Optimization Criteria and Geometric Algorithms for Motion and Structure Estimation *

Yi Ma Jana Košecká Shankar Sastry

Electronics Research Laboratory
University of California at Berkeley
Berkeley, CA 94720-1774
{mayi, janka, sastry}@robotics.eecs.berkeley.edu

May 14, 1999

Abstract

The prevailing efforts to study the standard formulation of motion and structure recovery have been recently focused on issues of sensitivity and 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 the problem: the choice of objective functions, optimization techniques and the 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", which can all be 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.

Using these analytical results 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 and other types of local minima leading to a consistent interpretations of simulation results over large range of signal-to-noise ratio and variety of configurations.

Key words: 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

*This work is supported by ARO under the MURI grant DAAH04-96-1-0341.

various perspectives. The proposed techniques have varied depending on the type of features they used, the types of assumptions they make about the environment, 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 sensitivity of different algorithms, search for intrinsic local minima and new algorithms are still subjects 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 often inappropriate 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, what 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 clearly reveal the effect of the bas relief ambiguity and other ambiguities on the sensitivity and robustness of the proposed algorithms.

The seminal work of Longuet-Higgins [12] 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 [17, 6, 10, 30] for overviews). The epipolar constraint has been formulated both in a discrete and a differential setting and the recent work of the authors [15] demonstrated the possibility of a parallel development of linear algorithms for both cases: namely using point feature correspondence 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 [13].

While the (analytic) geometrical aspects of the linear approach have been understood, the proposed solutions to the problem have been shown very 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 [3, 24] and experimental setting [28]. The appeal of linear algorithms which use the epipolar constraint (in the discrete case [30, 10, 12, 17] and in the differential case [9, 15, 27]) is the closed form solution to the problem which, in the absence of noise, provides true estimate of the motion. However, a further analysis of linear techniques reveals an inherent bias in the translation estimates [9]. Attempts made to compensate for the bias slightly improve the performance of the linear techniques [10].

Such attempts to remove the bias have led to different choice 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) [31, 13, 33, 8]. These techniques either require iterative numerical optimization [30, 22] or use Monte-Carlo simulations [9] to sample the space of the unknown parameters. Extensive experiments revealed problems with convergence when initialized far away from the true solution [28]. Since nonlinear objective functions have been obtained from quite different approaches, it is necessary to understand the relationship among all the existing objective functions. Although a preliminary comparison has been made in [33], 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)**, "minimizing epipolar constraint", "minimizing (geometrically or statistically[1]) normalized epipolar constraint" [31, 13, 33], "minimizing reprojection error" [31], and "triangulation" [7] can all be unified in a single geometric optimization procedure, the so called "optimal triangulation". As a by-product of this approach, a much simpler triangulation method than [7] is given along with the proposed algorithm. A highlight of our method is an iterative scheme between motion and structure without introducing any 3D scale (or depth).

Different objective functions have been used in different optimization techniques [8, 31, 26]. Horn [8] 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 [26] are some 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; and "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 [15] or in [22]. 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) [21, 5], 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 [23]). It also provides us 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 [22, 32]. For the differential case, recent studies [22] 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 (*i.e.*, these local extrema are independent of the choice of objective functions). The bas

[1]In the literature, they are respectively referred to as distance between points and epipolar lines, and gradient-weighted epipolar errors [33] or epipolar improvement [31].

relief ambiguity, in general, can characterized as the most sensitive direction in which the rotation and translation estimates are prone to be confound with each other (for example, see [1, 31, 22] for a more detailed analysis). Here we apply the same line of thought to the discrete case. Since the bas relief effect is evident only when the field of view and the depth variation of the scene are small, we here are more interested in characterizing, besides the bas relief ambiguity, other intrinsic extrema which may show up at a high noise level even for a general configuration, *i.e.*, with large base line, field of view and depth variation. 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 certain bifurcation of the objective function and usually result in a sudden 90^o 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, and under various motion and structure configurations.

Paper outline: Section 2, 3 and 4 rely on some familiarity with Edelman *et al*'s work [5] and some background of Riemannian geometry (good references for Riemannian geometry are [25, 11]).[2] Section 2 shows how to generalize optimization schemes on a single Riemannian manifold to their product space. Section 3 then studies the intrinsic Riemannian structure of the essential manifold (the space of all essential matrices). Section 4 outlines how to optimize a general objective function on the essential manifold using the (Riemannian) Newton's algorithm. Section 5 spells out in detail explicit formulae of gradient, Hessian and geodesics, which are needed by the (Riemannian) Newton's algorithm for optimizing various objective functions associated with the motion recovery problem. Different objective functions proposed in the literature are unified in Section 6 by a single optimization procedure proposed for estimating optimal structure and motion altogether. This procedure gives clear answers to both questions **(i)** and **(ii)**. Section 7 gives a geometric characterization of extrema of any function on the essential manifold. Among all the possible ambiguities, we characterize those which most likely occur in the motion and structure recovery problem. Sensitivity study and experimental comparison between different objective functions are given in Section 8. Section 7 and 8 give a detailed account of the question **(iii)**.

2 Optimization on Riemannian Manifold Preliminaries

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 [21] gave a detailed treatment of a theory of optimization on general Riemannian manifolds; Edelman, Arias and Smith [5] further studied the case of Stiefel and Grassmann manifolds,[3] 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 $g : TM \times TM \to C^\infty(M)$ is the Riemannian metric on M (and TM denotes the tangent space

[2] Readers who are not familiar with differential geometry terms may skip technical details in these sections without losing much continuity.

[3] 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 $\mathbb{R}^n$.

of M). An intuitive comparison between the Euclidean and Riemannian nonlinear optimization schemes is illustrated in Figure 1.

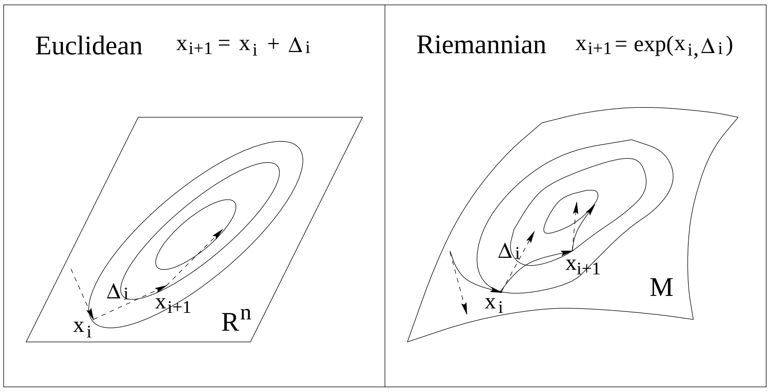

Figure 1: Comparison between the Euclidean and Riemannian nonlinear optimization schemes. At each step, an (optimal) updating vector $\Delta_i \in T_{x_i}M$ is computed using the Riemannian metric at x_i. Then the state variable is updated by following the geodesic from x_i in the direction Δ_i by a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

Conventional approaches for solving such an optimization problem are usually application 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, the new analysis of [5] shows that, for "nice" manifolds, *i.e.*, 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 so obtained algorithms is no longer parameterization dependent, and in addition they typically have polynomial complexity and super-linear (quadratic) rate of convergence [21]. An intuitive comparison between the conventional update-then-project approach and the Riemannian method is demonstrated in Figure 2 (where M is illustrated as the standard 2D sphere $\mathbb{S}^2 = \{x \in \mathbb{R}^3 \mid \|x\|^2 = 1\}$).

One of the purposes of this paper is to apply these new Riemannian optimization schemes to solve the nonlinear optimization problem of recovering 3D motion from image correspondences. As we will soon see the underlying Riemannian manifold for this problem (the so called essential manifold) is a product of Stiefel manifolds instead of a single one. We first need to generalize Edelman *et al*'s methods [5] to the product of Stiefel (or Grassmann) manifolds. Suppose (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

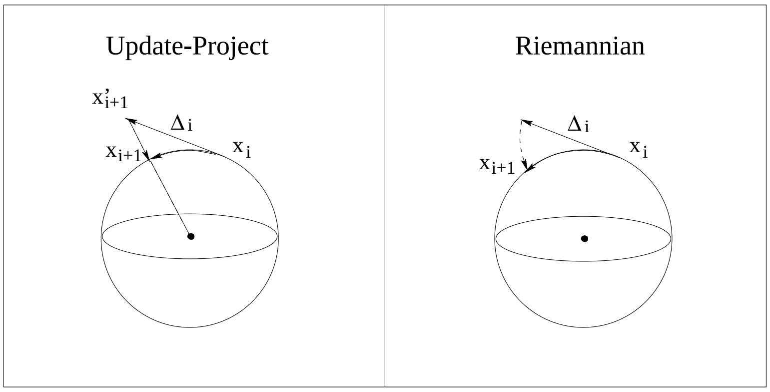

Figure 2: Comparison between the conventional update-then-project approach and the Riemannian scheme. For the conventional method, the state x_i is first updated to x'_{i+1} according to the updating vector Δ_i and then x'_{i+1} is projected back to the manifold at x_{i+1}. For the Riemannian scheme,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

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 $i_{1*}(TM_1)$ 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:

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

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* [5], 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 1 *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 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.

3 Riemannian Structure of the Essential Manifold

In this section we study the Riemannian structure of the essential manifold, which plays an important role in motion recovery from image correspondences (for details see [15]). To simplify notation, for any vector $u = (u_1, u_2, u_3)^T \in \mathbb{R}^3$, the notation $\widehat{u}$ means the associated skew-symmetric matrix:

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

Then for any two vectors $u, v \in \mathbb{R}^3$, the cross product $u \times v$ is equal to $\widehat{u}v$.

Camera motion is modeled as rigid body motion in $\mathbb{R}^3$. The displacement of the camera belongs to the special Euclidean group $SE(3)$:

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

where $SO(3) \in \mathbb{R}^{3\times 3}$ is the space of rotation matrices (orthogonal matrices with determinant $+1$). An element $g = (R, S)$ 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 $x = (x^1, x^2, x^3)^T \in \mathbb{R}^3$ and $y = (y^1, y^2, y^3)^T \in \mathbb{R}^3$ respectively. Then, x and y are associated by:

$$y = R^T x + S. \tag{2}$$

We here use a transpose on R to simply later notation. Without loss of generality, (perspective projection) images of x and y are given by $p = (p^1, p^2, 1)^T = (\frac{x^1}{x^3}, \frac{x^2}{x^3}, 1)^T \in \mathbb{R}^3$ and $q = (q^1, q^2, 1)^T = (\frac{y^1}{y^3}, \frac{y^2}{y^3}, 1)^T \in \mathbb{R}^3$ respectively.[4] The main purpose of this paper is to study the following:

> **Motion and structure recovery problem:** *For a given set of corresponding images points $\{(p_i, q_i)\}_{i=1}^N$, how to recover the camera motion (R, S) and the 3D coordinates (3D structure) of the points that these image points correspond to?*

It is well known in computer vision literature that two corresponding images p and q satisfy the so called *epipolar constraint* [12]:

$$p^T R \widehat{S} q = 0. \tag{3}$$

A good property of this constraint is that it decouples the problem of motion recovery from that of structure recovery. The first part of this paper will be devoted to recovering motion from directly

[4] We here assume the camera model is a perspective projection with focal length 1. The spherical projection case is similar and omitted here for simplicity.

using this constraint or its variations. In Section 6, we will see how this constraint has to be adjusted when we consider recovering motion and structure simultaneously.

The matrix $R\widehat{S}$ in the epipolar constraint is the so called *essential matrix*, and the *essential manifold* is defined to be the space of all such matrices, denoted by:

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

$SO(3)$ 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. In particular $\widehat{S} \in so(3)$. As we will show later in this paper, for the problem of recovering camera motion (R, S) from the corresponding image points p and q, the associated objective functions are usually functions of the epipolar constraint. Hence they are of the form $f(E) \in \mathbb{R}$ with $E \in \mathcal{E}$. Moreover such functions in general are homogeneous in E. Thus the problem of motion recovery is equivalent to optimize functions defined on the so called *normalized essential manifold*:

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

Note that $\frac{1}{2}tr(\widehat{S}^T\widehat{S}) = S^TS$. In order to study the optimization problem on such a manifold, it is crucial to understand the Riemannian structure of the normalized essential manifold. 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 [2] (such metric is unique up to a constant scale). In matrix form, this metric is given explicitly by:

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

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_*:

$$T_R(SO(3)) = R_*(so(3)) = \{R\widehat{S} | \widehat{S} \in so(3)\}.$$

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

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

Since the tangent bundle of a Lie group is trivial [25], $T(SO(3))$ is then equivalent to the product $SO(3) \times so(3)$. $T(SO(3))$ can then be expressed as:

$$T(SO(3)) = \{(R, R\widehat{S}) \mid R \in SO(3), \widehat{S} \in so(3)\} \cong SO(3) \times so(3).$$

If we identify the tangent space of $so(3)$ with itself, then the metric g_0 of $SO(3)$ induces a canonical metric on the tangent bundle $T(SO(3))$:

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

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 $T(SO(3))$,

$$T_1(SO(3)) \cong \{(R, R\widehat{S}) \mid R \in SO(3), \widehat{S} \in so(3), \frac{1}{2}tr(\widehat{S}^T\widehat{S}) = 1\}.$$

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* [5], $SO(3)$ 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 2 *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.*

However, 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 [15]). The natural covering map from $T_1(SO(3))$ to $\mathcal{E}_1$ is:

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

The inverse of this map is given by:

$$h^{-1}(R\widehat{S}) = \left\{(R, R\widehat{S}),\ (R\exp(-\widehat{S}\pi), R\widehat{S})\right\}.$$

Comment 1 *As we know, the two pairs of rotation and translation corresponding to the same normalized essential matrix $R\widehat{S}$ are $(R, \widehat{S})$ and $(R\exp(-\widehat{S}\pi), \exp(\widehat{S}\pi)\widehat{S})$. As pointed out by Weinstein, 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 $SO(3)$.*

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 $\mathcal{E}_1$ can be converted to optimizing $f(R, S)$ on $T_1(SO(3))$.[5] Generalizing Edelman *et al*'s methods to the product Riemannian manifolds, we may obtain intrinsic geometric Newton's or conjugate gradient algorithms for solving such an optimization problem. Due to Theorem 2, we can simply choose the induced Euclidean metric on $T_1(SO(3))$ and explicitly

[5] Although the topological structures of $\mathcal{E}_1$ and $T_1(SO(3))$ are different, the nonlinear optimization only relies on local Riemannian metric and this identification will not affect effectiveness of the search schemes.

give these intrinsic algorithms in terms of the matrix representation of $T_1(SO(3))$. Since this Euclidean metric is the same as the intrinsic metrics, 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 $T_1(SO(3))$.

4 Optimization on the Essential Manifold

Let $f(R, S)$ be a function defined on $T_1(SO(3)) \cong SO(3) \times \mathbb{S}^2$ with $R \in SO(3)$ represented by a 3×3 rotation matrix and $S \in \mathbb{S}^2$ a vector of unit length in $\mathbb{R}^3$. This section gives Newton's algorithm for optimizing a function defined on this manifold (please refer to [5] for the details of the Newton's or other conjugate gradient algorithms for general Stiefel or Grassmann manifolds).

In order to generalize Newton's algorithm to a Riemannian manifold, we need to know how to compute three things: the gradient, the Hessian of a 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 [5]. 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 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 $f(R, S)$ on $SO(3) \times \mathbb{S}^2$ is a vector field $G = \mathrm{grad}(f)$ on $SO(3) \times \mathbb{S}^2$ such that:

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

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:

$$G = (f_R - R f_R^T R,\ f_S - S f_S^T S) \quad \in T_R(SO(3)) \times T_S(\mathbb{S}^2)$$

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 $f_S \in \mathbb{R}^3$ is the vector of partial derivatives of f with respect to the elements of S, *i.e.*,

$$(f_R)_{ij} = \frac{\partial f}{\partial R_{ij}}, \quad (f_S)_k = \frac{\partial f}{\partial S_k}, \quad 1 \le i, j, k \le 3.$$

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.[6] 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 $f(R, S)$ is explicitly given by:

$$\begin{aligned} \text{Hess } f(X, Y) &= f_{RR}(X_1, Y_1) - tr f_R^T \Gamma_R(X_1, Y_1) \\ &+ f_{SS}(X_2, Y_2) - tr f_S^T \Gamma_S(X_2, Y_2) \\ &+ f_{RS}(X_1, Y_2) + f_{SR}(Y_1, X_2). \end{aligned}$$

[6] It is a fact in Riemannian geometry that there is a unique metric preserving and torsion-free covariant derivative.

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

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

and the other terms are:

$$\begin{aligned}f_{RR}(X_1, Y_1) &= \sum_{ij,kl} \frac{\partial^2 f}{\partial R_{ij} \partial R_{kl}} (X_1)_{ij} (Y_1)_{kl}, \quad f_{SS}(X_2, Y_2) = \sum_{i,j} \frac{\partial^2 f}{\partial S_i \partial S_j} (X_2)_i (Y_2)_j, \\ f_{RS}(X_1, Y_2) &= \sum_{ij,k} \frac{\partial^2 f}{\partial R_{ij} \partial S_k} (X_1)_{ij} (Y_2)_k, \quad f_{SR}(Y_1, X_2) = \sum_{i,jk} \frac{\partial^2 f}{\partial S_i \partial R_{jk}} (Y_1)_i (X_2)_{jk}\end{aligned}$$

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 $Z(\Delta) = (Z_1, Z_2) \in T_R(SO(3)) \times T_S(\mathbb{S}^2)$ (in terms of Δ) satisfying the linear equations:

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

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

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

with $R^T \Delta_1$ skew-symmetric and $S^T \Delta_2 = 0$. In the above expression, the notation skew(A) means the skew-symmetric part of the matrix A: $\text{skew}(A) = (A - A^T)/2$. 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, S) in direction Δ along geodesics to $(\exp(R, \Delta_1), \exp(S, \Delta_2))$, where $\exp(R, \cdot)$ stands for the exponential map from $T_R(SO(3))$ to $SO(3)$ at point R, similarly for $\exp(S, \cdot)$. Explicit expressions for the geodesics $\exp(R, \Delta_1 t)$ on $SO(3)$ and $\exp(S, \Delta_2 t)$ on $\mathbb{S}^2$ will be given in the next section. The overall algorithm can be summarized in the following:

Riemannian Newton's algorithm for minimizing $f(R, S)$ on the essential manifold:

- *At the point* (R, S),
  - *Compute the gradient* $G = (f_R - R f_R^T R,\ f_S - S f_S^T S)$,
  - *Compute* $\Delta = -\ \textit{Hess}^{-1} G$.
- *Move* (R, S) *in the direction* Δ *along the geodesic to* $(\exp(R, \Delta_1), \exp(S, \Delta_2))$.

- *Repeat if* $\|G\| \geq \epsilon$ *for pre-determined* $\epsilon > 0$.

Since the manifold $SO(3) \times \mathbb{S}^2$ is compact, this algorithm is guaranteed to converge to a (local) extremum of the objective function $f(R, S)$. 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 [21].

5 Optimal Motion Recovery

In this section, we apply the Riemannian Newton's algorithm to various objective functions associated with the motion recovery problem in computer vision. Relationship among different objective functions will be studied in detail in the section after.

5.1 Minimizing Epipolar Constraint

From preceding sections, we know that two corresponding image points $p, q \in \mathbb{R}^3$ satisfy the so called epipolar constraint:

$$p^T R \widehat{S} q \quad = \quad 0 \tag{4}$$

where $R \in SO(3)$ and $S \in \mathbb{S}^2$ are relative rotation and translation between the two image frames.[7] Thus to recover the motion R, S from a given set of image correspondences $p_i, q_i \in \mathbb{R}^3$, $i = 1, \ldots, N$, it is natural to minimize the following objective function:

$$F(R, S) \quad = \quad \sum_{i=1}^{N} (p_i^T R \widehat{S} q_i)^2, \quad p_i, q_i \in \mathbb{R}^3, (R, S) \in SO(3) \times \mathbb{S}^2. \tag{5}$$

In this section, we apply the Newton's algorithm introduced in the previous section to solve this problem. We will give explicit formulae for calculating all the ingredients needed: geodesics, gradient G, Hessian Hess F and the optimal updating vector $\Delta = -\mathrm{Hess}^{-1} G$ (and we will show later how these formulae can be extensively reused for obtaining corresponding formulae of all the other objective functions). It is well known that an explicit formula for the Hessian is also important for sensitivity analysis of the motion estimation [3]. Further, using this formula, we will be able to show that, under certain conditions, the Hessian is guaranteed non-degenerate, whence the Newton's algorithm has quadratic rate of convergence.

Instead of using formulae given in the previous section, the computation of the gradient and Hessian can also be carried out by using explicit formulae of geodesics on these manifolds. On $SO(3)$, the formula for the geodesic at R in the direction $\Delta_1 \in T_R(SO(3)) = R_*(so(3))$ is:

$$R(t) = \exp(R, \Delta_1 t) = R \exp \widehat{\omega} t = R(I + \widehat{\omega} \sin t + \widehat{\omega}^2 (1 - \cos t)) \tag{6}$$

where $t \in \mathbb{R}, \widehat{\omega} = R^T \Delta_1 \in so(3)$. The last equation is called the *Rodrigues' formula* (see [19]). $\mathbb{S}^2$ (as a Stiefel manifold) also has very simple expression of geodesics. At the point S along the direction $\Delta_2 \in T_S(\mathbb{S}^2)$ the geodesic is given by:

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

[7] In the literature, for different definitions of the rotation R, the matrix R in the above expression might differ by a transpose.

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

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

$$dF(\Delta) \;=\; \frac{dF(R(t),S(t))}{dt}\Big|_{t=0} = \sum_{i=1}^{N} p_i^T R\widehat{S}q_i(p_i^T\Delta_1\widehat{S}q_i + p_i^T R\widehat{\Delta}_2 q_i), \tag{8}$$

$$\begin{aligned}\text{Hess } F(\Delta,\Delta) \;&=\; \frac{d^2F(R(t),S(t))}{dt^2}\Big|_{t=0} \\ = \sum_{i=1}^{N}\left[p_i^T(\Delta_1\widehat{S} + R\widehat{\Delta}_2)q_i\right]^2 \;&+\; p_i^T R\widehat{S}q_i\left[p_i^T(-\Delta_1\Delta_1^T R\widehat{S} - \Delta_2^T\Delta_2 R\widehat{S} + 2\Delta_1\widehat{\Delta}_2)q_i\right].\end{aligned} \tag{9}$$

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

$$G = \sum_{i=1}^{N} p_i^T R\widehat{S}q_i\left(p_i q_i^T\widehat{S}^T - R\widehat{S}q_i p_i^T R,\; \widehat{q}_i R^T p_i - S p_i^T R\widehat{q}_i^T S\right) \tag{10}$$

It is direct to check that $R^T G_1 \in so(3)$ and $S^T G_2 = 0$, so the G given by the above expression is a vector in $T_R(SO(3)) \times T_S(\mathbb{S}^2)$.

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

$$\begin{aligned}\text{Hess } F(X,Y) \;&=\; \frac{1}{4}[\text{Hess } F(X+Y, X+Y) - \text{Hess } F(X-Y, X-Y)] \\ &=\; \sum_{i=1}^{N} p_i^T(X_1\widehat{S} + R\widehat{X}_2)q_i p_i^T(Y_1\widehat{S} + R\widehat{Y}_2)q_i \\ &+\; p_i^T R\widehat{S}q_i\left[p_i^T\left(-\frac{1}{2}(X_1Y_1^T + Y_1X_1^T)R\widehat{S} - X_2^T Y_2 R\widehat{S} + (X_1\widehat{Y}_2 + Y_1\widehat{X}_2)\right)q_i\right]\end{aligned} \tag{11}$$

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 quadratic rate of convergence by Theorem 3.4 of Smith [21].

Theorem 3 *Consider the objective function $F(R,S)$ as above. 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:*

$$p_i^T E q_i = 0, \quad E \in \mathbb{R}^{3\times 3}, \quad i = 1, \ldots, N.$$

If so, the Riemannian Newton's algorithm has 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 $p_i^T R\widehat{S}q_i \equiv 0$. The Hessian is then simplified to:

$$\text{Hess } F(\Delta,\Delta) = \sum_{i=1}^{N}\left[p_i^T(\Delta_1\widehat{S} + R\widehat{\Delta}_2)q_i\right]^2.$$

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

$$p_i^T(\Delta_1 \widehat{S} + R\widehat{\Delta}_2)q_i = 0, \quad i = 1, \ldots, N.$$

Since we also have

$$p_i^T R\widehat{S} q_i = 0, \quad i = 1, \ldots, N.$$

Then both $\Delta_1 \widehat{S} + R\widehat{\Delta}_2$ and $R\widehat{S}$ 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

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

■

Remark 1 *In the previous theorem, regarding the* 3×3 *matrix* E *in the equations* $p_i^T E q_i = 0$ *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* $\{(p_i, q_i)\}_{i=1}^N, N \geq 8$ *to guarantee the Hessian non-degenerate whence the iterative search algorithm 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 [17]. However, the five point technique usually leads to many (up to twenty) ambiguous solutions, as pointed out by Horn [8]. Moreover, numerical errors usually make the algorithm not work exactly on the essential manifold and the extra solutions for the equations* $p_i^T E q_i = 0$ *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 [17, 15]) to have a unique solution. A heuristic explanation is that the objective function here is a quadratic form of the epipolar constraint which the linear algorithm is directly based on.*

Returning to the Newton's algorithm, assume that the Hessian is non-degenerate, *i.e.*, invertible. Then, we need to solve for the optimal updating vector Δ such that $\Delta = \text{Hess}^{-1} G$, or equivalently:

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

Pick five linearly independent vectors, *i.e.*, a basis of $T_R(SO(3)) \times T_S(\mathbb{S}^2)$: $E^j, j = 1, \ldots, 5$. One then obtains five linear equations:

$$\text{Hess } F(E^j, \Delta) = -dF(E^j), \quad j = 1, \ldots, 5.$$

Since the Hessian is invertible, these five linear equations uniquely determine Δ. In particular, one can choose the simplest basis such that for $j = 1, 2, 3$: $E^j = (R\widehat{e}_j, 0)$ with e_j, $j = 1, 2, 3$ the standard basis for $\mathbb{R}^3$, and for $j = 4, 5$: $E^j = (0, e_j)$ such that $\{S, e_4, e_5\}$ form an orthonormal basis for $\mathbb{R}^3$. The vectors e_4, e_5 can be obtained using Gram-Schmidt 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:

$$(A)_{jk} = \text{Hess } F(E^j, E^k), \quad (\mathbf{b})_j = -dF(E^j), \quad j,k = 1,\ldots,5.$$

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 $\omega = (a_1, a_2, a_3)^T \in \mathbb{R}^3$ and $v = a_4 e_4 + a_5 e_5 \in \mathbb{R}^3$. Then for the optimal updating vector $\Delta = (\Delta_1, \Delta_2)$, we have $\Delta_1 = R\widehat{\omega}$ and $\Delta_2 = v$. We now summarize the Riemannian Newton algorithm for the optimal motion recovery, which can be directly implemented.

Riemannian Newton's algorithm for motion recovery from the objective function:

$$F(R,S) = \sum_{i=1}^{N} (p_i^T R\widehat{S} q_i)^2, \quad p_i, q_i \in \mathbb{R}^3, (R,S) \in SO(3) \times \mathbb{S}^2.$$

- *At the point $(R,S) \in SO(3) \times \mathbb{S}^2$, compute the optimal updating vector $\Delta = -Hess^{-1}G$:*
  - *Compute the vectors e_4, e_5 from S using Gram-Schmidt process and obtain the five basis tangent vectors $E^j \in T_R(SO(3)) \times T_S(\mathbb{S}^2), 1 \le j \le 5$ as defined in the above,*
  - *Compute the 5×5 matrix $(A)_{jk} = Hess\ F(E^j, E^k), 1 \le j,k \le 5$ using:*

$$\begin{aligned} Hess\ F(X,Y) &= \sum_{i=1}^{N} p_i^T (X_1\widehat{S} + R\widehat{X}_2) q_i p_i^T (Y_1\widehat{S} + R\widehat{Y}_2) q_i \\ &+ p_i^T R\widehat{S} q_i \left[p_i^T \left(-\frac{1}{2}(X_1 Y_1^T + Y_1 X_1^T) R\widehat{S} - X_2^T Y_2 R\widehat{S} + (X_1\widehat{Y}_2 + Y_1\widehat{X}_2) \right) q_i \right], \end{aligned}$$

  - *Compute the 5 dimensional vector $(\mathbf{b})_j = -dF(E^j), 1 \le j \le 5$ using:*

$$dF(X) = \sum_{i=1}^{N} p_i^T R\widehat{S} q_i (p_i^T X_1 \widehat{S} q_i + p_i^T R\widehat{X}_2 q_i),$$

  - *Compute the vector $\mathbf{a} = (a_1, a_2, a_3, a_4, a_5)^T \in \mathbb{R}^5$ such that $\mathbf{a} = A^{-1}\mathbf{b}$,*
  - *Define $\omega = (a_1, a_2, a_3)^T \in \mathbb{R}^3$ and $v = a_4 e_4 + a_5 e_5 \in \mathbb{R}^3$. Then the optimal updating vector*

$$\Delta = -Hess^{-1}G = (R\widehat{\omega}, v).$$

- *Move (R,S) in the direction Δ along the geodesic to $(\exp(R,\Delta_1), \exp(S,\Delta_2))$, using the formula for geodesics on $SO(3)$ and $\mathbb{S}^2$ respectively:*

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

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

- *Repeat if $\|\mathbf{b}\| \ge \epsilon$ for some pre-determined $\epsilon > 0$.*

Remark 2 *From calculations above, we note that one can consider a more general objective function with a (positive) weights $w_i \in \mathbb{R}^+$ associated with each image correspondence (p_i, q_i):*

$$F(R,S) = \sum_{i=1}^{N} w_i (p_i^T R\widehat{S} q_i)^2, \quad p_i, q_i \in \mathbb{R}^3, (R,S) \in SO(3) \times \mathbb{S}^2.$$

For example, one may choose $w_i^{-1} = \|p_i\|^2 \|q_i\|^2$ to convert the image points from perspective projection to spherical projection. Then, in the above algorithm, the expressions of the gradient, dF and the Hessian only need to be slightly modified.

5.2 Minimizing Normalized Epipolar Constraints

Although the epipolar constraint (4) gives the only necessary (depth independent) condition that image pairs have to satisfy, motion estimates obtained from minimizing the objective function (5):

$$F(R,S) \;=\; \sum_{i=1}^{N} (p_i^T R\widehat{S} q_i)^2, \quad p_i, q_i \in \mathbb{R}^3, (R,S) \in SO(3) \times \mathbb{S}^2. \tag{12}$$

are not necessarily statistically or geometrically optimal for the commonly used noise model of image correspondences. In general, in order to get less biased estimates, we need to *normalize* (or weight) the epipolar constraints properly, which has been initially observed in [31]. In this section, we will give a brief account of these normalized versions of epipolar constraints. These normalized versions in general are still functions defined on the essential manifold. The reason will become clear in the next section when we see that these normalizations in fact can be unified by a single procedure of getting optimal estimates of motion and structure.

We here discuss this issue for the perspective projection case.[8] In the perspective projection case, coordinates of image points p and q are of the form $p = (p^1, p^2, 1)^T \in \mathbb{R}^3$ and $q = (q^1, q^2, 1)^T \in \mathbb{R}^3$. Suppose that the actual measured image coordinates of N pairs of image points are:

$$p_i = \tilde{p}_i + x_i, \quad q_i = \tilde{q}_i + y_i, \quad i = 1, \ldots, N \tag{13}$$

where $\tilde{p}_i$ and $\tilde{q}_i$ are ideal (noise free) image coordinates, $x_i = (x_i^1, x_i^2, 0)^T \in \mathbb{R}^3$ and $y_i = (y_i^1, y_i^2, 0)^T \in \mathbb{R}^3$ and $x_i^1, x_i^2, y_i^1, y_i^2$ are independent Gaussian random variables of identical distribution $N(0, \sigma^2)$. Substituting p_i and q_i into the epipolar constraint (4), we obtain:

$$p_i^T R\widehat{S} q_i \;=\; x_i^T R\widehat{S} \tilde{q}_i + \tilde{p}_i^T R\widehat{S} y_i + x_i^T R\widehat{S} y_i.$$

Since the image coordinates p_i and q_i usually are magnitude larger than x_i and y_i, one can omit the last term in the equation above. Then $p_i^T R\widehat{S} q_i$ are independent random variables *approximately* of Gaussian distribution $N(0, \sigma^2(\|\widehat{e}_3 R\widehat{S} q_i\|^2 + \|p_i^T R\widehat{S}\widehat{e}_3\|^2))$ where $e_3 = (0,0,1)^T \in \mathbb{R}^3$. If we assume the *a prior* distribution of the motion (R,S) is uniform, the maximum *a posterior* (MAP) estimates of (R,S) is then the global minimum of the objective function:

$$F_s(R,S) \;=\; \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S} q_i)^2}{\|\widehat{e}_3 R\widehat{S} q_i\|^2 + \|p_i^T R\widehat{S}\widehat{e}_3\|^2}, \quad p_i, q_i \in \mathbb{R}^3, (R,S) \in SO(3) \times \mathbb{S}^2. \tag{14}$$

[8] The spherical projection case is similar and is omitted for simplicity.

We here use F_s to denote the *statistically normalized* objective function associated with the epipolar constraint. This objective function is also referred in the literature under the name *gradient criteria* [13] or *epipolar improvement* [30]. Therefore, we have:

$$(R, S)_{MAP} \quad \approx \quad \arg\min F_s(R, S) \tag{15}$$

Note that in the noise free case, F_s achieves zero, just like the unnormalized objective function F of equation (5). Asymptotically, MAP estimates approach the unbiased minimum mean square estimates (MMSE). So, in general, the MAP estimates give less biased estimates than the unnormalized objective function F.

Note that F_s is still a function defined on the manifold $SO(3) \times \mathbb{S}^2$. The discussion given in Section 4 about optimizing a general function defined on the essential manifold certainly applies to F_s. Moreover, note that 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:

$$\|\widehat{e}_3 R\widehat{S} q_i\|^2 + \|p_i^T R\widehat{S}\widehat{e}_3\|^2 = (e_1^T R\widehat{S} q_i)^2 + (e_2^T R\widehat{S} q_i)^2 + (p_i^T R\widehat{S} e_1)^2 + (p_i^T R\widehat{S} e_1)^2 \tag{16}$$

where $e_1 = (1, 0, 0)^T \in \mathbb{R}^3$ and $e_2 = (0, 1, 0)^T \in \mathbb{R}^3$. That is, 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 of the first and second order derivatives $dF(\Delta)$ and $\mathrm{Hess}F(\Delta, \Delta)$ of the unnormalized objective function F to express those for the normalized objective F_s by simply replacing p_i or q_i with e_1 or e_2 at proper places. This is one of the reasons 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 $(e_1^T R\widehat{S} q_i)^2, (e_2^T R\widehat{S} q_i)^2, (p_i^T R\widehat{S} e_1)^2, (p_i^T R\widehat{S} e_1)^2$ and $(p_i^T R\widehat{S} q_i)^2$, as oppose 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. But the normalized one gives statistically much better estimates, as we will demonstrate in the experiment section.

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

$$F_g(R, S) \quad = \quad \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S} q_i)^2}{\|\widehat{e}_3 R\widehat{S} q_i\|^2} + \frac{(p_i^T R\widehat{S} q_i)^2}{\|p_i^T R\widehat{S}\widehat{e}_3^T\|^2}, \quad p_i, q_i \in \mathbb{R}^3, (R, S) \in SO(3) \times \mathbb{S}^2. \tag{17}$$

We here use F_g to denote this *geometrically normalized* objective function. For a more detailed derivation and geometric meaning of this objective function see [13, 33]. 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.

The relationship between the statistically normalized objective function F_s and the geometrically normalized objective function F_g will be clearly revealed in the next section when we study the optimal motion and structure recovery as a constrained optimization problem. As we know from [16], 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 $(R, S) \in SE(3)$ with $S \in \mathbb{S}^2$ and $R = e^{\widehat{\omega}\theta}$ for some $\omega \in \mathbb{S}^2$ and $\theta \in \mathbb{R}$. If $\omega = (0, 0, 1)^T$ and $S = (s_1, s_2, 0)^T$, *i.e.*, the translation direction is in the image plane, then, since R and $\widehat{e}_3$ now commute, the expression $\|\widehat{e}_3 R\widehat{S} q_i\|^2$ simply becomes $\|S\|^2 = 1$. Similarly, $\|p_i^T R\widehat{S}\widehat{e}_3^T\|^2 = \|S\|^2 = 1$. 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, S).[9] 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.[10] Therefore, in certain cases, minimizing the objective function F which is directly related to the epipolar constraint is not necessarily a wrong thing to do.

6 Optimal Triangulation

Note that, in the presence of noise, for the motion (R, S) 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:

$$p_i^T R\widehat{S} q_i \neq 0, \quad i = 1, \ldots, N. \tag{18}$$

Consequently, if one directly uses p_i and q_i to recover the 3D location of the point to which the two images p_i and q_i correspond, the two rays corresponding to p_i and q_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 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 (13), in order to obtain the optimal (MAP) estimates of camera motion and a consistent 3D structure reconstruction, in principle we need to solve the following optimization problem:

Optimal Triangulation Problem: *Seek camera motion (R, S) and points $\tilde{p}_i \in \mathbb{R}^3$ and $\tilde{q}_i \in \mathbb{R}^3$ on the image plane such that they minimize the distance from p_i and q_i:*

$$F_t(R, S, \tilde{p}_i, \tilde{q}_i) = \sum_{i=1}^{N} \|\tilde{p}_i - p_i\|^2 + \|\tilde{q}_i - q_i\|^2 \tag{19}$$

subject to the conditions:

$$\tilde{p}_i^T R\widehat{S} \tilde{q}_i = 0, \quad \tilde{p}_i^T e_3 = 1, \quad \tilde{q}_i^T e_3 = 1, \quad i = 1, \ldots, N. \tag{20}$$

We here use F_t to denote the objective function for triangulation. This objective function is also referred in literature as the reprojection error. Unlike [7], we do not assume a known essential matrix $R\widehat{S}$. Instead we seek $\tilde{p}_i, \tilde{q}_i$ *and* (R, S) which minimize the objective function F_t given by (19). The objective function F_t then implicitly depends on the variables (R, S) through the constraints (20). Clearly, 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:

$$\min_{R,S,\tilde{p}_i,\tilde{q}_i} \sum_{i=1}^{N} \|\tilde{p}_i - p_i\|^2 + \|\tilde{q}_i - q_i\|^2 + \lambda_i \tilde{p}_i^T R\widehat{S} \tilde{q}_i + \beta_i(\tilde{p}_i^T e_3 - 1) + \gamma_i(\tilde{q}_i^T e_3 - 1). \tag{21}$$

[9] Around a small neighborhood of the actual (R, S), they only differ by high order terms.

[10] 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.

The necessary conditions for minima of this objective function are:

$$2(\tilde{p}_i - p_i) + \lambda_i R\widehat{S}\tilde{q}_i + \beta_i e_3 = 0 \tag{22}$$
$$2(\tilde{q}_i - q_i) + \lambda_i \widehat{S}^T R^T \tilde{p}_i + \gamma_i e_3 = 0 \tag{23}$$

Under the necessary conditions, we obtain:

$$\left\{ \begin{array}{lcl} \tilde{p}_i & = & p_i - \frac{1}{2}\lambda_i \widehat{e}_3^T \widehat{e}_3 R\widehat{S}\tilde{q}_i \\ \tilde{q}_i & = & q_i - \frac{1}{2}\lambda_i \widehat{e}_3^T \widehat{e}_3 \widehat{S}^T R^T \tilde{p}_i \\ \tilde{p}_i^T R\widehat{S}\tilde{q}_i & = & 0 \end{array} \right. \tag{24}$$

where λ_i is given by:

$$\lambda_i = \frac{2(p_i^T R\widehat{S}\tilde{q}_i + q_i^T \widehat{S}^T R^T \tilde{p}_i)}{\tilde{q}_i^T \widehat{S}^T R^T \widehat{e}_3^T \widehat{e}_3 R\widehat{S}\tilde{q}_i + \tilde{p}_i^T R\widehat{S}\widehat{e}_3^T \widehat{e}_3 \widehat{S}^T R^T \tilde{p}_i} \tag{25}$$

or

$$\lambda_i = \frac{2p_i^T R\widehat{S}\tilde{q}_i}{\tilde{q}_i^T \widehat{S}^T R^T \widehat{e}_3^T \widehat{e}_3 R\widehat{S}\tilde{q}_i} = \frac{2q_i^T \widehat{S}^T R^T \tilde{p}_i}{\tilde{p}_i^T R\widehat{S}\widehat{e}_3^T \widehat{e}_3 \widehat{S}^T R^T \tilde{p}_i}. \tag{26}$$

Substituting (24) and (25) into F_t, we obtain:

$$F_t(R, S, \tilde{p}_i, \tilde{q}_i) = \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S}\tilde{q}_i + \tilde{p}_i^T R\widehat{S} q_i)^2}{\|\widehat{e}_3 R\widehat{S}\tilde{q}_i\|^2 + \|\tilde{p}_i^T R\widehat{S}\widehat{e}_3^T\|^2} \tag{27}$$

and using (24) and (26) instead, we get:

$$F_t(R, S, \tilde{p}_i, \tilde{q}_i) = \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S}\tilde{q}_i)^2}{\|\widehat{e}_3 R\widehat{S}\tilde{q}_i\|^2} + \frac{(\tilde{p}_i^T R\widehat{S} q_i)^2}{\|\tilde{p}_i^T R\widehat{S}\widehat{e}_3^T\|^2}. \tag{28}$$

Geometrically, both expressions of F_t are the distances from the image points p_i and q_i to the epipolar lines specified by $\tilde{p}_i, \tilde{q}_i$ and (R, S). Equations (27) and (28) give explicit formulae of the residue of $\|\tilde{p}_i - p_i\|^2 + \|\tilde{q}_i - q_i\|^2$ as p_i, q_i being triangulated by $\tilde{p}_i, \tilde{q}_i$. Note that the terms in F_t are normalized *crossed epipolar constraints* between p_i and $\tilde{q}_i$ or between $\tilde{p}_i$ and q_i. These expressions of F_t can be further used to solve for (R, S) 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. *Initialize $\tilde{p}_i^*(R, S), \tilde{q}_i^*(R, S)$ as p_i, q_i.*

2. **Motion:** *Update (R, S) by minimizing $F_t^*(R, S) = F_t(R, S, \tilde{p}_i^*(R, S), \tilde{q}_i^*(R, S))$ given by (27) or (28) as a function defined on the manifold $SO(3) \times \mathbb{S}^2$.*

3. **Structure (Triangulation):** *Solve for $\tilde{p}_i^*(R, S)$ and $\tilde{q}_i^*(R, S)$ which minimize the objective function F_t (19) with respect to (R, S) computed in the previous step.*

4. *Back to step 2 until updates are small enough.*

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

$$F_t^*(R,S) \;=\; \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S}\tilde{q}_i^* + \tilde{p}_i^{*T} R\widehat{S} q_i)^2}{\|\widehat{e}_3 R\widehat{S}\tilde{q}_i^*\|^2 + \|\tilde{p}_i^{*T} R\widehat{S}\widehat{e}_3^T\|^2} \;=\; \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S}\tilde{q}_i^*)^2}{\|\widehat{e}_3 R\widehat{S}\tilde{q}_i^*\|^2} + \frac{(\tilde{p}_i^{*T} R\widehat{S} q_i)^2}{\|\tilde{p}_i^{*T} R\widehat{S}\widehat{e}_3^T\|^2} \tag{29}$$

is a sum of normalized crossed epipolar constraints. It is a function defined on the manifold $SO(3) \times \mathbb{S}^2$ again hence can be minimized using the Riemannian Newton's algorithm, which is essentially the same as minimizing the normalized epipolar constraint (14) studied in the preceding section. The algorithm ends when (R,S) is already a minimum of F_t^*. It can be shown that if (R,S) is a critical point of F_t^*, then $(R, S, \tilde{p}_i^*(R,S), \tilde{q}_i^*(R,S))$ is necessarily a critical point of the original objective function F_t given by (19).

At step 3, for a fixed (R,S), $\tilde{p}_i^*(R,S)$ and $\tilde{q}_i^*(R,S)$ can be computed by minimizing the distance $\|\tilde{p}_i - p_i\|^2 + \|\tilde{q}_i - q_i\|^2$ for each pair of image points. Let $t_i \in \mathbb{R}^3$ be the normal vector (of unit length) to the (epipolar) plane spanned by $(\tilde{q}_i, S)$. Given such a t_i, $\tilde{p}_i$ and $\tilde{q}_i$ are determined by:

$$\tilde{p}_i(t_i) = \frac{\widehat{e}_3 t_i' t_i'^T \widehat{e}_3^T p_i + \widehat{t'}_i^T \widehat{t'}_i e_3}{e_3^T \widehat{t'}_i^T \widehat{t'}_i e_3}, \quad \tilde{q}_i(t_i) = \frac{\widehat{e}_3 t_i t_i^T \widehat{e}_3^T q_i + \widehat{t}_i^T \widehat{t}_i e_3}{e_3^T \widehat{t}_i^T \widehat{t}_i e_3}, \tag{30}$$

where $t_i' = Rt_i$. Then the distance can be explicitly expressed as:

$$\|\tilde{q}_i - q_i\|^2 + \|\tilde{p}_i - p_i\|^2 \;=\; \|q_i\|^2 + \frac{t_i^T A_i t_i}{t_i^T B_i t_i} + \|p_i\|^2 + \frac{t_i'^T C_i t_i'}{t_i'^T D_i t_i'}, \tag{31}$$

where

$$\begin{array}{rcl} A_i &=& I - (\widehat{e}_3 q_i q_i^T \widehat{e}_3^T + \widehat{q}_i \widehat{e}_3 + \widehat{e}_3 \widehat{q}_i), \quad B_i = \widehat{e}_3^T \widehat{e}_3 \\ C_i &=& I - (\widehat{e}_3 p_i p_i^T \widehat{e}_3^T + \widehat{p}_i \widehat{e}_3 + \widehat{e}_3 \widehat{p}_i), \quad D_i = \widehat{e}_3^T \widehat{e}_3 \end{array}. \tag{32}$$

Then the problem of finding $\tilde{p}_i^*(R,S)$ and $\tilde{q}_i^*(R,S)$ becomes one of finding t_i^* which minimizes the function of a sum of two *singular Rayleigh quotients*:

$$\min_{t_i^T S = 0, t_i^T t_i = 1} V(t_i) \;=\; \frac{t_i^T A_i t_i}{t_i^T B_i t_i} + \frac{t_i^T R^T C_i R t_i}{t_i^T R^T D_i R t_i}. \tag{33}$$

This is an optimization problem on a unit circle $\mathbb{S}^1$ in the plane orthogonal to the vector S (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 S, n_1, n_2 form an orthonormal basis of $\mathbb{R}^3$, then $t_i = \cos(\theta) n_1 + \sin(\theta) n_2$ with $\theta \in \mathbb{R}$. We only need to find θ^* which minimizes the function $V(t_i(\theta))$. 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 $t_i(\theta_1)$ and $t_i(\theta_2)$ correspond to normal vectors of the two planes spanned by (q_i, S) and $(R^T p_i, S)$ respectively (if p_i, q_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 [7]. 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 $t_i(\theta)$, t_i^* can also be obtained by solving a 6-degree polynomial equation, as shown in [7] (and an approximate version results in solving a 4-degree polynomial equation [30]). However, the method given in [7] 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 $\mathbb{S}^1$:

$$\sin(2\theta) = \frac{2\lambda}{1+\lambda^2}, \quad \cos(2\theta) = \frac{1-\lambda^2}{1+\lambda^2}, \quad \lambda \in \mathbb{R}, \tag{34}$$

then it is straightforward to show from the Rayleigh quotient sum (33) that the necessary condition for minima of $V(t_i)$ is equivalent to a 6-degree polynomial equation in λ.[11] The triangulated pairs $(\tilde{p}_i, \tilde{q}_i)$ and the camera motion (R, S) obtained from the minimization automatically give a consistent (optimal) 3D structure reconstruction by two-frame stereo.

The optimal triangulation algorithm successfully resolves some mysteries about the epipolar geometry. First, it clarifies the relationship between previously obtained objective functions based on normalization, including F_s and F_g. In the expressions of F_t, if we simply approximate $\tilde{p}_i, \tilde{q}_i$ by p_i, q_i respectively, we may obtain the normalized versions of epipolar constraints for recovering camera motion. From (27) we get:

$$F_s(R, S) = \sum_{i=1}^{N} \frac{4(p_i^T R\widehat{S}q_i)^2}{\|\widehat{e}_3 R\widehat{S}q_i\|^2 + \|p_i^T R\widehat{S}\widehat{e}_3^T\|^2} \tag{35}$$

or from (28) we have:

$$F_g(R, S) = \sum_{i=1}^{N} \frac{(p_i^T R\widehat{S}q_i)^2}{\|\widehat{e}_3 R\widehat{S}q_i\|^2} + \frac{(p_i^T R\widehat{S}q_i)^2}{\|p_i^T R\widehat{S}\widehat{e}_3^T\|^2} \tag{36}$$

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 $p_i - \tilde{p}_i$ and $q_i - \tilde{q}_i$. Although such subtle difference between F_s, F_g and F_t has previously been pointed out in [33], 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 (29) appears as a key intermediate objective function in an approach which initially intends to minimize the so called reprojection error given by (19). The approach of minimizing reprojection error was previously considered in the computer vision literature as an alternative to methods which directly minimize epipolar constraints [31, 7]. We here see that they are in fact profoundly related. Further, the crossed epipolar constraint F_t^* given by (29) for motion estimation and the sum of singular Rayleigh quotients $V(t_i)$ given by (33) for triangulation are simply different expressions of the reprojection error under different conditions. In summary, "minimizing (normalized) epipolar constraints" [13, 33], "triangulation" [7] and "minimizing reprojection errors" [31] are all deeply related to each other. They are in fact different (approximate) versions of the same procedure of obtaining *the* optimal motion and structure estimates from image correspondences.

[11] Since there is no closed form solution to 6-degree polynomial equations, directly minimizing the Rayleigh quotient sum (33) avoids unnecessary transformations hence can be much more efficient.

7 Critical Values and Ambiguous Solutions

Note that all objective functions F, F_s, F_g and F_t^* that we have encountered are even functions in $S \in \mathbb{S}^2$.[12] 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. Although such an objective function could have numerous critical points, numbers of different types of critical points have to satisfy the so called *Morse inequalities*, which is associated to topological invariants of the underlying manifold (see [18]). 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 [18], by changing the local coordinates of a neighborhood around a critical point, say $q \in M$, the function f locally looks like:

$$-x_1^2 - \cdots - x_\lambda^2 + x_{\lambda+1}^2 + \cdots + x_n^2. \tag{37}$$

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 λ^{th} homology group $H_\lambda(M, \mathbb{K})$ of M over any field $\mathbb{K}$, the so called λ^{th} *Betti number*. Then the Morse inequalities are given by:

$$\begin{aligned}
\sum_{\lambda=i}^{0}(-1)^{i-\lambda}D_\lambda &\leq \sum_{\lambda=i}^{0}(-1)^{i-\lambda}C_\lambda, \quad i = 0, 1, 2, \ldots n-1 && (38)\\
\sum_{\lambda=0}^{n}(-1)^{\lambda}D_\lambda &= \sum_{\lambda=0}^{n}(-1)^{\lambda}C_\lambda. && (39)
\end{aligned}$$

Note that $\sum_{\lambda=0}^{n}(-1)^\lambda D_\lambda$ is the *Euler characteristic* $\chi(M)$ of the manifold M.

Now we compute the dimension of homology groups of $SO(3) \times \mathbb{RP}^2$. Since $SO(3) \simeq \mathbb{RP}^3$,[13] the manifold $SO(3) \times \mathbb{RP}^2$ is the same as $\mathbb{RP}^3 \times \mathbb{RP}^2$. From algebraic topology, if we pick the field $\mathbb{K}$ to be $\mathbb{Z}_2$,[14] homology groups of $\mathbb{RP}^n$ are given by:

$$H_\lambda(\mathbb{RP}^n, \mathbb{Z}_2) = \mathbb{Z}_2, \quad \lambda = 0, 1, \ldots, n. \tag{40}$$

Moreover, homology groups (over a field) of a product space $M \times M'$ are given by the so called *tensor formula of Künneth*: $H_\lambda(M \times M') = \bigoplus_{p+q=\lambda} H_p(M) \otimes H_q(M')$. This allows us to compute the homology groups of $SO(3) \times \mathbb{RP}^2$ over $\mathbb{Z}_2$:

$$H_\lambda(SO(3) \times \mathbb{RP}^2, \mathbb{Z}_2) = \begin{cases} \mathbb{Z}_2, & \lambda = 0 \\ \mathbb{Z}_2 \oplus \mathbb{Z}_2, & \lambda = 1 \\ \mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2, & \lambda = 2 \\ \mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2, & \lambda = 3 \\ \mathbb{Z}_2 \oplus \mathbb{Z}_2, & \lambda = 4 \\ \mathbb{Z}_2, & \lambda = 5 \\ 0, & \lambda \geq 6 \end{cases} \tag{41}$$

[12] A even function $f(S)$ on $\mathbb{S}^2$ satisfies $f(-S) = f(S)$.

[13] $\mathbb{RP}^3$ is the three dimensional real projective plane – the set of all one dimensional subspaces in $\mathbb{R}^4$. $SO(3)$ is diffeomorphic to $\mathbb{RP}^3$ is because the three dimensional sphere $\mathbb{S}^3$ is a double covering of $SO(3)$ which is clear from the quaternion representation of $SO(3)$.

[14] $\mathbb{Z}_2$ is the field of $\{0, 1\}$.

Then $D_\lambda = 1, 2, 3, 3, 2, 1$ for $\lambda = 0, 1, 2, 3, 4, 5$ respectively. In particular, this gives 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 $F(R, S)$.

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. The search process, if not properly initialized, may stop at any kind of the above critical points, especially the local minima.[15] Moreover, like any nonlinear system, when increasing the noise level, new critical points can be introduced through bifurcation (see [20]). An example of bifurcation is shown in Figure 3. The Morse inequalities

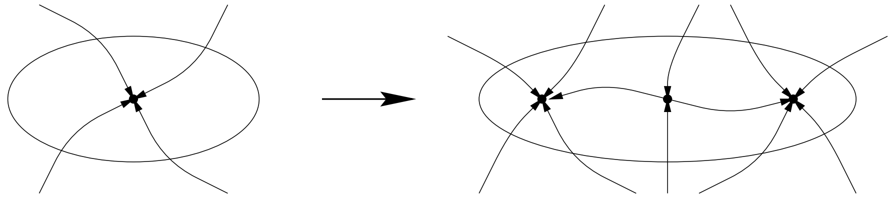

Figure 3: Bifurcation which preserves the Euler characteristic by introducing a pair of saddle and node. The indices of the two circled regions are both 1.

give necessary conditions of how the patterns of critical points may change from one to another. Although, in general, many different types of bifurcations may occur when increasing the noise level, the *fold bifurcation* illustrated in Figure 3 occurs most frequently in the motion and structure estimation problem. We therefore need to understand how such a bifurcation may occur and how it affects the motion estimates.

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. From the preceding section, we know all objective functions are more or less equivalent to the epipolar constraints, especially when the translation is parallel to the image plane. If we let $E = R\widehat{S}$ to be the essential matrix, then we can rewrite the epipolar constraint as $p_i^T E q_i = 0, i = 1, \ldots, N$. Then minimizing the objective function F is (approximately) equivalent to the following least square problem:

$$\min \|Ae\|^2 \tag{42}$$

where A is a $N \times 9$ matrix function of entries of p_i and q_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 $A^T A$. In the presence of noise, e is simply chosen to be the eigenvector corresponding to the least eigenvalue of $A^T A$. At a low noise level, this eigenvector in general gives a good initial estimate of the essential matrix.[16] However, at a certain high noise level, the smallest two eigenvalues may switch roles, as do the two corresponding eigenvectors – topologically, a bifurcation as shown in Figure 3 occurs. Let us denote these two eigenvectors as e and e'. Since they both are eigenvectors of the symmetric matrix $A^T A$, 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,

[15] Maxima and saddles have a at least one dimensional unstable submanifold hence the Newton's algorithm rarely ends at these points.

[16] Such estimate might be biased towards the bas relief ambiguity.

we have $tr(\widehat{S}^T R^T R' \widehat{S}') = 0$. It is well known that the rotation estimate R is usually much less sensitive to noise than the translation estimates S. Therefore, approximately, we have $R \approx R'$ hence $tr(\widehat{S}^T \widehat{S}') \approx 0$. That is, S and S' are almost orthogonal to each other. This phenomena is very common in the motion estimation problem: at a high noise level, the translation estimate may suddenly change direction by roughly 90^o, 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 [22]), the second eigenmotion ambiguity can usually be detected by checking the positive depth constraints. A similar situation of the 90^o flip in the motion estimates for the differential case has previously been reported in [4].

Figure 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^o, 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^o about the Y-axis and the actual translation is along the X-axis.[17] The ratio between translation and rotation is 2.[18] In the figures, "+" marks the actual translation, "$*$" marks the translation estimate from linear algorithm (see [17] 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^o away from the actual translation. 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 Figure 3. Clearly, in Figure 4, there are 2 maxima, 2 saddles and 1 minima on $\mathbb{RP}^2$; in Figure 5, there are 2 maxima, 3 saddles and 2 minima. Both patterns give the Euler characteristic of $\mathbb{RP}^2$ as 1.

From the Figure 5, we can see that the 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 pair and exist for general configuration even when both the FOV and depth variation are sufficiently large. This is a quite different interpretation of this effect which was previously attributed to the bas relief ambiguity. The bas relief effect is only evident when FOV and depth variation is small, but the second eigenmotion ambiguity may show up for general configurations. Bas relief estimates are statistically meaningful since they characterize a sensitive direction in which translation and rotation are the most likely to be confound. 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, Figure 6 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.[19] The (translation) estimates of 20 runs, marked as "$\circ$", 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.

[17] 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.

[18] Rotation and translation magnitudes are compared with respect to the center of the cloud of 3D points generated.

[19] This direction is given by the eigenvector of the Hessian associated with the smallest eigenvalue.

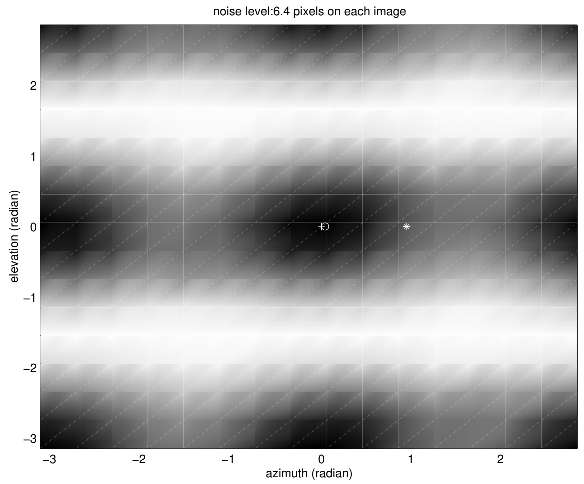

Figure 4: Value of objective function F_s for all S at noise level 6.4 pixels (rotation fixed at the estimate from the nonlinear optimization). Estimation errors: 0.014 in rotation estimate (in terms of the canonical metric on $SO(3)$) and 2.39^o in translation estimate (in terms of angle).

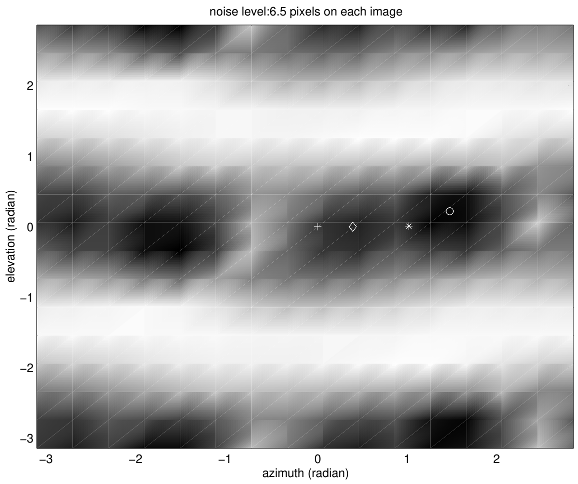

Figure 5: Value of objective function F_s for all S at noise level 6.5 pixels (rotation fixed at the estimate from the nonlinear optimization). Estimation errors: 0.227 in rotation estimate (in terms of the canonical metric on $SO(3)$) and 84.66^o in translation estimate (in terms of angle).

8 Experiments and Sensitivity Analysis

In this section, we clearly demonstrate by experiments the relationship among the linear algorithm (as in [17]), 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 algorithms' performance.

We follow the same line of thought as the analysis of the differential case in [22]. 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 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 base line is equivalent to changing the signal-to-noise ratio, we hence perform the simulations at a fixed base line 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 base line. 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

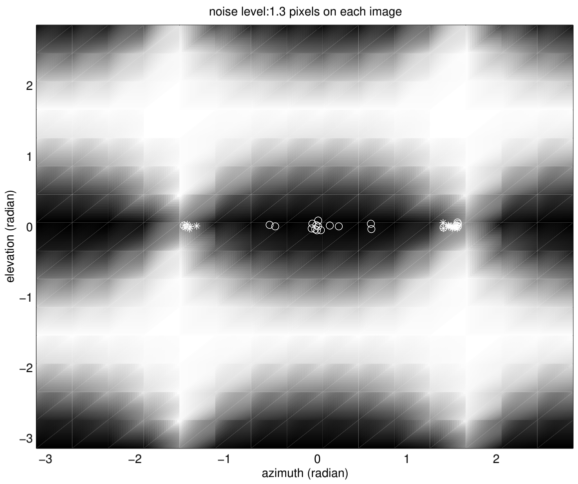

Figure 6: Bas relief ambiguity. FOV is 20^o and the random cloud depth varies from 100 to 150 units of focal length. Translation is along the X-axis and rotation around the Y-axis. Rotation magnitude is 2^o. T/R ratio is 2. 20 runs at the noise level 1.3 pixels.

of truncated pyramid with a field of view (FOV) 90^o, 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 [17]), instead of from the ground truth.[20] 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;[21] and 2. The smallest eigenvalue of the Hessian matrix is positive.[22]

8.1 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 things: 1. The signal-to-noise ratio and 2. Whether the simulation results are still statistically meaningful while varying the noise level.

Figure 7, 8, 9 and 10 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

[20] We like to point out that evaluation based on initializing from the ground truth is misleading for using these algorithms in real applications since it usually does not reveal correctly the relationship between the linear algorithm and nonlinear algorithms.

[21] Our current implementation of the algorithms in Matlab has a numerical accuracy at 10^{-8}.

[22] Since we have the explicit formulae for Hessian, this condition would keep the algorithms from stopping at saddle points.

algorithm minimizing the epipolar constraints F, "normal" is the Riemannian Newton's algorithm minimizing the normalized epipolar constraints F_s.

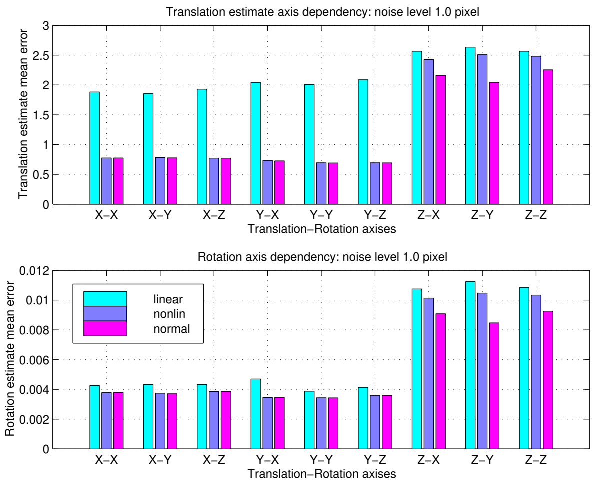

Figure 7: Axis dependency: estimation errors in rotation and translation at noise level 1.0 pixel. T/R ratio = 2 and rotation = 10^o.

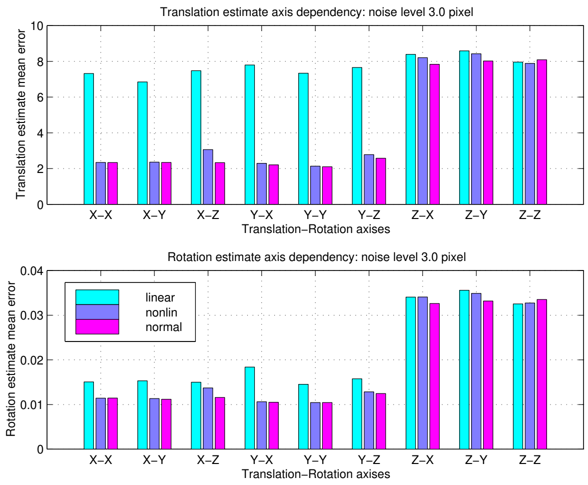

Figure 8: Axis dependency: estimation errors in rotation and translation at noise level 3.0 pixels. T/R ratio = 2 and rotation = 10^o.

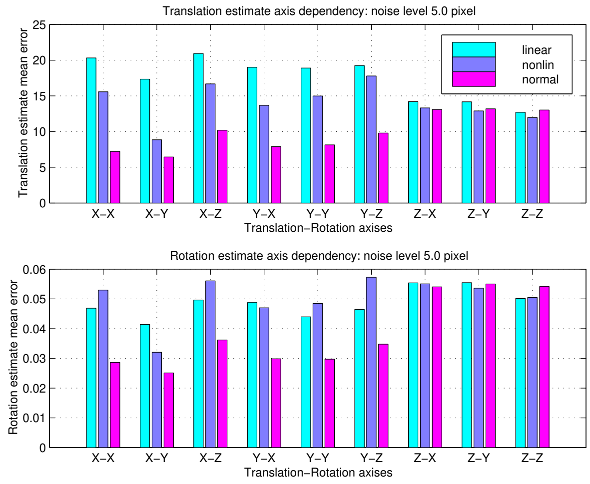

Figure 9: Axis dependency: estimation errors in rotation and translation at noise level 5.0 pixel. T/R ratio = 2 and rotation = 10^o.

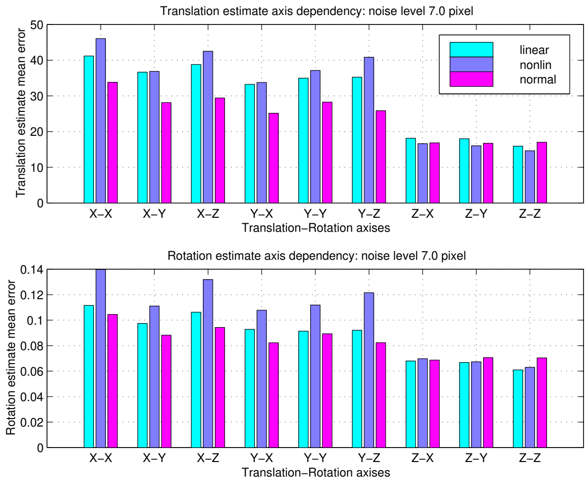

Figure 10: Axis dependency: estimation errors in rotation and translation at noise level 7.0 pixels. T/R ratio = 2 and rotation = 10^o.

By carefully comparing the simulation results in Figure 7, 8, 9 and 10, 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 (Figure 7 and 8). At higher noise levels, this is no longer true (Figure 9 and 10), 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 (Figure 7), 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 (Figure 8, 9 and 10), 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.

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

1. All three algorithms are the most robust to the increasing 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 (Figure 7, 8 and 9), 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 translating along the Z-axis.
3. At a noise level of 7 pixels (Figure 10), 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 [29]. 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 [29], 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 10 is therefore a generic picture of the axis dependency profile for the differential or small base-line case (for more details see [16]).

8.2 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 [33]. This is demonstrated in Figure 11 and 12 – "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 Figure 13 and 14 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.

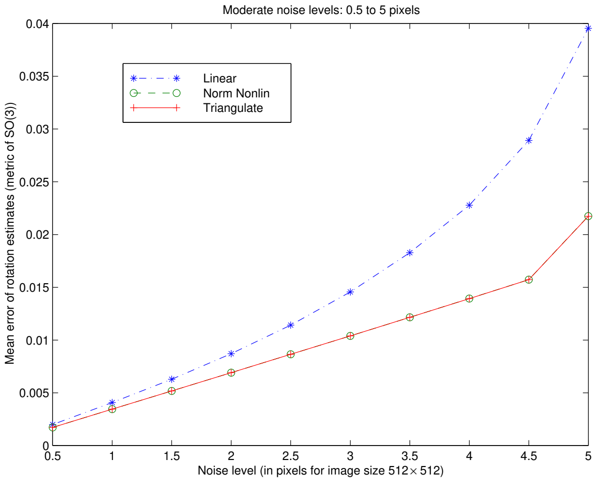

Figure 11: Estimation errors of rotation (in canonical metric on $SO(3)$). 50 trials, rotation 10 degree around Y-axis and translation along X-axis, T/R ratio is 2. Noises range from 0.5 to 5 pixels.

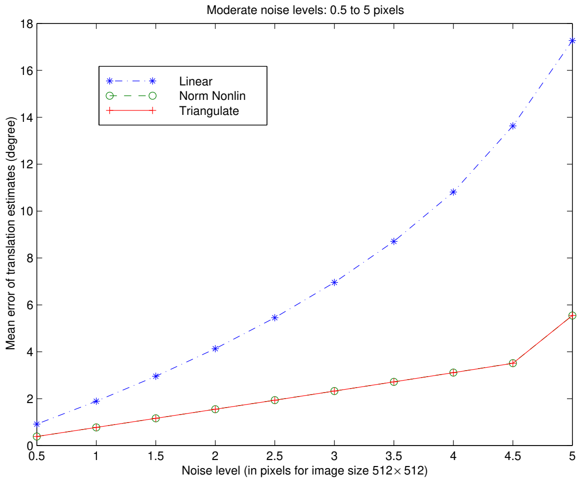

Figure 12: Estimation errors of translation (in degree). 50 trials, rotation 10 degree around Y-axis and translation along X-axis, T/R ratio is 2. Noises range from 0.5 to 5 pixels.

By comparing the simulation results in Figures 11, 12, 13 and 14, 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, out performs nonlinear algorithms. This is due to the more global nature of the linear algorithm. However, such high noise levels are barely realistic in real applications.

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

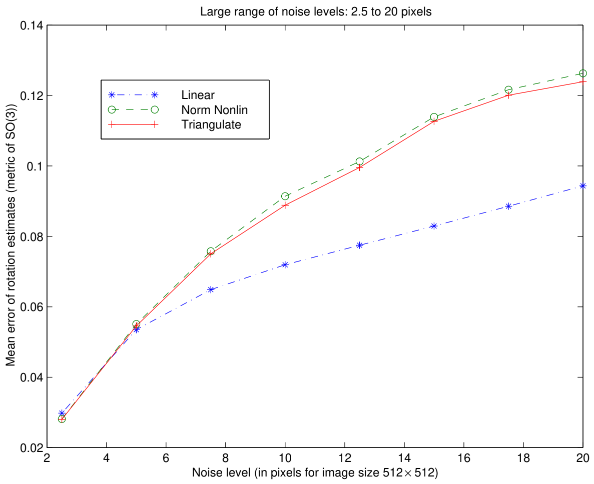

Figure 13: Estimation errors of rotation (in canonical metric on $SO(3)$). 40 points, 50 trials, rotation 10 degree around Y-axis and translation along Z-axis, T/R ratio is 2. Noises range from 2.5 to 20 pixels.

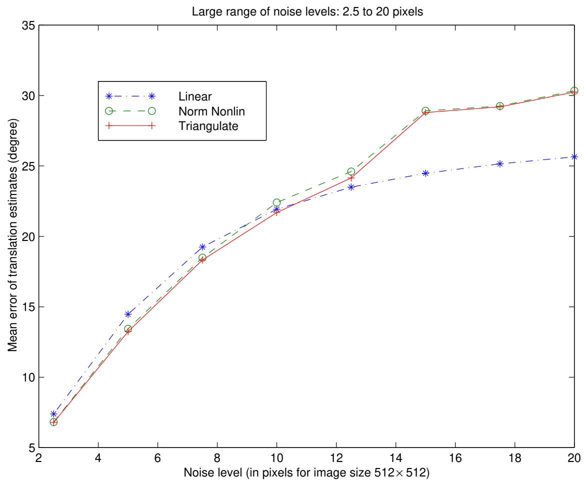

Figure 14: Estimation errors of translation (in degree). 40 points, 50 trials, rotation 10 degree around Y-axis and translation along Z-axis, T/R ratio is 2. Noises range from 2.5 to 20 pixels.

theoretical error bounds, such as the Cramer-Rao bound. This has been shown experimentally in [30]. 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.

8.3 Mutual Information Between Structure Estimates and Noises

So far, we have understood some of the difficulties in motion and structure estimation caused by various ambiguities, such as the bas relief ambiguity which is related to the sensitivity issue, or the second eigenmotion ambiguity which is related to the robustness issue. We here like to address, from an information theoretic viewpoint, another difficulty caused by noise in motion and structure estimation. More specifically, we like to ask the following questions:

> Is the (2-frame) motion and structure recovery problem well-defined from an estimation theoretic viewpoint?[23] If not, how much information can still be preserved in the presence of noise? Consequently, is there any simple criteria that a "good" estimation algorithm should achieve?

The answer to the first question is unfortunately negative due to following reasons. Let us assume the same noise model as given by (13).[24] As shown in Figure 15, given the noisy $p = p_0 + x$ where x is any isotropic noise. Then the valid estimate of p_0 is given by $\tilde{p}$, the projection of p onto the epipolar line. Therefore, the component of x which is parallel to the epipolar line is absorbed into the estimates. Without loss of generality, we assume the variance of the noise x is 1.[25] Then the

[23] It is certainly well defined geometrically: in the noise free case, the linear algorithm gives closed-from solutions.

[24] The Gaussian assumption is not necessary here. The following arguments hold for all isotropic noises.

[25] Note x is a vector, so here we mean the expectation $E(\|x\|^2) = 1$ where $\|\cdot\|$ is the 2-norm.

variance left in the residue $\Delta p = p - \tilde{p}$ is about 0.5. In other words, regardless of algorithms, at least half of the noise will always become part of the estimated 3D structure. Consequently, any good (2-frame) motion and structure estimation algorithm should have a residue variance (relative to the noise variance) close to 0.5. This is a very simple and important statistic for evaluating any structure and motion estimation algorithm. For the proposed optimal triangulation algorithm, we computed the average residue variance for all the runs which are presented in Figure 11 and 12. It gives 0.4988, very close to the theoretical value.

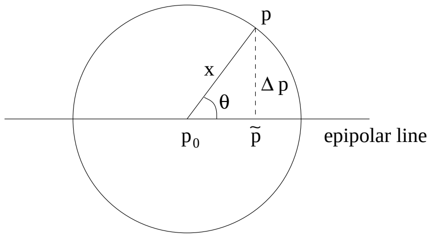

Figure 15: Estimate $\tilde{p}$ for given noise x.

9 Discussions 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 [17, 10]. Although these algorithms already have good performance in practice, the geometric concepts behind them have not yet 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 [23]), 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.

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 [5] 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 [15]. Optimization schemes for the differential case have also been studied by many researchers, including the most recent Bilinear Projection Algorithm (BPA) proposed in [22] and a robust algorithm proposed in [32]. 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 [14], 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 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.

10 Acknowledgment

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.

References

[1] G. Adiv. Inherent ambiguities in recovering 3-D motion and structure from a noisy flow field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):477–89, 1989.

[2] W. M. Boothby. *An Introduction to Differential Manifolds and Riemannian Geometry.* Academic Press, second edition, 1986.

[3] K. Danilidis. *Visual Navigation*, chapter "Understanding Noise Sensitivity in Structure from Motion". Lawrence Erlbaum Associates, 1997.

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

[5] A. Edelman, T. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Analysis Applications*, to appear.

[6] O. Faugeras. *There-Dimensional Computer Vision.* The MIT Press, 1993.

[7] R. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–57, 1997.

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

[9] A. D. Jepson and D. J. Heeger. Linear subspace methods for recovering translation direction. *Spatial Vision in Humans and Robots, Cambridge Univ. Press*, pages 39–62, 1993.

[10] K. Kanatani. *Geometric Computation for Machine Vision.* Oxford Science Publications, 1993.

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

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

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

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

[15] Y. Ma, J. Košecká, and S. Sastry. Motion recovery from image sequences: Discrete viewpoint vs. differential viewpoint. In *Proceeding of European Conference on Computer Vision, Volume II, (also Electronic Research Laboratory Memorandum M98/11, UC Berkeley)*, pages 337–53, 1998.

[16] Y. Ma, J. Košecká, and S. Sastry. Linear differential algorithm for motion recovery: A geometric approach. *Submitted to IJCV*, 1999.

[17] S. Maybank. *Theory of Reconstruction from Image Motion.* Springer-Verlag, 1993.

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

[19] R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation.* CRC press Inc., 1994.

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

[21] S. T. Smith. *Geometric Optimization Methods for Adaptive Filtering.* PhD thesis. Harvard University, Cambridge, Massachusetts, 1993.

[22] S. Soatto and R. Brockett. Optimal and suboptimal structure from motion. *Proceedings of International Conference on Computer Vision*, to appear.

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

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

[25] M. Spivak. *A Comprehensive Introduction to Differential Geometry: Volume II.* Publish or Perish, Inc., second edition, 1979.

[26] C. J. Taylor and D. J. Kriegman. Structure and motion from line segments in multiple images. *IEEE Transactions on PAMI*, 17(11):1021–32, 1995.

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

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

[29] J. Weng, T.S. Huang, and N. Ahuja. 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, 1989.

[30] J. Weng, T.S. Huang, and N. Ahuja. *Motion and Structure from Image Sequences.* Springer Verlag, 1993.

[31] J. Weng, T.S. Huang, and N. Ahuja. Optimal motion and structure estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):864–84, 1993.

[32] T. Zhang and C. Tomasi. Fast, robust and consistent camera motion estimation. In *to appear in Proceeding of CVPR*, 1999.

[33] Z. Zhang. Understanding the relationship between the optimization criteria in two-view motion analysis. In *Proceeding of International Conference on Computer Vision*, pages 772–77, Bombay, India, 1998.