Abstract

In this paper, we study closed-form optimal solutions to two-view triangulation with known internal calibration and pose. By formulating the triangulation problem as $L_1$ and $L_\infty$ minimization of angular reprojection errors, we derive the exact closed-form solutions that guarantee global optimality under respective cost functions. To the best of our knowledge, we are the first to present such solutions. Since the angular error is rotationally invariant, our solutions can be applied for any type of central cameras, be it perspective, fisheye or omnidirectional. Our methods also require significantly less computation than the existing optimal methods. Experimental results on synthetic and real datasets validate our theoretical derivations.

1. Introduction

Recovering the position of a 3D point given its projections in two or more cameras is called triangulation. It constitutes a fundamental building block in stereo vision [33], simultaneous localization and mapping (SLAM) [22] and structure-from-motion (SfM) pipelines [29]. For large problems, reconstructing thousands (or millions) of points is not uncommon, so achieving fast and accurate triangulation is important for the performance of such systems.

If one assumes the exact knowledge of camera matrices and noiseless feature measurements, triangulation amounts to intersecting two backprojected rays that correspond to the same 3D point. In practice, however, this assumption is unrealistic, and the rays do not necessarily intersect. Therefore, a nontrivial method is required even for just two views.

The standard approach is to find the 3D point that minimizes a chosen cost function given the feature measurements. The most common are the $L_1$ norm (sum of magnitude), $L_2$ norm (sum of squares) and $L_\infty$ norm (maximum) of image reprojection errors. While reasonable for perspective cameras, the image reprojection error does not generalize well to different camera types (e.g., omnidirectional or fully spherical panoramic cameras). This motivates the use of angular reprojection error, a rotationally invariant alternative to the image reprojection error that is generic and independent of the projection geometry [21, 24, 27].

In this work, we derive, for the first time to our knowledge, the exact closed-form solutions to the $L_1$ and $L_\infty$ optimal triangulation from two views based on the angular reprojection error. Unlike iterative methods (e.g., [13, 17]), the proposed methods guarantee global optimality without any iterations, and unlike polynomial methods (e.g., [10, 23, 32]), they do not involve finding the roots of a higher-degree polynomial. Hence, our methods simultaneously provide the global optimality, speed and simplicity. We also present our own derivation of the $L_2$ optimal solution that is much more compact and geometrically intuitive than the existing one [24]. Since all three methods are based on the angular error, they are not limited to standard perspective cameras and can also be used for fisheye, omnidirectional and fully spherical panoramic cameras.

The paper is organized as follows. In the next three sections, we discuss the related work and preliminaries. Section 5, 6 and 7 respectively present the closed-form solutions to the $L_1$, $L_2$ and $L_\infty$ optimal triangulation. To make our paper compact and easily accessible, we separated the proofs from our main findings and put them in the appendix. Section 8 addresses the cheirality constraint. Finally, experimental results are provided in Section 9, followed by the conclusions in Section 10.

2. Related Work

The most widespread approach to triangulation is to find the 3D point that minimizes the $L_2$ norm of image reprojection errors [8]. Assuming that image points are perturbed by Gaussian noise, the $L_2$ optimal solution gives the maximum likelihood estimate (MLE). This can be obtained in closed form by solving a polynomial of degree 6 for two views [10] and degree 47 for three views [32]. Such polynomial methods are, however, computationally expensive and susceptible to ill-conditioning [17]. Besides, an iterative search for the roots may converge to a local minimum [10].

Another two-view method by Kanatani et al. [13] iteratively corrects the 2D projections of the points. Although this method was shown to be faster than the one by Hartley
and Sturm [10], it does not satisfy the epipolar constraint [19] in each iteration. Lindstrom [17] solved this problem with an improved iterative algorithm that is even more stable and faster. However, neither his method nor Kanatani’s guarantee global optimality. Oliensis [24] showed that by formulating the problem as L2 minimization of the sine of angular reprojection errors, an exact closed-form solution can be derived for two-view triangulation.

Instead of minimizing the L2 norm, one may choose to minimize the L1 norm of reprojection errors. The advantage of L1 norm is that it is more robust to outliers as it places less emphasis on large errors [10, 11]. For two views, Hartley and Sturm [10] showed that the L1 optimal solution can be obtained in closed form by solving a polynomial of degree 8. They also found that the L1 optimization gives slightly more accurate 3D results than the L2 optimization.

In geometric problems, another popular norm is the L∞ norm. The L∞ optimal solution corresponds to the MLE under the assumption of uniform noise in the image points [6]. The advantage of the L∞ cost function over the L2 cost is that it is relatively simpler and has a single minimum [9]. For the case of two views, Nister [23] showed that the optimal solution can be obtained in closed form by keeping the reprojection errors equal in the two views and solving the resulting quartic equation. A main drawback of the L∞ cost is that it is relatively more sensitive to outliers [9]. This being said, such sensitivity was shown to be useful for outlier removal [30, 26, 16].

While most of the aforementioned works formulate their optimization problem in terms of the image reprojection error, the angular reprojection error is another popular choice. It embodies a better noise model for fisheye or omnidirectional cameras [24, 20]. Even for perspective cameras, the assumption of Gaussian noise is not justified [6], and the angular reprojection error is just as valid as the image reprojection error, if not more so. In the literature, it has been proposed to minimize the sine of angular reprojection errors in L2 norm [24], the tangent in L2 or L∞ norm [7, 9, 12], and the cosine in negative L1 norm [28, 3]. In contrast to these methods, our L1 and L∞ optimization do not involve trigonometric functions.

3. Preliminaries on 3D Geometry

Throughout the paper, we adopt the following notation: We use bold letters for vectors and matrices, and light letters for scalars. The Euclidean norm of a vector v is denoted by ||v||, and the unit vector by \( \hat{v} = v/||v|| \). The angle between two lines \( L_0 \) and \( L_1 \) is denoted by \( \angle(L_0, L_1) \in [0, \pi/2] \).

The following vector identities will come in handy later:

\[
\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = \mathbf{b} \cdot (\mathbf{c} \times \mathbf{a}) = \mathbf{c} \cdot (\mathbf{a} \times \mathbf{b}) \]  
\[
||\hat{a} \times \hat{b}||^2 = 1 - (\hat{a} \cdot \hat{b})^2
\]

We also make frequent use of the following formulas:

1. The distance between a point \( \mathbf{p} \) and a plane \( \Pi_0(\mathbf{x}) = \mathbf{n}_0 \cdot (\mathbf{x} - \mathbf{c}_0) = 0 \) is given by \( ||\mathbf{p} - \mathbf{r}_0|| \) where \( \mathbf{r}_0 \) is the projection of \( \mathbf{p} \) onto \( \Pi_0 \). This is computed as follows:

\[
||\mathbf{p} - \mathbf{r}_0|| = |\hat{\mathbf{n}}_0 \cdot (\mathbf{p} - \mathbf{c}_0)|.
\]

2. The distance between two skew lines \( L_0(s_0) = \mathbf{c}_0 + s_0 \mathbf{m}_0 \) and \( L_1(s_1) = \mathbf{c}_1 + s_1 \mathbf{m}_1 \) is given by \( ||\mathbf{r}_0 - \mathbf{r}_1|| \) where \( \mathbf{r}_0 \) and \( \mathbf{r}_1 \) are the points on each line that form the closest pair. Letting \( \mathbf{t} = \mathbf{c}_0 - \mathbf{c}_1 \) and \( \mathbf{q} = \mathbf{m}_0 \times \mathbf{m}_1 \), this is computed as follows:

\[
||\mathbf{r}_0 - \mathbf{r}_1|| = |\mathbf{t} \cdot \hat{\mathbf{q}}|.
\]

The two points can also be obtained individually [14]:

\[
\mathbf{r}_0 = \mathbf{c}_0 + \frac{\mathbf{q} \cdot (\mathbf{m}_0 \times \mathbf{t})}{||\mathbf{q}||^2} \mathbf{m}_0,
\]

\[
\mathbf{r}_1 = \mathbf{c}_1 + \frac{\mathbf{q} \cdot (\mathbf{m}_1 \times \mathbf{t})}{||\mathbf{q}||^2} \mathbf{m}_1.
\]

Equation (4) can be interpreted as the minimum amount of translation required for the two lines to intersect. In this work, it will be also important to know the minimum amount of rotation (or pivot) required for the two lines to intersect. We answer this question in the following lemma:

Lemma 1 (Minimum Pivot Angle for Intersection)

Given two skew lines \( L_0(s_0) = \mathbf{c}_0 + s_0 \mathbf{m}_0 \) and \( L_1(s_1) = \mathbf{c}_1 + s_1 \mathbf{m}_1 \), let \( \mathbf{L}_0 \) be the line that forms the smallest angle \( \theta_0 \in [0, \pi/2] \) to \( \mathbf{L}_0 \) among all possible lines that intersect both point \( \mathbf{c}_0 \) and line \( \mathbf{L}_1 \). Then, \( \mathbf{L}_0 \) is the projection of \( \mathbf{L}_0 \) onto the plane that contains \( \mathbf{c}_0 \) and \( \mathbf{L}_1 \). Furthermore, letting \( \mathbf{t} = \mathbf{c}_0 - \mathbf{c}_1 \) and \( \mathbf{n}_1 = \mathbf{m}_1 \times \mathbf{t} \),

\[
\sin \theta_0 = ||\hat{\mathbf{n}}_1 \cdot \hat{\mathbf{m}}_0||.
\]

We call \( \theta_0 \) the minimum pivot angle for intersection, as it represents the smallest angle required for pivoting line \( \mathbf{L}_0 \) at \( \mathbf{c}_0 \) to make it intersect \( \mathbf{L}_1 \).

Proof. Refer to Appendix A.

4. Preliminaries on Two-View Triangulation

Consider two cameras \( \mathbf{C}_0 \) and \( \mathbf{C}_1 \) observing the same 3D world point \( \mathbf{x}_w \). Let \( \mathbf{c}_0 \) and \( \mathbf{c}_1 \) be their positions in the world frame, and let \( \mathbf{R} \) and \( \mathbf{t} \) be the rotation matrix and translation vector that together transform a point from the camera frame \( \mathbf{C}_0 \) to \( \mathbf{C}_1 \), i.e., \( \mathbf{x}_1 = \mathbf{R} \mathbf{x}_0 + \mathbf{t} \), where \( \mathbf{x}_0 = [x_0, y_0, z_0]^T \) and \( \mathbf{x}_1 = [x_1, y_1, z_1]^T \) correspond to \( \mathbf{x}_w \) in camera frame \( \mathbf{C}_0 \) and \( \mathbf{C}_1 \), respectively. Since triangulation is impossible for zero translation, we set \( ||\mathbf{t}|| = ||\mathbf{c}_0 - \mathbf{c}_1|| = 1 \) without loss of generality. Let \( \mathbf{u}_0 = [u_0, v_0, 1]^T \) and \( \mathbf{u}_1 = [u_1, v_1, 1]^T \) be the homogeneous pixel coordinates of the estimated correspondence to \( \mathbf{x}_w \) in each frame. Given the camera calibration matrix \( \mathbf{K} \), the normalized image coordinates \( \mathbf{f}_0 = [x_0/z_0, y_0/z_0, 1]^T \) and \( \mathbf{f}_1 = [x_1/z_1, y_1/z_1, 1]^T \) are related to \( \mathbf{u}_0 \) and \( \mathbf{u}_1 \) by \( \mathbf{u}_0 = \mathbf{Kf}_0 \) and \( \mathbf{u}_1 = \mathbf{Kf}_1 \).
The two backprojected rays in frame $C_1$, i.e., $r_1(s_1) = s_1f_1$ and $r_0(s_0) = s_0Rf_0 + t$, do not necessarily intersect due to inaccuracies in the image measurements and camera matrices. For the rays to intersect, $f_0$ and $f_1$ must be corrected to $f'_0$ and $f'_1$ such that the epipolar constraint [19] is satisfied. It enforces the coplanarity of $f'_1$, $Rf'_0$ and $t$, and is given by

$$f'_1 \cdot (t \times Rf'_0) = 0.$$  

(8)

The goal of the optimal triangulation is to \textit{minimally} correct the feature rays so that they satisfy (8) and intersect at some point $x'_1$ in frame $C_1$. What is meant by “minimal” depends on the chosen cost function and error criterion. Fig. 1 illustrates two most popular error criteria, namely the image reprojection error and the angular reprojection error. Formally, they are defined as follows:

$$d_i := ||u_i - u'_i|| = ||K(f_i - f'_i)||,$$  

(9)

$$\theta_i := \angle (f_i, f'_i) = \angle (K^{-1}u_i, K^{-1}u'_i),$$  

(10)

In this work, we minimize the latter in $L_1$, $L_2$ and $L_\infty$ norms. Once we have the optimal $f'_0$ and $f'_1$, the point of intersection $x'_1$ can be obtained using either (5) or (6):

$$x'_1 = t + \frac{z \cdot (t \times f'_1)}{||z||^2} Rf'_0 = \frac{z \cdot (t \times Rf'_0)}{||z||^2} f'_1$$  

(11)

with $z = f'_1 \times Rf'_0$, where $\lambda_i$ equals the depth multiplied by $||f'_i||$ for $i = 0, 1$.

Note that the epipolar constraint (8) is a necessary condition for intersecting the two rays, but not a sufficient one. Fig. 2 illustrates scenarios where the two rays are coplanar, but do not intersect. This happens when the intersection requires negative depth(s), violating the \textit{cheirality} constraint [8]. In the following analysis (until Section 8), we will temporarily assume that satisfying the epipolar constraint (8) is sufficient for intersecting the rays.

5. Closed-Form $L_1$ Triangulation

The $L_1$ triangulation based on the angular reprojection error (10) finds the feature rays $f'_0$ and $f'_1$ that minimize $\theta_0 + \theta_1$ subject to the epipolar constraint (8). The following lemma reveals a surprising fact that $(\theta_0 + \theta_1)_{mn}$ is achieved by correcting either one of $f_0$ or $f_1$, but not both:

\textbf{Lemma 2 ($L_1$ Angle Minimization)}

Given two skew lines $L_0(s_0) = c_0 + s_0m_0$ and $L_1(s_1) = c_1 + s_1m_1$, consider any two intersecting lines that also pass $c_0$ and $c_1$, respectively, i.e., $L_0(s'_0) = c'_0 + s'_0m'_0$ and $L_1(s'_1) = c'_1 + s'_1m'_1$. Let $t = c_0 - c_1$, $m_0 = m_0 \times t$, $m_1 = m_1 \times t$, $\theta_0 = \angle(L_0, L_0')$ and $\theta_1 = \angle(L_1, L_1')$. Then, $(\theta_0 + \theta_1)$ is minimized for the following $m'_0$ and $m'_1$:

$$m'_0 = m_0 - (m_0 \cdot \mathbf{n}) \mathbf{n} \quad \text{and} \quad m'_1 = m_1.$$  

(12)

Otherwise,

$$m'_0 = m_0 \quad \text{and} \quad m'_1 = m_1 - (m_1 \cdot \mathbf{n}) \mathbf{n}.$$  

(13)

\textbf{Proof.} Refer to Appendix B.

By substituting $Rf_0$ and $f_1$ into $m_0$ and $m_1$ in the above lemma, the resulting $m'_0$ and $m'_1$ become the corrected rays $Rf'_0$ and $f'_1$ that satisfy the $L_1$ optimality, and $m_0$ (or $m_1$) becomes the normal of the corresponding epipolar plane.

6. Closed-Form $L_2$ Triangulation

Considering that the angular errors are small in practice, the “relaxed” $L_2$ triangulation finds the feature rays $f'_0$ and $f'_1$ that minimize $\sin^2(\theta_0) + \sin^2(\theta_1)$ (instead of $\theta_0^2 + \theta_1^2$) subject to the epipolar constraint (8). Note that the small-angle approximation by $\sin(\theta)$ is more accurate than by $\tan(\theta)$ or $1 - \cos(\theta)$ that have been used in literature [7, 9, 12, 28, 3]. This is easily seen by comparing their Maclaurin expansions. As will be shown in the following lemma (and previously in [24]), the relaxation with the sine function allows us to derive the $L_2$ optimal solution in closed form.

\textbf{Lemma 3 ($L_2$ Angle Minimization)}

Given two skew lines $L_0(s_0) = c_0 + s_0m_0$ and $L_1(s_1) = c_1 + s_1m_1$, consider any two intersecting lines that also pass $c_0$ and $c_1$, respectively, i.e., $L_0(s'_0) = c'_0 + s'_0m'_0$ and $L_1(s'_1) = c'_1 + s'_1m'_1$. Let $t = c_0 - c_1$, $\theta_0 = \angle(L_0, L_0')$ and $\theta_1 = \angle(L_1, L_1')$. Then, $(\sin^2 \theta_0 + \sin^2 \theta_1)$ is minimized for

$$m'_i = m_i - (m_i \cdot \mathbf{n}) \mathbf{n} \quad \text{for} \quad i = 0, 1,$$  

(14)

where $\mathbf{n}$ is the second column of the $3 \times 3$ matrix $V$ from
that satisfy the error constraint. In case (a), both rays have negative depths at the optimal intersection. Increasing the allowed angular reprojective error, the first intersection with positive depths occurs when the two corrected rays become parallel, resulting in a point at infinity. This point cannot be triangulated, so it should be discarded.

In the remaining cases, the optimal intersection involves only one of the rays having a negative depth. Following the same procedure, the first intersection with positive depths occurs either at infinity (case (b)), at one of the camera centers (case (c)), along the ray parallel to the translation (case (d)), or at a point somewhere else (case (e)).

In case (b), (c) and (d), the newly triangulated point has either infinite, zero or ambiguous depth, so it is reasonable to discard it. In case (e), we found that reattempting the triangulation with positive depths yields either a very large error, a point near the epipole or a low parallax angle. Typically, these are the indicators of low accuracy or an outlier [10, 8], so a reasonable choice is to discard the match. This procedure is outlined in Step 4–6 of Tab. 1.

### 8. Closed-Form $L_\infty$ Triangulation

The $L_\infty$ triangulation based on the angular reprojective error (10) finds the feature rays $f'_0$ and $f'_1$ that minimize $\max(\theta_0, \theta_1)$ subject to the epipolar constraint (8). The following lemma states that this is achieved when $\theta_0 = \theta_1$.

**Lemma 4 ($L_\infty$ Angle Minimization)**

Given two skew lines $L_0(s_0) = c_0 + s_0 m_0$ and $L_1(s_1) = c_1 + s_1 m_1$, consider any two intersecting lines that also pass $c_0$ and $c_1$, respectively, i.e., $L'_0(s'_0) = c_0 + s'_0 m'_0$ and $L'_1(s'_1) = c_1 + s'_1 m'_1$. Let $t = c_0 - c_1$, $n_a = (m_0 + m_1) \times t$, $n_b = (m_0 - m_1) \times t$, $\theta_0 = \angle(L_0, L'_0)$ and $\theta_1 = \angle(L_1, L'_1)$. Then, $\max(\theta_0, \theta_1)$ is minimized when $\theta_0 = \theta_1$. This is achieved for $m'_i = m_i - (m_i \cdot \hat{n'}) \hat{n'}$ for $i = 0, 1$, (16) where $\hat{n'} = \begin{cases} n_a & \text{if } \|n_a\| \geq \|n_b\| \\ n_b & \text{otherwise} \end{cases}$ (17)

**Proof.** Refer to Appendix D.

**Table 1.** Summary of the proposed methods.

<table>
<thead>
<tr>
<th>Input: Calib. matrix ($K$), relative pose ($R$, $t$), and a match ($u_0$, $u_1$) from two views ($C_0$, $C_1$).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Output:</strong> Triangulated 3D point ($x'_i$) in ref. frame $C_1$.</td>
</tr>
<tr>
<td><strong>1.</strong> $f_0 \leftarrow K^{-1} u_0$, $f_1 \leftarrow K^{-1} u_1$, $m_0 \leftarrow Rf_0$, $m_1 \leftarrow f_1$.</td>
</tr>
<tr>
<td><strong>2.</strong> For $L_1$ triangulation:</td>
</tr>
<tr>
<td>* If $|m_0 \times t| \leq |m_1 \times t|$, use (12) to obtain $m'_0$ and $m'_1$.</td>
</tr>
<tr>
<td>* Otherwise, use (13).</td>
</tr>
<tr>
<td><strong>3.</strong> $Rf'_0 \leftarrow m'_0$ and $f'_1 \leftarrow m'_1$.</td>
</tr>
<tr>
<td><strong>4.</strong> Check cheirality:</td>
</tr>
<tr>
<td>* Obtain $\lambda_0$ and $\lambda_1$ from (11).</td>
</tr>
<tr>
<td>* Discard the point and terminate if either $\lambda_0 \leq 0$ or $\lambda_1 \leq 0$.</td>
</tr>
<tr>
<td><strong>5.</strong> Check angular reprojective errors:</td>
</tr>
<tr>
<td>* $\theta_0 \leftarrow \angle(Rf_0, Rf'_0)$ and $\theta_1 \leftarrow \angle(f_1, f'_1)$.</td>
</tr>
<tr>
<td>* Discard the point and terminate if $\max(\theta_0, \theta_1) &gt; \epsilon_1$ for some small $\epsilon_1$.</td>
</tr>
<tr>
<td><strong>6.</strong> Check parallax:</td>
</tr>
<tr>
<td>* $\beta \leftarrow \angle(Rf'_0, f'_1)$.</td>
</tr>
<tr>
<td>* Discard the point and terminate if $\beta &lt; \epsilon_2$ for some small $\epsilon_2$.</td>
</tr>
<tr>
<td><strong>7.</strong> Compute and return $x'_i$ from (11).</td>
</tr>
</tbody>
</table>

9. Experimental Results

We evaluate the proposed methods in comparison to the midpoint method [2, 10], Hartley and Sturm’s $L_1$ and $L_2$ method [10], Lindström’s $L_2$ method with five iterations [17], and Nister’s $L_\infty$ method [23]. The evaluation was performed on both synthetic and real datasets. We generated the synthetic datasets as follows: A set of $8 \times 4$ point clouds of $2,500$ points each are generated with a Gaussian radial distribution $N(0, (d/4)^2)$ where $d$ is the distance from the world origin. Each point cloud is centered at $[0, 0, d]^T$ for $d = 2^n$ with $n = -1, 0, ..., +6$, and their image projections are perturbed by Gaussian noise $N(0, \sigma^2)$ for $\sigma = 0.5, 1, 2, 4, 8$. The size and the focal length of the images are $1,024 \times 1,024$ pixels and $512$ pixel, respectively. We have three configurations for the camera poses: (1) “orbital” - the cameras at $[\pm 0.5, 0, 0]^T$ pointing at the point cloud center, (2) “lateral” - the cameras at $[\pm 0.5, 0, 0]^T$ pointing at $[0, 0, \infty]^T$, and (3) “forward” -
the cameras at \([0, 0, \pm 0.5]^T\) pointing at the point cloud center. The poses are slightly perturbed with uniform noise \(U(0, 0.01)\). For real datasets, we used the Oxford Dinosaur, Model House and Corridor [1], Notre Dame [31] and Fountain [4, 25] dataset. In total, the synthetic and real datasets provide over 5.5 million unique triangulation problems in a wide variety of geometric configurations.

Tab. 2 provides the percentage of the total matches (from both synthetic and real datasets) for which each method yields the lowest error in given criterion. In 100 % of the total triangulation problems, all three of our methods yield the lowest errors in their corresponding optimal criterion. We also see that minimizing \(\sin^2(\theta_0) + \sin^2(\theta_1)\) is very close to minimizing \(\theta_0^2 + \theta_1^2\), as discussed in Section 6. Since our \(L_1\) angular method is numerically stable, it sometimes finds better solutions than Hartley-Sturm’s closed-form \(L_1\) method [10] even in the \(L_1\) image error criterion \((d_0 + d_1)\).

In Fig. 5, histograms are given for the 3D reconstruction errors on the synthetic datasets. It shows that 1) discarding low-parallax points (Step 6 of Tab. 1) helps to remove large 3D errors, and 2) all methods then exhibit similar 3D accuracy. Qualitatively, we also found that the reconstructions of the real datasets look similar for all methods. Fig. 3 shows the reconstruction results using the proposed \(L_1\) method.

We compare the speed of each algorithm in Tab. 3. The midpoint method is the fastest, as it directly computes the 3D point using (11) without correcting the feature rays or image points. Among the optimal methods, our \(L_1\) and \(L_\infty\) methods are significantly faster than the rest, i.e., at least 1–2 orders of magnitude faster than the state-of-the-art [17].
Table 2. Percentage of the total matches (from all synthetic and real datasets) for which each method yields the lowest error in given criterion. “img/ang”: optimal in the image/angular errors. See the supplementary material for the results from individual datasets.

<table>
<thead>
<tr>
<th>Error Criterion</th>
<th>Midpoint</th>
<th>$L_1$ img</th>
<th>$L_2$ img</th>
<th>$L_∞$ img</th>
<th>$L_1$ ang</th>
<th>$L_2$ ang</th>
<th>$L_∞$ ang</th>
</tr>
</thead>
<tbody>
<tr>
<td>$θ_0 + θ_1$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>100 %</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$θ_0^2 + θ_1^2$</td>
<td>-</td>
<td>-</td>
<td>7e-5 %</td>
<td>5e-5 %</td>
<td>-</td>
<td>-</td>
<td>99.9999 %</td>
</tr>
<tr>
<td>$\sin^2(θ_0) + \sin^2(θ_1)$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>100 %</td>
<td>-</td>
</tr>
<tr>
<td>$\max(θ_0, θ_1)$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>100 %</td>
</tr>
<tr>
<td>$d_0 + d_1$</td>
<td>-</td>
<td>70.84%</td>
<td>0.002%</td>
<td>0.002%</td>
<td>-</td>
<td>29.16 %</td>
<td>-</td>
</tr>
<tr>
<td>$d_0^2 + d_1^2$</td>
<td>-</td>
<td>-</td>
<td>23.14%</td>
<td>76.86%</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$\max(d_0, d_1)$</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>100 %</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3. Speed of computing a 3D point. The relative speed is normalized by that of the midpoint method. Note that this does not take into account Step 4–6 of Tab. 1. All algorithms were implemented in C++ and run on a laptop CPU (Intel i7-4810MQ, 2.8 GHz).

<table>
<thead>
<tr>
<th>Points/sec</th>
<th>$L_1$ img</th>
<th>$L_2$ img</th>
<th>$L_∞$ img</th>
<th>$L_2$ img</th>
<th>$L_1$ ang</th>
<th>$L_2$ ang</th>
<th>$L_∞$ ang</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[2, 10]</td>
<td>[10]</td>
<td>[10]</td>
<td>[23]</td>
<td>5 it.</td>
<td>[17]</td>
<td>5 it.</td>
</tr>
</tbody>
</table>

10. Conclusions

In this work, we derived optimal closed-form solutions to the $L_1$, $L_2$ and $L_∞$ stereo triangulation based on the angular reprojection error. The proposed triangulation methods are extremely simple and fast, and they guarantee global optimality under respective cost functions. We believe that our findings will be particularly useful for large-scale SfM and real-time visual SLAM algorithms.

Acknowledgement

This work was partially supported by the Spanish government (project PGC2018-096367-B-I00) and the Aragón regional government (Grupo DGA-T45.17R/FSE).

Appendix

A. Proof of Lemma 1

Consider a right circular cone with apex $c_0$ and axis $L_0$, lying sideways on a plane $Π$ that contains $c_0$ and line $L_1$ (see Fig. 6). The equation of the plane is given by

$$Π(x) = n_1 \cdot (x - c_0) = 0 \quad \text{with} \quad n_1 = m_1 \times t.$$  \hspace{1cm} (18)

The line of intersection between the plane and the cone forms the smallest angle to $L_0$ among all possible lines on the plane that pass $c_0$. That is, it forms the smallest angle to $L_0$ among all possible lines that pass both $c_0$ and $L_1$. Hence, this line of intersection must be $L_0'$. Now, consider a point $a_0 = c_0 + \tilde{m}_0$ located one unit away from $c_0$ along $L_0$. Let $r$ be the projection of $a_0$ onto plane $Π$. According to lemma 5 in Appendix E, the point $r$ must be located along $L_0'$. Let $d = \|a_0 - r\|$, i.e., the distance between $a_0$ and plane $Π$. Then, we obtain $\sin(θ_a)$ as follows:

$$\sin(θ_a) = d \overset{(*)}{=} |\tilde{m}_1 \cdot (a_0 - c_0)| = |\tilde{m}_1 \cdot \tilde{m}_0|.$$  \hspace{1cm} (19)

B. Proof of Lemma 2

One of the following is true when $(θ_0 + θ_1)$ is minimized:

1. $L_0' \neq L_0$ and $L_0' = L_1 \iff θ_0 > 0$ and $θ_1 = 0$.
2. $L_0' = L_0$ and $L_0' \neq L_1 \iff θ_0 = 0$ and $θ_1 > 0$.
3. $L_0' \neq L_0$ and $L_0' \neq L_1 \iff θ_0 > 0$ and $θ_1 > 0$.

Suppose, for the sake of argument, that one of the first two statements is true. In the first case, lemma 1 states that $\tilde{m}_0'$ is obtained by projecting $m_0$ onto the plane with the normal $m_1 \times t$, which leads to (12) and

$$\sin(θ_a) = \frac{|\tilde{m}_0 \cdot (m_1 \times t)|}{\|m_1 \times t\|} = \frac{|\tilde{m}_0 \cdot (\tilde{m}_1 \times t)|}{\|\tilde{m}_1 \times t\|}.$$  \hspace{1cm} (19)

Likewise, in the second case, lemma 1 leads to (13), and

$$\sin(θ_1) = \frac{|\tilde{m}_1 \cdot (m_0 \times t)|}{\|m_0 \times t\|} \overset{(*)}{=} \frac{|\tilde{m}_0 \cdot (\tilde{m}_1 \times t)|}{\|\tilde{m}_0 \times t\|}. \hspace{1cm} (20)$$
Now, the question is how to determine which of the two statements is true. Comparing the right-hand side of (19) and (20), we find that if \( \| \mathbf{\hat{m}}_0 \times \mathbf{t} \| \leq \| \mathbf{\hat{m}}_1 \times \mathbf{t} \| \), then

\[
\min_{\theta_0 | \theta_1 = 0} \theta_0 \leq \min_{\theta_1 | \theta_0 = 0} \theta_1, \tag{21}
\]

and \( \min(\theta_0 + \theta_1) \) is equal to the left-hand side of (21), indicating that the first statement is true. Naturally, the second statement is true otherwise. Note that there is an ambiguity if \( \| \mathbf{\hat{m}}_0 \times \mathbf{t} \| = \| \mathbf{\hat{m}}_1 \times \mathbf{t} \| \), and the solution is optimal whichever case is considered. This concludes the proof of lemma 2 for the first two cases.

We will now prove that the third case never occurs. Given some angle \( \theta_1 \), minimizing \( (\theta_0 + \theta_1) \) is equivalent to minimizing \( \theta_0 \). This is the identical situation as the first case if we replace \( \mathbf{L}_0 \) by \( \mathbf{L}'_0 \). We know from the proof of lemma 1 that pivoting a line to intersect another with minimum angle can be modeled by a cone lying sideways on a plane. The top of Fig. 7 illustrates this. Similarly, the bottom of Fig. 7 illustrates the minimization of \( \theta_1 \) with respect to \( \mathbf{L}'_0 \) given \( \theta_0 \). Now, since both planes touching each cone contain the same two intersecting lines \( \mathbf{L}'_0 \) and \( \mathbf{L}'_1 \), they must be the same plane. Let this plane be \( \Pi' \). According to lemma 1, \( \mathbf{L}'_0 \) is the projection of \( \mathbf{L}_0 \) onto plane \( \Pi' \).

Therefore,

\[
\mathbf{\hat{m}}'_0 = \frac{\mathbf{m}_0 - (\mathbf{\hat{m}}_0 \cdot \mathbf{n}') \mathbf{n}'}{\| \mathbf{m}_0 - (\mathbf{\hat{m}}_0 \cdot \mathbf{n}') \mathbf{n}' \|}, \tag{22}
\]

where \( \mathbf{n}' \) is the unit normal of plane \( \Pi' \). Since \( \Pi' \) contains both \( \mathbf{c}_0 \) and \( \mathbf{c}_1 \), \( \mathbf{n}' \) is perpendicular to \( \mathbf{t} = \mathbf{c}_0 - \mathbf{c}_1 \). Hence, computing the dot product with \( \mathbf{t} \) on each side of (22) yields

\[
\mathbf{t} \cdot \mathbf{\hat{m}}'_0 = \frac{\mathbf{t} \cdot \mathbf{m}_0}{\| \mathbf{m}_0 - (\mathbf{\hat{m}}_0 \cdot \mathbf{n}') \mathbf{n}' \|}. \tag{23}
\]

Note that \( \| \mathbf{m}_0 - (\mathbf{\hat{m}}_0 \cdot \mathbf{n}') \mathbf{n}' \| \) corresponds to the magnitude of the projection of \( \mathbf{m}_0 \) onto plane \( \Pi' \) for non-zero \( \theta_0 \), so it must be smaller than \( \| \mathbf{m}_0 \| = 1 \). Thus

\[
| \mathbf{t} \cdot \mathbf{\hat{m}}'_0 | = \frac{| \mathbf{t} \cdot \mathbf{m}_0 |}{\| \mathbf{m}_0 - (\mathbf{\hat{m}}_0 \cdot \mathbf{n}') \mathbf{n}' \|} > | \mathbf{t} \cdot \mathbf{m}_0 |. \tag{24}
\]

Using (2), this inequality can be written as

\[
| \mathbf{t} \times \mathbf{\hat{m}}'_0 | < | \mathbf{t} \times \mathbf{\hat{m}}_0 |. \tag{25}
\]

Analogously, we can also derive

\[
| \mathbf{t} \times \mathbf{\hat{m}}'_1 | < | \mathbf{t} \times \mathbf{\hat{m}}_1 |. \tag{26}
\]

Now, suppose that

\[
\min_{\theta_0, \theta_1} (\theta_0 + \theta_1) = \theta_0^* + \theta_1^* \quad \text{with} \quad \theta_0^*, \theta_1^* > 0. \tag{27}
\]

Without loss of generality, let us assume that \( | \mathbf{t} \times \mathbf{\hat{m}}'_0 | \leq | \mathbf{t} \times \mathbf{\hat{m}}_1 | \). Then, (25) gives \( | \mathbf{t} \times \mathbf{\hat{m}}'_0 | < | \mathbf{t} \times \mathbf{\hat{m}}_1 | \). As we discussed for the first two cases, this means that pivoting \( \mathbf{L}'_0 \) to intersect \( \mathbf{L}_1 \) takes smaller angle than pivoting \( \mathbf{L}_1 \) to intersect \( \mathbf{L}'_0 \), i.e., \( \theta_0^* < \theta_1^* \). Thus

\[
\theta_0^* + \theta_1^* < \theta_0^* + \theta_1^*. \tag{28}
\]

According to lemma 6 in Appendix E, pivoting a line twice for intersection takes equal or greater angle than the single minimum pivot angle. Therefore,

\[
\min_{\theta_0 | \theta_1 = 0} \theta_0 \leq \theta_0^* + \theta_1^* < \theta_0^* + \theta_1^*, \tag{29}
\]

which contradicts (27). Therefore, \((\theta_0 + \theta_1)\) is minimized when either \(\theta_0\) or \(\theta_1\) is zero.

\[\blacksquare\]

**C. Proof of Lemma 3**

Given some angle \( \theta_1 \), \( (\sin^2(\theta_0) + \sin^2(\theta_1))_{\text{min}} \) is achieved by minimizing \( \theta_0 \) and vice versa. As discussed in the proof of lemma 2, this means that the underlying geometry at \( (\sin^2(\theta_0) + \sin^2(\theta_1))_{\text{min}} \) can be represented by the two cones with apex \( \mathbf{c}_0, \mathbf{c}_1 \) and skew axes \( \mathbf{L}_0, \mathbf{L}_1 \), respectively, touching each side of the same plane on their lateral surface. This is visualized in Fig. 7. Let \( \mathbf{n}' \) be the normal of plane \( \Pi' \). From Lemma 1, we know that

\[
\sin(\theta_0) = |\mathbf{n}' \cdot \mathbf{\hat{m}}_0| \quad \text{and} \quad \sin(\theta_1) = |\mathbf{n}' \cdot \mathbf{\hat{m}}_1|. \tag{30}
\]

Combining these two equations, we get

\[
\sin^2(\theta_0) + \sin^2(\theta_1) = \| \mathbf{M}^T \mathbf{n}' \|^2 \quad \text{with} \quad \mathbf{M} = [\mathbf{\hat{m}}_0 \mathbf{\hat{m}}_1]. \tag{31}
\]

Since plane \( \Pi' \) contains both \( \mathbf{c}_0 \) and \( \mathbf{c}_1 \), \( \mathbf{n}' \) is perpendicular to \( \mathbf{t} = \mathbf{c}_0 - \mathbf{c}_1 \). Therefore, minimizing \( (\sin^2(\theta_0) + \sin^2(\theta_1)) \) is equivalent to solving the following equality-constrained quadratic programming problem:

\[
\arg\min_{\mathbf{n}'} \| \mathbf{M}^T \mathbf{n}' \|^2, \text{ s.t. } \| \mathbf{n}' \| = 1 \text{ and } \mathbf{t} \cdot \mathbf{n}' = 0. \tag{32}
\]

In [5], it was shown that this problem can be solved using the method of Lagrange multipliers, and \( \| \mathbf{M}^T \mathbf{n}' \|^2 \) is
minimized when \( \hat{n}' \) is the eigenvector corresponding to the smallest nontrivial eigenvalue of \( A = (I - t \hat{t} \hat{t}^\top)MM^\top \).

Letting \( P = (I - t \hat{t} \hat{t}^\top) \), it can be easily shown that \( P = P^\top = P^\top P = P P^\top \). Hence, \( A = P M M^\top = P P^\top M M^\top \).

Note that for any square matrix \( X \) and \( Y \), the eigen-decomposition of \( X Y \) is the same as that of \( Y X \).

This means that the eigenvectors of \( A = P (P^\top M M^\top) \) are the same as those of \( (P^\top M M^\top) P = (M^\top P)^\top (M^\top P) \), i.e., the right-singular vectors of \( M^\top P \).

Therefore, letting \( U S V^\top = S V D (M^\top P) \) with the diagonal entries of \( S \) in descending order, the optimal \( \hat{n}' \) is given by the second column of \( V \). Finally, projecting \( m_0 \) and \( m_1 \) onto plane \( \Pi' \) leads to \( (14) \).

D. Proof of Lemma 4

First, we show that \( \theta_0 = \theta_1 \) when \( \max(\theta_0, \theta_1) \) is minimized:

Consider two cones with apex \( c_0, c_1 \) and skew axes \( L_0, L_1 \). Constrain both their apertures to be \( 2\theta \). When \( \theta = 0 \), they are simply two skew lines. As we gradually increase \( \theta \), they will grow at the same rate, and eventually, touch one another. Let \( \theta = \theta' \) at this point. Now, suppose

\[
\theta^* := \min_{\theta_0, \theta_1} \max(\theta_0, \theta_1) < \theta'.
\]

(33)

The definition of \( \theta^* \) implies that setting \( \theta_0 = \theta_1 = \theta^* \) will make the two cones partially overlap in space (or at least meet at a point). However, the they do not meet when \( \theta_0 = \theta_1 < \theta' \). This is a contradiction, so the inequality in \( (33) \) must be false, and \( \theta^* \) must be equal to \( \theta' \). That is, \( \theta_0 = \theta_1 = \theta' \) in order for \( \max(\theta_0, \theta_1) \) to be minimized.

We can now represent the underlying geometry at \( (\max(\theta_0, \theta_1))_{\min} \) as two congruent cones with skew axes, touching each side of the same plane \( \Pi' \) on their lateral surface. This is the situation shown in Fig. 7 for \( \theta_0 = \theta_1 \). Let \( \hat{n}' \) be the normal of plane \( \Pi' \). Then, from lemma 1, we get

\[
\sin(\theta_0) = \sin(\theta_1) = |\hat{n}' \cdot m_0| = |\hat{n}' \cdot \hat{m}_0|.
\]

(34)

The last equality in \( (34) \) can be written as

\[
(\hat{m}_0 + w\hat{m}_1) \cdot \hat{n}' = 0,
\]

(35)

where \( w \) is \(-1 \) or \( 1 \), depending on the signs of \( \hat{n}' \cdot \hat{m}_0 \) and \( \hat{n}' \cdot \hat{m}_1 \). On the other hand, since plane \( \Pi' \) contains both \( c_0 \) and \( c_1 \), \( \hat{n}' \) is perpendicular to \( t = c_0 - c_1 \):

\[
t \cdot \hat{n}' = 0.
\]

(36)

Combining \( (35) \) and \( (36) \), \( \hat{n}' \) can be expressed as

\[
\hat{n}' = \lambda(\hat{m}_0 + w\hat{m}_1) \times t
\]

(37)

where \( \lambda \) is the normalizing factor. Evaluating \( (37) \) at \( w = 1 \) and \( w = -1 \) gives two candidates for optimal \( \hat{n}' \). The optimal solution is then determined by comparing the values of \( (34) \) with each candidate \( \hat{n}' \), which amounts to choosing the solution with smaller \( \lambda \). This procedure corresponds to \( (17) \). Finally, projecting \( m_0 \) and \( m_1 \) onto plane \( \Pi' \) with optimal \( \hat{n}' \) leads to \( (16) \).

E. Other Geometric Lemmas

Lemma 5 (Cone-On-Plane Perpendicularity)

When a plane is tangent to a right circular cone, the line of intersection is the projection of cone’s axis onto the plane.

Proof.

Consider a cone with axis \( L_0 \) and plane \( \Pi \) tangent to this cone. They are both symmetric with respect to the plane that contains \( L_0 \) and the normal of \( \Pi \). Let this plane of symmetry be \( \Pi_{sym} \). For any circular cross-section of the cone, there is a single point touching \( \Pi \). Therefore, this point must lie on \( \Pi_{sym} \), and so must the line of intersection, \( L_0 \). It follows that \( \Pi_{sym} \) contains \( L_0 \) and \( L_0 \). Since \( \Pi_{sym} \) is perpendicular to \( \Pi \), \( L_0 \) is a projection of \( L_0 \) onto \( \Pi \).

Lemma 6 (Single vs Multi-Pivot for Intersection)

Given two skew lines \( L_0(s_0) = c_0 + s_0 m_0 \) and \( L_1(s_1) = c_1 + s_1 m_1 \), let \( L'_0 \) be the line that forms the smallest angle \( \theta_0 \in [0, \pi/2] \) to \( L_0 \) among all possible lines that intersect both point \( c_0 \) and line \( L_1 \). For any positive integer \( N \), consider the following arbitrary lines passing \( c_0 \) such that

\[
L_i(s_i) = \begin{cases} L_0 & \text{for } i = 0 \\ c_0 + s_i m_i^* & \text{for } i = 1, 2, \ldots, N \end{cases}
\]

where only \( L_N \) intersects \( L_1 \). Then,

\[
\theta_0 \leq \sum_{i=1}^{N} \angle(L_i^*, L_{i-1}^*).
\]

(38)

Proof.

The right-hand side of \( (38) \) corresponds to the sum of \( N \) pivot angles that make line \( L_0 \) to intersect \( L_1 \). Fig. 8a depicts such operation for \( N = 2 \). Let \( \phi = \angle (L_0^*, L_1^*) \), \( \psi = \angle (L_1^*, L_2^*) \) and \( \tau = \angle (L_0^*, L_2^*) \). Now, consider three arbitrary points \( p, q_1 \) and \( q_2 \) on \( L_0^* \), \( L_1^* \) and \( L_2^* \), respectively. A tetrahedron formed by these three points and \( c_0 \) are shown in Fig. 8b. At a vertex of a tetrahedron, the three edges form three angles such that the sum of any two angles is greater than the third one [18, 15]. Thus, \( \tau \leq \phi + \psi \). Since \( \theta_0 \) is the minimum pivot angle for intersection, we have \( \theta_0 \leq \tau \leq \phi + \psi \), which proves \( (38) \) for \( N = 2 \). Now, for \( N > 2 \), we know that replacing the last two pivots by the corresponding single minimum pivot will produce \( N - 1 \) pivots that take equal or smaller angle. Repeating this process until \( N = 1 \) proves \( (38) \) for any \( N > 2 \).
References