Camera Pose Estimation Using First-Order Curve Differential Geometry

Ricardo Fabbri[1], Benjamin B. Kimia[1], and Peter J. Giblin[2]

[1] Brown University
Division of Engineering
Providence RI 02912, USA
{rfabbri,kimia}@lems.brown.edu
[2] University of Liverpool
Liverpool, UK
pjgiblin@liverpool.ac.uk

Abstract. This paper considers and solves the problem of estimating camera pose given a pair of point-tangent correspondences between the 3D scene and the projected image. The problem arises when considering curve geometry as the basis of forming correspondences, computation of structure and calibration, which in its simplest form is a point augmented with the curve tangent. We show that while the standard resectioning problem is solved with a minimum of three points given the intrinsic parameters, when points are augmented with tangent information only two points are required, leading to substantial computational savings, *e.g.*, when used as a minimal engine within RANSAC. In addition, computational algorithms are developed to find a practical and efficient solution shown to effectively recover camera pose using both synthetic and realistic datasets. The resolution of this problem is intended as a basic building block of future curve-based structure from motion systems, allowing new views to be incrementally registered to a core set of views for which relative pose has already been computed.

Keywords: Pose Estimation, Camera Resectioning, Differential Geometry.

1 Introduction

A key problem in the reconstruction of structure from multiple views is the determination of relative pose among cameras as well as the intrinsic parameters for each camera. The classical method is to rely on a set of corresponding points across views to determine each camera's intrinsic parameter matrix $\mathcal{K}_{im}$ as well as the relative pose between pairs of cameras [11]. The set of corresponding points can be determined using a calibration jig, but, more generally, using isolated keypoints such as Harris corners [10] or SIFT/HOG [17] features which remain somewhat stable over view and other variations. As long as there is a sufficient number of keypoints between two views, a random selection of a few feature correspondences using RANSAC [7,11] can be verified by measuring the number of inlier features. This class of isolated feature point-based methods are currently in popular and successful use through packages such as the Bundler and used in applications such as Phototourism [1].

A. Fitzgibbon et al. (Eds.): ECCV 2012, Part IV, LNCS 7575, pp. 231–244, 2012.
© Springer-Verlag Berlin Heidelberg 2012

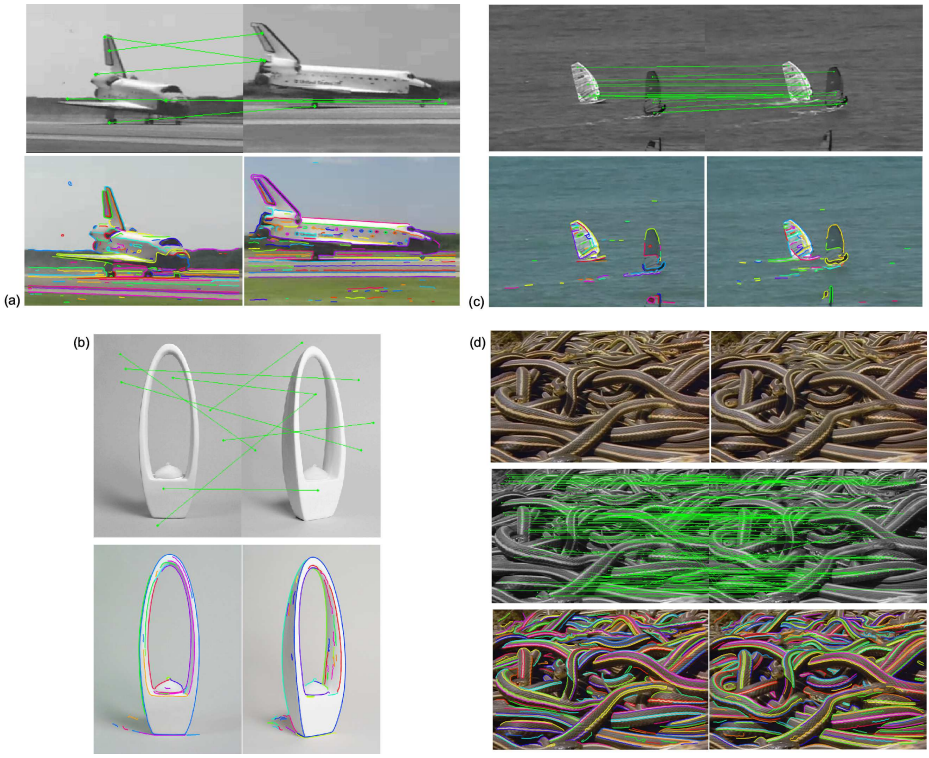

Fig. 1. (a) Views with wide baseline separation may not have enough interest points in common, but they often do share common curve structure. (b) There may not always be sufficient interest points matching across views of homogeneous objects, such as for the sculpture, but there is sufficient curve structure. (c) Each moving object requires its own set of features, which may not be sufficient without a richly textured surface. (d) Non-rigid structures face the same issue.

Two major drawbacks limit the applicability of interest points. First, it is well-known that in practice the correlation of interest points works for views with a *limited baseline*, according to some estimates no greater than 30° [18], Figure 1(a). In contrast, certain image curve fragments, *e.g.*, those corresponding to sharp ridges, reflectance curves, *etc*, persist stably over a much larger range of views. Second, the success of interest point-based methods is based on the presence of an abundance of features so that a sufficient number of them survive the various variations between views. While this is true in many scenes, as evidenced by the popularity of this approach, in a non-trivial number of scenes this is not the case, such as *(i)* Homogeneous regions, *e.g.*, from man-made objects, corridors, *etc*., Figure 1(b); *(ii)* Multiple moving objects require their own set of features which may not be sufficiently abundant without sufficient texture, Figure 1(c); *(iii)* Non-rigid objects require a rich set of features per roughly non-deforming patch, Figure 1(d). In all these cases, however, there is often sufficient **image curve structure**, motivating augmenting the use of interest points by developing a parallel technology for the use of image curve structure.

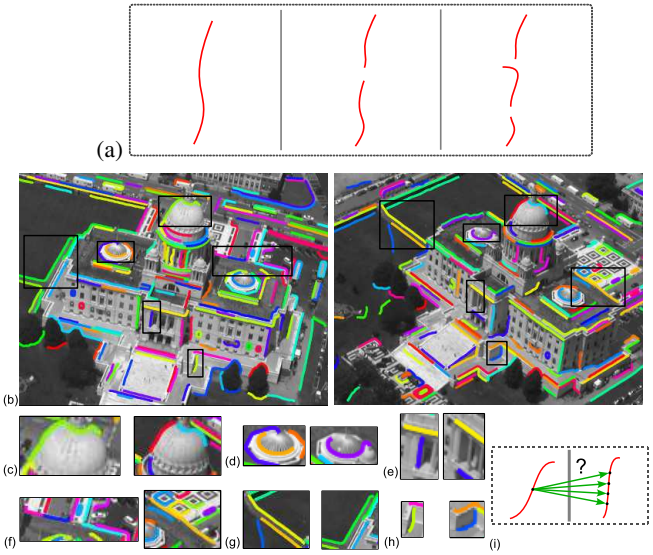

Fig. 2. Real challenges in using curve fragments in multiview geometry: (a) instabilities with slight changes in viewpoint, shown for two views in (b) and zoomed in (c-h), such as a curve in one view broken into two in another, a curve linked onto background, a curve detected in one view but absent in another, a curve fragmented into several pieces at junctions in one view but fully linked in another, different parts of a curve occluded in different views, and a curve undergoing deformation from one view to the other. (i) Point correspondence ambiguity along the curve.

The use of image curves in determining camera pose has generally been based on epipolar tangencies, but these techniques assume that curves are closed or can be described as conics or other algebraic curves [14, 15, 19, 21]. The use of image curve fragments as the basic structure for auto-calibration under general conditions is faced with two significant challenges. First, current edge linking procedures do not generally produce curve segments which persist stably across images. Rather, an image curve fragment in one view may be present in broken form and/or or grouped with other curve fragments. Thus, while the underlying curve geometry correlates well across views, the individual curve fragments do not, Figure 2(a-h). Second, even when the image curve fragments correspond exactly, there is an intra-curve correspondence ambiguity, Figure 2(i). This ambiguity prevents the use of corresponding curve points to solve for the unknown pose and intrinsic parameters. Both these challenges motivate the use of *small curve fragments*.

The paradigm explored in this paper is that small curve fragments, or equivalently points augmented with differential-geometric attributes[1], can be used as the basic image structure to correlate across views. The intent is to use curve geometry as a complementary approach to the use of interest points in cases where these fail or are not available. The value of curve geometry is in correlating structure across *three* frames or more

[1] Previous work in exploring local geometric groupings [22] has shown that tangent and curvature as well as the sign of curvature derivative can be reliably estimated.

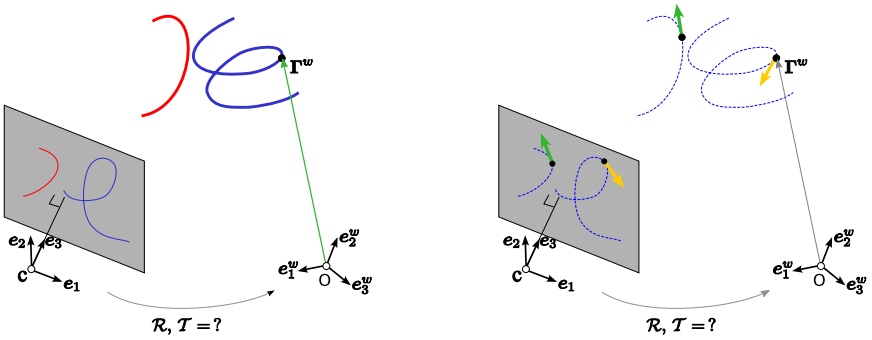

Fig. 3. The problem of determining camera pose $\mathcal{R}$, $\mathcal{T}$ given space curves in a world coordinate system and their projections in an image coordinate system (left), and an approach to that consisting of (right) determining camera pose $\mathcal{R}$, $\mathcal{T}$ given 3D point-tangents (*i.e.*, local curve models) in a world coordinate system and their projections in an image coordinate system.

since the correspondence geometry in two views is unconstrained. The differential geometry at two corresponding points in two views reconstruct the differential geometry of the space curve they arise from [4] and this constrains the differential geometry of corresponding curves in a third view.

The fundamental questions underlying the use of points augmented with differential-geometric attributes are: how many such points are needed, what order of differential geometry is required, *etc*. This paper explores the use of first-order differential geometry, namely points with tangent attributes, for determining the pose of a single camera with respect to the coordinates of observed 3D point-tangents. It poses and solves the following:

Problem: For a camera with known intrinsic parameters, how many corresponding pairs of point-tangents in space specified in world coordinates, and point-tangents in 2D specified in image coordinates, are required to establish the pose of the camera with respect to the world coordinates, Figure 3.

The solution to the above problem is useful under several scenarios. First, when many views of the scene are available and there is a reconstruction available from two views, *e.g.*, as in [5]. In this case a pair of point-tangents in the reconstruction can be matched under a RANSAC strategy to a pair of point-tangents in the image to determine pose. The advantage as compared to using three points from unorganized point reconstruction and resectioning is that *(i)* there are fewer edges than surface points and *(ii)* the method uses two rather than three points in RANSAC, requiring about half the number of runs for the same level of robustness, *e.g.*, 32 runs instead of 70 to achieve 99.99% probability of not hitting an outlier in at least one run, assuming 50% outliers (in practical systems it is often necessary to do as many runs as possible, to maximize robustness). Second, the 3D model of the object may be available from CAD or other sources, *e.g.*, civilian or military vehicles. In this case a strategy similar to the first scenario can be used. Third, in stereo video sequences obtained from precisely calibrated binocular cameras, the reconstruction from one frame of the video can be used to determine the camera pose in subsequent frames.

2 Related Work

Previous work has generally relied on matching *epipolar tangencies* on *closed curves*. Two corresponding points $\boldsymbol{\gamma}^1$ in image 1 and $\boldsymbol{\gamma}^2$ in image 2 are related by $\boldsymbol{\gamma}^{2\top} E \boldsymbol{\gamma}^1 = 0$, where E is the *essential matrix* [16]. This can be extended to the differential geometry of two curves, $\boldsymbol{\gamma}^1(s)$ in the first view and a curve $\boldsymbol{\gamma}^2(s)$ in a second view, *i.e.*,

$$\boldsymbol{\gamma}^{1\top}(s) E \boldsymbol{\gamma}^2(s) = 0. \tag{2.1}$$

The tangents $\boldsymbol{t}^1(s)$ and $\boldsymbol{t}^2(s)$ are related by differentiation

$$g^1(s)\boldsymbol{t}^{1^\top}(s) E \boldsymbol{\gamma}^2(s) + \boldsymbol{\gamma}^{1\top}(s) E g^2(s) \boldsymbol{t}^2(s) = 0, \tag{2.2}$$

where $g^1(s)$ and $g^2(s)$ are the respective speeds of parametrization of the curves $\boldsymbol{\gamma}^1(s)$ and $\boldsymbol{\gamma}^2(s)$. It is clear that when one of the tangents $\boldsymbol{t}^1(s)$ is along the epipolar plane, *i.e.*, $\boldsymbol{t}^{1^\top}(s) E \boldsymbol{\gamma}^2(s) = 0$ at a point s, then $\boldsymbol{\gamma}^{1\top}(s) E \boldsymbol{t}^2(s) = 0$. Thus, epipolar tangency in image 1 implies tangency in image 2 at the corresponding point, Figure 4.

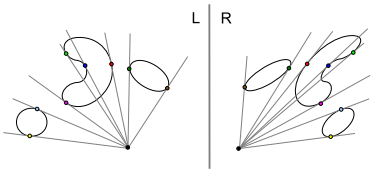

Fig. 4. Correspondence of epipolar tangencies in curve-based camera calibration. An epipolar line on the left must correspond to the epipolar line on the right having tangency on the corresponding curve, marked with the same color. This works for both static curves and occluding contours.

The epipolar tangency constraint was first proposed in [19] who use linked edges and a coarse initial estimate E to find a sparse set of epipolar tangencies, including those at corners, in each view. They are matched from one view to another manually. This is then used to refine the estimate E, see Figure 5, by minimizing $\boldsymbol{\gamma}^{1\top}(s) E \boldsymbol{\gamma}^2(s)$ over all matches in an iterative two-step scheme: the corresponding points are kept fixed and E is optimized in the first step and then E is kept fixed and the points are updated in a

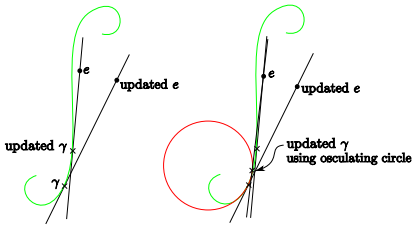

Fig. 5. The differential update of epipolar tangencies through curvature information

second step using a closed form solution based on an approximation of the curve as the osculating circle. This assumes that closed curves are available.

Kahl and Heyden [14] consider the special case when four corresponding conics are available in two views with unknown intrinsic parameters. In this approach, each pair of corresponding conics provides a pair of tangencies and therefore two constraints. Four pairs of conics are needed. If the intrinsic parameters are available, then the absolute conic is known giving two constraints on the epipolar geometry, so that only 3 conic correspondences are required. This approach is only applied to synthetic data which shows the scheme to be extremely sensitive even when a large number of conics (50) is used. Kaminski and Shashua [15] extended this work to general algebraic curves viewed in multiple uncalibrated views. Specifically, they extend Kruppa's equations to describe the epipolar constraint of two projections of a general algebraic curve. The drawback of this approach is that algebraic curves are restrictive.

Sinha *et. al.* [21] consider a special configuration where multiple static cameras view a moving object. Since the epipolar geometry between any pair of cameras is fixed, each hypothesized pair of epipoles representing a point in 4D is then probed for a pair of epipolar tangencies across video frames. Specifically, two pairs of tangencies in one frame in time and a single pair of tangencies in another frame provide a constraint in that they must all intersect in the same point. This allows for an estimation of epipolar geometry for each pair of cameras, which are put together for refinement using bundle adjustment, providing intrinsic parameters and relative pose. This approach, however, is restrictive in assuming well-segmentable silhouettes.

We should briefly mention the classic results that three 2D-3D point correspondences are required to determine camera pose [7], in a procedure known as *camera resectioning* in the photogrammetry literature (and by Hartley and Zisserman [11]), also known as *camera calibration* when this is used with the purpose of obtaining the intrinsic parameter matrix $\mathcal{K}_{im}$, where the camera pose relative to the calibration jig is not of interest. This is also related to the perspective n-point problem (PnP) originally introduced in [7] which can be stated as the recovery of the camera pose from n corresponding 3D-2D point pairs [12] or alternatively of depths [9].

Notation: Consider a sequence of n 3D points $(\boldsymbol{\Gamma}_1^w, \boldsymbol{\Gamma}_2^w, \ldots, \boldsymbol{\Gamma}_n^w)$, described in the world coordinate system and their corresponding projected image points $(\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_n)$ described as points in the 3D camera coordinate system. Let the rotation $\mathcal{R}$ and translation $\mathcal{T}$ relate the camera and world coordinate systems through

$$\boldsymbol{\Gamma} = \mathcal{R}\boldsymbol{\Gamma}^w + \mathcal{T}, \tag{2.3}$$

where $\boldsymbol{\Gamma}$ and $\boldsymbol{\Gamma}^w$ are the coordinates of a point in the camera and world coordinate systems, respectively. Let $(\rho_1, \rho_2, \ldots, \rho_n)$ be the depth defined by

$$\boldsymbol{\Gamma}_i = \rho_i \boldsymbol{\gamma}_i, \qquad i = 1, \ldots, n. \tag{2.4}$$

In general we assume that each point $\boldsymbol{\gamma}_i$ is a sample from an image curve $\boldsymbol{\gamma}_i(s_i)$ which is the projection of a space curve $\boldsymbol{\Gamma}_i(S_i)$, where s_i and S_i are arclengths along the image and space curves, resp.

The direct solution to P3P, also known as the *triangle pose problem*, given in 1841 [8], equates the sides of the triangle formed by the three points with those of the vectors in the camera domain, *i.e.*,

$$\begin{cases} \|\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2\|^2 = \|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2 \\ \|\rho_2\boldsymbol{\gamma}_2 - \rho_3\boldsymbol{\gamma}_3\|^2 = \|\boldsymbol{\Gamma}_2^w - \boldsymbol{\Gamma}_3^w\|^2 \\ \|\rho_3\boldsymbol{\gamma}_3 - \rho_1\boldsymbol{\gamma}_1\|^2 = \|\boldsymbol{\Gamma}_3^w - \boldsymbol{\Gamma}_1^w\|^2 \end{cases} \tag{2.5}$$

This gives a system of three quadratics (conics) in unknowns ρ_1, ρ_2, and ρ_3. Following traditional methods going back to the German mathematician Grunert in 1841 [8] and later Finsterwalder in 1937 [6], by factoring out one depth, say ρ_1, this can be reduced to a system of two quadratics in two unknowns – depth ratios $\frac{\rho_2}{\rho_1}$ and $\frac{\rho_3}{\rho_1}$. Grunert further reduced this to a single quartic equation and Finsterwalder proposed an analytic solution.

Table 1. The number of 3D–2D point correspondences needed to solve for camera pose and intrinsic parameters

Case	Unknowns	Min. # of Point Corresp.	Min. # of Pt-Tgt Corresp.
Calibrated ($\mathcal{K}_{im}$ known)	Camera pose $\mathcal{R}$, $\mathcal{T}$	3	2 (this paper)
Focal length unknown	Pose $\mathcal{R}$, $\mathcal{T}$ and f	4	3 (conjecture)
Uncalibrated ($\mathcal{K}_{im}$ unknown)	Camera model $\mathcal{K}_{im}$, $\mathcal{R}$, $\mathcal{T}$	6	4 (conjecture)

In general, the camera resectioning problem can be solved using **three** 3D $\leftrightarrow$ 2D point correspondences when the intrinsic parameters are known, and **six** points when the intrinsic parameters are not known. It can be solved using **four** point correspondences when only the focal length is unknown, but all the other intrinsic parameters are known [3], Table 1. We now show that when intrinsic parameters are known, **only a pair of point-tangent correspondences are required to estimate camera pose.** We conjecture that future work will show that 3 and 4 points, respectively, are required for the other two cases, Table 1. This would represent a significant reduction for a RANSAC-based computation.

3 Determining Camera Pose from a Pair of 3D–2D Point-Tangent Correspondences

Theorem 1. *Given a pair of 3D point-tangents* $\{(\boldsymbol{\Gamma}_1^w, \boldsymbol{T}_1^w), (\boldsymbol{\Gamma}_2^w, \boldsymbol{T}_2^w)\}$ *described in a world coordinate system and their corresponding perspective projections, the 2D point-tangents* $(\boldsymbol{\gamma}_1, \boldsymbol{t}_1)$, $(\boldsymbol{\gamma}_2, \boldsymbol{t}_2)$, *the pose of the camera* $\mathcal{R}$, $\mathcal{T}$ *relative to the world coordinate system defined by* $\boldsymbol{\Gamma} = \mathcal{R}\boldsymbol{\Gamma}^w + \mathcal{T}$ *can be solved up to a finite number of solutions*[2], *by solving the system*

$$\begin{cases} \boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1\,\rho_1^2 - 2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2\,\rho_1\rho_2 + \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2\,\rho_2^2 = \|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2, \\ Q(\rho_1, \rho_2) = 0, \end{cases} \tag{3.1}$$

[2] assuming that the intrinsic parameters $\mathcal{K}_{im}$ are known

where $\mathcal{R}\boldsymbol{\Gamma}_1^w + \mathcal{T} = \boldsymbol{\Gamma}_1 = \rho_1\boldsymbol{\gamma}_1$ *and* $\mathcal{R}\boldsymbol{\Gamma}_2^w + \mathcal{T} = \boldsymbol{\Gamma}_2 = \rho_2\boldsymbol{\gamma}_2$*, and* $Q(\rho_1, \rho_2)$ *is an eight degree polynomial. This then solves for* $\mathcal{R}$ *and* $\mathcal{T}$ *as*

$$\begin{cases} \mathcal{R} = \left[(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)\ \boldsymbol{T}_1^w\ \boldsymbol{T}_2^w\right]^{-1} \cdot \\ \qquad \left[\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2\ \rho_1\frac{g_1}{G_1}\boldsymbol{t}_1 + \frac{\rho_1'}{G_1}\boldsymbol{\gamma}_1\ \rho_2\frac{g_2}{G_2}\boldsymbol{t}_2 + \frac{\rho_2'}{G_2}\boldsymbol{\gamma}_2\right] \\ \mathcal{T} = \rho_1\boldsymbol{\gamma}_1 - \mathcal{R}\boldsymbol{\Gamma}_1^w, \end{cases}$$

where expressions for four auxiliary variables $\frac{g_1}{G_1}$ *and* $\frac{g_2}{G_2}$*, the ratio of speeds in the image and along the tangents, and* ρ_1 *and* ρ_2 *are available.*

Proof. We take the 2D-3D point-tangents as samples along 2D-3D curves, respectively, where the speed of parametrization along the image curves are g_1 and g_2 and along the space curves G_1 and G_2. The proof proceeds by *(i)* writing the projection equations for each point and its derivatives in the simplest form involving $\mathcal{R}$, $\mathcal{T}$, depths ρ_1 and ρ_2, depth derivatives ρ_1' and ρ_2', and speed of parametrizations G_1 and G_2, respectively; *(ii)* eliminating the translation $\mathcal{T}$ by subtracting point equations; *(iii)* eliminating $\mathcal{R}$ using dot products among equations. This gives six equations in six unknowns: $(\rho_1, \rho_2, \rho_1\frac{g_1}{G_1}, \rho_2\frac{g_2}{G_2}, \frac{\rho_1'}{G_1}, \frac{\rho_2'}{G_2})$; *(iv)* eliminating the unknowns ρ_1' and ρ_2' gives four quadratic equations in four unknowns: $(\rho_1, \rho_2, \rho_1\frac{g_1}{G_1}, \rho_2\frac{g_2}{G_2})$. Three of these quadratics can be written in the form:

$$\begin{cases} Ax_1^2 + Bx_1 + C = 0 & (3.2) \\ Ex_2^2 + Fx_2 + G = 0 & (3.3) \\ H + Jx_1 + Kx_2 + Lx_1x_2 = 0, & (3.4) \end{cases}$$

where $x_1 = \rho_1\frac{g_1}{G_1}$ and $x_2 = \rho_2\frac{g_2}{G_2}$ and where A through L are only functions of the two unknowns ρ_1 and ρ_2. Now, Eq. 3.4 represents a rectangular hyperbola, Fig. 6, while Eqs. 3.2 and 3.3 vertical and horizontal lines in the (x_1, x_2) space. Fig. 6 illustrates that only one solution is possible which is then analytically written in terms of variables A–L (not shown here). This allows expressing $\rho_1\frac{g_1}{G_1}$ and $\rho_2\frac{g_2}{G_2}$ in terms of ρ_1 and ρ_2 – a degree 16 polynomial – but this is in fact divisible by $\rho_1^4\rho_2^4$, leaving a polynomial Q of degree 8. Furthermore, we find that $Q(-\rho_1, -\rho_2) = Q(\rho_1, \rho_2)$, using the symmetry of the original equations. This, together with the unused equation (the remaining one of four) gives the system 3.1. The detailed proof is given in the supplementary material.

Proposition 1. *The algebraic solutions to the system* (3.1) *of Theorem 1 are also required to satisfy the following inequalities arising from imaging and other requirements enforced by*

$$\rho_1 > 0,\ \rho_2 > 0 \tag{3.5}$$

$$\frac{g_1}{G_1} > 0,\ \frac{g_2}{G_2} > 0 \tag{3.6}$$

$$\frac{\det[\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2\ \ \rho_1\frac{g_1}{G_1}\boldsymbol{t}_1 + \frac{\rho_1'}{G_1}\boldsymbol{\gamma}_1\ \ \rho_2\frac{g_2}{G_2}\boldsymbol{t}_2 + \frac{\rho_2'}{G_2}\boldsymbol{\gamma}_2]}{\det\left[\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\ \boldsymbol{T}_1^w\ \boldsymbol{T}_2^w\right]} > 0. \tag{3.7}$$

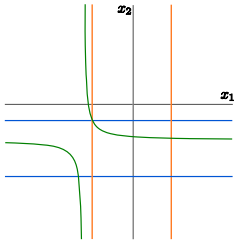

Fig. 6. Diagram of the mutual intersection of Equations 3.2-3.4 in the x_1–x_2 plane

Proof. There are multiple solutions for ρ_1 and ρ_2 in Eq. 3.1. Observe that if $\rho_1, \rho_2, \mathcal{R}, \mathcal{T}$ are a solution, then so are $-\rho_1, -\rho_2, -\mathcal{R}$, and $-\mathcal{T}$. Only one of these two solutions are valid, as the camera geometry enforces positive depth, $\rho_1 > 0$ and $\rho_2 > 0$; solutions are sought only in the top right quadrant of the ρ_1–ρ_2 space. In fact, the imaging geometry further restricts the points to lie in front of the camera.

Second, observe that the matrix $\mathcal{R}$ can only be a rotation matrix if it has determinant $+1$ and is a reflection if it has determinant -1. Using (3.2), $\det(\mathcal{R})$ can be written as

$$\det \mathcal{R} = \frac{\det\left[\rho_1 \boldsymbol{\gamma}_1 - \rho_2 \boldsymbol{\gamma}_2 \;\; \rho_1 \frac{g_1}{G_1}\boldsymbol{t}_1 + \frac{\rho_1'}{G_1}\boldsymbol{\gamma}_1 \;\; \rho_2 \frac{g_2}{G_2}\boldsymbol{t}_2 + \frac{\rho_2'}{G_2}\boldsymbol{\gamma}_2\right]}{\det\left[\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w \;\; \boldsymbol{T}_1^w \;\; \boldsymbol{T}_2^w\right]}.$$

Finally, the space curve tangent $\boldsymbol{T}$ and the image curve tangent $\boldsymbol{t}$ must point in the same direction: $\boldsymbol{T} \cdot \boldsymbol{t} > 0$, or, as in the supplementary material, $\frac{g_1}{G_1} > 0$ and $\frac{g_2}{G_2} > 0$.

4 A Practical Approach to Computing a Solution

Equations 3.1 can be viewed as the intersection of two curves in the $\rho_1 - \rho_2$ space. Since one of the curves to be intersected is shown to be an ellipse, it is possible to parametrize it by a bracketed parameter and then look for intersections with the second curve which is of degree 8. This gives a higher-order polynomial in a *single* unknown which can be solved more readily than simultaneously solving the two equations of degree 2 and 8.

Proposition 2. *Solutions ρ_1 and ρ_2 to the quadratic equation in (3.1) can be parametrized as*

$$\begin{cases} \rho_1(t) = \dfrac{2\alpha t \cos\theta + \beta(1-t^2)\sin\theta}{1+t^2} \\ \rho_2(t) = \dfrac{-2\alpha t \sin\theta + \beta(1-t^2)\cos\theta}{1+t^2}, \end{cases} \qquad -1 \le t \le 1$$

where

$$\tan(2\theta) = \frac{2(1+\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2)}{\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2}, \qquad 0 \le 2\theta \le \pi,$$

and

$$\alpha = \frac{\sqrt{2}\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|}{\sqrt{(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2) + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\cos(2\theta) + 2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2\sin(2\theta)}}, \quad \alpha > 0,$$
$$\beta = \frac{\sqrt{2}\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|}{\sqrt{(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2) - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\cos(2\theta) - 2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2\sin(2\theta)}}, \quad \beta > 0.$$

Proof. An ellipse centered at the origin with semi-axes of lengths $\alpha > 0$ and $\beta > 0$ and parallel to the coordinates x and y can be parametrized as

$$x = \frac{2t}{1+t^2}\alpha, \quad y = \frac{(1-t^2)}{1+t^2}\beta, \qquad t \in (-\infty, \infty), \tag{4.1}$$

with ellipse vertices identified at $t = -1,\ 0,\ 1$ and ∞, as shown in Figure 7. For a general ellipse centered at the origin, the coordinates must be multiplied with the rotation matrix for angle θ, obtaining

$$\begin{cases} \rho_1 = \dfrac{2\alpha t\cos\theta + \beta(1-t^2)\sin\theta}{1+t^2} \\ \rho_2 = \dfrac{-2\alpha t\sin\theta + \beta(1-t^2)\cos\theta}{1+t^2}. \end{cases} \qquad -1 \le t \le 1$$

Figure 7 illustrates this parametrization. Notice that the range of values of t we need to consider certainly lies in $[-1, 1]$ and in fact in a smaller interval where $\rho_1 > 0$ and $\rho_2 > 0$. Note that t and $-\frac{1}{t}$ correspond to opposite points on the ellipse.

The parameters α, β, and θ for the ellipse in (3.1) can then be found by substitution of ρ_1 and ρ_2, details of which are found in the supplementary material.

Both equations in (3.1) are symmetric with respect to the origin in the (ρ_1, ρ_2)-plane and the curves will intersect in at most $2 \times 8 = 16$ real points, at most 8 of which will be in the positive quadrant, as we in fact require $\rho_1 > 0$ and $\rho_2 > 0$.

The parametrization of the ellipse given in Proposition 2 allows us to reduce the two Equations 3.1 to a single polynomial equation in t. Substituting for ρ_1, ρ_2 in terms of t

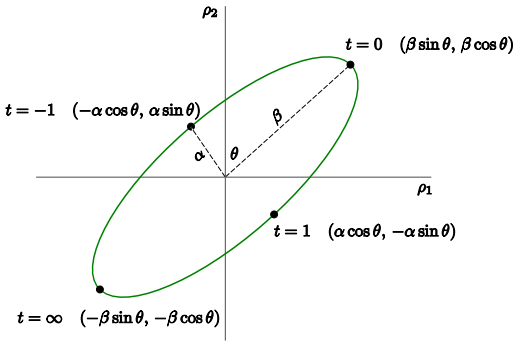

Fig. 7. Diagram illustrating a parametrization of the ellipse by a parameter t

into $Q = 0$ gives an equation in t for which, in fact, all the denominators are $(1+t^2)^{12}$, so that these can be cleared leaving a polynomial in $\tilde{Q}(t)$ of degree 16. The symmetry with respect to the origin in the (ρ_1, ρ_2)-plane becomes, in terms of t, a symmetry with respect to the substitution $t \to -1/t$, which gives diametrically opposite points of the ellipse. This implies that $\tilde{Q}$ has the special form

$$\tilde{Q}(t) = q_0 + q_1 t + q_2 t^2 + \cdots + q_{16} t^{16}, \tag{4.2}$$

where $q_i = -q_{16-i}$ for i odd. At most 8 solutions will lie in the range $-1 < t \leq 1$, and indeed we are only interested in solutions which make $\rho_1 > 0$ and $\rho_2 > 0$.

5 Experiments

We use two sets of experiments to probe camera pose recovery using 2D-3D point-tangent correspondences. First, we use a set of synthetically generated 3D curves consisting of a variety of curves (helices, parabolas, ellipses, straight lines, and saddle curves), as shown in Figure 8. Second, we use realistic data.

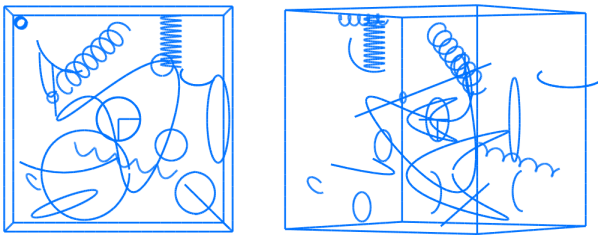

Fig. 8. Sample views of the synthetic dataset. Real datasets have also been used in our experiments, reported in further detail in the **supplemental material.**

The synthetic 3D curves of Figure 8 are densely sampled and projected to a single 500×400 view, and their location and tangent orientation are perturbed to simulate measurement noise in the range of $0 - 2$ pixels in location and $0 - 10^\circ$ in orientation. Our expectation in practice using the publically available edge detector [22] is that the edges can be found with subpixel accuracy and edge orientations are accurate to less than 5°.

In order to simulate the intended application, pairs of 2D-3D point-tangent correspondences are selected in a RANSAC procedure from among 1000 veridical ones, to which 50% random spurious correspondences were added. The practical method discussed in Section 4 is used to determine the pose of the camera $(\mathcal{R}, \mathcal{T})$ inside the RANSAC loop. Each step takes $90ms$ in Matlab on a standard 2GHz dual-core laptop. What is most significant, however, is that only 17 runs are sufficient to get 99% probability of hitting an outlier-free correspondence pair, or 32 runs for 99.99% probability. In practice more runs can easily be used depending on computational requirements. To assess the output of the algorithm, we could have measured the error of the estimated

pose compared to the ground truth pose. However, what is more meaningful is the impact of the measured pose on the *measured* reprojection error, as commonly used in the field to validate the output of RANSAC-based estimation. Since this is a controlled experiment, we measure final reprojection error not just to the inlier set, but to the entire pool of 1000 true correspondences. In practice, a bundle-adjustment would be run to refine the pose estimate using all inliers, but we chose to report the raw errors without nonlinear least-squares refinement. The distribution of reprojection error is plotted for various levels of measurement noise, Figure 9. These plots show that the relative camera pose can be effectively determined for a viable range of measurement errors, specially since these results are typically optimized in practice through bundle adjustment. Additional information can be found in the supplemental material.

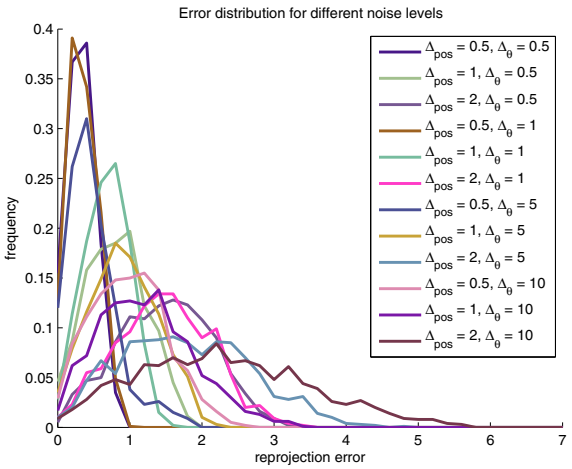

Fig. 9. Distributions of reprojection error for synthetic data without bundle adjustment, for increasing levels of positional and tangential perturbation in the measurements. **Additional results are reported in the supplemental material.**

Second, we use data from a real sequence, the "Capitol sequence", which is a set of 256 frames covering a 90° helicopter fly-by from the Rhode Island State Capitol, Figure 2, using a High-Definition camera (1280×720). Intrinsic parameters were initialized using the Matlab Calibration toolbox from J. Bouguet (future extension of this work would allow for an estimation of intrinsic parameters as well). The camera parameters were obtained by running Bundler [1] essentially out-of-the-box, with calibration accuracy of $1.3px$. In this setup, a pair of fully calibrated views are used to reconstruct a 3D cloud of 30 edges from manual correspondences. Pairs of matches from 3D edges to observed edges in novel views are used with RANSAC to compute the camera pose with respect to the frame of the 3D points, and measure reprojection error. One can then either use multiple pairs or use bundle adjustment to improve the reprojection error resulting from our initial computation of relative pose. Figure 10 shows the reprojection error distribution of our method for a single point-tangent pair after RANSAC, before and after running bundle-adjustment, versus the dataset camera from bundler (which is

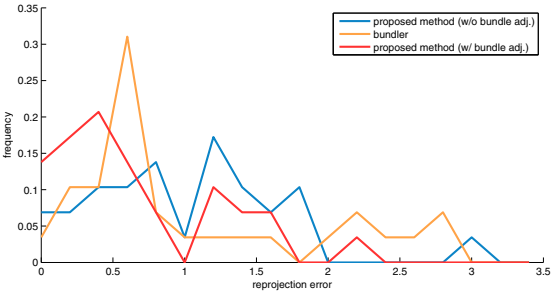

Fig. 10. The reprojection error distribution for real data (Capitol sequence) using only two point-tangents, before and after bundle adjustment. **Additional results are reported in the supplemental material.**

bundle-adjusted), for the Capitol sequence. The proposed approach achieved an average error of $1.1px$ and $0.76px$ before and after a metric bundle adjustment, respectively, as compared to $1.3px$ from Bundler. Additional information and results can be found in the supplemental material.

6 Future Directions

The paper can be extended to consider the case when intrinsic parameters are unknown. Table 1 conjectures that four pairs of corresponding 3D-2D point-tangents are sufficient to solve this problem. Also, we have been working on the problem of determining trinocular relative pose from corresponding point-tangents across 3 views. We conjecture that three triplets of correspondences among the views are sufficient to establish relative pose. This would allow for a complete curve-based structure from motion system starting from a set of images without any initial calibration.

Acknowledgments. The support of NSF grant 1116140, CNPq/Brazil proc. 200875/2004-3, FAPERJ/Brazil E26/112.082/2011, E26/190.180/2010, and the UERJ visiting professor grant are gratefully acknowledged.

References

1. Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building Rome in a day. In: ICCV 2009 (2009)
2. Ayache, N., Lustman, L.: Fast and reliable passive trinocular stereovision. In: ICCV 1987 (1987)
3. Bujnak, M., Kukelova, Z., Pajdla, T.: A general solution to the p4p problem for camera with unknown focal length. In: CVPR 2008 (2008)
4. Fabbri, R., Kimia, B.B.: High-Order Differential Geometry of Curves for Multiview Reconstruction and Matching. In: Rangarajan, A., Vemuri, B.C., Yuille, A.L. (eds.) EMMCVPR 2005. LNCS, vol. 3757, pp. 645–660. Springer, Heidelberg (2005)
5. Fabbri, R., Kimia, B.B.: 3D curve sketch: Flexible curve-based stereo reconstruction and calibration. In: CVPR 2010 (2010)

6. Finsterwalder, S., Scheufele, W.: Das ruckwartseinschneiden im raum. Sebastian Finsterwalder zum 75, 86–100 (1937)
7. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM 24(6), 381–395 (1981)
8. Grunert, J.A.: Das pothenotische problem in erweiterter gestalt nebst Über seine anwendungen in der geodäsie. Archiv der für Mathematik and Physik 1, 238–248 (1841)
9. Haralick, R.M., Lee, C.-N., Ottenberg, K., Nölle, M.: Review and analysis of solutions of the three point perspective pose estimation problem. IJCV 13(3), 331–356 (1994)
10. Harris, C., Stephens, M.: A combined edge and corner detector. In: Alvey Vision Conference (1988)
11. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)
12. Horaud, R., Conio, B., Leboulleux, O., Lacolle, B.: An analytic solution for the p4p problem. CVGIP 47(1), 33–44 (1989)
13. Hu, Z.Y., Wu, F.C.: A note on the number of solutions of the noncoplanar p4p problem. PAMI 24(4), 550–555 (2002)
14. Kahl, F., Heyden, A.: Using conic correspondence in two images to estimate the epipolar geometry. In: ICCV 1998 (1998)
15. Kaminski, J.Y., Shashua, A.: Multiple view geometry of general algebraic curves. IJCV 56(3), 195–219 (2004)
16. Longuet-Higgins, H.C.: A computer algorithm for reconstructing a scene from two projections. Nature 293, 133–135 (1981)
17. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)
18. Moreels, P., Perona, P.: Evaluation of features detectors and descriptors based on 3D objects. IJCV 73(3), 263–284 (2007)
19. Porrill, J., Pollard, S.: Curve matching and stereo calibration. IVC 9(1), 45–50 (1991)
20. Robert, L., Faugeras, O.D.: Curve-based stereo: figural continuity and curvature. In: CVPR 1991 (1991)
21. Sinha, S.N., Pollefeys, M., McMillan, L.: Camera network calibration from dynamic silhouettes. In: CVPR 2004 (2004)
22. Tamrakar, A., Kimia, B.B.: No grouping left behind: From edges to curve fragments. In: ICCV 2007 (2007)

Supplementary Material to Camera Pose Estimation Using First-Order Curve Differential Geometry, ECCV 2012

Complement to Submission 1034

1 Overview of this document

In Section 2 we present additional results for *(i)* our synthetic experiments, first clarifying the plot shown in the paper, Figure 1, and then using bundle adjustment, Figures 2–3, together with *(ii)* results for the standard Dino sequence from the Middlebury multiview stereo dataset [1], Figure 4. In the remaining sections of this document we supply additional details in the proofs of theorems and propositions from the paper. All references to equations and figures are to objects in the present document unless otherwise stated.

2 Additional Results

2.1 Synthetic experiments

Figure 1 clarifies Figure 9 of the paper, by splitting it into two plots, one for fixed tangential perturbation (top), and another for fixed positional perturbation. We also ran bundle adjustment on top of our RANSAC results, which is standard practice in applications, and recorded the distribution of reprojection errors, shown in Figures 2–3.

2.2 Dino sequence

Results: We also tested the proposed method on the standard Dino sequence from the Middlebury multiview stereo dataset [1], Figure 4. The Cameras sample 363 views at 640×480 on a hemisphere around the object. The data is low resolution compared to our Capitol dataset. The calibration accuracy in this case is hard to determine objectively, but it is "on the order of a pixel" or about 1-2px according to the authors (see a description of the calibration process below). We note that even though this is a carefully constructed dataset, the average reprojection error using our method are $1.03px$ and $0.66px$ before and after bundle adjustment, respectively, while the average error using the dataset camera is $0.88px$. This was obtained as follows. As for the Capitol sequence, we picked a set of manual edge correspondences (in this case 10) across 3 views, and reconstructed a 3D cloud of edges from the first two views using the dataset cameras. This gives a set of 3D-2D correspondences with which we seek to determine the pose of the third view and compare to the dataset pose. The third view plays the role of novel views to be iteratively integrated and registered/calibrated by a structure from motion system. We added 50% outliers to the set of manual correspondences, in order to be realistic, and ran RANSAC to select two point-tangents giving the pose which is most consistent with the data. Bundle adjustment can then be optionally run to refine this pose. The distributions of reprojection error before and after bundle adjustment, as compared to that of the dataset camera, are shown in Figure 1.

Details of the dataset calibration process: We note that the dataset calibration of the Dino sequence was performed as follows [1]: the images were captured using the Stanford Spherical Gantry which enables moving a camera on a sphere. To calibrate the cameras, they took images of a planar grid from 68 viewpoints and used a combination of Jean-Yves Bouguet's Matlab toolbox and their own software to find grid points and estimate camera intrinsics and extrinsics. From these parameters, they computed the gantry radius and camera orientation, hence enabling a map of any gantry position to camera parameters. The authors then scanned the object from several orientations using a laser scanner and merged the results. The cameras were then aligned with the resulting mesh.

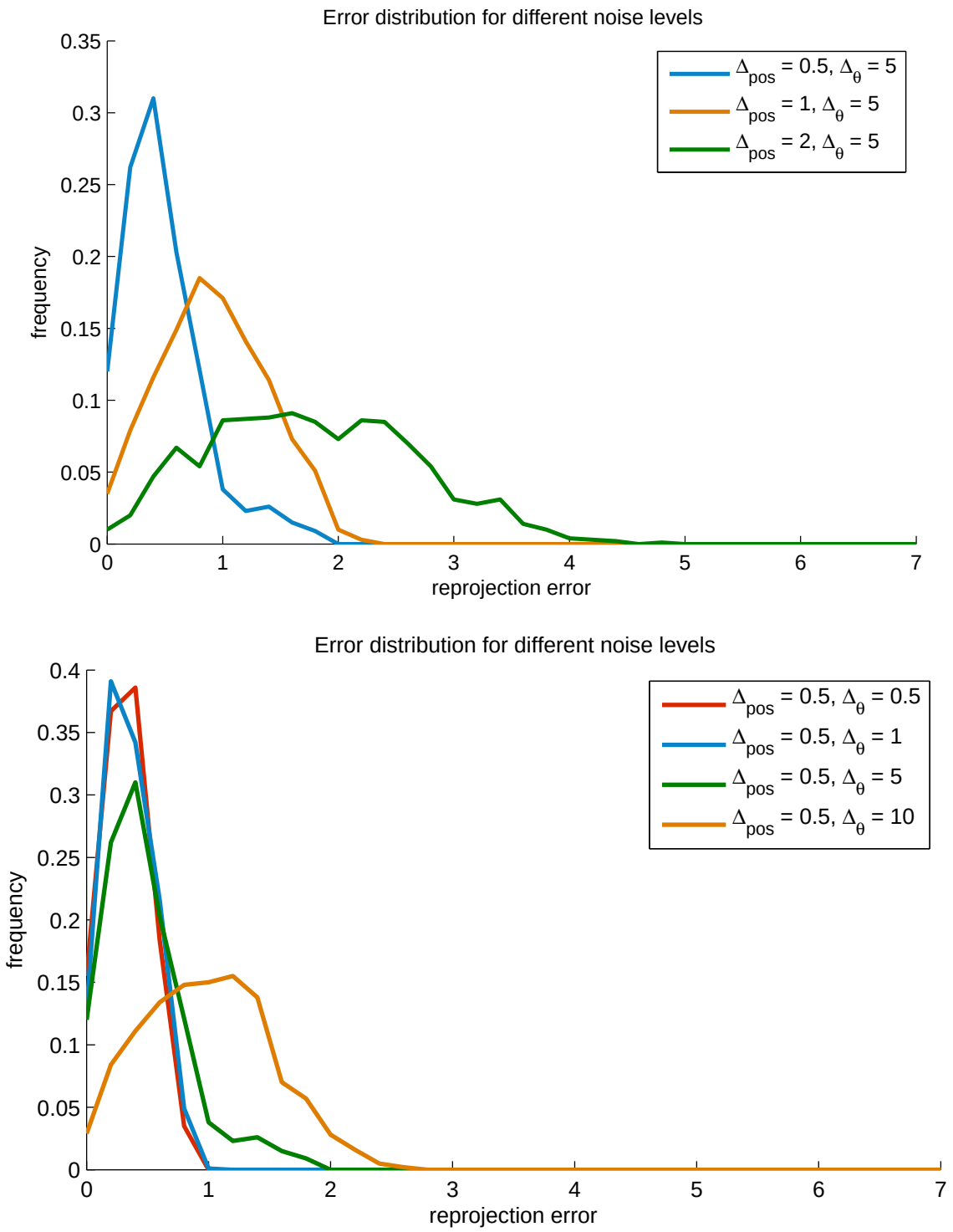

Figure 1: Distributions of reprojection error for synthetic data results **without bundle adjustment**, for (top) increasing levels of positional perturbation while keeping tangential orientation perturbation fixed; and (bottom) increasing levels of tangential orientation perturbation while keeping positional perturbation fixed. This is the same as in the paper, but split into two different plots for clarity.

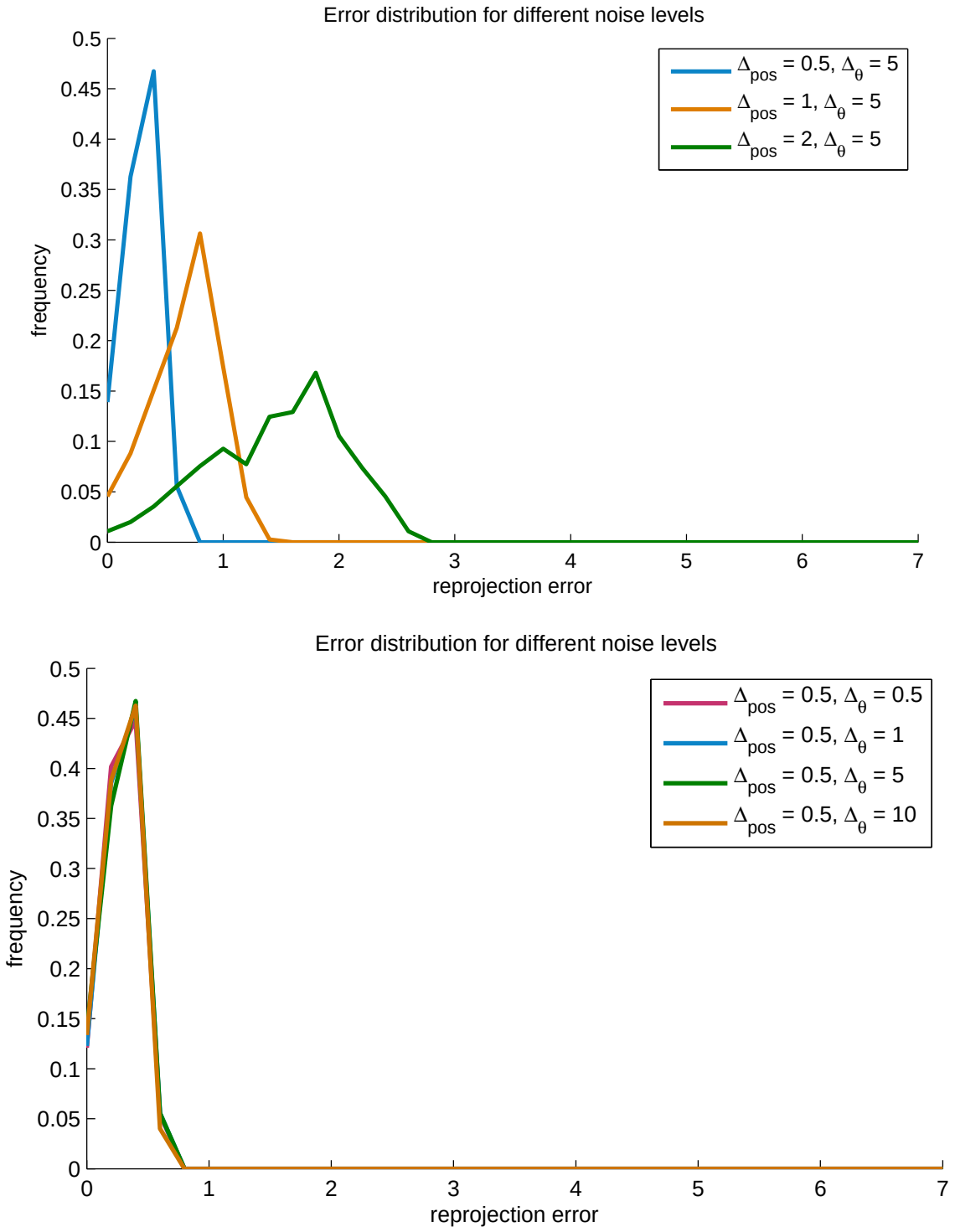

Figure 2: Distributions of reprojection error for synthetic data results **with bundle adjustment**, for (top) increasing levels of positional perturbation while keeping tangential orientation perturbation fixed; and (bottom) increasing levels of tangential orientation perturbation while keeping positional perturbation fixed.

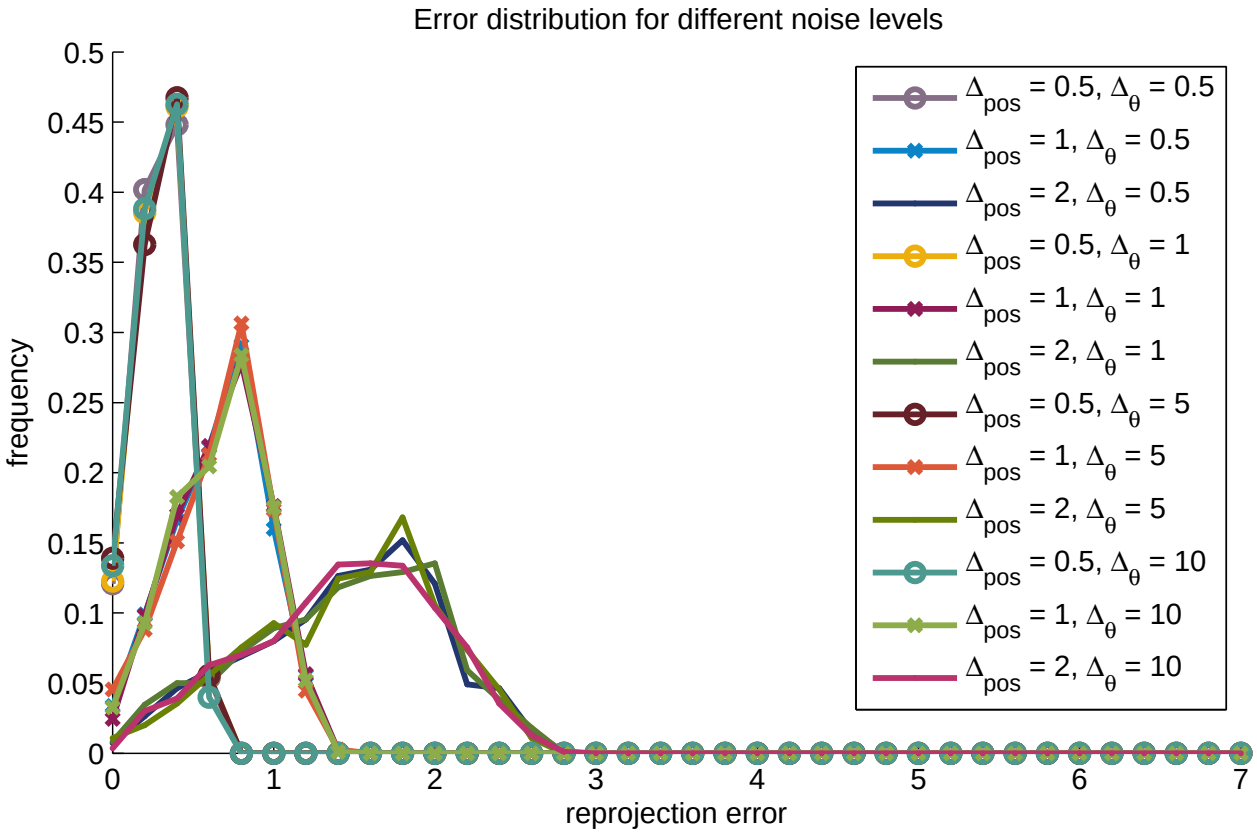

Figure 3: Full set of distributions of reprojection error for synthetic data results **with bundle adjustment**, for increasing levels of positional perturbation and tangential orientation perturbation. This is the same experiment as in Figure 2.

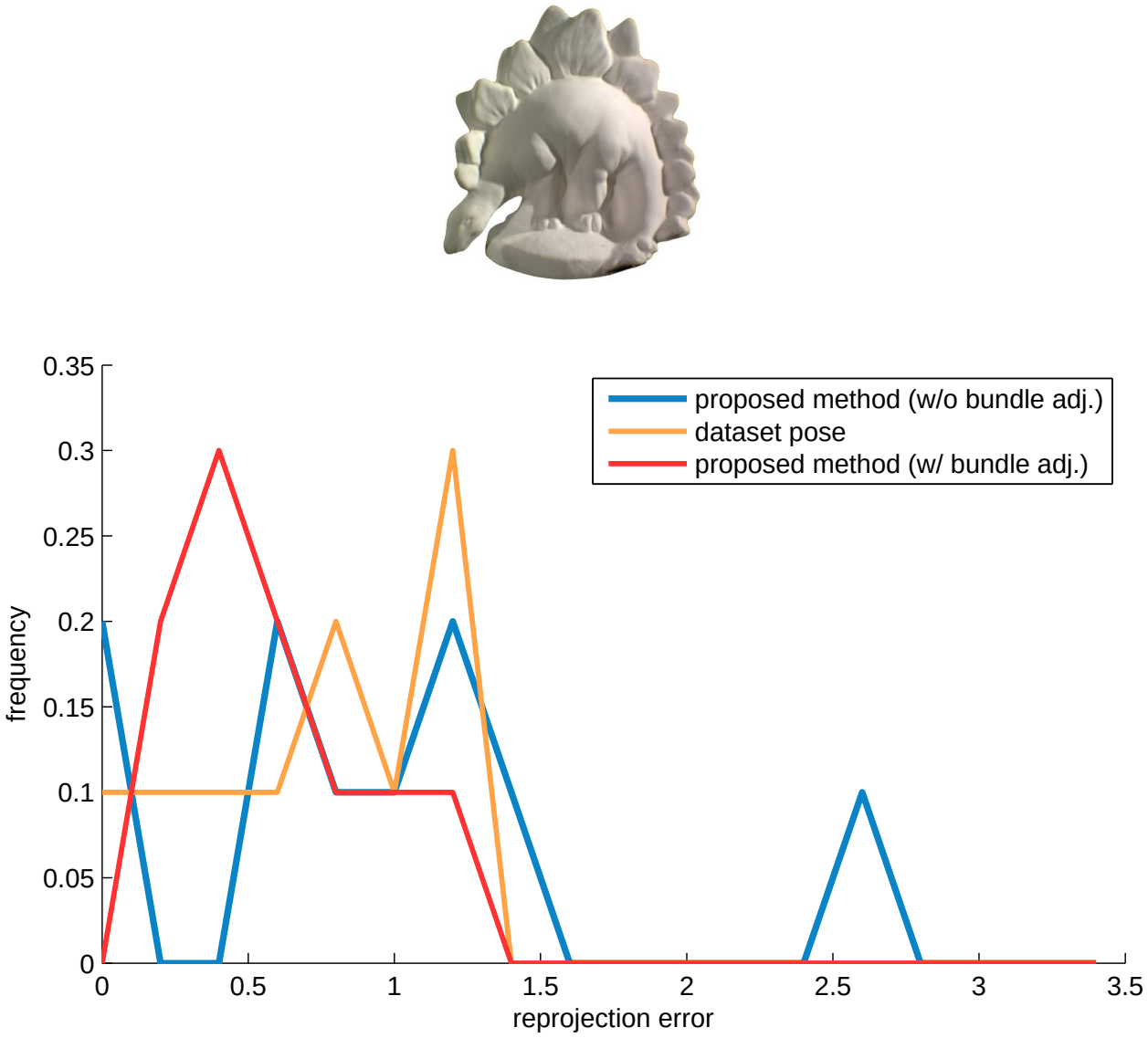

Figure 4: The reprojection error distributions for the standard Dino sequence from the Middlebury multiview stereo database [1], with a sample image shown at the top, using only two point-tangents selected within a RANSAC framework from 10 manual correspondences plus 50% outliers, before and after bundle adjustment. The average reprojection error for the proposed method are $1.03px$ and $0.66px$ before and after bundle adjustment, respectively, while the average error using the dataset camera is $0.88px$.

3 Detailed proof of Theorem 3.1

In the course of proving Theorem 3.1, we will *also* show that

$$\begin{aligned} Q(\rho_1,\rho_2) &= A^3(EH^2-FHK+GK^2)^2 + AC^2(EJ^2-FJL+GL^2)^2 \\ &- 2A^2C(EH^2-FHK+GK^2)(EJ^2-FJL+GL^2) + [-AB(EH^2-FHK+GK^2) \\ &+ BC(EJ^2-FJL+GL^2)]\,[A(2EHJ-FHL-FJK+2GKL) - B(EJ^2-FJL+GL^2)] \\ &+ C[A(2EHJ-FHL-FJK+2GKL) - B(EJ^2-FJL+GL^2)]^2 = 0 \end{aligned} \tag{1}$$

where the parameters A through L are defined as

$$\begin{cases} A = 1 - 2\boldsymbol{\gamma}_1^\top \boldsymbol{t}_1 B_1 + \boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 B_1^2 \\ B = [2(\boldsymbol{\gamma}_1^\top \boldsymbol{t}_1) - 2\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 B_1]A_1 \\ C = (\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1)A_1^2 - 1 \\ E = 1 - 2\boldsymbol{\gamma}_2^\top \boldsymbol{t}_2 B_2 + \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2 B_2^2 \\ F = [2(\boldsymbol{\gamma}_2^\top \boldsymbol{t}_2) - 2\boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2 B_2]A_2 \\ G = (\boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2)A_2^2 - 1 \\ H = \boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 A_1 A_2 - (\boldsymbol{T}_1^w)^\top \boldsymbol{T}_2^w \\ J = [\boldsymbol{\gamma}_2^\top \boldsymbol{t}_1 - \boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 B_1]A_2 \\ K = [\boldsymbol{\gamma}_1^\top \boldsymbol{t}_2 - \boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 B_2]A_1 \\ L = \boldsymbol{t}_1^\top \boldsymbol{t}_2 - \boldsymbol{\gamma}_2^\top \boldsymbol{t}_1 B_2 - \boldsymbol{\gamma}_1^\top \boldsymbol{t}_2 B_1 + \boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 B_1 B_2, \end{cases} \tag{2}$$

where

$$\begin{cases} A_1 = \dfrac{(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top \boldsymbol{T}_1^w}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top \boldsymbol{\gamma}_1} \\ A_2 = \dfrac{(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top \boldsymbol{T}_2^w}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top \boldsymbol{\gamma}_2} \end{cases} \qquad \begin{cases} B_1 = \dfrac{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top \boldsymbol{t}_1}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top \boldsymbol{\gamma}_1} \\ B_2 = \dfrac{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top \boldsymbol{t}_2}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top \boldsymbol{\gamma}_2}, \end{cases} \tag{3}$$

and where

$$\begin{cases} \rho_1 \dfrac{g_1}{G_1} = -\dfrac{A(EH^2-FHK+GK^2) - C(EJ^2-FJL+GL^2)}{A(2EHJ-FHL-FJK+2GKL) - B(EJ^2-FJL+GL^2)}. \\ \rho_2 \dfrac{g_2}{G_2} = -\dfrac{E(AH^2-BHJ+CJ^2) - G(AK^2-BKL+CL^2)}{E(2AHK-BHL-BKJ+2CJL) - F(AK^2-BKL+CL^2)}, \end{cases} \tag{4}$$

and

$$\begin{cases} \dfrac{\rho_1'}{G_1} = A_1 - B_1\rho_1\dfrac{g_1}{G_1} \\ \dfrac{\rho_2'}{G_2} = A_2 - B_2\rho_2\dfrac{g_2}{G_2}. \end{cases} \tag{5}$$

Proof. (*Of Theorem 3.1 and the above statements*) An image point $\boldsymbol{\gamma}$ is related to the underlying space point $\boldsymbol{\Gamma}$ through $\boldsymbol{\Gamma} = \rho\boldsymbol{\gamma}$, where ρ is depth. A space point $\boldsymbol{\Gamma}$ in local coordinates is related to $\boldsymbol{\Gamma}^w$ in the world coordinates by a rotation matrix $\mathcal{R}$ and translation $\mathcal{T}$ through $\boldsymbol{\Gamma} = \mathcal{R}\boldsymbol{\Gamma}^w + \mathcal{T}$. Equating these at each of the two points gives

$$\begin{cases} \rho_1\boldsymbol{\gamma}_1 = \mathcal{R}\boldsymbol{\Gamma}_1^w + \mathcal{T} \\ \rho_2\boldsymbol{\gamma}_2 = \mathcal{R}\boldsymbol{\Gamma}_2^w + \mathcal{T}, \end{cases} \tag{6}$$

where ρ_1 and ρ_2 are the depth at image points $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$, respectively. By differentiating with respect to the parameters of γ_1 and $\boldsymbol{\gamma}_2$ we have:

$$\begin{cases} \rho_1 g_1 \boldsymbol{t}_1 + \rho_1'\boldsymbol{\gamma}_1 = \mathcal{R}G_1\boldsymbol{T}_1^w \\ \rho_2 g_2 \boldsymbol{t}_2 + \rho_2'\boldsymbol{\gamma}_2 = \mathcal{R}G_2\boldsymbol{T}_2^w, \end{cases} \tag{7}$$

where ρ_1 and ρ_2 are depth derivatives with respect to the curve parameter, g_1 and g_2 are speeds of parametrization of $\boldsymbol{\gamma}_1$ and $\boldsymbol{\gamma}_2$, respectively, and G_1 and G_2 are the speeds of parametrization of the space curves $\boldsymbol{\Gamma}_1$ and $\boldsymbol{\Gamma}_2$, respectively. The vector Equations 6 and 7 represent 3 scalar equations for each point, so that there are 12 equations in all. The parametrization speeds g_1 and g_2 are arbitrary and can be set to 1 uniformly, but we keep them in general form. The given quantities are $\boldsymbol{\gamma}$, $\boldsymbol{t}$, and $\boldsymbol{\Gamma}^w$, $\boldsymbol{T}^w$ at each point. The unknowns are $\mathcal{R}$, $\mathcal{T}$ (6 unknowns), ρ, ρ' (4 unknowns), and the two speeds of the curve Γ at the two points, 12 unknowns in all. Therefore, in principle, two points should provide enough constraints to solve the problem.

First, $\mathcal{T}$ is eliminated by subtracting the two Equations (6)

$$\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2 = \mathcal{R}(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w), \tag{8}$$

which together with Equation 7 gives a system of equations

$$\begin{cases} \rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2 = \mathcal{R}(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w) & (9)\\ \rho_1\dfrac{g_1}{G_1}\boldsymbol{t}_1 + \dfrac{\rho_1'}{G_1}\boldsymbol{\gamma}_1 = \mathcal{R}\boldsymbol{T}_1^w & (10)\\ \rho_2\dfrac{g_2}{G_2}\boldsymbol{t}_2 + \dfrac{\rho_2'}{G_2}\boldsymbol{\gamma}_2 = \mathcal{R}\boldsymbol{T}_2^w. & (11) \end{cases}$$

At this stage, the unknowns are ρ_1, ρ_2, $\frac{\rho_1'}{G_1}$, $\frac{\rho_2'}{G_2}$, $\rho_1\frac{g_1}{G_1}$, $\rho_2\frac{g_2}{G_2}$, and $\mathcal{R}$, nine numbers in all, which can potentially be solved through the three vector equations (nine scalar equations) in (9)–(11). The number of unknowns can be reduced by eliminating $\mathcal{R}$ in a second step. The matrix $\mathcal{R}$ rotates three known vectors, $(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)$, $\boldsymbol{T}_1^w$, and $\boldsymbol{T}_2^w$ to the three unknown vectors on the left side of these equations, requiring a preservation of vector lengths and mutual angles. The length and relative angles are obtained from the known dot products, which do not involve $\mathcal{R}$ at all. This provides six equations for the six unknowns $\{\rho_1, \rho_2, \frac{g_1}{G_1}, \frac{g_2}{G_2}, \frac{\rho_1'}{G_1}, \frac{\rho_2'}{G_2}\}$. Alternatively, we write these three equations in matrix form composed from the three vector equations (9)–(11), *i.e.*,

$$\left[\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2 \quad \rho\tfrac{g_1}{G_1}\boldsymbol{t}_1 + \tfrac{\rho_1'}{G_1}\boldsymbol{\gamma}_1 \quad \rho_2\tfrac{g_2}{G_2}\boldsymbol{t}_2 + \tfrac{\rho_2'}{G_2}\boldsymbol{\gamma}_2\right] = \mathcal{R}\left[(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w) \quad \boldsymbol{T}_1^w \quad \boldsymbol{T}_2^w\right] \tag{12}$$

This is a system of six equations. Note that a clear geometric condition for the problem to have a solution is that the vectors $\{(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w), \boldsymbol{T}_1^w, \boldsymbol{T}_2^w\}$ be non-coplanar. Using product of the left hand matrix with its transpose, and using $\mathcal{R}^\top\mathcal{R} = I$, gives

$$\begin{cases} (\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2) = (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)\\ (\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top(\rho_1\dfrac{g_1}{G_1}\boldsymbol{t}_1 + \dfrac{\rho_1'}{G_1}\boldsymbol{\gamma}_1) = (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top\boldsymbol{T}_1^w\\ (\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top(\rho_2\dfrac{g_2}{G_2}\boldsymbol{t}_2 + \dfrac{\rho_2'}{G_2}\boldsymbol{\gamma}_1) = (\boldsymbol{\Gamma}_2^w - \boldsymbol{\Gamma}_2^w)^\top\boldsymbol{T}_2^w\\ (\rho_1\dfrac{g_1}{G_1}\boldsymbol{t}_1 + \dfrac{\rho_1'}{G_1}\boldsymbol{\gamma}_1)^\top(\rho_1\dfrac{g_1}{G_1}\boldsymbol{t}_1 + \dfrac{\rho_1'}{G_1}\boldsymbol{\gamma}_1) = 1\\ (\rho_2\dfrac{g_2}{G_2}\boldsymbol{t}_2 + \dfrac{\rho_2'}{G_2}\boldsymbol{\gamma}_2)^\top(\rho_2\dfrac{g_2}{G_2}\boldsymbol{t}_2 + \dfrac{\rho_2'}{G_2}\boldsymbol{\gamma}_2) = 1\\ (\rho_1\dfrac{g_1}{G_1}\boldsymbol{t}_1 + \dfrac{\rho_1'}{G_1}\boldsymbol{\gamma}_1)^\top(\rho_2\dfrac{g_2}{G_2}\boldsymbol{t}_2 + \dfrac{\rho_2'}{G_2}\boldsymbol{\gamma}_2) = (\boldsymbol{T}_1^w)^\top\boldsymbol{T}_2^w. \end{cases} \tag{13}$$

The first equation is a quadratic in ρ_1 and ρ_2

$$\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1\,\rho_1^2 - 2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2\,\rho_1\rho_2 + \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2\,\rho_2^2 = (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w), \tag{14}$$

which as a conic in the ρ_1–ρ_2 plane with negative discriminant

$$(\boldsymbol{\gamma}_1\cdot\boldsymbol{\gamma}_2)^2 - (\boldsymbol{\gamma}_1\cdot\boldsymbol{\gamma}_1)(\boldsymbol{\gamma}_2\cdot\boldsymbol{\gamma}_2) = -\|\boldsymbol{\gamma}_1\times\boldsymbol{\gamma}_2\|^2 < 0 \tag{15}$$

is an ellipse. The ellipse is centered at the origin so we can check that it has real points by solving for ρ_1 when $\rho_2 = 0$, giving $\rho_1^2\|\boldsymbol{\gamma}_1\|^2 = \|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2$, or real roots $\rho_1 = \pm\frac{\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|}{\|\boldsymbol{\gamma}_1\|}$.

The remaining five equations involve the additional unknowns $\{\rho_1\frac{g_1}{G_1}, \rho_2\frac{g_2}{G_2}, \frac{\rho_1'}{G_1}, \frac{\rho_2'}{G_2}\}$. The latter appear in a linear form in the second and third equations, and in quadratic form in the last three equations. Thus, the terms $\frac{\rho_1'}{G_1}$ and $\frac{\rho_2'}{G_2}$ can be isolated from the second and third equations and then used in the last three equations

$$\begin{cases} [(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{\gamma}_1]\dfrac{\rho_1'}{G_1} = (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top\boldsymbol{T}_1^w - [(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{t}_1]\rho_1\dfrac{g_1}{G_1} \\ [(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{\gamma}_2]\dfrac{\rho_2'}{G_2} = (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top\boldsymbol{T}_2^w - [(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{t}_2]\rho_2\dfrac{g_2}{G_2}, \end{cases} \tag{16}$$

or

$$\begin{cases} \dfrac{\rho_1'}{G_1} = \dfrac{(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top\boldsymbol{T}_1^w}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{\gamma}_1} - \left[\dfrac{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{t}_1}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{\gamma}_1}\right]\rho_1\dfrac{g_1}{G_1} \quad = A_1 - B_1\rho_1\dfrac{g_1}{G_1} \\ \dfrac{\rho_2'}{G_2} = \dfrac{(\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top\boldsymbol{T}_2^w}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{\gamma}_2} - \left[\dfrac{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{t}_2}{(\rho_1\boldsymbol{\gamma}_1 - \rho_2\boldsymbol{\gamma}_2)^\top\boldsymbol{\gamma}_2}\right]\rho_2\dfrac{g_2}{G_2} \quad = A_2 - B_2\rho_2\dfrac{g_2}{G_2}, \end{cases} \tag{17}$$

noting that A_1, A_2, B_1, and B_2 depend on only two of the unknowns ρ_1 and ρ_2. The last three equations in (13) can be expanded as

$$\begin{cases} \left(\rho_1\dfrac{g_1}{G_1}\right)^2 + 2(\boldsymbol{\gamma}_1^\top\boldsymbol{t}_1)\left(\rho_1\dfrac{g_1}{G_1}\right)\left(\dfrac{\rho_1'}{G_1}\right) + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)\left(\dfrac{\rho_1'}{G_1}\right)^2 = 1 \\ \left(\rho_2\dfrac{g_2}{G_2}\right)^2 + 2(\boldsymbol{\gamma}_2^\top\boldsymbol{t}_2)\left(\rho_2\dfrac{g_2}{G_2}\right)\left(\dfrac{\rho_2'}{G_2}\right) + (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\left(\dfrac{\rho_2'}{G_2}\right)^2 = 1 \\ (\boldsymbol{t}_1^\top\boldsymbol{t}_2)\left(\rho_1\dfrac{g_1}{G_1}\right)\left(\rho_2\dfrac{g_2}{G_2}\right) + (\boldsymbol{\gamma}_2^\top\boldsymbol{t}_1)\left(\rho_1\dfrac{g_1}{G_1}\right)\left(\dfrac{\rho_2'}{G_2}\right) + (\boldsymbol{\gamma}_1^\top\boldsymbol{t}_2)\left(\rho_2\dfrac{g_2}{G_2}\right)\left(\dfrac{\rho_1'}{G_1}\right) + \\ \qquad (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)\left(\dfrac{\rho_1'}{G_1}\right)\left(\dfrac{\rho_2'}{G_2}\right) = (\boldsymbol{T}_1^w)^\top\boldsymbol{T}_2^w. \end{cases}$$

Substituting $\frac{\rho_1'}{G_1}$ and $\frac{\rho_2'}{G_2}$ from Equations 17 gives

$$\begin{cases} \left(\rho_1\dfrac{g_1}{G_1}\right)^2 + 2(\boldsymbol{\gamma}_1^\top\boldsymbol{t}_1)\left(\rho_1\dfrac{g_1}{G_1}\right)\left(A_1 - B_1\left(\rho_1\dfrac{g_1}{G_1}\right)\right) + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)\left(A_1 - B_1\left(\rho_1\dfrac{g_1}{G_1}\right)\right)^2 = 1 \\ \left(\rho_2\dfrac{g_2}{G_2}\right)^2 + 2(\boldsymbol{\gamma}_2^\top\boldsymbol{t}_2)\left(\rho_2\dfrac{g_2}{G_2}\right)\left(A_2 - B_2\left(\rho_2\dfrac{g_2}{G_2}\right)\right) + (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\left(A_2 - B_2\left(\rho_2\dfrac{g_2}{G_2}\right)\right)^2 = 1 \\ (\boldsymbol{t}_1^\top\boldsymbol{t}_2)\left(\rho_1\dfrac{g_1}{G_1}\right)\left(\rho_2\dfrac{g_2}{G_2}\right) + (\boldsymbol{\gamma}_2^\top\boldsymbol{t}_1)\left(\rho_1\dfrac{g_1}{G_1}\right)\left(A_2 - B_2\left(\rho_2\dfrac{g_2}{G_2}\right)\right) + \\ (\boldsymbol{\gamma}_1^\top\boldsymbol{t}_2)\left(\rho_2\dfrac{g_2}{G_2}\right)\left(A_1 - B_1\left(\rho_1\dfrac{g_1}{G_1}\right)\right) + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)\left(A_1 - B_1\left(\rho_1\dfrac{g_1}{G_1}\right)\right)\left(A_2 - B_2\left(\rho_2\dfrac{g_2}{G_2}\right)\right) \\ = (\boldsymbol{T}_1^w)^\top\boldsymbol{T}_2^w. \end{cases}$$

These three equations can be written in summary form using $x_1 = \rho_1\frac{g_1}{G_1}$ and $x_2 = \rho_2\frac{g_2}{G_2}$,

$$\begin{cases} Ax_1^2 + Bx_1 + C = 0 & (18) \\ Ex_2^2 + Fx_2 + G = 0 & (19) \\ H + Jx_1 + Kx_2 + Lx_1x_2 = 0, & (20) \end{cases}$$

and where A through L are only functions of the two unknowns ρ_1 and ρ_2. Thus, the three Equations 18–20 after solving for x_1 and x_2 express a relationship between ρ_1 and ρ_2, which together with Equation 14 can lead to a solution for ρ_1 and ρ_2.

Equation 20, with given values for ρ_1 and ρ_2, represents a rectangular hyperbola in the x_1–x_2 plane, as illustrated in the paper, and each of the Equations 18 and 19 represents a pair of (real) lines in the same plane, parallel respectively to the x_2 and x_1 axes. In general there will not be more than one intersection between the aforementioned curves.

Specifically, the variables x_1 and x_2 can be solved by rewriting Equation 20 as

$$(H + Jx_1) + (K + Lx_1)x_2 = 0, \tag{21}$$

giving

$$x_2 = -\frac{H + Jx_1}{K + Lx_1}. \tag{22}$$

Using this expression in Equation 19 gives

$$E\frac{(H + Jx_1)^2}{(K + Lx_1)^2} - F\frac{H + Jx_1}{K + Lx_1} + G = 0, \tag{23}$$

or

$$E(H + Jx_1)^2 - F(H + Jx_1)(K + Lx_1) + G(K + Lx_1)^2 = 0. \tag{24}$$

Reorganizing as a quadratic in x_1, this solves for x_1 which together with Equation 18 gives a constraint on the parameters depending on ρ_1 and ρ_2,

$$\begin{cases} (EJ^2 - FJL + GL^2)x_1^2 + (2EHJ - FHL - FJK + 2GKL)x_1 & (25) \\ \quad +(EH^2 - FHK + GK^2) = 0 & \\ \qquad\qquad Ax_1^2 + Bx_1 + C = 0. & (26) \end{cases}$$

The quadratic term is eliminated by multiplying the first equation by A and the second equation by $(EJ^2 - FJL + GL^2)$ and subtracting, giving

$$\begin{aligned} &[A(2EHJ - FHL - FJK + 2GKL) - B(EJ^2 - FJL + GL^2)]x_1 + \\ &\qquad [A(EH^2 - FHK + GK^2) - C(EJ^2 - FJL + GL^2)] = 0, \end{aligned} \tag{27}$$

so that

$$x_1 = -\frac{A(EH^2 - FHK + GK^2) - C(EJ^2 - FJL + GL^2)}{A(2EHJ - FHL - FJK + 2GKL) - B(EJ^2 - FJL + GL^2)}. \tag{28}$$

Substituting back into Equation 26 gives

$$\begin{aligned} &A\left[\frac{A(EH^2 - FHK + GK^2) - C(EJ^2 - FJL + GL^2)}{A(2EHJ - FHL - FJK + 2GKL) - B(EJ^2 - FJL + GL^2)}\right]^2 + \\ &-B\frac{A(EH^2 - FHK + GK^2) - C(EJ^2 - FJL + GL^2)}{A(2EHJ - FHL - FJK + 2GKL) - B(EJ^2 - FJL + GL^2)} + C = 0, \end{aligned} \tag{29}$$

or

$$\begin{aligned} &A^3(EH^2 - FHK + GK^2)^2 + AC^2(EJ^2 - FJL + GL^2)^2 \\ &- 2A^2C(EH^2 - FHK + GK^2)(EJ^2 - FJL + GL^2) + [-AB(EH^2 - FHK + GK^2) \\ &+ BC(EJ^2 - FJL + GL^2)]\,[A(2EHJ - FHL - FJK + 2GKL) - B(EJ^2 - FJL + GL^2)] \\ &+ C[A(2EHJ - FHL - FJK + 2GKL) - B(EJ^2 - FJL + GL^2)]^2 = 0 \end{aligned} \tag{30}$$

The equation, after expressions for A, B, …, L are substituted in, can be divided by $\rho_1^4\rho_2^4$, giving an 8^{th} order polynomial equation in ρ_1 and ρ_2, *i.e.*, $Q(\rho_1, \rho_2) = 0$. This equation together with Equation 14 represents a system of two equations in two unknowns

$$\begin{cases} \boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1\, \rho_1^2 - 2\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2\, \rho_1\rho_2 + \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2\, \rho_2^2 = (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w)^\top (\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w), \\ Q(\rho_1, \rho_2) = 0, \end{cases} \tag{31}$$

and gives a number of solutions for ρ_1, and ρ_2 which in turn solve for the unknowns $\rho_1\frac{g_1}{G_1}$, $\rho_2\frac{g_2}{G_2}$, $\frac{\rho_1'}{G_1}$, and $\frac{\rho_2}{G_2}$. Once these unknowns are solved for, the rotation $\mathcal{R}$ can be obtained from the matrix equation (12). The translation $\mathcal{T}$ is then solved from Equations 6 as

$$\mathcal{T} = \rho_1\boldsymbol{\gamma}_1 - \mathcal{R}\boldsymbol{\Gamma}_1^w. \tag{32}$$

□

4 Details in the Proof of Proposition 3.2

The parametrization we have assumed in the space curve projects $\boldsymbol{T}$ to the same half plane as $\boldsymbol{t}$ in each view so that $\boldsymbol{T}$ and $\boldsymbol{t}$ need to point in the same direction, *i.e.*, $\boldsymbol{T}\cdot\boldsymbol{t} > 0$, or from Equations 10 and 11, $\frac{g_1}{G_1} > 0$ and $\frac{g_2}{G_2} > 0$.

5 Details in the Proof of Proposition 4.1

The parameters α, β, and θ for the ellipse in Equation 14 can be found by substitution of ρ_1 and ρ_2 in the parametric form (given in the paper) into Equation 14. Specifically, writing

$$\begin{aligned}
&\frac{\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1}{(1+t^2)^2}[4\alpha^2t^2\cos^2\theta + \beta^2(1-t^2)^2\sin^2\theta + 4\alpha\beta t(1-t^2)\sin\theta\cos\theta] + \\
&-\frac{2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2}{(1+t^2)^2}[-4\alpha^2t^2\sin\theta\cos\theta + 2\alpha\beta t(1-t^2)\cos^2\theta - 2\alpha\beta t(1-t^2)\sin^2\theta] + \beta^2(1-t^2)^2\sin\theta\cos\theta \\
&\frac{-2\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2}{(1+t^2)^2}[4\alpha^2t^2\sin^2\theta + \beta^2(1-t^2)^2\cos^2\theta - 4\alpha\beta t(1-t^2)\sin\theta\cos\theta] = \|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2.
\end{aligned} \tag{33}$$

Simplifying the equation as

$$\begin{aligned}
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)4\alpha^2t^2 - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2) + (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\beta^2(1-t^2)^2]\cos^2\theta + \\
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)\beta^2(1-t^2)^2 + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2)(\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)4\alpha^2t^2]\sin^2\theta + \\
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)4\alpha\beta t(1-t^2) + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2 8\alpha^2t^2 - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)2\beta^2(1-t^2)^2 - (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2)]\sin\theta\cos\theta \\
&= (1+t^2)^2\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2
\end{aligned} \tag{34}$$

and using simple trigonometric identities $\cos^2\theta = \frac{1+\cos(2\theta)}{2}$ and $\sin^2\theta = \frac{1-\sin(2\theta)}{2}$, $\cos^2\theta - \sin^2\theta = \cos(2\theta)$ and $\sin(2\theta) = 2\sin\theta\cos\theta$, this equation can be better simplified to

$$\begin{aligned}
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)4\alpha^2t^2 - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2) + (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\beta^2(1-t^2)^2](1+\cos(2\theta)) + \\
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)\beta^2(1-t^2)^2 + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2) + (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)4\alpha^2t^2](1-\cos(2\theta)) + \\
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1)4\alpha\beta t(1-t^2) + (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)8\alpha^2t^2 - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)2\beta^2(1-t^2)^2 - (\boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2)]\sin(2\theta) \\
&= 2(1+t^2)^2\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2.
\end{aligned} \tag{35}$$

which is an equation only involving the unknown θ,

$$\begin{aligned}
&(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)[4\alpha^2t^2 + \beta^2(1-t^2)] + \\
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)[4\alpha^2t^2 - \beta^2(1-t^2)^2] - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2)8\alpha\beta t(1-t^2)]\cos(2\theta) \\
&[(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)4\alpha\beta t(1-t^2) + 2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2[4\alpha^2t^2 - \beta^2(1-t^2)^2]]\sin(2\theta) \\
&= 2(1+t^2)^2\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|^2.
\end{aligned} \tag{36}$$

This equation holds for all values of t. For $t = 0$,

$$(\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\beta^2 - (\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2 - \boldsymbol{\gamma}_2^\top\boldsymbol{\gamma}_2)\beta^2\cos(2\theta) - 2\boldsymbol{\gamma}_1^\top\boldsymbol{\gamma}_2\beta^2\sin(2\theta) = 2\|\boldsymbol{\Gamma}_1^w - \boldsymbol{\Gamma}_2^w\|, \tag{37}$$

giving

$$\beta^2 = \frac{2\|\mathbf{\Gamma}_1^w - \mathbf{\Gamma}_2^w\|^2}{(\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2) - (\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2)\cos(2\theta) - 2\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 \sin(2\theta)}. \tag{38}$$

Similarly, at $t = 1$,

$$(\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2)4\alpha^2 + (\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2)4\alpha^2 \cos(2\theta) + 2\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 4\alpha^2 \sin(2\theta) = 8\|\mathbf{\Gamma}_1^w - \mathbf{\Gamma}_2^w\|^2, \tag{39}$$

giving

$$\alpha^2 = \frac{2\|\mathbf{\Gamma}_1^w - \mathbf{\Gamma}_2^w\|^2}{(\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 + \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2) + (\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_1 - \boldsymbol{\gamma}_2^\top \boldsymbol{\gamma}_2)\cos(2\theta) + 2\boldsymbol{\gamma}_1^\top \boldsymbol{\gamma}_2 \sin(2\theta)}. \tag{40}$$

6 Additional Remarks

We plan to provide the Matlab source code for our pose estimation approach to the public once this paper gets accepted.

References

[1] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR'06*, pages 519–528. IEEE Computer Society, 2006. 1, 5