconstruction from two views.

Comment 2 (*Stability of the Optimal Triangulation Algorithm*). *The* (*local*) *stability of the optimal triangulation algorithm follows directly from the fact that, in either step* 2 *or* 3 *of each iteration, the value of the crossed epipolar objective function* $F_t(R, T, \tilde{\mathbf{x}}_2^i, \tilde{\mathbf{x}}_1^i)$ *in* (29) *always decreases. Let us denote the estimates after kth iteration as* $(R(k), T(k), \tilde{\mathbf{x}}_2^i(k), \tilde{\mathbf{x}}_1^i(k))$. *Then* F_t *certainly decreases at step* 2, *i.e.*,

$$\begin{aligned} &F_t\big(R(k+1), T(k+1), \tilde{\mathbf{x}}_1^i(k), \tilde{\mathbf{x}}_2^i(k)\big) \\ &\quad \leq F_t\big(R(k), T(k), \tilde{\mathbf{x}}_1^i(k), \tilde{\mathbf{x}}_2^i(k)\big). \end{aligned}$$

To see that F_t *must decrease after step* 3, *we notice that what step* 3 *does is to directly minimize the original objective* (21) *subject to the epipolar constraint* (22) *with the motion* $(R(k+1), T(k+1))$ *fixed. For this constrained optimization problem, we apply the Lagrangian method in a similar way as the case when* (R, T) *are unknown. According to the Lagrangian method, if* $\tilde{\mathbf{x}}_1^i(k+1), \tilde{\mathbf{x}}_2^i(k+1)$ *solve the constrained optimization problem, it is necessary that, for some Lagrangian multipliers,* $(\tilde{\mathbf{x}}_1^i(k+1), \tilde{\mathbf{x}}_2^i(k+1))$ *should be minimizing the function* $F_t(R(k+1), T(k+1), \cdot, \cdot)$ *which is of exactly the same form as in* (29). *Hence we* *have*:

$$
\begin{aligned}
&F_t\big(R(k+1), T(k+1), \tilde{\mathbf{x}}_1^i(k+1), \tilde{\mathbf{x}}_2^i(k+1)\big)\\
&\quad\leq F_t\big(R(k+1), T(k+1), \tilde{\mathbf{x}}_1^i(k), \tilde{\mathbf{x}}_2^i(k)\big)\\
&\quad\leq F_t\big(R(k), T(k), \tilde{\mathbf{x}}_1^i(k), \tilde{\mathbf{x}}_2^i(k)\big).
\end{aligned}
$$

Hence the algorithm is at least locally stable, i.e., it is guaranteed to converge to a critical point of the function F_t given in (29).

Comment 3 (*Sensitivity of the Optimal Triangulation Algorithm*). *As one may have noticed, the crossed epipolar objective function F_t in* (29) *is a function on both the motion* (R, T) *and structure* $(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2)$. *Hence the Hessian of the function F_t around the ground truth* $(R^*, S^*, \tilde{\mathbf{x}}_1^*, \tilde{\mathbf{x}}_2^*)$ *provides information about the sensitivity of joint estimates of motion and structure together.*

The optimal triangulation method clarifies the relationship between previously obtained objective functions based on normalization, including F_s and F_g as well constraints captured by epipolar geometry. In the expressions of F_t, if we simply approximate $\tilde{\mathbf{x}}_1^i, \tilde{\mathbf{x}}_2^i$ by $\mathbf{x}_1^i, \mathbf{x}_2^i$ respectively, we may obtain the normalized versions of epipolar constraints for recovering camera motion. From (29) we get:

$$
F_s(R, T) = \sum_{i=1}^{N} \frac{4\big(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\big)^2}{\big\|\hat{e}_3\hat{T}R\mathbf{x}_1^i\big\|^2 + \big\|\mathbf{x}_2{}^{iT}\hat{T}R\hat{e}_3^T\big\|^2} \qquad (37)
$$

or from (30) we have:

$$
F_g(R, T) = \sum_{i=1}^{N} \frac{\big(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\big)^2}{\big\|\hat{e}_3\hat{T}R\mathbf{x}_1^i\big\|^2} + \frac{\big(\mathbf{x}_2^{iT}\hat{T}R\mathbf{x}_1^i\big)^2}{\big\|\mathbf{x}_2{}^{iT}\hat{T}R\hat{e}_3^T\big\|^2} \qquad (38)
$$

The first function (divided by 4) is exactly the same as the statistically normalized objective function F_s introduced in the preceding section; and the second one is exactly the geometrically normalized objective function F_g. From the above derivation, we see that there is essentially no difference between these two objective functions—they only differ by a second order term in terms of $\mathbf{x}_1^i - \tilde{\mathbf{x}}_1^i$ and $\mathbf{x}_2^i - \tilde{\mathbf{x}}_2^i$. Although such subtle differences between F_s, F_g and F_t has previously been pointed out in Zhang (1998), our approach discovers that all these three objective functions can be unified in the same optimization procedure—they are just slightly different approximations of the same objective function F_t^*. Practically speaking, using either normalized objective function F_s or F_g, one can already get camera motion estimates which are very close to the optimal ones.

Secondly, as we noticed, the epipolar constraint type objective function F_t^* given by (31) appears as a key intermediate objective function in an approach which initially intends to minimize the so-called reprojection error given by (21). The approach of minimizing reprojection error was previously considered in computer vision literature as an alternative to methods which directly minimize epipolar constraints (Weng et al., 1993b; Hartley and Sturm, 1997). We see here that they are in fact profoundly related. Further, the crossed epipolar constraint F_t^* given by (31) for motion estimation and the sum of singular Rayleigh quotients $V(t_2^i)$ given by (35) for triangulation are simply different expressions of the reprojection error under different conditions. In summary, "minimizing (normalized) epipolar constraints" (Luong and Faugeras, 1996; Zhang, 1998), "triangulation" (Hatley and Sturm, 1997) and "minimizing reprojection errors" (Weng et al., 1993b) are in fact different (approximate) versions of the same procedure of obtaining *the* optimal motion and structure estimates from image correspondences.

4. Critical Values and Ambiguous Solutions

We devote the remainder of this paper to the study of the robustness and sensitivity of motion and structure estimation problem in the presence of large levels of noise. We emphasize here the role of the linear techniques for initialization and utilize the characterization of the space of essential matrices and the intrinsic optimization techniques on the essential manifold for characterization of the critical points of the presented objective functions. We make a distinction between the robustness issue (behavior of the objective function in general configuration in the presence of large levels of noise) and sensitivity issue, which is more related to sensitive configurations of motion/structure.

Like any nonlinear system, when increasing the noise level, new critical points of the objective function can be introduced through bifurcation (Sastry, 1999). Although in general an objective function could have numerous critical points, numbers of different types of critical points have to satisfy the so-called *Morse inequalities*, which are associated to topological invariants of the underlying parameter space manifold (see Milnor, 1969). A study of these inequalities will help us to understand how patterns of the objective function's critical points may switch from one to another when the noise level varies.

Given a Morse function f (i.e., critical points are all non-degenerate) defined on a n-dimensional compact manifold M, according to the Morse lemma (Milnor, 1969), by changing the local coordinates of a neighborhood around a critical point, say $q \in M$, the function f locally looks like:

$$-x_1^2 - \cdots - x_\lambda^2 + x_{\lambda+1}^2 + \cdots + x_n^2. \quad (39)$$

The number λ is called the *index* of the critical point q. Note that q is a local minimum when $\lambda = 0$ and a maximum when $\lambda = n$. Let C_λ denote the number of critical points with index λ. Let D_λ denote the dimension of the λth homology group $H_\lambda(M, \mathbb{K})$ of M over any field $\mathbb{K}$, the so-called λth *Betti number*. Then the Morse inequalities are given by:

$$\sum_{\lambda=i}^{0}(-1)^{i-\lambda}D_\lambda \leq \sum_{\lambda=i}^{0}(-1)^{i-\lambda}C_\lambda, \quad i = 0, 1, 2, \ldots n-1 \quad (40)$$

$$\sum_{\lambda=0}^{n}(-1)^{\lambda}D_\lambda = \sum_{\lambda=0}^{n}(-1)^{\lambda}C_\lambda. \quad (41)$$

Note that $\sum_{\lambda=0}^{n}(-1)^{\lambda}D_\lambda$ is the *Euler characteristic* $\chi(M)$ of the manifold M. In our case, all objective functions F, F_s, F_g and F_t^* that we have encountered are even functions in $S \in \mathbb{S}^2$.[13] We can then view them as functions on the manifold $SO(3) \times \mathbb{RP}^2$ instead of $SO(3) \times \mathbb{S}^2$, where $\mathbb{RP}^2$ is the two dimensional real projective plane. Computing the dimension of homology groups of $SO(3) \times \mathbb{RP}^2$ we obtain $D_\lambda = 1, 2, 3, 3, 2, 1$ for $\lambda = 0, 1, 2, 3, 4, 5$ respectively, whence the Euler characteristic $\chi(SO(3) \times \mathbb{RP}^2) = 0$.

All the Morse inequalities above give necessary constraints on the numbers of different types of critical points of the function $F(R, T)$. Among all the critical points, those belonging to type 0 are called (local) *minima*, type n are (local) *maxima*, and types 1 to $n - 1$ are *saddles*. Since, from the above computation, the Euler characteristic of the manifold $SO(3) \times \mathbb{RP}^2$ is 0, any Morse function defined on it must have all three kinds of critical values. The nonlinear search algorithms proposed in the above are trying to find the global minimum of given objective functions. When increasing the noise level, new critical points can be introduced through bifurcation. Although, in general, many different types of bifurcations may occur when increasing the noise level, the *fold bifurcation* illustrated in Fig. 3 occurs most generically (see Sastry 1999, Chapter 7) in the motion and structure estimation problem. We therefore need to understand how such a bifurcation occurs and demonstrate how it affects the motion estimates. The study of the intrinsic local minima and ambiguities in the discrete setting is extremely difficult due to the need of visualizing the space of unknown parameters. Previous approaches resorted to approximations (Oliensis, 1999) and demonstrated the presence of local minima and bias towards optical axis. Here we undertake the study of the problem of local minima in the context of initialization by linear algorithms.

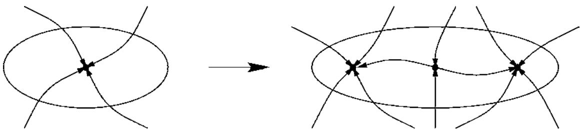

Figure 3. Bifurcation which preserves the Euler characteristic by introducing a pair of saddle and node. The indices of the two circled regions are both 1.

Since the nonlinear search schemes are usually initialized by the linear algorithm, not all the local minima are equally likely to be reached by the proposed algorithms. In the preceding section we showed that all the objective functions presented here are approximately equivalent to the epipolar constraints, especially when the translation is parallel to the image plane. If we let $E = \hat{T}R$ to be the essential matrix, then we can rewrite the epipolar constraint as $\mathbf{x}_2^{iT}E\mathbf{x}_1^i = 0, i = 1, \ldots, N$. Then minimizing the objective function F is (approximately) equivalent to the following least square problem:

$$\min \|Ae\|^2 \quad (42)$$

where A is a $N \times 9$ matrix function of entries of $\mathbf{x}_1^i$ and $\mathbf{x}_2^i$, and $e \in \mathbb{R}^9$ is a vector of the nine entries of E. Then e is the (usually one dimensional) null space of the 9×9 symmetric matrix A^TA. In the presence of noise, e is simply chosen to be the eigenvector corresponding to the least eigenvalue of A^TA. At a low noise level, this eigenvector in general gives a good initial estimate of the essential matrix. However, at a certain high noise level, the smallest two eigenvalues may switch roles, as do the two corresponding eigenvectors—topologically and a bifurcation as shown in Fig. 3 occurs. Let us denote these two eigenvectors as e and e'. Since they both are eigenvectors of the symmetric matrix A^TA,

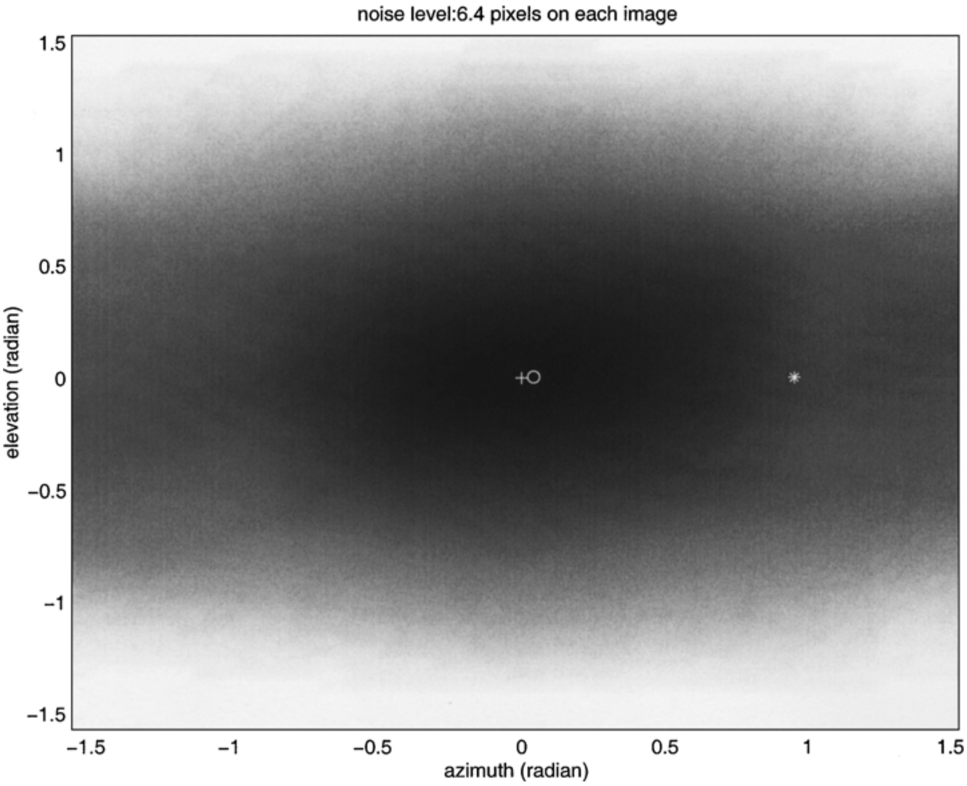

Figure 4. Value of objective function F_s for all S at noise level 6.4 pixels (rotation fixed at the estimate from the nonlinear optimization). Estimation errors: 0.014 in rotation estimate (in terms of the canonical metric on $SO(3)$) and 2.39° in translation estimate (in terms of angle).

they must be orthogonal to each other, i. e., $e^T e' = 0$. In terms of matrix notation, we have $tr(E^T E') = 0$. For the motions recovered from E and E' respectively, we have $tr(R^T \hat{T}^T \hat{T}' R') = 0$. It is well known that the rotation estimate R is usually much less sensitive to noise than the translation estimates T. Therefore, approximately, we have $R \approx R'$ hence $tr(\hat{T}^T \hat{T}') \approx 0$, That is T and T' are almost orthogonal to each other. This phenomena is very common for linear techniques for the motion estimation problem: at a high noise level, the translation estimate may suddenly change direction by roughly 90°, especially in the case when translation is parallel to the image plane. We will refer to such estimates as the *second eigenmotion*. Similar to detecting local minima in the differential case (see Soatto and Brockett, 1998), the second eigenmotion ambiguity can be usually detected by checking the positive depth constraints. A similar situation of the 90° flip in the motion estimates for the differential case and small field of view has previously been reported in Danilidis and Nagel (1990).

Figures 4 and 5 demonstrate such a sudden appearance of the second eigenmotion. They are the simulation results of the proposed nonlinear algorithm of minimizing the function F_s for a cloud of 40 randomly generated pairs of image correspondences (in a field of view 90°, depth varying from 100 to 400 units of focal length.). Gaussian noise of standard deviation of 6.4 or 6.5 pixels is added on each image point (image size 512 × 512 pixels). To make the results comparable, we used the same random seeds for both runs. The actual rotation is 10° about the Y-axis and the actual translation is along the X-axis.[14] The ratio between translation and rotation is 2.[15] In the figures, "$+$" marks the actual translation, "$*$" marks the translation estimate from linear algorithm (see Maybank, 1993 for detail) and "$\circ$" marks the estimate from nonlinear optimization. Up to the noise level of 6.4 pixels, both rotation and translation estimates are very close to the actual motion. Increasing the noise level further by 0.1 pixel, the translation estimate suddenly switches to one which is roughly 90° away from the actual translation.

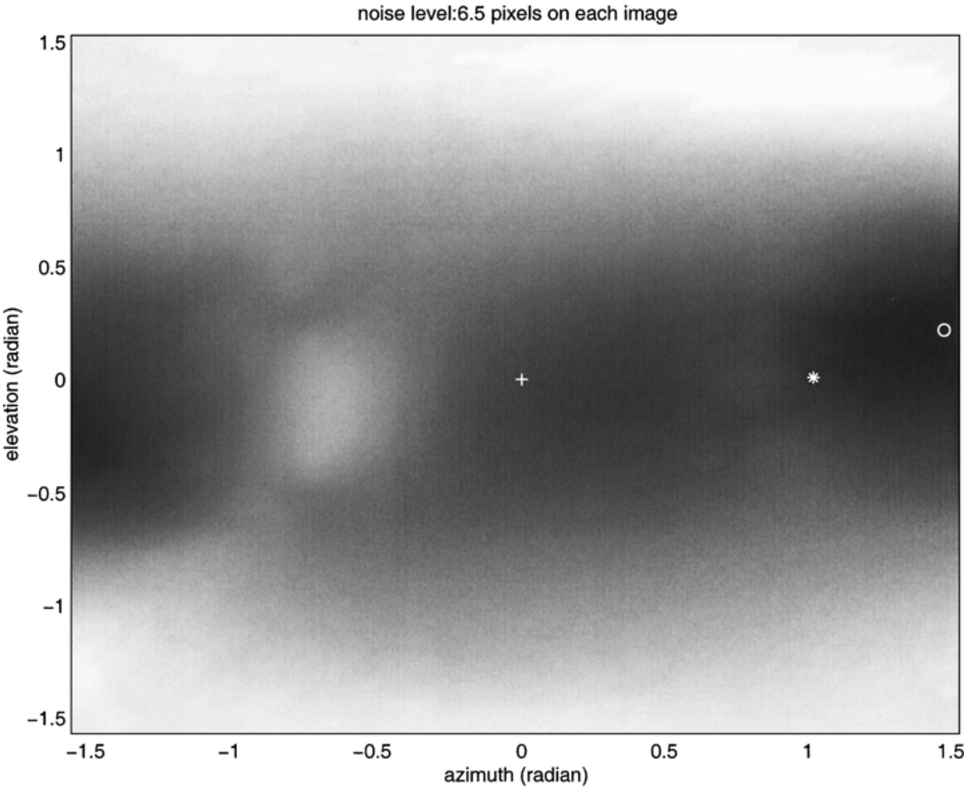

Figure 5. Value of objective function F_s for all S at noise level 6.5 pixels (rotation fixed at the estimate from the nonlinear optimization). Estimation errors: 0.227 in rotation estimate (in terms of the canonical metric on $SO(3)$) and 84.66° in translation estimate (in terms of angle).

Geometrically, this estimate corresponds to the second smallest eigenvector of the matrix $A^T A$ as we discussed before. Topologically, this estimate corresponds to the local minimum introduced by a bifurcation as shown by Fig. 3. Clearly, in Fig. 4, there are 2 maxima, 2 saddles and 1 minima on $\mathbb{RP}^2$; in Fig. 5, there are 2 maxima, 3 saddles and 2 minima. Both patterns give the Euler characteristic of $\mathbb{RP}^2$ as 1.

From the Fig. 5, we can see that the second eigenmotion ambiguity is even more likely to occur (at certain high noise level) than the other local minimum marked by "$\diamond$" in the figure which is a legitimate estimate of the actual one. These two estimates always occur in pairs and exist for general configuration even when both the FOV and depth variation are sufficiently large. We propose a way for resolving the second eigenmotion ambiguity at the initialization stage by linear algorithm. An indicator of the configuration being close to critical is the ratio of the two smallest eigenvalues of $A^T A$ σ_9 and σ_8 . By using both eigenvectors v_9 and v_8 for computing the linear motion estimates and choosing the one which satisfies the positive depth constraint by a larger margin (i.e., larger number of points satisfies the positive depth constraint) leads to the motion estimates closer to the true one. The motion estimate (R, T) which satisfies the positive depth constraint should make the following inner product:

$$\left(\hat{T}\mathbf{x}_1^i\right)^T \left(\hat{\mathbf{x}}_1^i R^T \mathbf{x}_2^i\right) > 0 \tag{43}$$

greater then 0 for all the corresponding points. While for low noise level all the points satisfy the positive depth constraint, with the increasing noise level some of the points fail to satisfy it. We therefore chose the solution where a majority of points satisfies the positive depth constraint. Simple re-initialization then guarantees convergence of the nonlinear techniques to the true solution. Figures 6 and 7 depict a slice of the objective function for varying translation and for the rotation estimate obtained by linear algorithm using v_9 and v_8 as two different estimates of the essential matrix.

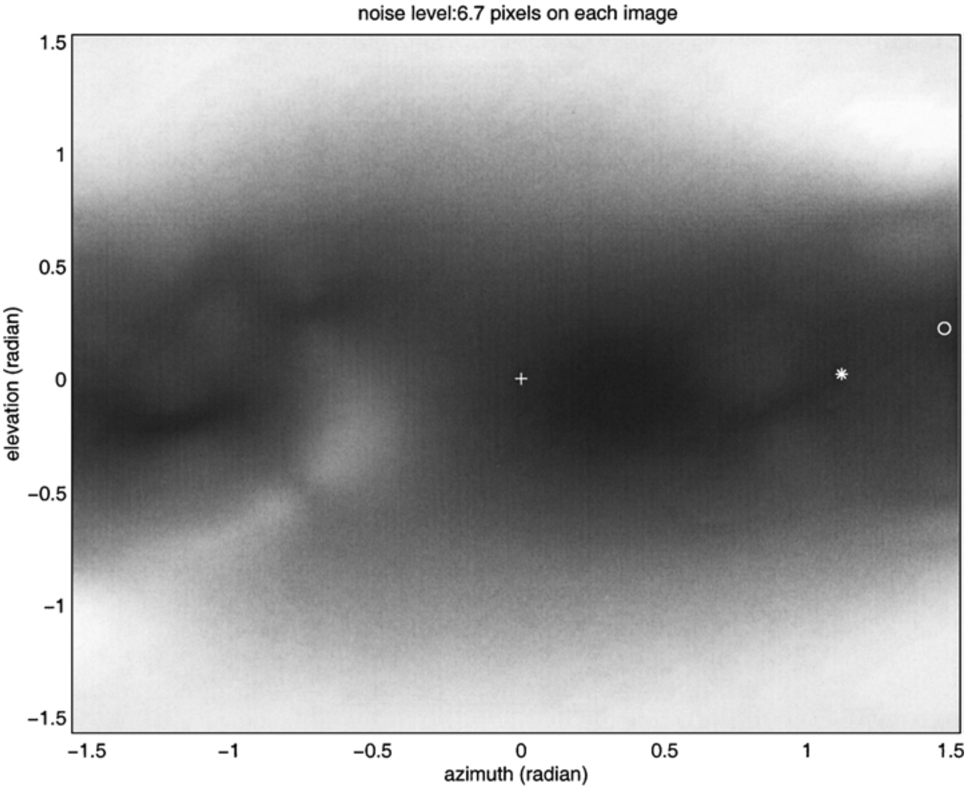

Figure 6. Value of objective function F_s for all S at noise level 6.7 pixels. Rotation is fixed at the estimate from the linear algorithm from the eigenvector v_9 associated with the smallest eigenvalue. Note the verge of the bifurcation of the objective function.

The second eigenmotion, however, is not statistically meaningful: it is an artifact introduced by a bifurcation of the objective function; it occurs only at a high noise level and this critical noise level gives a measure of the *robustness* of the given algorithm. For comparison, Fig. 8 demonstrates the effect of the bas-relief ambiguity: the long narrow valley of the objective function corresponds to the direction that is the most sensitive to noise.[16] The (translation) estimates of 20 runs, marked as "○", give a distribution roughly resembling the shape of this valley—the actual translation is marked as "+" in the center of the valley which is covered by circles. This second eigenmotion effect has a quite different interpretation then bas-relief ambiguity. The bas-relief effect is only evident when FOV and depth variation is small, but the second eigenmotion ambiguity appears at higher noise levels for general configurations.

5. Experiments and Sensitivity Analysis

In this section, we demonstrate by experiments the relationship among the linear algorithm (as in Maybank (1993)), nonlinear algorithm (minimizing F), normalized nonlinear algorithm (minimizing F_s) and optimal triangulation (minimizing F_t). Due to the nature of the second eigenmotion ambiguity, it gives statistically meaningless estimates. Such estimates should be treated as "outliers" if one wants to properly evaluate a given algorithm and compare simulation results. In order for all the simulation results to be statistically meaningful and comparable to each other, in following simulations, we usually keep the noise level below the critical level at which the second eigenmotion ambiguity occurs unless we need to comment on its effect on the evaluation of algorithm's performance.

We follow the same line of thought as the analysis of the differential case in Soatto and Brockett (1998). We will demonstrate by simulations that seemingly conflicting statements in the literature about the performance of existing algorithms can in fact be given a *unified* explanation if we systematically compare the simulation results with respect to a *large range* of noise levels (as long as the results are statistically meaningful). Some existing evaluations of the algorithms

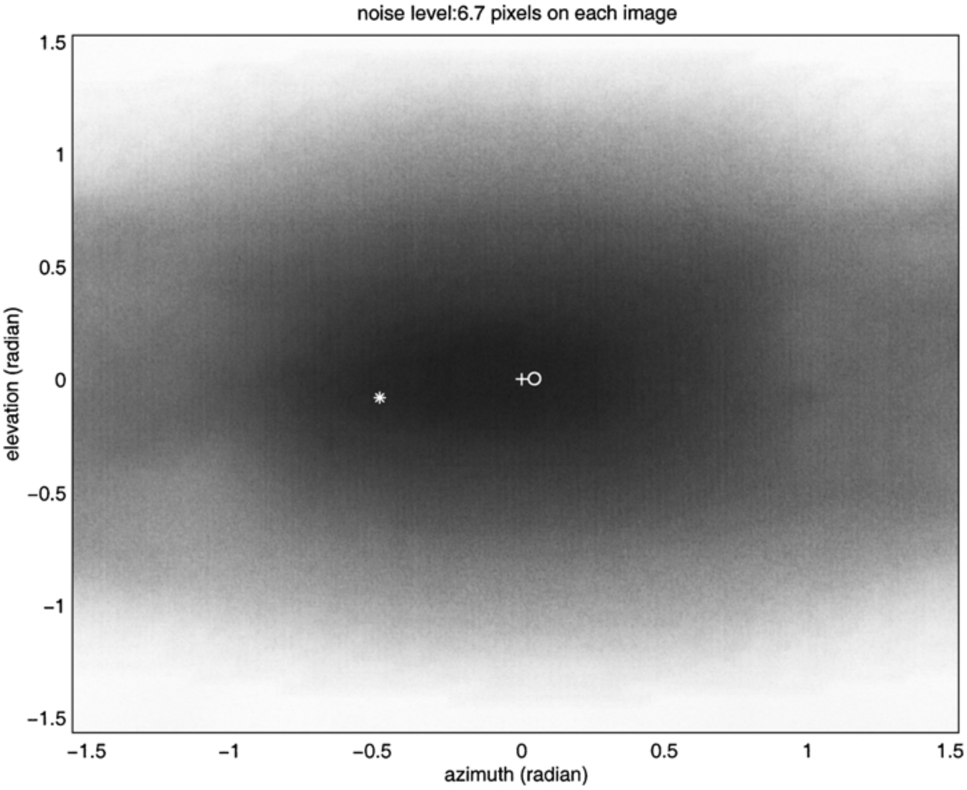

Figure 7. Value of objective function F_s for all S at noise level 6.7 pixels. Rotation is fixed at the estimate from the linear algorithm from the eigenvector v_8 associated with the second smallest eigenvalue. The objective function is well shaped and the nonlinear algorithm refined the linear estimate closer to the true solution.

turn out to be valid only for a certain small range of signal-to-noise ratio. In particular, algorithms' behaviors at very high noise levels have not yet been well understood or explained. Since, for a fixed noise level, changing baseline is equivalent to changing the signal-to-noise ratio, we hence perform the simulations at a fixed baseline but the noise level varies from very low (<1 pixels) to very high (tens of pixels for a typical image size of 512 × 512 pixels). The conclusions therefore hold for a large range of baselines. In particular, we emphasize that some of the statements given below are valid for the differential case as well.

In following simulations, for each trial, a random cloud of 40 3D points is generated in a region of truncated pyramid with a field of view (FOV) 90°, and a depth variation from 100 to 400 units of the focal length. Noises added to the image points are i.i.d. 2D Gaussian with standard deviation of the given noise level (in pixels). Magnitudes of translation and rotation are compared at the center of random cloud. This will be denoted as the translation-to-rotation ratio, or simply the T/R ratio. The algorithms will be evaluated for different combinations of translation and rotation directions. We here use the convention that Y-axis is the vertical direction of the image and X-axis is the horizontal direction and the Z-axis coincides with the optical axis of the camera. All nonlinear algorithms are initialized by the estimates from the standard 8-point linear algorithm (see Maybank, 1993). The criteria for all nonlinear algorithms to stop are: 1. The norm of gradient is less than a given error tolerance, which usually we pick as 10^{-8} unless otherwise stated;[17] and 2. The smallest eigenvalue of the Hessian matrix is positive.[18]

5.1. Axis Dependency Profile

It has been well known that the sensitivity of the motion estimation depends on the camera motion. However, in order to give a clear account of such a dependency, one has to be careful about two points: 1. The signal-to-noise ratio and 2. Whether the simulation results are

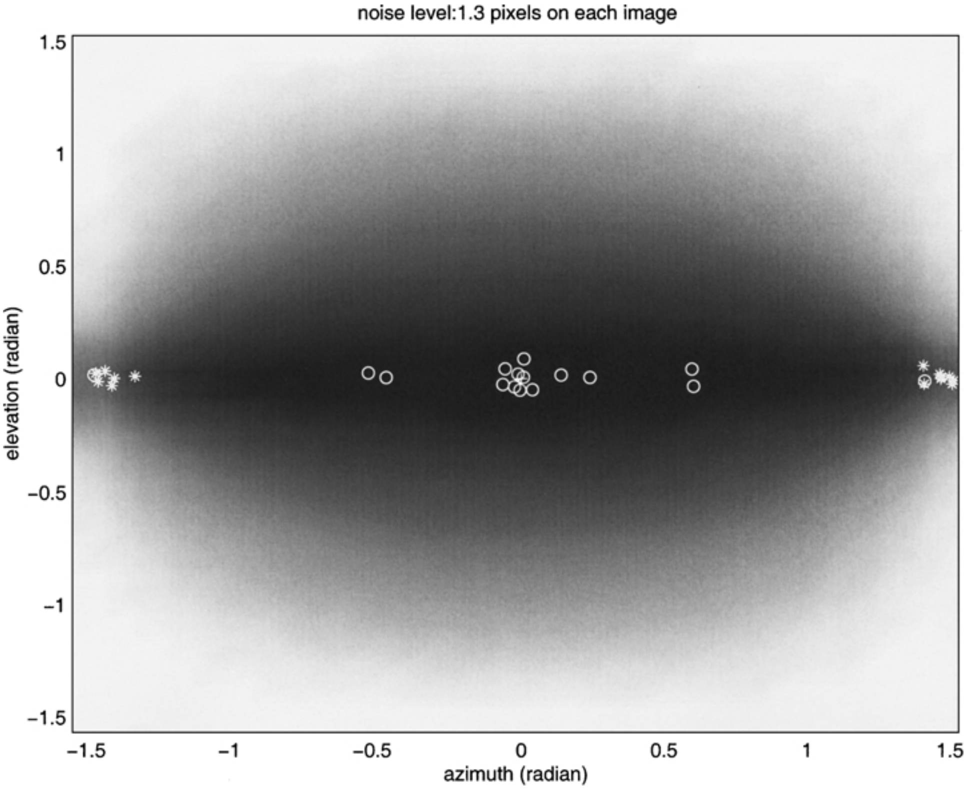

Figure 8. Bas-relief ambiguity. FOV is 20° and the random cloud depth varies from 100 to 150 units of focal length. Translation is along the X-axis and rotation around the Y-axis. Rotation magnitude is 2°. T/R ratio is 2. 20 runs at the noise level 1.3 pixels.

still statistically meaningful while varying the noise level.

Figures 9–12 give simulation results of 100 trials for each combination of translation and rotation ("T-R") axes, for example, "X-Y" means translation is along the X-axis and the rotation axis is the Y-axis. Rotation is always 10^o about the axis and the T/R ratio is 2. In the figures, "linear" stands for the standard 8-point linear algorithm; "nonlin" is the Riemannian Newton's algorithm minimizing the epipolar constraints F, "normal" is the Riemannian Newton's algorithm minimizing the normalized epipolar constraints F_s.

By carefully comparing the simulation results in Figs. 9–12, we can draw the following conclusions:

- *Optimization Techniques (linear vs. nonlinear)*

  1. Minimizing F in general gives better estimates than the linear algorithm at low noise levels (Figs. 9 and 10). At higher noise levels, this is no longer true (Figs. 11 and 12), due to the more global nature of the linear technique.
  2. Minimizing the normalized F_s in general gives better estimates than the linear algorithm at moderate noise levels (all figures). Very high noise level case will be studied in the next section.

- *Optimization Criteria* (F vs. F_s)

  1. At relatively low noise levels (Fig. 9), normalization has little effect when translation is parallel to the image plane; and estimates are indeed improved when translation is along the Z-axis.
  2. However, at moderate noise levels (Figs. 10–12), things are quite the opposite: when translation is along the Z-axis, little improvement can be gained by minimizing F_s instead of F since estimates are less sensitive to noise in this case (in fact all three algorithms perform very close); however, when translation is parallel to the image plane, F is more sensitive to noise and minimizing the statistically less biased F_s consistently improves the estimates.

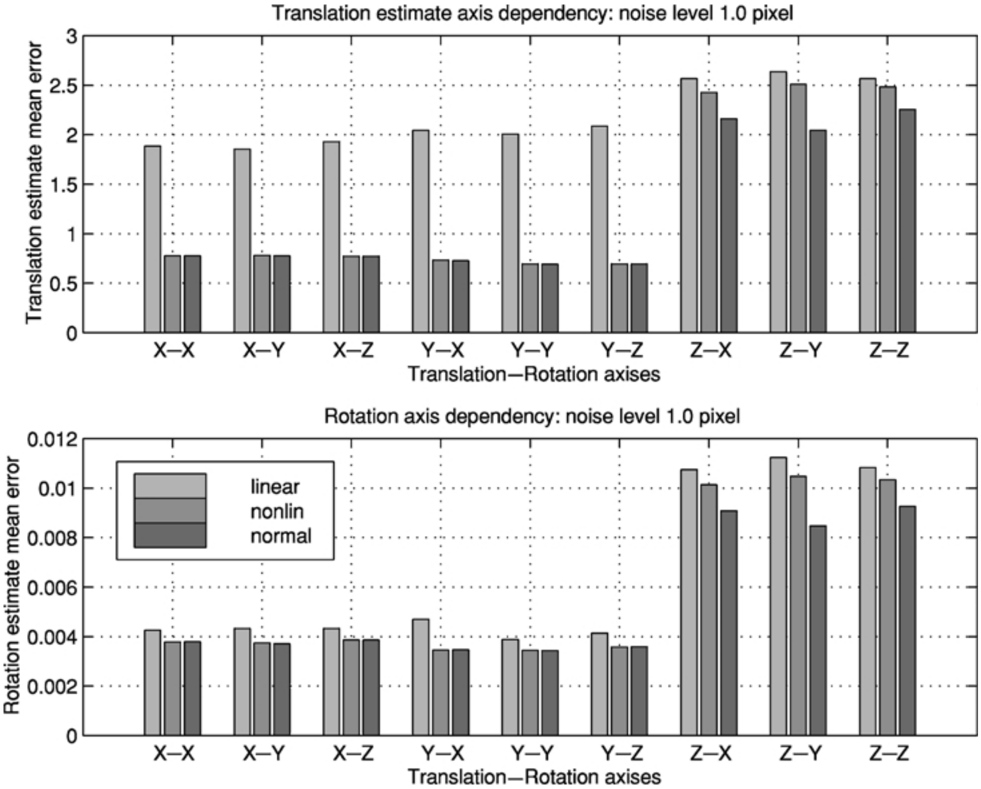

Figure 9. Axis dependency: estimation errors in rotation and translation at noise level 1.0 pixel. T/R ratio = 2 and rotation = 10°.

- *Axis Dependency (translation parallel to image plane vs. along Z-axis)*

1. All three algorithms are the most robust to the increase of noise when the translation is along Z. At moderate noise levels (all figures), their performances are quite close to each other.
2. Although, at relatively low noise levels (Figs. 9–11), estimation errors seem to be larger when the translation is along the Z-axis, estimates are in fact much less sensitive to noise and more robust to increasing of noise in this case. The larger estimation error in case of translation along Z-axis is because the displacements of image points are smaller than those when translation is parallel to the image plane. Thus, with respect to the same noise level, the signal-to-noise ratio is in fact smaller in the case of translation along the Z-axis.
3. At a noise level of 7 pixels (Fig. 12), estimation errors seem to become smaller when the translation is along Z-axis. This is not only because estimates are less sensitive to noise for this case, but also due to the fact that, at a noise level of 7 pixels, the second eigenmotion ambiguity already occurs in some of the trials when the translation is parallel to the image plane. Outliers given by the second eigenmotion are averaged in the estimation errors and make them look even worse.

The second statement about the axis dependency supplements the observation given in Weng et al. (1989). In fact, the motion estimates are both robust and less sensitive to increasing of noise when translation is along the Z-axis. Due to the exact reason given in Weng et al. (1989), smaller signal-to-noise ratio in this case makes the effect of robustness not to appear in the mean estimation error until at a higher noise level. As we have claimed before, for a fixed base line, high noise level results resemble those for a smaller base line at a moderate noise level. Figure 12 is therefore a generic picture of the axis dependency profile for the differential or small base-line case (for more details see Ma et al., 2000).

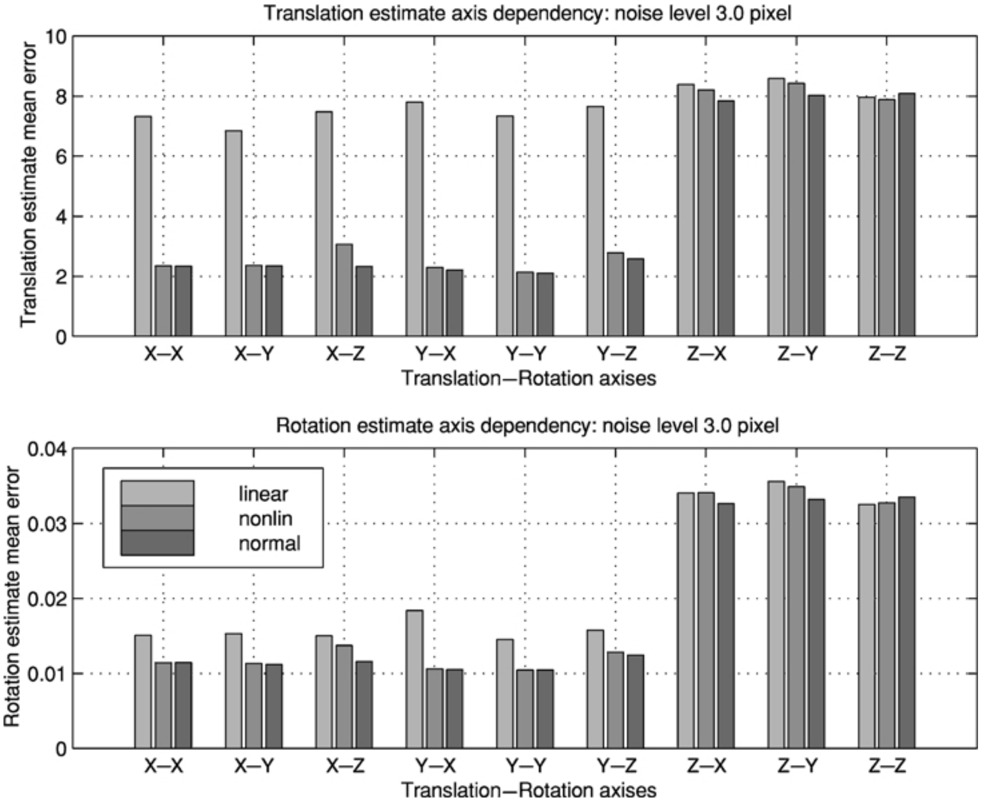

Figure 10. Axis dependency: estimation errors in rotation and translation at noise level 3.0 pixels. T/R ratio = 2 and rotation = 10°.

5.2. *Non-iterative vs. Iterative*

In general, the motion estimates obtained from directly minimizing the normalized epipolar constraints F_s or F_g are already very close to the solution of the optimal triangulation obtained by minimizing F_t *iteratively* between motion and structure. It is already known that, at low noise levels, the estimates from the non-iterative and iterative schemes usually differ by less than a couple of percent (Zhang, 1998). This is demonstrated in Figs. 13 and 14—"linear" stands for the linear algorithm; "norm nonlin" for the Riemannian Newton's algorithm minimizing normalized epipolar constraint F_s; "triangulate" for the iterative optimal triangulation algorithm. For the noise level from 0.5 to 5 pixels, at the error tolerance 10^{-6}, the iterative scheme has little improvement over the non-iterative scheme—the two simulation curves overlap with each other. Simulation results given in Figs. 15 and 16 further show that the improvements of the iterative scheme become a little bit more evident when noise levels are very high, but still very slim. Due to the second eigenmotion ambiguity, we can only perform high noise level simulation properly for the case when the translation direction is along the Z-axis.

By comparing the simulation results in Figs. 13–16, we can therefore draw the following conclusions:

- Although the iterative optimal triangulation algorithm usually gives better estimates (as it should), the non-iterative minimization of the normalized epipolar constraints F_s or F_g gives motion estimates with only a few percent larger errors for all range of noise levels. The higher the noise level, the more evident the improvement of the iterative scheme is.
- Within moderate noise levels, normalized nonlinear algorithms consistently give significantly better estimates than the standard linear algorithm, especially when the translation is parallel to the image plane. At very high noise levels, the performance of the standard linear algorithm outperforms nonlinear algorithms. This is due to the more global nature of the linear algorithm.

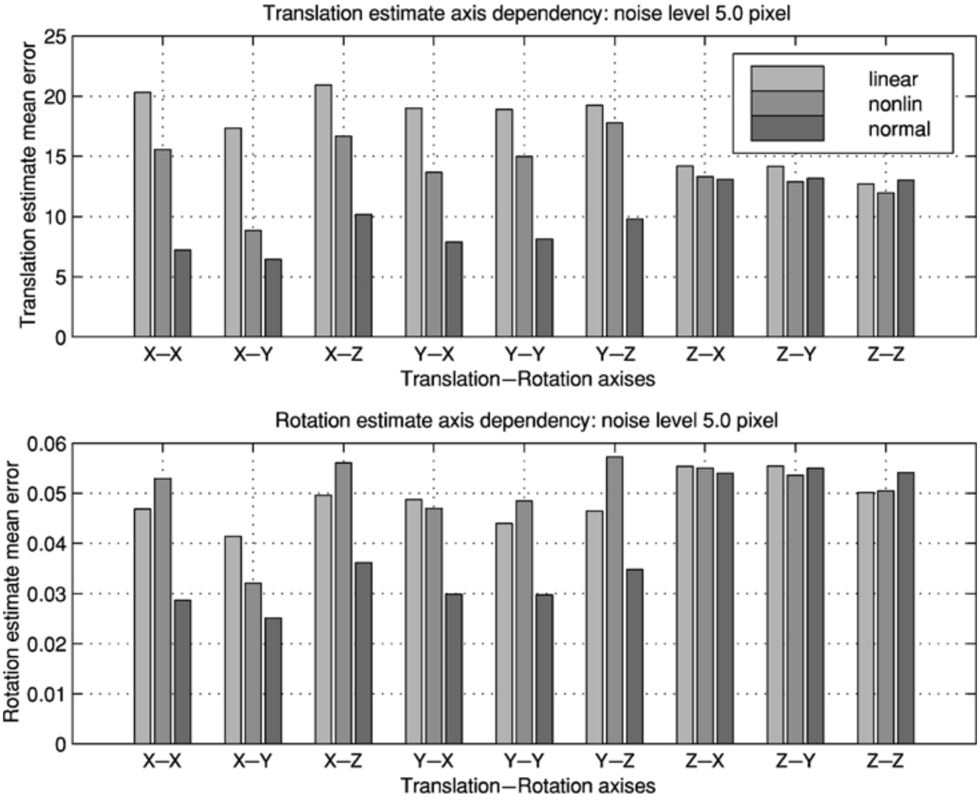

Figure 11. Axis dependency: estimation errors in rotation and translation at noise level 5.0 pixel. T/R ratio = 2 and rotation = 10°.

For low level Gaussian noises, the iterative optimal triangulation algorithm gives the MAP estimates of the camera motion and scene structure, the estimation error can be shown close to the theoretical error bounds, such as the Cramer-Rao bound. This has been shown experimentally in Weng et al. (1993a). Consequently, minimizing the normalized epipolar constraints F_s or F_g gives motion estimates close to the error bound as well. At very high noise levels, linear algorithm is certainly more robust and gives better estimates. Due to numerous local minima, running nonlinear algorithms to update the estimate of the linear algorithm does not necessarily reduce the estimation error further.

6. Discussion and Future Work

The motion and structure recovery problem has been studied extensively and many researchers have proposed efficient nonlinear optimization algorithms. One may find historical reviews of these algorithms in Maybank (1993) and Kanatani (1993). Although these algorithms already have good performance in practice, the geometric concepts behind them had not been completely revealed. The non-degeneracy conditions and convergence speed of those algorithms are usually not explicitly addressed. Due to the recent development of optimization methods on Riemannian manifolds, we now can have a better mathematical understanding of these algorithms, and propose new geometric algorithms or filters (for example, following (Soatto and Perona, 1996), which exploit the intrinsic geometric structure of the motion and structure recovery problem. As shown in this paper, regardless of the choice of different objectives, the problem of optimization on the essential manifold is common and essential to the optimal motion and structure recovery problem. Furthermore, from a pure optimization theoretic viewpoint, most of the objective functions previously used in the literature can be unified in a single optimization procedure. Consequently, “minimizing (normalized) epipolar constraints,” “triangulation,” “minimizing reprojection errors” are all different (approximate) versions of the same simple optimal triangulation algorithm.

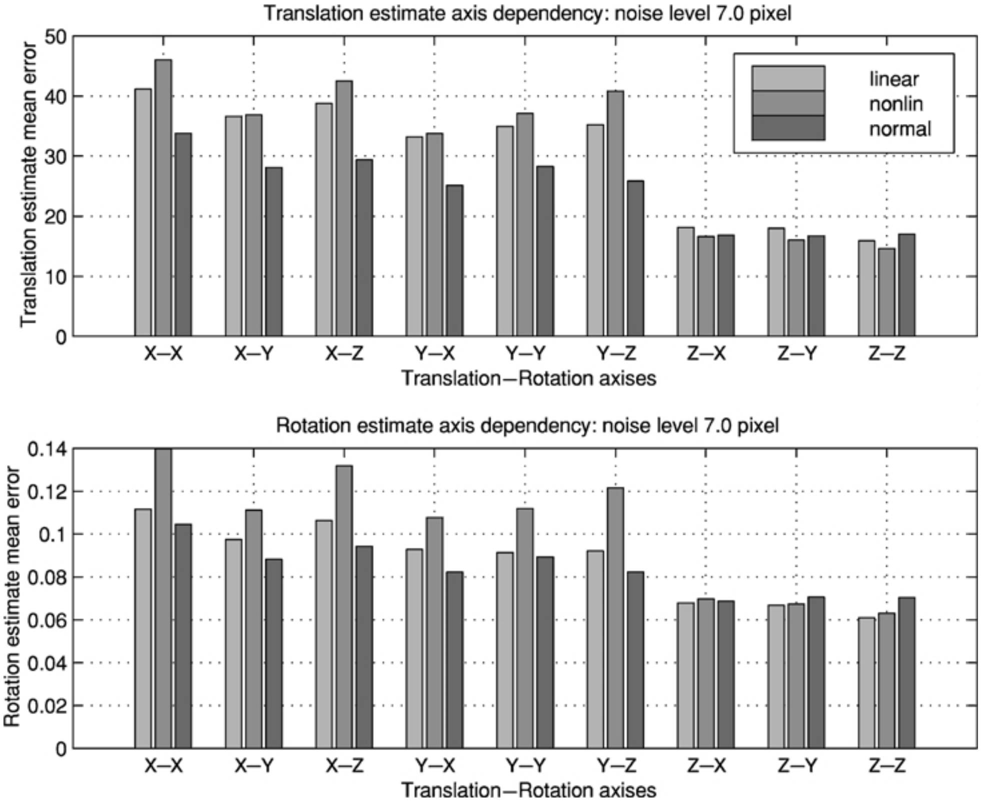

Figure 12. Axis dependency: estimation errors in rotation and translation at noise level 7.0 pixels. T/R ratio = 2 and rotation = 10°.

We have applied only Newton's algorithm to the motion and structure recovery problem since it has the fastest convergence rate (among algorithms using second order information, see Edelman et al. (1998) for the comparison). In fact, the application of other conjugate gradient algorithms would be easier since they usually only involve calculation of the first order information (the gradient, not Hessian), at the cost of a slower convergence rate. Like most iterative search algorithms, Newton's and conjugate gradient algorithms are local methods, i.e., they do not guarantee convergence to the global minimum. Due to the fundamental relationship between the motion recovery objective functions and the epipolar constraints discovered in this paper, at high noise levels all the algorithms unavoidably will suffer from the second eigenmotion (except the case when translation is along the Z-axis). Such an ambiguity is intrinsic to the problem of motion and structure recovery and independent of the choice of objective functions.

In this paper, we have studied in detail the problem of recovering a discrete motion (displacement) from image correspondences. Similar ideas certainly apply to the differential case where the rotation and translation are replaced by angular and linear velocities respectively (Ma et al., 2000). Optimization schemes for the differential case have also been studied by many researchers, including the most recent Bilinear Projection Algorithm (BPA) proposed in Soatto Brockett (1998) and a robust algorithm proposed in Zhang and Tomasi (1999). Similarly, one can show that they all in fact minimize certain normalized versions of the differential epipolar constraint. We hope the Riemannian optimization theoretic viewpoint proposed in this paper will provide a different perspective to revisit these schemes. Although the study of the proposed algorithms is carried out in a calibrated camera framework, due to a clear geometric connection between the calibrated and uncalibrated case (Ma et al., 1998), the same approach and optimization schemes can be generalized with little effort to the uncalibrated case as well. Details will be presented in future work. As we pointed out in this paper, Riemannian optimization algorithms can be easily generalized to products of manifolds. Thus, although

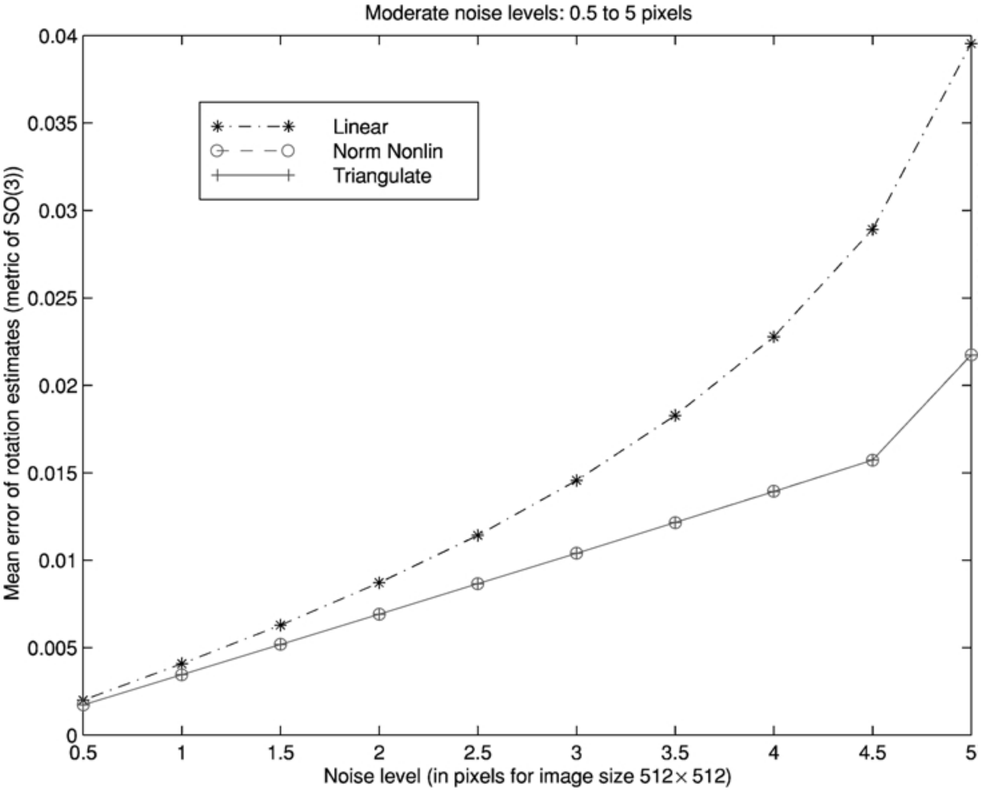

Figure 13. Estimation errors of rotation (in canonical metric on $SO(3)$). 50 trials, rotation 10 degree around Y-axis and translation along X-axis, T/R ratio is 2. Noises range from 0.5 to 5 pixels.

the proposed Newton's algorithm is for 2-frame and a single rigid body motion, it can be easily generalized to multi-frame and multi-body cases. Only the underlying search spaces of optimization will be replaced by products of Lie groups instead of Stiefel manifolds. Comparing to other existing algorithms and conjugate gradient algorithms, the Newton's algorithm involves more computational cost in each iteration step. However, it has the fastest rate of convergence. This is very important when the dimension of the search space is high, for instance, multi-body motion recovery problem. This is because the number of search steps usually increases with the dimension, and each step becomes more costly. We will study these issues in future work.

Appendix A: Optimization on a Product of Riemannian Manifolds

In this appendix, we discuss how to generalize Edelman et al.'s methods (to appear) to the product of Stiefel (or Grassmann) manifolds. Suppose that (M_1, g_1) and (M_2, g_2) are two Riemannian manifolds with Riemannian metrics:

$$g_1(\cdot,\cdot): TM_1 \times TM_1 \to C^\infty(M_1),$$
$$g_2(\cdot,\cdot): TM_2 \times TM_2 \to C^\infty(M_2)$$

where TM_1 is the tangent bundle of M_1, similarly for TM_2. The corresponding Levi-Civita connections (i.e., the unique metric preserving and torsion-free connection) of these manifolds are denoted as:

$$\nabla_1: \mathcal{X}(M_1) \times \mathcal{X}(M_1) \to \mathcal{X}(M_1),$$
$$\nabla_2: \mathcal{X}(M_2) \times \mathcal{X}(M_2) \to \mathcal{X}(M_2)$$

where $\mathcal{X}(M_1)$ stands for the space of smooth vector fields on M_1, similarly for $\mathcal{X}(M_2)$.

Now let M be the product space of M_1 and M_2, i.e., $M = M_1 \times M_2$. Let $i_1: M_1 \to M$ and $i_2: M_2 \to M$ be the natural inclusions and $\pi_1: M \to M_1$ and $\pi_2: M \to M_2$ be the projections. To simplify the notation, we identify TM_1 and TM_2 with $i_{1*}(TM_1)$

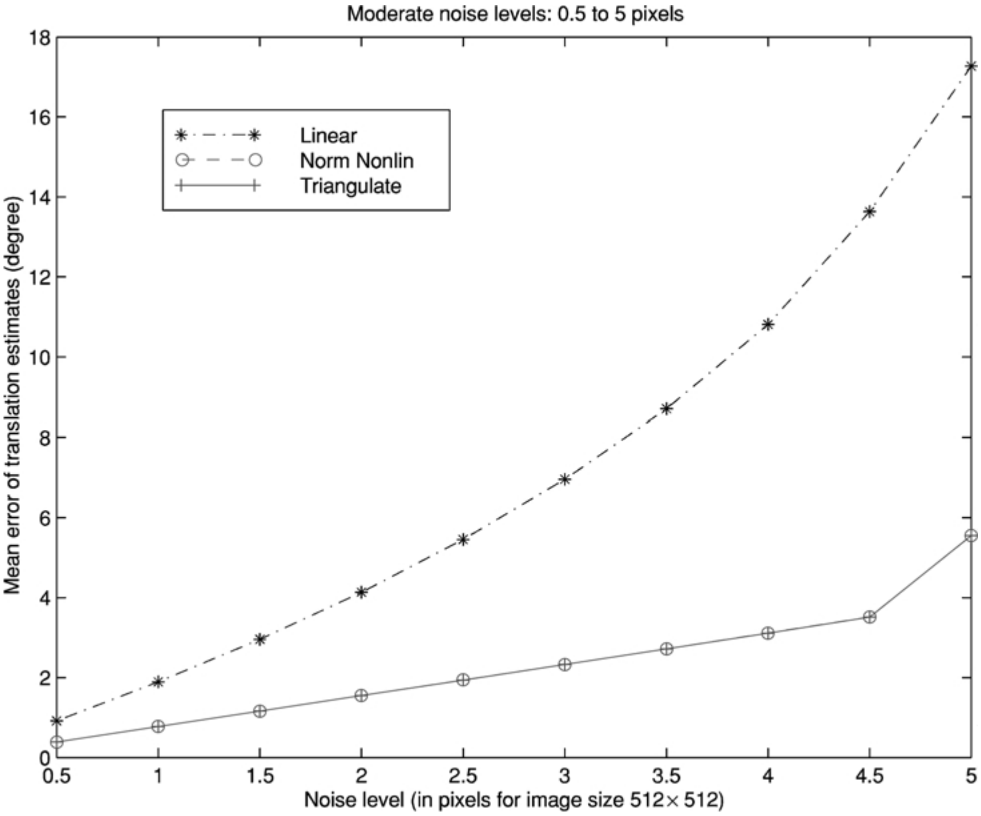

Figure 14. Estimation errors of translation (in degree). 50 trials, rotation 10 degree around Y-axis and translation along X-axis, T/R ratio is 2. Noises range from 0.5 to 5 pixels.

and $i_{2*}(TM_2)$ respectively. Then $TM = TM_1 \times TM_2$ and $\mathcal{X}(M) = \mathcal{X}(M_1) \times \mathcal{X}(M_2)$. For any vector field $X \in \mathcal{X}(M)$ we can write X as the composition of its components in the two subspaces TM_1 and TM_2: $X = (X_1, X_2) \in TM_1 \times TM_2$. The canonical Riemannian metric $g(\cdot, \cdot)$ on M is determined as:

$$g(X, Y) = g_1(X_1, Y_1) + g_2(X_2, Y_2), \quad X, Y \in \mathcal{X}(M).$$

Define a connection ∇ on M as:

$$\nabla_X Y = (\nabla_{1X_1} Y_1, \nabla_{2X_2} Y_2) \in \mathcal{X}(M_1) \times \mathcal{X}(M_2), \\ X, Y \in \mathcal{X}(M).$$

One can directly check that this connection is torsion free and compatible with the canonical Riemannian metric g on M (i.e., preserving the metric) hence it is the Levi-Civita connection for the product Riemannian manifold (M, g). From the construction of ∇, it is also canonical.

According to Edelman et al. (to appear), in order to apply Newton's or conjugate gradient methods on a Riemannian manifold, one needs to know how to explicitly calculate parallel transport of vectors on the manifolds and an explicit expression for geodesics. The reason that Edelman et al.'s methods can be easily generalized to any product of Stiefel (or Grassmann) manifolds is because there are simple relations between the parallel transports on the product manifold and its factor manifolds. The following theorem follows directly from the above discussion of the Levi-Civita connection on the product manifold.

Theorem 3. *Consider $M = M_1 \times M_2$ the product Riemannian manifold of M_1 and M_2. Then for two vector fields $X, Y \in \mathcal{X}(M)$, Y is parallel along X if and only if Y_1 is parallel along X_1 and Y_2 is parallel along X_2.*

As a corollary to this theorem, the geodesics in the product manifold are just the products of geodesics in

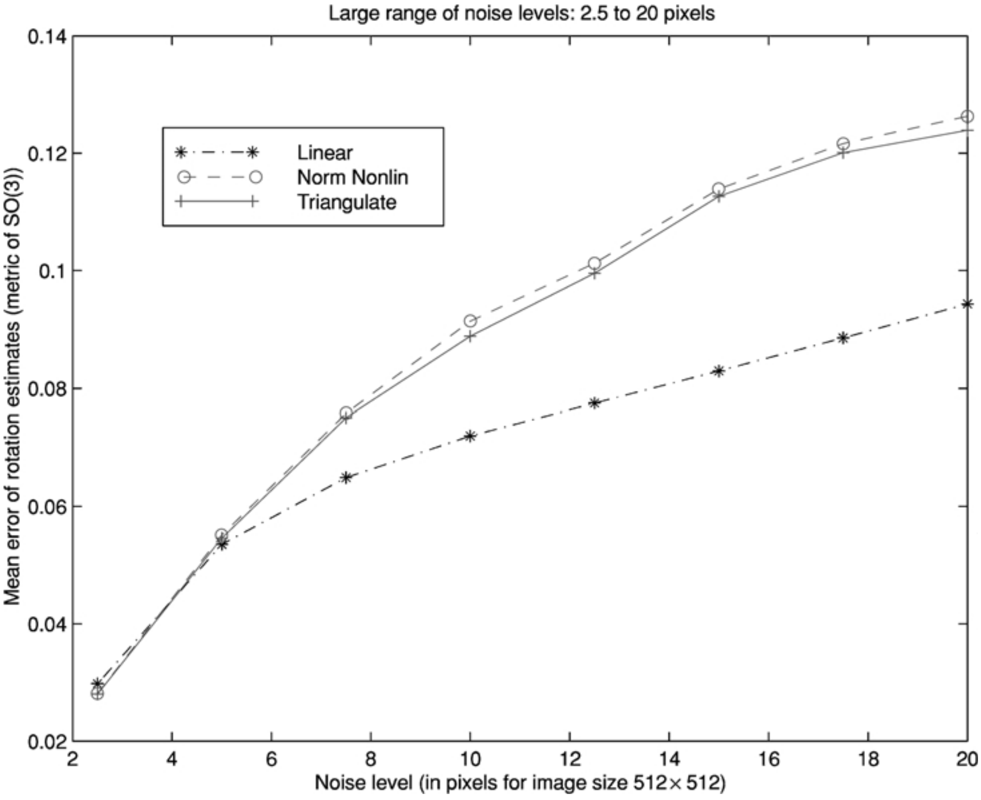

Figure 15. Estimation errors of rotation (in canonical metric on $SO(3)$). 40 points, 50 trials, rotation 10 degree around Y-axis and translation along Z-axis, T/R ratio is 2. Noises range from 2.5 to 20 pixels.

the two factor manifolds. Consequently, the calculation of parallel transport and geodesics in the product space can be reduced to those in each factor manifold.

Appendix B: Optimization on the Essential Manifold

This appendix outlines Newton's algorithm for optimizing a general function defined on the essential manifold, as a product of Stiefel manifolds. For the details of the Newton's or other conjugate gradient algorithms for general Stiefel or Grassmann manifolds please refer to Edelman et al. (to appear).

Generally speaking, in order to generalize Newton's algorithm to a Riemannian manifold, we at least need to know how to compute three things: the gradient, the Hessian of the given function and the geodesics of the manifold. Since the metric of the manifold is no longer the standard Euclidean metric, the computation for these three needs to incorporate the new metric. In the following, we will give general formulae for the gradient and Hessian of a function defined on $SO(3) \times \mathbb{S}^2$ using results from Edelman et al. (to appear). In the next section, we will however give an alternative approach for directly computing these ingredients by using the explicit expression of geodesics on this manifold.

Let $f(R, T)$ be a function defined on the essential manifold or, equivalently, $T_1(SO(3)) \cong SO(3) \times \mathbb{S}^2$ with $R \in SO(3)$ represented by a 3×3 rotation matrix and $T \in \mathbb{S}^2$ a vector of unit length in $\mathbb{R}^3$. Let g_1 and g_2 be the canonical metrics for $SO(3)$ and $\mathbb{S}^2$ respectively and ∇_1 and ∇_2 be the corresponding Levi-Civita connections. Let g and ∇ be the induced Riemannian metric and connection on the product manifold $SO(3) \times \mathbb{S}^2$. The gradient of the function $f(R, T)$ on $SO(3) \times \mathbb{S}^2$ is a vector field $G = \text{grad}(f)$ on $SO(3) \times \mathbb{S}^2$ such that:

$$df(Y) = g(G, Y), \quad \text{for all vector fields } Y \text{ on } SO(3) \times \mathbb{S}^2.$$

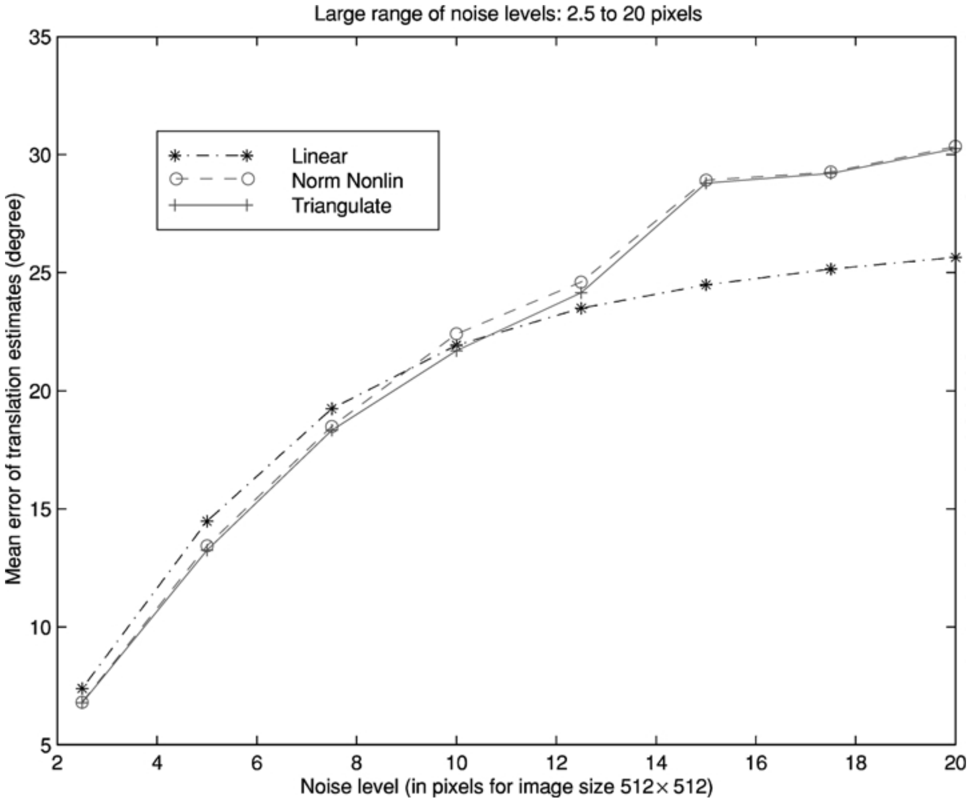

Figure 16. Estimation errors of translation (in degree). 40 points, 50 trials, rotation 10 degree around Y-axis and translation along Z-axis, T/R ratio is 2. Noises range from 2.5 to 20 pixels.

Geometrically, so defined gradient G has the same meaning as in the standard Euclidean case, i.e., G is the direction in which the function f increases the fastest. On $SO(3) \times \mathbb{S}^2$, it can be shown that the gradient is explicitly given as:

$$\begin{aligned} G &= \left(f_R - Rf_R^T R,\ f_T - Tf_T^T T\right)\\ &\in T_R(SO(3)) \times T_T(\mathbb{S}^2)\end{aligned}$$

where $f_R \in \mathbb{R}^{3\times 3}$ is the matrix of partial derivatives of f with respect to the elements of R and $f_T \in \mathbb{R}^3$ is the vector of partial derivatives of f with respect to the elements of T:

$$(f_R)_{ij} = \frac{\partial f}{\partial R_{ij}}, \qquad (f_T)_k = \frac{\partial f}{\partial T_k}, \qquad 1 \leq i, j, k \leq 3.$$

Geometrically, the Hessian of a function is the second order approximation of the function at a given point. However, when computing the second order derivative, unlike the Euclidean case, one should take the *covariant derivative* with respect to the Riemannian metric g on the given manifold.[19] On $SO(3) \times \mathbb{S}^2$, for any $X = (X_1, X_2), Y = (Y_1, Y_2) \in T(SO(3)) \times T(\mathbb{S}^2)$, the Hessian of $f(R, T)$ is explicitly given by:

$$\begin{aligned}\operatorname{Hess} f(X, Y) = {} & f_{RR}(X_1, Y_1) - tr\, f_R^T \Gamma_R(X_1, Y_1)\\ & + f_{TT}(X_2, Y_2) - tr\, f_T^T \Gamma_T(X_2, Y_2)\\ & + f_{RT}(X_1, Y_2) + f_{TR}(Y_1, X_2).\end{aligned}$$

where the Christoffel functions Γ_R for $SO(3)$ and Γ_T for $\mathbb{S}^2$ are:

$$\begin{aligned}\Gamma_R(X_1, Y_1) &= \frac{1}{2} R\left(X_1^T Y_1 + Y_1^T X_1\right),\\ \Gamma_T(X_2, Y_2) &= \frac{1}{2} T\left(X_2^T Y_2 + Y_2^T X_2\right)\end{aligned}$$

and the other terms are:

$$f_{RR}(X_1, Y_1) = \sum_{ij,kl} \frac{\partial^2 f}{\partial R_{ij} \partial R_{kl}} (X_1)_{ij} (Y_1)_{kl},$$

$$f_{TT}(X_2, Y_2) = \sum_{i,j} \frac{\partial^2 f}{\partial T_i \partial T_j} (X_2)_i (Y_2)_j,$$

$$f_{RT}(X_1, Y_2) = \sum_{ij,k} \frac{\partial^2 f}{\partial R_{ij} \partial T_k} (X_1)_{ij} (Y_2)_k,$$

$$f_{TR}(Y_1, X_2) = \sum_{i,jk} \frac{\partial^2 f}{\partial T_i \partial R_{jk}} (Y_1)_i (X_2)_{jk}$$

For Newton's algorithm, we need to find the *optimal updating* tangent vector Δ such that:

$$\text{Hess}\, f(\Delta, Y) = g(-G, Y) \quad \text{for all tangent vectors } Y.$$

Δ is then well-defined and independent of the choice of local coordinate chart. In order to solve for Δ, first find the tangent vector $Z(\Delta) = (Z_1, Z_2) \in T_R(SO(3)) \times T_T(\mathbb{S}^2)$ (in terms of Δ) satisfying the linear equations (see Edelman et al., to appear for a more detailed derivation of the equations):

$$\begin{aligned} &f_{RR}(\Delta_1, Y_1) + f_{TR}(Y_1, \Delta_2) = g_1(Z_1, Y_1) \\ &\qquad \text{for all tangent vectors } Y_1 \in T(SO(3)) \\ &f_{TT}(\Delta_2, Y_2) + f_{RT}(\Delta_1, Y_2) = g_2(Z_2, Y_2) \\ &\qquad \text{for all tangent vectors } Y_2 \in T(\mathbb{S}^2) \end{aligned}$$

From the expression of the gradient G, the vector $\Delta = (\Delta_1, \Delta_2)$ then satisfies the linear equations:

$$\begin{aligned} Z_1 - R\,\text{skew}\big(f_R^T \Delta_1\big) - \text{skew}\big(\Delta_1 f_R^T\big) R &= -\big(f_R - R f_R^T R\big) \\ Z_2 - f_T^T T \Delta_2 &= -\big(f_T - T f_T^T T\big) \end{aligned}$$

with $\Delta_1 R^T$ skew-symmetric and $T^T \Delta_2 = 0$. In the above expression, the notation skew(A) means the skew-symmetric part of the matrix A: skew$(A) = (A - A^T)/2$. For this system of linear equations to be solvable, the Hessian has to be non-degenerate, in other words the corresponding Hessian matrix in local coordinates is invertible. This non-degeneracy depends on the chosen objective function f.

According to Newton's algorithm, knowing Δ, the search state is then updated from (R, T) in direction Δ along geodesics to $(\exp(R, \Delta_1), \exp(T, \Delta_2))$, where $\exp(R, \cdot)$ stands for the exponential map from $T_R(SO(3))$ to $SO(3)$ at point R, similarly for $\exp(T, \cdot)$. Explicit expressions for the geodesics $\exp(R, \Delta_1 t)$ on $SO(3)$ and $\exp(T, \Delta_2 t)$ on $\mathbb{S}^2$ are given in (6) and (7). The overall algorithm can be summarized in the following:

Riemannian Newton's Algorithm for Minimizing $f(R, T)$ on the Essential Manifold

- *At the point* (R, T),
  - *Compute the gradient* $G = (f_R - R f_R^T R,\ f_T - T f_T^T T)$,
  - *Compute* $\Delta = -Hess^{-1}\, G$.
- *Move* (R, T) *in the direction* Δ *along the geodesic to* $(\exp(R, \Delta_1), \exp(T, \Delta_2))$.
- *Repeat if* $\|G\| \geq \epsilon$ *for pre-determined* $\epsilon > 0$.

Since the manifold $SO(3) \times \mathbb{S}^2$ is compact, the Newton algorithm is guaranteed to converge to a (local) extremum of the objective function $f(R, T)$. Note that this algorithm works for any objective function defined on $SO(3) \times \mathbb{S}^2$. For an objective function with non-degenerate Hessian, the Riemannian Newton's algorithm has quadratic (super-linear) rate of convergence (Smith, 1993).

Acknowledgments

We would like to thank Steven T. Smith (at MIT Lincoln Laboratory) for his pioneering work in the optimization techniques on Riemannian manifolds and his valuable suggestions during the preparation of this manuscript. We want to thank Alan Weinstein (at Berkeley Mathematics Department) for proof-reading the manuscript and his insightful comments on some of the differential geometric characterization of the essential manifold. Finally, we thank Stefano Soatto (at Washington University) for inspiring discussions on the topic of estimation on geometric spaces during his stay at Berkeley in the summer of 1998.

Notes

1. In the literature, they are respectively referred to as distance between points and epipolar lines, and gradient-weighted epipolar errors (Zhang, 1998) or epipolar improvement (Weng et al., 1993 a).
2. Stiefel manifold $V(n, k)$ is the set of all orthonormal k-frames in $\mathbb{R}^n$; Grassmann manifold $G(n, k)$is the set of all k dimensional subspaces in $\mathbb{R}^n$. Then canonically, $V(n, k) = O(n)/O(n - k)$

and $G(n,k) = O(n)/O(k) \times O(n-k)$ where $O(n)$ is the orthogonal group of $\mathbb{R}^n$.

3. Without loss of generality we here assume the camera model is a perspective projection with focal length 1. The development for spherical projection case is similar.
4. Given a vector $u = [u_1, u_2, u_3]^T \in \mathbb{R}^3$, the notation $\hat{u}$ denotes the associated skew-symmetric matrix:

$$\hat{u} = \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix} \in \mathbb{R}^{3\times 3}.$$

   Then for any two vectors $u, v \in \mathbb{R}^3$, the cross product $u \times v$ is equal to $\hat{u}v$.
5. This fact has been pointed out by Professor A. Weinstein, Mathematics Department, UC Berkeley.
6. $\mathcal{E}_1$ and $T_1(SO(3))$ have the same local Riemannian structure according to the covering map.
7. The exact formulae for the gradient and Hessian of those objective functions would be extensively long. Hence such a reduction is quite necessary.
8. For a symmetric bilinear form: $b(\cdot,\cdot)$, we only need to know $b(x,x)$ for all x to know the form $b(\cdot,\cdot)$ since we can always evaluate $b(x,y)$ for all x, y using the so-called *polarization* scheme: $b(x,y) = \frac{1}{4}[b(x+y,x+y) - b(x-y,x-y)]$.
9. The spherical projection case is similar and is omitted for simplicity as before.
10. Around a small neighborhood of the actual (R, T), they only differ by higher order terms.
11. Strictly speaking, this is the case only when the noise level is low, i.e., corrupted objective functions are not yet so different from the noise-free one.
12. Since there is no closed form solution to 6-degree polynomial equations, directly minimizing the Rayleigh quotient sum (35) avoids unnecessary transformations hence can be much more efficient.
13. A even function $f(T)$ on $\mathbb{S}^2$ satisfies $f(-T) = f(T)$.
14. We here use the convention that Y-axis is the vertical direction of the image and X-axis is the horizontal direction and the Z-axis coincides with the optical axis of the camera.
15. Rotation and translation magnitudes $\|\omega\|$ and $\|T\|$ are compared with respect to the average depth $\bar{Z}$ of the cloud of 3D points generated; $\|T\| = \bar{Z} * \|\omega\| * ratio$. The translation is expressed in units of focal length.
16. This direction is given by the eigenvector of the Hessian associated with the smallest eigenvalue.
17. Our current implementation of the algorithms in Matlab has a numerical accuracy at 10^{-8}.
18. Since we have the explicit formulae for Hessian, this condition would keep the algorithms from stopping at saddle points.
19. It is a fact in Riemannian geometry that there is a unique metric preserving and torsion-free covariant derivative.

References

Adiv, G. 1989. Inherent ambiguities in recovering 3-D motion and structure from a noisy flow field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):477–489.

Boothby, W.M. 1986. *An Introduction to Differential Manifolds and Riemannian Geometry.* 2nd edn. Academic Press: San Diego.

Danilidis, K. 1997. *Visual Navigation.* Lawrence Erlbaum Associates.

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

Edelman, A., Arias, T., and Smith, S.T. 1998. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Analysis Applications*, 20(2):303-353.

Faugeras, O. 1993. *Three-dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press: Cambridge, MA, USA.

Hartley, R. and Sturm, P. 1997. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157.

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

Jepson, A.D. and Heeger, D.J. 1993. *Spatial Vision in Humans and Robots.* Cambridge University Press: Cambridge. pp. 39–62.

Kanatani, K. 1993. *Geometric Computation for Machine Vision*. Oxford Science Publications: Oxford.

Kobayashi, S. and Nomizu, T. 1996. *Foundations of Differential Geometry: Volume I.* John Wiley & Sons, Inc.: New York.

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

Luong, Q.-T. and Faugeras, O. 1996. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1):43–75.

Ma, Y., Košecká, J., and Sastry, S. 1998. A mathematical theory of camera self-calibration. Electronic Research Laboratory Memorandum, UC Berkeley, UCB/ERL M98/64.

Ma, Y., Košecká, J., and Sastry, S. 2000. Linear differential algorithm for motion recovery: A geometric approach. *IJCV*, 36(1):71–89.

Maybank, S. 1993. *Theory of Reconstruction from Image Motion.* Springer-Verlag: Berlin.

Milnor, J. 1969. *Morse Theory. Annals of Mathematics Studies no. 51.* Princeton University Press: Princeton.

Murray, R.M., Li, Z., and Sastry, S.S. 1994. *A Mathematical Introduction to Robotic Manipulation*. CRC press Inc.: Boca Raton, FL.

Oliensis, J. 1999. A new structure from motion ambiguity. In *IEEE Proceedings from CVPR*, pp. 185–191.

Sastry, S.S. 1999. *Nonlinear Systems: Analysis, Stability and Control.* Springer-Verlag: Berlin.

Smith, S.T. 1993. Geometric optimization methods for adaptive filtering. Ph.D. Thesis, Harvard University, Cambridge, Massachusetts.

Soatto, S. and Brockett, R. 1998. Optimal and suboptimal structure from motion. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*.

Soatto, S. and Perona, P. 1996. Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, 41(3):393–413.

Spetsakis, M. 1994. Models of statistical visual motion estimation. *CVIPG: Image Understanding*, 60(3):300–312.

Spivak, M. 1979. *A Comprehensive Introduction to Differential Geometry:* 2nd ed. Publish or Perish, Inc. Berkeley.

Taylor, C.J. and Kriegman, D.J. 1995. Structure and motion from line segments in multiple images. *IEEE Transactions on PAMI*, 17(11):1021–1032.

Thomas, I. and Simoncelli, E. 1995. Linear structure from motion. Ms-cis-94-61, Grasp Laboratory, University of Pennsylvania.

Tian, T.Y., Tomasi, C., and Heeger, D. 1996. Comparison of approaches to egomotion computation. In *CVPR*.

Weng, J., Huang, T.S., and Ahuja, N. 1989. Motion and structure from two perspective views: Algorithms, error analysis, and error estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):451–475.

Weng, J., Huang, T.S., and Ahuja, N. 1993a. *Motion and Structure from Image Sequences.* Springer Verlag: Berlin.

Weng, J., Huang, T.S., and Ahuja, N. 1993b. Optimal motion and structure estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):864–884.

Zhang, T. and Tomasi, C. 1999. Fast, robust and consistent camera motion estimation. In *Proceedings of CVPR*.

Zhang, Z. 1998. Understanding the relationship between the optimization criteria in two-view motion analysis. In *Proceedings of International Conference on Computer Vision*, Bombay, India, pp. 772–777.