Noname manuscript No.
(will be inserted by the editor)

Verifying Global Minima for L_2 Minimization Problems in Multiple View Geometry

Richard Hartley, Fredrik Kahl, Carl Olsson, and Yongduek Seo[3]

the date of receipt and acceptance should be inserted later

Abstract We consider the least-squares (L2) minimization problems in multiple view geometry for triangulation, homography, camera resectioning and structure-and-motion with known rotatation, or known plane. Although optimal algorithms have been given for these problems under an L-infinity cost function, finding optimal least-squares solutions to these problems is difficult, since the cost functions are not convex, and in the worst case may have multiple minima. Iterative methods can be used to find a good solution, but this may be a local minimum. This paper provides a method for verifying whether a local-minimum solution is globally optimal, by providing a simple and rapid test involving the Hessian of the cost function. The basic idea is that by showing that the cost function is convex in a restricted but large enough neighbourhood, a sufficient condition for global optimality is obtained.

The method is tested on numerous problem instances of real data sets. In the vast majority of cases we are able to verify that the solutions are optimal, in particular, for small to medium-scale problems.

1 Introduction

Modern methods for the solution of geometric reconstruction problems in Computer Vision may be viewed as starting with the paper of Longuet-Higgins [9] in 1981. However, after nearly three decades of research, the problem of finding guaranteed optimal least-squares (L_2) solutions to such problems is not solved. In fact, papers such as [11,12] suggest that such algorithms do not exist.

Even for the simplest of geometric vision problems, the *triangulation problem*, no ideal method has been given that guarantees an optimal least-squares solution. This paper, however, tries a totally different approach, by giving a procedure for verifying whether a solution obtained through standard methods such as bundle-adjustment actually is the global optimum. Usually, in fact, it is. Though not useful for all geometric Vision problems, the method given in this paper is applicable to a substantial number of the common reconstruction problems, as will be shown theoretically and by experiment in this paper.

Perhaps the simplest of all 3D reconstruction problems is the triangulation problem, which nevertheless shares many of the difficulties of more complex tasks. We begin our description of our approach by considering the triangulation problem, before generalizing later to a wider class of problems. The condition we provide for the verification is a sufficient but not necessary condition for the solution to be a global optimum. However, for triangulation it works in almost all cases. In the rare cases where the condition fails it is usually because the point has large noise, in which case in a large-scale reconstruction problem, the best option is just to remove the point from consideration. Alternatively it may be possible to apply one of the recent (considerably more time-consuming) optimal algorithms ([10,6]) for solving the problem; see also the survey [1].

How hard is the triangulation problem, really? It was shown in [3] that the least-squares two-view triangulation problem is solvable in closed form. However, up to three local minima may exist. Much more recently, it has been shown ([16]) that the solution for three views involves the solution of a polynomial of degree 47, and higher degree polynomials are required with more views. The degree of the polynomial involved in the solution translates into numbers of possible local minima. It is certainly possible to find explicit examples with multiple local minima [3]. This suggests that the problem is difficult.

[3]Department of Media Technology, Sogang University, Seoul, Korea

On the other hand, it was shown in [2] that a single local (and hence global) minimum occurs if an L_∞ (minimax) solution is sought, instead of the least-squares solution. However, the least-squares problem remains the problem of primary interest. The L_∞ solution is useful as an initialization procedure for the least squares, but it does not guarantee an optimal L_2 solution. In a real data test it seems that common algorithms get the right answer most of the time. So, perhaps the problem is not so hard after all – if only one knows whether one has the right solution. That problem is effectively solved in this paper.

Working algorithms with practical accuracy have been reported for geometric vision problems since the eight-point algorithm for two view reconstruction [5]. Although these known methods do not guarantee a globally optimal solution, nevertheless, simple methods based on initialization, followed by iterative refinement usually work very well. They depend on the initial solution being within the basin of attraction of the optimal solution. However, until now there has been no way of verifying this requirement. In this paper, we address this problem and give a fast and very satisfactory method of verifying that a given local minimum is the global minimum. By experiments on a very large set of triangulation problems, the test is seen to have a 99.9% success rate and averages 0.5ms per triangulated point on a modern 2.66 GHz laptop computer, substantially faster for small problems.

The new technique for proving convexity and thus the global optimality is applicable to not only triangulation but also other geometric vision problems. We show that the same analysis for the triangulation problem applies to any optimization problem with quasi-convex residual functions [8,7], since the location of the global minimum can be constrained to a convex set.

The class of problems that our framework covers is the same as the quasiconvex problems in the L_∞-norm [8,7], which includes n-view triangulation, homography estimation, camera resectioning to name a few.

Our main contributions may be listed as follows:

- An easy and fast method to verify the global optimality of the solution under the L_2 norm for a whole class of multiview geometry problems.
- We perform an extensive set of experiments on several data sets to explore the limitations of the method. Results are given for triangulation, homography estimation, camera resectioning and structure and motion with known rotations. Our main conclusion is that in most cases the basic verification test succeeds, especially for small scale problems.

2 Triangulation – Simple Convexity Test

We will derive conditions for optimality of the solutions for a class of cost functions encountered in the solution of geometric vision problems. These are the cost functions composed of sum of squares of measurement errors, where each error term is a quasi-convex cost function of Second Order Cone Programming (SOCP) type [7]. This type of cost function is encountered in many different problems, many of which are listed in the survey [1]. However, to allow a simple introduction to the techniques introduced in this paper, we will first apply them to the triangulation problem, which has been shown to be representative of the more general problems considered later.

2.1 Problem Description

We consider a calibrated camera model and think of image points as being represented by points on a sphere, that is unit vectors, rather than points in an image plane. The triangulation problem is characterized by a set of unit measurement vectors $\mathbf{w}_i$ based at projection centres $\mathbf{C}_i$, all pointing to a common point $\mathbf{X}$ in space, which is to be determined. Thus, nominally $\mathbf{X} = \mathbf{C}_i + d_i\mathbf{w}_i$, where d_i is the depth of the point $\mathbf{X}$ with respect to the i-th projection.

With error in the measurements, the rays from centre $\mathbf{C}_i$ in direction d_i do not exactly intersect, so the required point $\mathbf{X}$ is the one that minimizes the sum of squares error. The image error is normally the angular difference between a measured direction $\mathbf{w}$ and the direction vector from the camera centre to the point $\mathbf{X}$. For simplification, our cost function will be the sum of squares of the *tangent* of the angular error, rather than the squared angle itself. Since errors are usually quite small, the difference is insignificant, and the analysis is simpler. Eventually, we will extend the result to minimization of squared angle error (see the appendix).

Considered more generally, the desired solution is a point $\mathbf{X}_{\rm opt}$ that represents a global minimum of the cost function $C(\mathbf{X}) = \sum_{i=1}^{N} f_i^2(\mathbf{X})$, where f_i^2 represents a squared residual error measurement in the i-th of N images. If we use the tangent of the angle to measure the residual error, then the total cost function is given by

$$\begin{aligned} C(\mathbf{X}) &= \sum_{i=1}^{N} f_i^2(\mathbf{X}) \\ &= \sum_{i=1}^{N} \tan^2 \angle(\mathbf{w}_i, \mathbf{X} - \mathbf{C}_i) \\ &= \sum_{i=1}^{N} \frac{\|\mathbf{w}_i \times (\mathbf{X} - \mathbf{C}_i)\|^2}{\left(\mathbf{w}_i{}^\top (\mathbf{X} - \mathbf{C}_i)\right)^2} \,. \end{aligned} \tag{1}$$

Bounding the solution. Now, suppose that $\mathbf{X}_0$ is a proposed solution to this triangulation problem with residual given by $C(\mathbf{X}_0) = \sum_{i=1}^{N} f_i^2(\mathbf{X}_0) = \epsilon^2$. This observation gives a constraint on the position of the optimal solution, which must satisfy

$$C(\mathbf{X}_{\rm opt}) = \sum_{i=1}^{N} f_i^2(\mathbf{X}_{\rm opt}) \leq \epsilon^2 \ . \tag{2}$$

Since the sum of terms $f_i^2(\mathbf{X}_{\rm opt})$ must be less than ϵ^2, so must each individual term. Thus, for all $i,\ f_i^2(\mathbf{X}_{\rm opt}) \leq \epsilon^2$. The set of points $\mathbf{X}$ in $\mathbb{R}^3$ satisfying this condition for some i consists of those points that map into a circular neighbourhood of the image point. This constitutes a cone in $\mathbb{R}^3$, as observed in [2]. Since this condition must be satisfied by each of the projections, it follows that $\mathbf{X}_{\rm opt}$ must lie in the intersection of all the cones. This is a convex region of $\mathbb{R}^3$, since each cone is a convex set.

Let us define the convex set $\mathcal{D}$ of points for the optimal solution as

$$\mathcal{D} = \{\, \mathbf{X} \,|\, f_i^2(\mathbf{X}) \leq \epsilon^2;\ {\mathbf{w}_i}^\top(\mathbf{X} - \mathbf{C}_i) \geq 0, \quad i = 1, ..., N \,\} \ , \tag{3}$$

where the second condition ${\mathbf{w}_i}^\top(\mathbf{X} - \mathbf{C}_i) \geq 0$ restricts the domain to points in front of the camera.

Our overall strategy of finding $\mathbf{X}_{\rm opt}$ (or veryfying $\mathbf{X}_0 = \mathbf{X}_{\rm opt}$) is to provide tests that allow us to prove that the cost function must be convex on this convex region. If this is so, then finding the optimal solution $\mathbf{X}_{\rm opt}$ may be carried out using standard convex optimization methods. More significantly, if $\mathbf{X}_0$ already is a local minimum of the cost function, found by any geometric optimization technique, then it must be a global minimum.

The Hessian of the cost function. A sufficiently differentiable function is convex on a convex region if and only if its Hessian is positive semi-definite. Hence, we are led in this section to consider the Hessian of the cost function. The total cost function is a sum of several different terms, one for each measurement. We will first consider a single term in the cost function, corresponding to a single image measurement.

To begin with, we consider the Hessian expressed in a suitable camera-centred coordinate frame. Consider a vector $\mathbf{w}$ pointing from the origin in the direction of the Z-axis, and let $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z})$ be a point lying close to the positive Z axis, such that the vector from the origin to $\mathbf{X}$ makes an angle ϕ from the vector $\mathbf{w}$. Consider the *error function* given by

$$f^2(\mathrm{X}, \mathrm{Y}, \mathrm{Z}) = \tan^2\phi = (\mathrm{X}^2 + \mathrm{Y}^2)/\mathrm{Z}^2 \ . \tag{4}$$

This represents the squared projection error of the point $\mathbf{X}$ with respect to the "measured direction", $\mathbf{w}$.

The Hessian matrix of f^2 with respect to X, Y and Z is easily computed to be

$$\mathtt{H} = \frac{2}{\mathrm{Z}^2} \begin{bmatrix} 1 & 0 & -2\mathrm{X}/\mathrm{Z} \\ 0 & 1 & -2\mathrm{Y}/\mathrm{Z} \\ -2\mathrm{X}/\mathrm{Z} & -2\mathrm{Y}/\mathrm{Z} & 3(\mathrm{X}^2+\mathrm{Y}^2)/\mathrm{Z}^2 \end{bmatrix}, \text{ or} \tag{5}$$

$$\mathtt{H} = \frac{2}{d^2} \begin{bmatrix} 1 & 0 & -2\tau_{\mathrm{X}} \\ 0 & 1 & -2\tau_{\mathrm{Y}} \\ -2\tau_{\mathrm{X}} & -2\tau_{\mathrm{Y}} & 3\tau^2 \end{bmatrix}, \tag{6}$$

where we introduce the notation τ_{X}, τ_{Y}, τ and d defined by the terms in corresponding position. Note that $\tau^2 = \tau_{\mathrm{X}}^2 + \tau_{\mathrm{Y}}^2$, and d represents the depth of the point in the viewing direction. Neglecting the constant factor $2/d^2$, the eigenvalues of this matrix are easily computed to be 1 and $a \pm \sqrt{a^2 + \tau^2}$, where $a = (3\tau^2 + 1)/2$. There are two positive and one negative eigenvalue.

A Bound on the Hessian. As the point $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z})$ moves over the region of interest $\mathcal{D}$, the Hessian changes. It is our purpose to find a lower bound for the Hessian matrix for all points $\mathbf{X}$ in $\mathcal{D}$.

Consider the matrix given by

$$\mathtt{H}' = \frac{2}{d^2} \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 1/3 & 0 \\ 0 & 0 & -3\tau^2 \end{bmatrix} \ . \tag{7}$$

The reasoning that led to the choice of this matrix $\mathtt{H}'$ will be explained later, but its properties are simple enough. The key property of this matrix is the observation that

$$\mathtt{H} - \mathtt{H}' = \frac{4}{3d^2} \begin{bmatrix} 1 & 0 & -3\tau_{\mathrm{X}} \\ 0 & 1 & -3\tau_{\mathrm{Y}} \\ -3\tau_{\mathrm{X}} & -3\tau_{\mathrm{Y}} & 9\tau^2 \end{bmatrix}$$

is positive semi-definite. Indeed, it is easily seen that $\mathtt{H} - \mathtt{H}'$ has eigenvalues $(4/3d^2)$ times the following:

$$\begin{aligned} 1 \quad & \text{with eigenvector } (\tau_{\mathrm{Y}}, -\tau_{\mathrm{X}}, 0)^\top \\ 0 \quad & \text{with eigenvector } (3\tau_{\mathrm{X}}, 3\tau_{\mathrm{Y}}, 1)^\top \\ 1 + 9\tau^2 \quad & \text{with eigenvector } (\tau_{\mathrm{X}}, \tau_{\mathrm{Y}}, -3\tau^2)^\top \ . \end{aligned}$$

All eigenvalues are non-negative. We write $\mathtt{H} - \mathtt{H}' \succeq \mathtt{0}$, or $\mathtt{H} \succeq \mathtt{H}'$. Matrix $\mathtt{H}'$ is a lower bound for the Hessian.

The eigenvalues of $\mathtt{H}'$ are obviously the diagonal entries. We see that $\mathtt{H}'$ has two positive and one negative eigenvalue. The eigenvector corresponding to the negative eigenvalue is directed along the Z axis, and the other two eigenvectors are in the plane perpendicular to the Z axis. However, since the two corresponding eigenvalues are equal, the eigenvectors may be taken as any two orthogonal vectors in the XY plane.

The matrix $\mathtt{H}'$ is more easily handled than H, since it is dependent on the point $\mathbf{X}$ only through the values of d and

τ, both of which can be bounded, as we shall see. We may write $\mathtt{H}'$ as

$$\mathtt{H}' = (2/3d^2)\,\mathrm{diag}(1, 1, -9\tau^2) \ .$$

We assume that the point $\mathbf{X}$ lies in a cone with angle $\arctan\tau_{\max}$. Thus, $\tau \leq \tau_{\max}$, and we see that

$$\begin{aligned}\mathtt{H} \succeq \mathtt{H}' &= (2/3d^2)\,\mathrm{diag}(1, 1, -9\tau^2) \\ &\succeq (2/3d^2)\,\mathrm{diag}(1, 1, -9\tau_{\max}^2) \ .\end{aligned}$$

Note that this matrix is a lower bound (in the semi-definite partial ordering) for the Hessian of f at any point $\mathbf{X}$ lying in the cone. It depends only on the depth d of the point from the vertex of the cone.

Generally placed cone. The above computation was carried out for the case where the vector $\mathbf{w}$ points directly along the $\mathtt{Z}$ axis. We now consider a cone with axis represented by an arbitrary unit vector $\mathbf{w}$ and with vertex at a point $\mathbf{C}$. This corresponds to rotating the camera by $\mathtt{R}$ to align the $\mathbf{Z}$ direction to $\mathbf{w}$, followed by a translation of the origin. The depth $d(\mathbf{X})$ of a point $\mathbf{X}$ relative to the vertex is given by $\mathbf{w}^\top(\mathbf{X} - \mathbf{C})$. The Hessian for the function (4) can be transformed easily to this rotated coordinate frame, according to

$$\mathtt{H} \mapsto \mathtt{R}\mathtt{H}\mathtt{R}^\top \succeq \frac{2}{3d(\mathbf{X})^2}\left(\mathbf{u}\mathbf{u}^\top + \mathbf{v}\mathbf{v}^\top - 9\tau_{\max}^2\mathbf{w}\mathbf{w}^\top\right), \quad (8)$$

where $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ are the three columns of the rotation matrix. This gives a lower bound for the Hessian, depending only on the point $\mathbf{X}$ through its depth $d(\mathbf{X})$ from the vertex.

Bounded depths. We may remove this final dependency of the Hessian on the point $\mathbf{X}$ if we assume that the point $\mathbf{X}$ lies in a bounded domain. Thus, suppose the depth $d(\mathbf{X})$ is bounded on the feasible domain $\mathcal{D}$, and that we can find bounds L and U such that for all $\mathbf{X} \in \mathcal{D}$,

$$L \leq 1/d(\mathbf{X}) \leq U \ . \quad (9)$$

In this case, one can find a bound on the Hessian that holds over the complete domain $\mathcal{D}$, and yet is independent of $\mathbf{X}$, namely (8) and (9) together give

$$\mathtt{H} \succeq (2/3)\left(L^2(\mathbf{u}\mathbf{u}^\top + \mathbf{v}\mathbf{v}^\top) - 9\,U^2\,\tau_{\max}^2\mathbf{w}\mathbf{w}^\top\right) \ , \quad (10)$$

where the inequality holds for the Hessian of f^2 computed at any point in $\mathcal{D}$. Thus, the matrix on the right may be thought of as a lower bound for the Hessian evaluated in this region.

Summing the Hessians. Now, we consider a point $\mathbf{X}$, subject to several measurements, represented by vectors $\mathbf{w}_i$. We do not care where the vertex of the cone (corresponding to the camera centre) is located, but only that the depth of the point $\mathbf{X}$ in the i-th cone is d_i. We suppose that the point $\mathbf{X}$ is situated in the intersection of cones with angle $\arctan\tau_{\max}$. Let $f_i^2(\mathbf{X})$ be τ_i^2 where $\arctan\tau_i$ is the angle of $\mathbf{X}$ from the axis of the i-th cone. The L_2 error associated with the point $\mathbf{X}$ is given by $f^2(\mathbf{X}) = \sum_i f_i^2(\mathbf{X})$ and the Hessian of f^2 is $\mathtt{H} = \sum_i \mathtt{H}_i$. Now applying the inequality (8) to each $\mathtt{H}_i$, we get

$$\mathtt{H} \succeq \sum_i (2/3d_i(\mathbf{X})^2)\left(\mathbf{u}_i\mathbf{u}_i{}^\top + \mathbf{v}_i\mathbf{v}_i{}^\top - 9\tau_{\max}^2\mathbf{w}_i\mathbf{w}_i{}^\top\right) \ . \quad (11)$$

Although each of the individual summands $\mathtt{H}_i$ in this expression cannot be positive semi-definite, we hope that by adding up the contributions for different i, the positive eigenvalues will dominate the negative eigenvalues, resulting in a positive definite or semi-definite matrix.

Summarizing the previous discussion, we may state our basic convexity theorem for the 3D triangulation problem.

Theorem 1 *Let the convex domain $\mathcal{D}$ be contained in the intersection of a set of N cones with vertices $\mathbf{C}_i$ and with axes represented by unit vectors $\mathbf{w}_i$, and angle bounded by $\arctan\tau_{\max}$. Let $\mathbf{u}_i$ and $\mathbf{v}_i$ be unit vectors orthogonal to each other and to $\mathbf{w}_i$. For a point $\mathbf{X} \in \mathcal{D}$, let $d_i(\mathbf{X}) = \mathbf{w}_i{}^\top(\mathbf{X} - \mathbf{C}_i)$ represent its depth from the vertex of the i-th cone in the direction $\mathbf{w}_i$. Let U_i and L_i be upper and lower bounds for the value of $1/d_i(\mathbf{X})$ on $\mathcal{D}$. Then*

$$\sum_{i=1}^{N}\left(L_i^2(\mathbf{u}_i\mathbf{u}_i{}^\top + \mathbf{v}_i\mathbf{v}_i{}^\top) - 9U_i^2\,\tau_{\max}^2\mathbf{w}_i\mathbf{w}_i{}^\top\right) \succeq \mathtt{0} \quad (12)$$

is a sufficient condition for the least-squares error function (1) to be convex on $\mathcal{D}$. □

3 Weighted Depths

It is possible to obtain a better condition than that given in Theorem 1, as will be shown next. Observe that the summand in (12) consists of two parts, which are multiplied by L_i^2 and U_i^2 respectively. Whereas $\mathbf{u}_i\mathbf{u}_i{}^\top + \mathbf{v}_i\mathbf{v}_i{}^\top$ is positive semi-definite, $-9\tau_{\max}^2\mathbf{w}_i\mathbf{w}_i{}^\top$ is negative semi-definite. In order for the total sum to be positive semi-definite, it is advantageous if the factor U_i^2 is kept as small as possible. Though U_i can be no smaller than L_i, it is best if the bounds L_i and U_i are as close to each other (tight) as possible.

Looked at another way, the condition (12) resulted from obtaining a lower bound for the Hessian over the whole region $\mathcal{D}$. Since the Hessian of each individual term f_i^2 scales as $1/d^2$ along any ray through $\mathbf{C}_i$, any constant lower bound

over the region $\mathcal{D}$ will not be a good bound if the depth of the region varies greatly, so that L_i and U_i are widely different. An alternative is to introduce some positive function $\alpha(\mathbf{X})$ and find a lower bound for the scaled Hessian $\alpha(\mathbf{X})\mathtt{H}$ over the region $\mathcal{D}$. If this lower bound is positive definite, then so is $\mathtt{H}$ everywhere in $\mathcal{D}$.

Consider a positive function $\alpha(\mathbf{X})$ and suppose that there exist constants L_i^α and U_i^α, such that the following conditions holds for all i, and all $\mathbf{X} \in \mathcal{D}$:

$$L_i^\alpha \leq \frac{\alpha(\mathbf{X})}{d_i(\mathbf{X})} \leq U_i^\alpha \ . \tag{13}$$

Then, referring to (11), we find that

$$\sum_i \left(\frac{\mathbf{u}_i \mathbf{u}_i{}^\top + \mathbf{v}_i \mathbf{v}_i{}^\top}{d_i^2(\mathbf{X})} - 9\tau_{\max}^2 \frac{\mathbf{w}_i \mathbf{w}_i{}^\top}{d_i^2(\mathbf{X})} \right) \succeq \tag{14}$$
$$\frac{1}{\alpha(\mathbf{X})^2} \sum_i \left((L_i^\alpha)^2 \left(\mathbf{u}_i \mathbf{u}_i{}^\top + \mathbf{v}_i \mathbf{v}_i{}^\top \right) - 9(U_i^\alpha)^2 \tau_{\max}^2 \mathbf{w}_i \mathbf{w}_i{}^\top \right)$$

So to establish convexity it is sufficient to verify that

$$\sum_i \left((L_i^\alpha)^2 \left(\mathbf{u}_i \mathbf{u}_i{}^\top + \mathbf{v}_i \mathbf{v}_i{}^\top \right) - 9(U_i^\alpha)^2 \tau_{\max}^2 \mathbf{w}_i \mathbf{w}_i{}^\top \right) \succeq \mathtt{0} \ . \tag{15}$$

Observe that this condition is identical with (12) except that bounds L_i and U_i are replaced by L_i^α and U_i^α. If these new bounds are tighter (that is more nearly equal) than (L_i, U_i), then we can expect a better result; we are more likely to be able to establish convexity using condition (15) than by using (12).

If L_i^α and U_i^α are not greatly different, this condition specifies that all the depths $d_i(\mathbf{X})$ are approximately proportional to some "average" depth value $\alpha(\mathbf{X})$.

A useful choice for the function $\alpha(\mathbf{X})$ in the above discussion is the average depth of point $\mathbf{X}$ with respect to all views, that is

$$\alpha(\mathbf{X}) = (1/N) \sum_{i=1}^{N} d_i(\mathbf{X}) \ .$$

If the depth $d_i(\mathbf{X})$ of a point is reasonably approximated by the average depth over all views, then $\alpha(\mathbf{X})/d_i(\mathbf{X})$ does not vary greatly over the domain $\mathcal{D}$, so the bounds L_i^α and U_i^α are close together. Even for domains $\mathcal{D}$ reaching to infinity, the ratio $\alpha(\mathbf{X})/d_i(\mathbf{X})$ will remain within reasonable maximum and minimum bounds, since the average depth $\alpha(\mathbf{X})$ increases to infinity at the same rate as any specific depth d_i. This is illustrated in Fig 1.

Average depth is a good choice for the function $\alpha(\mathbf{X})$ in the case where all the points have approximately the same depth in all views. In the case when the depths are substantially different, then it may be better to choose a weighted depth average. For instance, let $\mu_i = d_i(\mathbf{X}_0)$, where $\mathbf{X}_0 \in \mathcal{D}$ is a proposed solution to the problem. Then, a good choice for the function $\alpha(\mathbf{X})$ is

$$\alpha(\mathbf{X}) = \frac{1}{N} \sum_{i=1}^{N} \frac{d_i(\mathbf{X})}{\mu_i} \ .$$

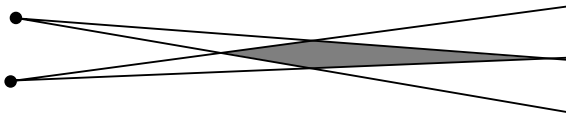


Fig. 1 *When the baseline is small compared to depth, the region $\mathcal{D}$ is elongated, and the ratio U_i/L_i can be large. However, over the region of interest, the depth of a point $\mathbf{X}$ is approximately the same in all views, so $\alpha(\mathbf{X})/d_i(\mathbf{X})$ is close to constant when $\alpha(\mathbf{X})$ is the average depth over all views.*

We now may summarize the previous discussion by stating the general convexity theorem.

Theorem 2 *With the notation as in Theorem 1, let $\alpha(\mathbf{X})$ be any positive real valued function defined on $\mathcal{D}$, and let U_i^α and L_i^α be upper and lower bounds for the value of $\alpha(\mathbf{X})/d_i(\mathbf{X})$ on $\mathcal{D}$. Then*

$$\sum_{i=1}^{N} \left((L_i^\alpha)^2 \left(\mathbf{u}_i \mathbf{u}_i{}^\top + \mathbf{v}_i \mathbf{v}_i{}^\top \right) - 9(U_i^\alpha)^2 \tau_{\max}^2 \mathbf{w}_i \mathbf{w}_i{}^\top \right) \succeq \mathtt{0} \tag{16}$$

is a sufficient condition for the least-squares error function (1) to be convex on $\mathcal{D}$. □

3.1 Computing the Bounds

For conditions of this type to be useful, it is necessary that the bounds L_i^α and U_i^α can be calculated. We can find U_i^α by maximizing $\alpha(\mathbf{X})/d_i(\mathbf{X})$ for $\mathbf{X} \in \mathcal{D}$. Any value U_i^α no smaller than this maximum will be a suitable bound.

The region $\mathcal{D}$ of $\mathbb{R}^3$ over which we need to minimize and maximize $\alpha(\mathbf{X})/d_i(\mathbf{X})$ is the intersection of second order cones. Using second order cones leads us into Second Order Cone Programming (SOCP), which for efficiency (and the non-availability of public domain code in C++) we wish to avoid. It is easier to allow $\mathcal{D}$ to be a polygonal region (if necessary by enlarging it slightly). This polygonal region can be expressed in terms of linear constraints, resulting in the use of Linear Programming to optimize $\alpha(\mathbf{X})/d_i(\mathbf{X})$. More details of this are given in the algorithm description in section 3.2.

In the case $\alpha(\mathbf{X}) = 1$, the problem reduces to finding the maximum and minimum values of depth on $\mathcal{D}$. Thus, let

$$B = \min_{\mathbf{X} \in \mathcal{D}} d_i(\mathbf{X}) = \min_{\mathbf{X} \in \mathcal{D}} \mathbf{w}_i{}^\top (\mathbf{X} - \mathbf{C}_i) \ .$$

Then U_i^α can be chosen as any value greater than $1/B$. If the region $\mathcal{D}$ is polygonal, the value of B may be found by linear programming. If $\mathcal{D}$ is an interesction of cones, then we may use SOCP to find the minimum.

If $\alpha(\mathbf{X})$ is a linear function of the different depths, say $\alpha(\mathbf{X}) = \alpha_0 + \sum_{j=1}^{n} \alpha_j d_j(\mathbf{X})$, then it is also possible to find the bounds U_i^α and L_i^α. The required optimization problem to find U_i^α is

$$\begin{array}{ll} \text{maximize} & B \\ \text{subject to} & \alpha(\mathbf{X}) \geq d_i(\mathbf{X})B \text{ and } \mathbf{X} \in \mathcal{D} \end{array}$$

where the maximization takes place over all values of B and $\mathbf{X} \in \mathcal{D}$, and U_i^α is subsequently chosen as any value no smaller than B. In this case, however, writing $d_i(\mathbf{X}) = {\mathbf{w}_i}^\top(\mathbf{X} - \mathbf{C}_i)$ results in a constraint involving both B and the point $\mathbf{X}$; this is not a linear constraint. Nevertheless, for a fixed value of B, the constraint is linear, and instead of maximizing B, we may solve a feasibility problem to determine whether there exists $\mathbf{X}$ to satisfy the constraints. The problem can be rewritten, essentially equivalently, as

$$\begin{array}{ll} \text{minimize} & B \\ \text{such that} & \alpha(\mathbf{X}) \geq d_i(\mathbf{X})\, B;\ \mathbf{X} \in \mathcal{D} \text{ is infeasible.} \end{array}$$

By a process of binary search over the range of B one may solve the optimization problem to determine the optimal B. This requires repeated solution of the feasibility problem. We write the problem in this way to emphasize that the required value of B is one in which the constraint problem is infeasible. It is of some importance not to stop the bisection process at a value B where the constraints are satisfied.

A good starting point is the point $\mathbf{X}_0 \in \mathcal{D}$, which can be used to specify a value B for which the problem is feasible. Since we may not need a tight upper bound B, the bisection process may not need to continue to high accuracy. One may set U_i^α to some value of B for which the constraint problem is infeasible.

3.2 Algorithm

We give a summary of the algorithm for proving convexity of the cost function.

Consider a set of cameras with centres $\mathbf{C}_i$, and let $\mathbf{w}_i$ be a direction vector representing the measured direction of an observed point $\mathbf{X}$ from $\mathbf{C}_i$. Let $\mathbf{u}_i$ and $\mathbf{v}_i$ be two unit vectors orthogonal to $\mathbf{w}_i$ constituting (along with $\mathbf{w}_i$) an orthogonal coordinate frame.

1. **Bounding the region.** Let $\mathbf{X}$ be a 3D point constituting a potential solution to the triangulation problem. The cost of this point is the value of the cost function

$$\begin{aligned} C(\mathbf{X}) &= \sum_{i=1}^{N} f_i^2(\mathbf{X}) \\ &= \sum_{i=1}^{N} \left(\frac{{\mathbf{u}_i}^\top(\mathbf{X} - \mathbf{C}_i)}{{\mathbf{w}_i}^\top(\mathbf{X} - \mathbf{C}_i)} \right)^2 + \left(\frac{{\mathbf{v}_i}^\top(\mathbf{X} - \mathbf{C}_i)}{{\mathbf{w}_i}^\top(\mathbf{X} - \mathbf{C}_i)} \right)^2 . \end{aligned} \tag{17}$$

   Let the value of this cost for the given point $\mathbf{X}$ be ϵ^2. We then define a region of space in which the optimal point $\mathbf{X}_{\text{opt}}$ lies according to the inequalities.

$$-\epsilon \ \leq \ \frac{{\mathbf{u}_i}^\top(\mathbf{X}_{\text{opt}} - \mathbf{C}_i)}{{\mathbf{w}_i}^\top(\mathbf{X}_{\text{opt}} - \mathbf{C}_i)} \ \leq \ \epsilon \tag{18}$$

   and similar inequalities involving $\mathbf{v}_i$ instead of $\mathbf{u}_i$. Since ${\mathbf{w}_i}^\top(\mathbf{X}_{\text{opt}} - \mathbf{C}_i) > 0$ (the cheirality constraint that the point must lie in the direction it is observed), we can multiply out by ${\mathbf{w}_i}^\top(\mathbf{X}_{\text{opt}} - \mathbf{C}_i)$ to obtain a total of four linear inequalities in the positions of the point $\mathbf{X}_{\text{opt}}$, constraining it to a polyhedral region of space, $\mathcal{D}$.
2. **Finding depth bounds.** The next step is to find minimum and maximum values of d_i on the region $\mathcal{D}$. Since d_i is defined to be ${\mathbf{w}_i}^\top(\mathbf{X} - \mathbf{C}_i)$, determining its minimum and maximum over the polyhedral region $\mathcal{D}$ is simply a pair of linear programming problems.
3. **Performing the test.** We can now form the matrix in (12) and test its smallest eigenvalue to determine if it is positive-definite. If it is, then the cost function is convex on the region $\mathcal{D}$. If the initial estimate $\mathbf{X}$ is a local minimum, then it is also a global minimum.
4. **α-test.** If the test in the previous step failed, then repeat steps 2 and 3 to find bounds (13), and then condition (16) to test for convexity.

4 General SOCP Cost Functions

We now consider the same problem in the more general context of the type of cost function that arises in many geometric Vision problems. The type of cost function that we consider is of the type that arises in SOCP. Thus, let $f : \mathbb{R}^m \to \mathbb{R}$ be a function of the form

$$f^2(\mathbf{X}) = \frac{\|\mathtt{A}^\top\mathbf{X} + \mathbf{b}\|^2}{(\mathbf{c}^\top\mathbf{X} + k)^2} . \tag{19}$$

where $\mathtt{A}$ is an $m \times n$ matrix. Suppose that for $\mathbf{X} \in \mathcal{D}$ the value of the function is bounded, so that $f^2(\mathbf{X}) < \tau_{\max}^2$ for $\mathbf{X} \in \mathcal{D}$.

Define the vector $\mathbf{v} = (\mathtt{A}^\top\mathbf{X} + \mathbf{b})/(\mathbf{c}^\top\mathbf{X} + k)$, and observe that $\|\mathbf{v}\|^2 = f^2(\mathbf{X}) < \tau_{\max}^2$. The Hessian of $f^2(\mathbf{X})$ can now be computed to equal

$$\mathtt{H} = \frac{2}{(\mathbf{c}^\top\mathbf{X} + k)^2} \begin{bmatrix} \mathtt{A} & \mathbf{c} \end{bmatrix} \begin{bmatrix} \mathtt{I}_{n\times n} & -2\mathbf{v} \\ -2\mathbf{v}^\top & 3\|\mathbf{v}\|^2 \end{bmatrix} \begin{bmatrix} \mathtt{A}^\top \\ \mathbf{c}^\top \end{bmatrix} .$$

(20)

Apart from the multiplication by $[\mathtt{A}\ \mathbf{c}]$ and its transpose, this is the same as what was obtained previously in (5). As before, we see that

$$\begin{bmatrix} \mathtt{I}_{n\times n} & -2\mathbf{v} \\ -2\mathbf{v}^\top & 3\|\mathbf{v}\|^2 \end{bmatrix} \succeq \frac{1}{3}\begin{bmatrix} \mathtt{I}_{n\times n} & 0 \\ 0^\top & -9\|\mathbf{v}\|^2 \end{bmatrix} \frac{1}{3}\begin{bmatrix} \mathtt{I}_{n\times n} & 0 \\ 0^\top & -9\tau_{\max}^2 \end{bmatrix}.$$

Indeed, it is straight-forward to verify that the difference between these two matrices,

$$\frac{2}{3}\begin{bmatrix} \mathtt{I}_{n\times n} & -3\mathbf{v} \\ -3\mathbf{v}^\top & 9\|\mathbf{v}\|^2 \end{bmatrix}$$

has eigenvectors $0, 2/3$ (repeated $n-1$ times) and $(2/3)(1+9\|\mathbf{v}\|^2)$. It results from this that

$$\mathtt{H} \succeq \frac{2}{3d(\mathbf{X})^2}(\mathtt{A}\mathtt{A}^\top - 9\tau_{\max}^2\mathbf{c}\mathbf{c}^\top). \tag{21}$$

where $d(\mathbf{X}) = \mathbf{c}^\top\mathbf{X} + k$. If we can bound $1/d(\mathbf{X})^2$ so that $L^2 \leq 1/d(\mathbf{X})^2 \leq U^2$, then we obtain a bound that does not depend at all on the point $\mathbf{X}$, namely

$$\mathtt{H} \succeq (2/3)\left(L^2\mathtt{A}\mathtt{A}^\top - 9U^2\,\tau_{\max}^2\mathbf{c}\mathbf{c}^\top\right). \tag{22}$$

In addition, as with the triangulation problem, we may define a function $\alpha(\mathbf{X})$ on $\mathcal{D}$ to obtain a general condition.

Theorem 3 *Let $\mathcal{D}$ be a domain in $\mathbb{R}^m$ and for $i = 1, \ldots, N$ let $f_i^2 : \mathcal{D} \to \mathbb{R}$ be a function of the form*

$$f_i^2(\mathbf{X}) = \frac{\|\mathtt{A}_i{}^\top\mathbf{X} + \mathbf{b}_i\|^2}{(\mathbf{c}_i{}^\top\mathbf{X} + k_i)^2}$$

such that $0 \leq f_i^2(\mathbf{X}) \leq \tau_i^2$ on $\mathcal{D}$. Let α be a positive real valued function defined on $\mathcal{D}$ and (L_i^α, U_i^α) be lower and upper bounds for $\alpha(\mathbf{X})/(\mathbf{c}_i{}^\top\mathbf{X} + k_i)$ on $\mathcal{D}$. Then

$$\sum_{i=1}^{N}\left((L_i^\alpha)^2\mathtt{A}_i\mathtt{A}_i{}^\top - 9\,(U_i^\alpha)^2\,\tau_i^2\mathbf{c}_i\mathbf{c}_i{}^\top\right) \succeq \mathtt{0} \tag{23}$$

is a sufficient condition for the function $\sum_{i=1}^N f_i^2(\mathbf{X})$ to be convex on $\mathcal{D}$.

An important special case is when α is identically equal to unity, and (L_i, U_i) are bounds for the inverse "depth", $1/(\mathbf{c}_i{}^\top\mathbf{X} + k_i)$. Also note that if $\tau_{\max} = \max_i \tau_i$, then we can replace τ_i by $\tau_{\max}$ in the (23) to get a slightly weaker (but more simple) sufficient condition.

4.1 Application to Other Problems

We now consider the way many other problems in Multiple-View Geometry may be expressed in terms of cost functions of the form given in (19), and hence are susceptible to the methods developed in this paper. Problems that can be addressed in this way are essentially the same as those that can be solved in L_∞ norm using SOCP or related methods. A survey of such problems was given in [1]. We consider the most important such problems here. In all cases we consider, it is a simple exercise to write the cost functions as a sum of terms of the form (19).

4.2 Multiple View Reconstruction

The cost function (17) involves only one point $\mathbf{X}$. Suppose we have several points $\mathbf{X}_j;\ j = 1, \ldots, M$ and also several points $\mathbf{C}_i;\ i = 1, \ldots, N$, and that a direction vector $\mathbf{w}_{ij}$ nominally pointing from $\mathbf{C}_i$ to $\mathbf{X}_j$ is known for some subset $\mathcal{S}$ of pairs (i, j). One may write a cost-function

$$\sum_{(i,j)\in\mathcal{S}} f_{ij}^2(\mathbf{C}_i, \mathbf{X}_j) = \sum_{i,j}\frac{\left(\mathbf{u}_{ij}{}^\top(\mathbf{X}_j - \mathbf{C}_i)\right)^2 + \left(\mathbf{v}_{ij}{}^\top(\mathbf{X}_j - \mathbf{C}_i)\right)^2}{\left(\mathbf{w}_{ij}{}^\top(\mathbf{X}_j - \mathbf{C}_i)\right)^2} \tag{24}$$

which is to be minimized over all choices of the $\mathbf{X}_j$ and $\mathbf{C}_i$. To remove the guage freedoms from this problem, it is best to set one of the $\mathbf{C}_i$ to be at the origin, and for one point $\mathbf{X}_j$ to be chosen to satisfy $\mathbf{w}_{ij}{}^\top(\mathbf{X}_j - \mathbf{C}_i) = 1$.

Now, if we define a vector $\mathbf{X}$ to be a concatenation of all the $\mathbf{X}_j$ and $\mathbf{C}_i$, then each of the terms in (24) is easily expressible in the form (19). The matrix $\mathtt{A}_{ij}$, and the vector $\mathbf{c}_{ij}{}^\top$ that will appear in the place of $\mathtt{A}_i$ and $\mathbf{c}_i$ in (19) will have $3(M + N)$ columns, but will be quite sparse, having non-zero entries only in columns corresponding to the points $\mathbf{X}_i$ and $\mathbf{C}_j$ involved in the term.

Defined in this way, the problem satisfies the conditions of Theorem 3, and the bound (23) may be used to check convexity of the cost function.

For large problems, the number of measurements is quite large and the total cost $\sum f_{ij}^2(\mathbf{X}_0) = \epsilon^2$ will be high. This value determines the value of τ that must be used to verify optimality. For this reason, the region $\mathcal{D}$ can be large, and the test can fail. An alternative is to set a smaller value of τ. If the convexity test succeeds for this smaller value, then we cannot rule out the existence of an optimum $\mathbf{X}_{\text{opt}}$ not in the region $\mathcal{D}$; however, it such an optimum exists, then the residual $f_{ij}^2(\mathbf{X}_{\text{opt}})$ must be greater than τ for at least one measurement. If τ is reasonably large, then this may be considered an unlikely occurrence, particularly if the measurement set is outlier-free.

4.3 Reconstruction Using Image Measurements

Consider a projective camera with camera matrix $\mathtt{P}_i$ acting on a point $\mathbf{X}_j$, and let the corresponding measured image point be $\mathbf{x}_{ij}$, defined for $(i,j) \in \mathcal{S}$, that is for some subset of all point – image pairs. If we decompose the camera matrix as

$$\mathtt{P}_i = \begin{bmatrix} \mathtt{A}_i & \mathbf{b}_i \\ \mathbf{a}_i{}^\top & b_i \end{bmatrix}$$

where $\mathtt{A}_i$ is a 2×3 matrix and $\mathbf{a}_i$ and $\mathbf{b}_i$ are vectors, then the projection of a point $\mathbf{X}_j$ is given by $(\mathtt{A}_i\mathbf{X}_j + \mathbf{b}_i)/(\mathbf{a}_i{}^\top\mathbf{X}_j + b_i)$. The squared image error is then given by

$$f_{ij}^2(\mathbf{X}_j, \mathbf{b}_i, b_i) = \left\| \frac{\mathtt{A}_i\mathbf{X}_j + \mathbf{b}_i}{\mathbf{a}_i{}^\top\mathbf{X}_j + b_i} - \mathbf{x}_{ij} \right\|^2$$

which is easily put into the form (19).

If the camera matrices are completely known, and there is only one point $\mathbf{X}_j$ that needs to be found, then this is simply the triangulation problem, expressed in terms of image coordinates.

With several points and cameras, we suppose that the first three columns of $\mathtt{P}_i$, consisting of matrix $\mathtt{A}_i$ and vector $\mathbf{a}^\top$ are known, whereas $\mathbf{X}_j$, $\mathbf{b}_i$ and b_i are unknown. Then, $\mathtt{A}_i\mathbf{X}_j + \mathbf{b}_i$ and $\mathbf{a}_i{}^\top\mathbf{X}_j + b_i$ are linear in the unknowns $\mathbf{X}_j$, $\mathbf{b}_i$ and b_i. As in section 4.2 we define a vector $\mathbf{X}$ by concatenating all the $\mathbf{X}_j$, $\mathbf{b}_i$ and b_i. Each term $f_{ij}^2(\mathbf{X})$ of the cost function is then of the required form (19) and (23) may be used to prove convexity of the cost function.

Together, $\mathtt{A}_i$ and $\mathbf{a}_i{}^\top$ make up the left-hand 3×3 block of the camera matrix. There are two important situations in which this left-hand block of $\mathtt{P}_i$ may be known. The first is when the cameras are calibrated, and the block constitutes the rotation matrix associated with the camera. This rotation matrix may have been obtained from a prior reconstruction step in which rotations are computed. This is the same problem as was considered in section 4.2, expressed now in image coordinates.

The second situation is that considered by [14], in which a plane in the scene allows us to compute inter-image homographies induced by that plane. These homographies are represented by the left-hand block of the camera matrices. Note that the solution given in [14] was a linear solution to the reconstruction problem, not the sort of least-squares solution that we wish to verify here.

4.4 Homographies and Camera Resectioning

In the camera resectioning problem, we are given 3-dimensional points $\mathbf{X}_i$ and measured corresponding points $\mathbf{x}_i$ related by an unknown 3×4 projection matrix $\mathtt{P}$, which is to be estimated. This is also known as the projective camera pose estimation problem. For no apparent reason, this has also recently become known as the PnP problem, a neologism which we abjure. The homography estimation is essentially the same, except that the points $\mathbf{X}_i$ are 2-dimensional points, and the matrix $\mathtt{P}$ is a 3×3 matrix. Since these two problems are essentially the same, we will concentrate on the pose-estimation problem.

If we denote the k-th row of $\mathtt{P}$ by $\mathbf{p}_k$, then the squared error associated with the measurement of point $\mathbf{x}_i$ is given by

$$f^2(\mathtt{P}) = \left(\frac{\mathbf{p}_1{}^\top\tilde{\mathbf{X}}_i}{\mathbf{p}_3{}^\top\tilde{\mathbf{X}}_i} - x_i\right)^2 + \left(\frac{\mathbf{p}_2{}^\top\tilde{\mathbf{X}}_i}{\mathbf{p}_3{}^\top\tilde{\mathbf{X}}_i} - y_i\right)^2$$

where $\tilde{\mathbf{X}}_i$ represents the homogeneous vector $(\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1)$. Here, the situation is slightly different from the previous cases of triangulation and multiple view reconstruction, in that the entries of the matrix $\mathtt{P}$ are the unknowns, and not the coordinates of the points $\mathbf{X}_i$. Nevertheless, it is easily observed that the squared error term may be written in the usual form (19).

5 Projective Coordinate Change

If the convexity test described previously should fail to give a positive answer, then there are further things that can be tried to attempt to prove convexity. Our first method of attack is to try a projective coordinate change.

By changing projective coordinate systems, we will reparametrize the domain of the error function $f^2(\mathbf{X})$. This may turn a non-convex function into a convex function. It is easy to see that affine coordinate changes do not affect convexity so it is enough to restrict our attention to transformations that change the plane at infinity. Cheirality conditions also apply, that is, all 3D-points must remain in front of the cameras. See [5] for more details on projective geometry.

We consider the use of the inequalities given previously to demonstrate that the function is positive semi-definite on a region. The matrix $\mathtt{H}'$ defined in (7) has one small negative eigenvalue in the direction along the principal ray of the camera, and two larger positive eigenvalues oriented in directions orthogonal to the principal ray. It gives a uniform lower bound on the Hessian in a region of space. The general idea is that if a point is seen from two different directions, then the Hessian of the combined error function is simply the sum of the individual Hessians. If the two principal rays are not aligned, then the idea is that the negative eigenvalue in the principal direction for one camera will be outweighed by the contribution of the positive eigenvalues for the other camera.

This will be true if the view directions for the different cameras are sufficiently different. If on the other hand, the view directions for the different views are the same, then

no cancelling will take place, and we will not be able to conclude that the Hessian is positive definite in any region around the point $\mathbf{X}$ of interest.

In this case, the situation can be saved by the application of a projective transformation. If the view directions for several cameras are similar, this implies that the point $\mathbf{X}$ is near to infinity. We apply a projective transformation that maps this point to a point closer to the cameras, so that the viewing rays are no longer near parallel. Then, consider the Hessian of the error function in this new coordinate system. The general idea is best illustrated by an example.

Consider cameras with matrices $\mathtt{P}_a = [\mathtt{I}|(a,0,0)^\top]$, where a is a variable. This camera is placed at the position $\mathtt{X} = -a$ on the X-axis, and the principal rays for all these cameras point in the direction of the Z-axis. Suppose we want to find the Hessian of the error function on the intersection of error-cones, defined by τ, around the principal rays of the cameras. This situation will occur when the correct triangulation point lies near infinity. Now, according to (7), the Hessian at a point with depth d may be bounded as follows:

$$H_a \succeq 2/3d^2 \,\mathrm{diag}(1,1,-9\tau^2)$$

and this bound is the same for all a. Adding any number of such Hessians $\mathtt{H}_a$ for different cameras $\mathtt{P}_a$ will not result in a positive-definite matrix. Hence, we cannot easily conclude that the Hessian is positive-definite.

It is instructive to note here in passing that since the region of interest $\mathcal{D}$ stretches to infinity, the bound L_a in condition (12) will be zero. Hence the matrix in (12) will be primitive. This problem will be partly alleviated by using the α-test (16) with $\alpha(\mathbf{X})$ equalling the average depth of a point over different images. Since the depth is the same in all images, the ratio $\alpha(\mathbf{X})/d(\mathbf{X}) = 1$, so $L_i^\alpha = U_i^\alpha = 1$, and the matrix to test is $\sum_{i=1}^{N} 2/3d^2 \,\mathrm{diag}(1,1,-9\tau^2)$, which as remarked above is still not positive-definite. The problem is that the axes of the cones are parallel, which gives no scope for the positive eigenvalues of the individual Hessians to cancel the negative ones.

However, if we apply a projective transformation represented by matrix

$$\mathtt{T} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

which takes the point $(0,0,1,0)^\top$, the point at infinity in the Z direction, to $(0,0,1,1)^\top$, a finite point. The axes of different cones pointing to this point will no longer be parallel.

This action transforms the camera matrix $\mathtt{P}_a$ to

$$\mathtt{P}_a\mathtt{T}^{-1} = \begin{bmatrix} 1 & 0 & -a & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} .$$

Note that the point $(0,0,1,1)$ still maps to the origin in all images. Now, according to (21) the Hessian for a point at depth d will be bounded by

$$\mathtt{H}_a = \frac{2}{3d^2} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -a & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -9\tau^2 \end{bmatrix} \begin{bmatrix} 1 & 0 & -a \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \frac{2}{3d^2} \begin{bmatrix} 1 & 0 & -a \\ 0 & 1 & 0 \\ -a & 0 & a^2 - 9\tau^2 \end{bmatrix} .$$

Now, summing two such Hessians for $a = 1$ and $a = -1$, we find that

$$\mathtt{H}_1 + \mathtt{H}_{-1} = \frac{4}{3d^2} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & a^2 - 9\tau^2 \end{bmatrix} ,$$

where we use the fact that (as before) the depth d is the same for all images. This matrix is obviously positive definite even for moderately large values of τ. This proves that the Hessian is positive definite on the intersection of τ-cones, and shows that the region over which it is certain that there is a single minimum of the cost function is relatively large.

Algorithm. Even if there exists a transformation $T : \mathbb{P}^m \mapsto \mathbb{P}^m$ that makes the error function $f^2(\mathbf{X})$ convex over the transformed domain $T(\mathcal{D})$ (where m is the dimension of the unknown), it may be hard to find it automatically. Note that the maximum and minimum depths change when the plane at infinity is transformed, so it is required that the bounding depths are recomputed as well.

In addition, there is an essential subtlety to observe in selecting candidates for the plane at infinity π_∞. We limit the discussion for the triangulation problems, but the similar considerations apply for other problems. First, it is not advisable to select π_∞ to be a plane cutting the region of interest $\mathcal{D}$, for in this case, points in $\mathcal{D}$ will be moved to infinity, which defeats our purpose of selecting non-parallel viewing directions. More important is a second consideration that if the selected plane at infinity separates a camera centre from the region of interest $\mathcal{D}$, then cheirality of points in $\mathcal{D}$ with respect to that camera is reversed ([4]). This causes complications in defining depth with the right sign.

In the experiments, we have found that by choosing the candidate plane at infinity as a plane parallel to one of the cameras' image planes, and with an offset such that *all* 3D points and camera centres are just in front of this plane, then good verification rates are obtained. This choice corresponds to picking a principal plane to one of the cameras and moving it slightly backwards. Recall that the coordinates of a principal plane is extracted from the last row of the camera matrix.

Suppose the candidate plane at infinity is parametrized by $\mathbf{v}^\top\mathbf{X} + 1 = 0$ where $\mathbf{v}$ is a m-vector (for triangulation

$m = 3$). Then, the the matrices $\mathtt{A}_i$ and vectors $\mathbf{c}_i$ of Theorem 3 should be updated according to $\mathtt{A}_i \mapsto \mathtt{A}_i - \mathbf{v}\mathbf{b}_i{}^\top$ and $\mathbf{c}_i \mapsto \mathbf{c}_i - k_i\mathbf{v}$. The next step is to recompute the bounding depths, and finally test whether condition (16) holds.

6 Experiments

Test Procedure. In all experiments described below, the following test procedure is run until a definitive answer is returned (unless otherwise stated) in order to determine convexity and hence to verify global optimality.

1. Compute a local minimizer $\mathbf{x}_{\text{local}}$ with bundle adjustment.
2. Compute lower and upper bounds, L_i and U_i, respectively, according to Theorem 1.
3. Test if the convexity condition in (12) holds (referred to as *primary test*).
4. Compute lower and upper bounds, L_i^α and U_i^α, respectively, according to Theorem 2.
5. Test if the stronger convexity condition in (16) holds (referred to as α*-test*).
6. Apply a projective transformation and test convexity (see Section 5). This is only done for triangulation.

Notre Dame. This data set was originally created in [15]. It consists of 595 images with $277,877$ reconstructed 3D-points. Each point is visible in at least 2 images up to as many as 216 images according to the following distribution: 2 images 58%, 3 images 19%, $4-10$ images 18% and more than 10 images 5%.

Out of $277,887$ triangulated 3D-points, 265 instances were unsuccessful with our primary test in (12), and after applying the stronger α-test derived in Section 3, 158 cases remained. By transforming the plane at infinity, another 156 cases could be proven to be optimal. Hence, only 2 cases could not be verified to be globally optimal.

See Table 1 for a summary of the results. Each problem instance takes on the average less than 0.5 milliseconds to verify on a 2.66GHz Pentium written in C++.

Dinosaur. This turn-table sequence consists of 36 images with 328 given point features. The complete 3D reconstruction was computed with standard structure from motion routines (including refinements by bundle adjustment).

The multiview geometry problems tested were triangulation, camera resectioning and structure and motion with known rotations. See Table 2 for a summary of results. Both camera resectioning and triangulation work very well for this type of scene and camera motion. It was not possible to prove optimality for the whole sequence (assuming known rotations), only for the first 22 images. Note that this configuration is still a large structure and motion problem: 22 cameras and several hundreds of 3D points. When going beyond 22 images, the primary test was not sufficient. We did not apply the α-test due to the sheer size of the configuration.

The large number of image measurements in this data set makes the total cost $C(\mathbf{x}_0)$ relatively large. And since this value determines the value of τ for the test, it is not surprising that the convexity test fails. One may ask to what pixel threshold is it possible to verify optimality? We ran a series of tests with different values τ's using bisection and found that it is possible to verify the whole configuration for up to 2.59 pixels. It cannot be ruled out the existence of an optimum $\mathbf{x}_{\text{opt}}$ with lower total cost $C(\mathbf{x}_{\text{opt}})$, but at the same time such a solution must have at least one residual with more than 2.59 pixels of error. Given that the data set is outlier free and typical residual errors are much lower, this seems like an unlikely occurrence. Hence, we may conclude that with high probability we have verified the globally optimal solution. Similar ideas have recently been developed for one-dimensional vision problems in [13].

We also tried verifying 100 random pairs of images with common feature points in this pair, and similarly 100 random triples with all common feature points, with 100% success rate. Only pairs and triples with at least 10 feature points were selected.

The Christ Statue. The images were collected from various tourist photographs of this well-known statue and reconstructed with standard structure and motion routines. Similar experiments as for the Dinosaur sequences were performed with similar success rates, see Table 3.

Corridor. The forward camera motion for this 11-image sequence is quite different from the other sequences. Verifying global optimality for triangulation and camera resectioning turned out to be no problem, but for the other multiview problems it was more difficult, see Table 4. We did not apply the α-test to the known rotation cases due to the large size of the configurations.

There are several 3D planes in the scene and we took all the point features on the left frontal wall and computed all pairwise 3×3 homographies for this plane using bundle adjustment. Out of the $\binom{11}{2} = 55$ image pairs, 27 were successfully verified with the primary test, and another 3 with the α-test.

The known rotation case turned out to be more difficult, which is consistent with the findings in [17]. Not a single test was successful for the primary test. We did not apply the other methods due to the size of the setup. The whole sequence has 11 cameras and more than 4000 image points, so just computing all the depth tests takes time. (The size for the LP constraint set is approximately $4 \times 4000 = 16000$ since each image coordinate gives rise to two inequality constraints).

Arnold. A final experiment with 6 images of a poster was tested, see Fig 2 and Table 5. In this scene, all 3D feature points are on a plane, and hence uncalibrated camera resectioning is not possible. Triangulation of all 3D points (in total, 1639 points) as well as all pairwise homographies (15 in total) induced by the plane were successfully verified by the primary test. The results with known rotation were as follows: 9 out of 15 pairwise images were successfully verified and 17 out 20 image triples were verified. Again, no other steps in the test procedure were applied due to the size of these configurations.

Discussion. The method works very successfully (with almost 100% success) on the triangulation and resection problems. However for large problems, such as full 3D reconstruction the method is less reliable. The main reason for this is the approximation made after (2), where it is deduced that each $f_i(\mathbf{x}_{\text{opt}}) \leq \epsilon^2$. This is the worst-case hypothesis, that all the error is incurred by one measurement. The effect is to set an unrealistically high bound $\tau_{\max}$ in the convexity conditions, which causes the tests to fail. For this reason, the tests derived here do not work well for very large problems. It is only possible to verify subproblems for optimality.

The other main failure case for the primary and α-tests is when points are a long way away compared to the base line, that is, close to infinity. This is because the two bounds L and U in (10) or L_i^α and U_i^α in (15) are far apart – the intersection volume is too elongated. Indeed, it is easily seen that if U or U_i^α is too large then the matrix in (10) or (15) cannot be positive definite. However, in this case, as shown, the projective change method succeeds in all but two cases.

In practice, the three tests, the primary test, the α-test and projective change are run sequentially until one of them returns a positive answer. Since the tests are successively more expensive in terms of time, this is the most efficient approach. The total time for triangulation verification of all the points in the Notre Dame sequence was 2m 12s (less than 0.5ms per point) on a 2.66GHz laptop, single threaded, programmed in C++.

Comparing the primary test and the α-test, it was observed (on the Notre Dame set) that the α-test is more accurate. There were only two cases where the α-test failed but the primary test succeeded. However the α-test takes about 5 times as long to run as the primary test (9m 40s). Since in practice it is only run when the primary test fails (about one test in 1000, this longer run time is not important).

As a tool to verify the optimality of the triangulation and resection results for a 3D reconstruction, this method requires insignificant time in the context of the complete reconstruction task. Even for a large reconstruction of 10^6 points, the time to verify that all points and cameras are triangulated or resectioned correctly will take no more than 500 seconds.

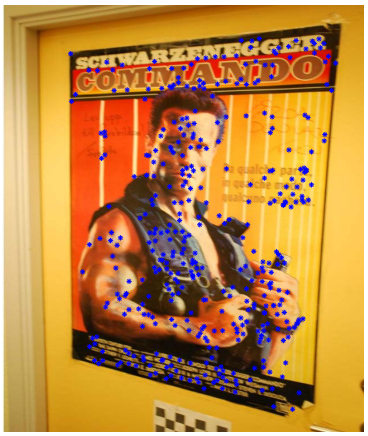


Fig. 2 One image of the Arnold sequence with feature points.

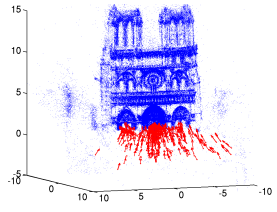
To summarize, reconstruction problems with small dimensions can be handled well with a small computational effort (the primary test generally works) while for larger problems the picture is mixed. The success rate depends on the geometry of the scene as well as the motion of the camera. Images taken with a wide baseline seem to be more easily verifiable compared to other camera motions. As the number of residuals increases, it becomes harder to verify optimality – just as expected.

Appendix A: Minimizing Squared Angle

In the derivations in Section 2, the error function used was the sum of squares of the *tangents* of the angles, as given by (4). This leads to a relatively simple result in terms of computing and bounding the Hessian. On the other hand, it would be more natural to wish to minimize the sum of squares of the error angles, and not their tangents. The difference is very small, but for exactness, we now derive a similar result for this error function. It is worth noting here that although in many computer-vision applications, pixel error is the natural metric for evaluating a solution, this is not clearly true of omnidirectional or wide-angle images.

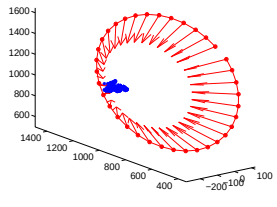
In a different application, target tracking and localization, the position of a target may be measured from several observation points resulting in angular (alt-azimuth) readings. The task of determining the position of the target is precisely the orientation problem. In this context, it is natural to wish to minimize squared angular error, which is what we do in this appendix.

In the following discussion, we give the outline of the argument. To verify the details, the reader may need to use a computer algebra system, such as Mathematica. Now, under



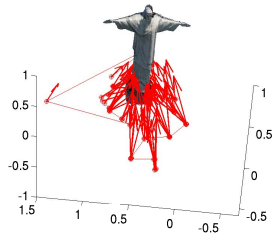
Problem	Instances	Method	Unsuccessful tests
Triangulation	277887 points	Primary test (12)	265 / 277887
		α-test (16)	158 / 265
		Projective change	2 / 158

Table 1: Results for Notre Dame.



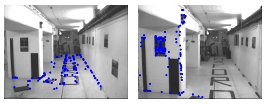
Problem	Instances	Method	Unsuccessful tests
Triangulation	328 points	Primary test (12)	0 / 328
Resectioning	36 cameras	Primary test (12)	0 / 36
Known rotation	1 set of 36 images	Primary test (12)	1 / 1
	1 set of 22 images	Primary test (12)	0 / 1
	100 random pairs	Primary test (12)	0 / 100
	100 random triples	Primary test (12)	0 / 100

Table 2: Results for Dinosaur.



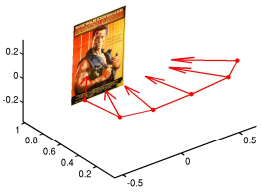
Problem	Instances	Method	Unsuccessful tests
Triangulation	185 points	Primary test (12)	0 / 185
Resectioning	76 cameras	Primary test (12)	0 / 76
Known rotation	100 random pairs	Primary test (12)	12 / 100
	100 random triples	Primary test (12)	3 / 100

Table 3: Results for the Christ Statue.



Problem	Instances	Method	Unsuccessful tests
Triangulation	737 points	Primary test (12)	0 / 737
Resectioning	11 cameras	Primary test (12)	0 / 11
Homography	55 image pairs	Primary test (12)	28 / 55
		α-test (16)	25 / 28
Known rotation	1 set of 11 images	Primary test (12)	1 / 1
	55 sets of image pairs	Primary test (12)	55 / 55
	165 sets of image triples	Primary test (12)	165 / 165

Table 4: Results for Corridor.



Problem	Instances	Method	Unsuccessful tests
Triangulation	1639 points	Primary test (12)	0 / 1639
Homography	15 image pairs	Primary test (12)	0 / 15
Known rotation	15 sets of image pairs	Primary test (12)	6 / 15
	20 sets of image triples	Primary test (12)	3 / 20

Table 5: Results for Arnold.

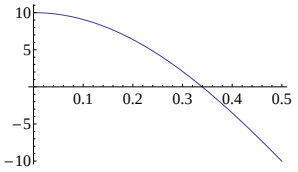


Fig. 4 *Plot of the function $D(\phi)/\phi^2$ demonstrating that the product of the eigenvalues of $\mathtt{H}-\mathtt{H}'$ is positive for $\phi < 0.3$.*

the same conditions as before, we define the error function

$$f^2(\mathrm{X},\mathrm{Y},\mathrm{Z}) = \phi^2 = \arctan\left(\frac{\sqrt{\mathrm{X}^2+\mathrm{Y}^2}}{\mathrm{Z}}\right)^2 . \tag{25}$$

We compute the Hessian at a point with $(\mathrm{X},\mathrm{Y},\mathrm{Z})$ with respect to the coordinates X, Y and Z. Subsequently, evaluating with $\mathrm{X} \geq 0$ and $\mathrm{Y} = 0$, and making the substitutions ϕ for $\arctan(\mathrm{X}/\mathrm{Z})$, radial distance r for $\sqrt{\mathrm{X}^2+\mathrm{Z}^2}$ and $\tan\phi$ for X/Z, we arrive after some computation at the expression for the Hessian given in Fig 3. This matrix has eigenvalues $2\phi/(r^2\tan(\phi))$ and $\left(1 \pm \sqrt{1+4\phi^2}\right)/r^2$, namely two positive and one negative eigenvalue.

Similarly as before[1], let $\mathtt{H}'$ be the matrix $(2/r^2)\mathrm{diag}(1/4, 1/4, -4\phi^2)$. We claim that $\mathtt{H}-\mathtt{H}'$ is positive semi-definite. Unfortunately, the proof is a little more intricate than before.

First, we observe that $(0,1,0)^\top$ is an eigenvector of $\mathtt{H}-\mathtt{H}'$, with eigenvalue $(2/r^2)(\phi/\tan(\phi) - 1/4)$, which is positive at least for $\phi < 1$. The other two eigenvalues are the eigenvalues of the reduced-size matrix obtained by eliminating the second row and column from $\mathtt{H}-\mathtt{H}'$. This matrix will have two positive eigenvalues, as long as its trace and determinant are both positive. Apart from the factor $1/r^2$ the trace is equal to $8\phi^2 + 3/2$ which is positive. The determinant is equal to

$$D(\phi) = -1 + (1+16\phi^2)(\cos(2\phi) - 2\phi\sin(2\phi)) .$$

This function is positive for $\phi < 0.3$ as may be shown by plotting the function $D(\phi)/\phi^2$ (see Fig 4). A more formal proof can be given by computing the series expansion of this function.

The result of this computation is the following result.

Lemma 1 *If $\mathtt{H}$ is the Hessian of the error function (25) evaluated at a point with error less than angle $\phi < 0.3$, then*

$$\mathtt{H} \succeq (2/r^2)\mathrm{diag}(1/4, 1/4, -4\phi^2) .$$

From here on, conditions for convexity of the error function in the intersection of a set of cones with angle bounded by $\phi_{\max}$ proceeds just the same as previously.

[1] By analogy with section 2.1 one may be tempted to define $\mathtt{H}' = (2/r^2)\mathrm{diag}(1/3, 1/3, -3\phi^2)$ which gives a slightly better bound, but this does not work in this case.

Appendix B: Minimum Bounds for Hessians

The matrix $\mathtt{H}'$ given in (7) was presented without motivation. In this section, we will give some theory that justifies the choice. This will be developed in a more general context, to allow for possible applications to other problems. In what follows, all matrices are assumed to be symmetric, even if not stated.

Notation. We need to develop some properties of the relationship $\mathtt{H} \succeq \mathtt{0}$ which has been used throughout this paper. It means that $\mathtt{H}$ is positive semi-definite, possibly zero. The notation $\mathtt{H} \succeq \mathtt{H}'$ means $\mathtt{H}-\mathtt{H}' \succeq \mathtt{0}$. This defines a partial ordering relationship between matrices. We use the notation $\mathtt{H} \succ 0$ to mean that $\mathtt{H}$ is positive semi-definite, but not zero. It could be interpreted as meaning that $\mathtt{H}$ is positive definite, but not semi-definite, but that is not what we mean. If we need to specify that $\mathtt{H}$ is strictly positive-definite, this will be specifically stated.

Consider a family of symmetric matrices $\mathtt{H}_\mathbf{v}$ defined in terms of a parameter vector $\mathbf{v}$. We say that a matrix $\mathtt{G}$ is a lower bound for this set if $\mathtt{H}_\mathbf{v} \succeq \mathtt{G}$ for all $\mathbf{v}$, and it is a maximal lower bound if there is no other lower bound $\mathtt{G}'$ such that $\mathtt{G}' \succ \mathtt{G}$. There may be more than one maximal lower bound for a family of matrices. Clearly if we want to define a lower bound for a set of Hessian matrices, it is advantageous to identify the maximal lower bounds.

We are particularly interested in the lower bounds for matrices of the form

$$\mathtt{H}_\mathbf{v} = \begin{bmatrix} \mathtt{I}_{n\times n} & -2\mathbf{v} \\ -2\mathbf{v}^\top & 3\tau_{\max}^2 \end{bmatrix} \tag{26}$$

as in (5) or (20). We may determine the complete set of maximal lower bounds for the set of all matrices $\mathtt{H}_\mathbf{v}$ where $\|\mathbf{v}\|$ is bounded.

Theorem 4 *Consider the set of matrices*

$$\mathcal{H} = \{\mathtt{H}_\mathbf{v} \mid \|\mathbf{v}\|^2 \leq \tau_{\max}^2\} . \tag{27}$$

The maximal lower bounds for this set are the matrices of the form $\mathtt{G}_\lambda = \mathrm{diag}(1-1/\lambda, 1-1/\lambda, (3-4\lambda)\tau_{\max}^2)$ with $\lambda > 0$.

This theorem will follow from a more general theorem, which we will state shortly, in terms of maximal ellipsoids in $\mathbb{R}^n$.

The matrices of interest are of the form

$$\mathtt{A}_0 - \begin{bmatrix} \mathtt{0}_{n\times n} & \mathbf{v} \\ \mathbf{v}^\top & 0 \end{bmatrix} = \mathtt{A}_0 - \mathtt{S}(\mathbf{v}), \tag{28}$$

where $\mathbf{v}$ ranges over a set V of vectors in $\mathbb{R}^n$, and $\mathtt{A}_0$ is a symmetric matrix. Compare this with the form of (26).

The following simple observation reduces finding lower bounds of such matrices $\mathtt{A}_0 - \mathtt{S}(\mathbf{v})$ to finding bounds for the set $\mathtt{S}(\mathbf{v})$.

$$\mathtt{H} = \frac{1}{r^2}\begin{bmatrix} 1+\cos(2\phi)-2\phi\sin(2\phi) & 0 & -2\phi\cos(2\phi)-\sin(2\phi) \\ 0 & 2\phi/\tan(\phi) & 0 \\ -2\phi\cos(2\phi)-\sin(2\phi) & 0 & 1-\cos(2\phi)+2\phi\sin(2\phi) \end{bmatrix}$$

Fig. 3 *Hessian for cost function (25)*

Lemma 2 *The (maximal) lower bounds of* $\mathtt{A}_0 - \mathtt{S}(\mathbf{v})$ *for all* $\mathbf{v} \in V$ *are exactly those matrices* $\mathtt{A}_0 - \mathtt{G}$*, where* $\mathtt{G}$ *is a (minimal) upper bound for the matrices* $\mathtt{S}(\mathbf{v})$*.*

We now consider a connection between matrices and ellipsoids in $\mathbb{R}^n$ that will help us identify a connection between maximal lower bounds for sets of matrices and ellipsoids. Let $\mathtt{G}$ be a matrix of the form

$$\mathtt{G} = \begin{bmatrix} \mathtt{A} & \mathbf{r} \\ \mathbf{r}^\top & q \end{bmatrix} \text{ with } \mathtt{A} \succeq 0 \text{ and } q \geq 0 \tag{29}$$

Define also

$$\mathtt{G}_\lambda = \begin{bmatrix} \mathtt{A}/\lambda & \mathbf{r} \\ \mathbf{r}^\top & q\lambda \end{bmatrix} \text{ for } \lambda > 0. \tag{30}$$

Definition 1 Let $\mathtt{G}$ be a matrix of the form (29). We define the ellipsoid $C(\mathtt{G})$ associated with $\mathtt{G}$ to be the set

$$C(\mathtt{G}) = \{\mathtt{K}\mathbf{w} + \mathbf{r} \mid \|\mathbf{w}\|^2 \leq q\}$$

where $\mathtt{K}$ is a real symmetric $n \times n$ matrix such that $\mathtt{A} = \mathtt{K}^2$.

Thus, $C(\mathtt{G})$ is the image of a ball in $\mathbb{R}^n$ under an affine mapping, and hence an ellipsoid, possibly degenerate (lying in a subspace of dimension less than n) if $\mathtt{A}$ does not have full rank. It is clear that $C(\mathtt{G})$ and $C(\mathtt{G}_\lambda)$ define the same ellipsoid for all λ.

Even when $\mathtt{A}$ is not of full rank such a matrix $\mathtt{K}$ exists, since writing $\mathtt{A} = \mathtt{U}\mathtt{D}^2\mathtt{U}^\top$, where $\mathtt{D}$ is diagonal and $\mathtt{U}$ orthogonal, we can choose $\mathtt{K} = \mathtt{U}\mathtt{D}\mathtt{U}^\top$, which is symmetric. It is easy to verify that the set $C(\mathtt{G})$ is independent of the choice of $\mathtt{K}$, depending only on $\mathtt{A} = \mathtt{K}^2$.

We give another useful characterization of ellipsoids. For a matrix $\mathtt{A}$, define the span of $\mathtt{A}$ to be $\mathrm{Sp}(\mathtt{A}) = \{\mathtt{A}\mathbf{w}\}$ for all vectors $\mathbf{w}$. Further, given a symmetric matrix $\mathtt{A}$ with Singular Value Decomposition $\mathtt{A} = \mathtt{U}\mathtt{D}\mathtt{U}^\top$, define the pseudo-inverse $\mathtt{A}^+ = \mathtt{U}\mathtt{D}^+\mathtt{U}^\top$, where $\mathtt{D}^+_{ii} = \mathtt{D}^{-1}_{ii}$ when $\mathtt{D}_{ii} = 0$ and $\mathtt{D}^+_{ij} = 0$ otherwise. Then, for a matrix of the form (29),

$$C(\mathtt{G}) = \{\mathbf{x} \mid (\mathbf{x}-\mathbf{r}) \in \mathrm{Sp}(\mathtt{A}) \text{ and } (\mathbf{x}-\mathbf{r})^\top\mathtt{A}^+(\mathbf{x}-\mathbf{r}) \leq q\}. \tag{31}$$

Our main theorem on lower bounds of matrices can now be stated.

Theorem 5 *Let* V *be a bounded set of points in* $\mathbb{R}^n$*. The maximal lower bounds for the set of matrices* $\mathtt{A}_0 - \mathtt{S}(\mathbf{v})$ *with* $\mathbf{v} \in V$*, are the matrices of the form* $\mathtt{A}_0 - \mathtt{G}$*, where* $\mathtt{G}$ *is of the form (29) and* $C(\mathtt{G})$ *is a minimal ellipsoid containing the set* V*.*

Application. Before giving the proof of this theorem, we show how it may be used to prove Theorem 4 as a special case. Thus, let $\mathtt{A}_0 = \mathrm{diag}(\mathtt{I}_{n\times n}, 3\tau^2_{\max})$. According to Theorem 5, the maximal lower bounds for the set of matrices $\mathcal{H}$ are those matrices of the form $\mathtt{A}_0 - \mathtt{G}_\lambda$ where $C(\mathtt{G})$ is a minimal ellipsoid containing the set of vectors $V = \{2\mathbf{v} \mid \|\mathbf{v}\| \leq \tau^2\}$. The minimal ellipsoid containing V is clearly a sphere of radius 2τ, having equation $\mathbf{x}^\top\mathtt{I}\mathbf{x}^\top \leq 4\tau^2$. Therefore, $\mathtt{G}_\lambda = \mathrm{diag}(\mathtt{I}/\lambda, 4\lambda\tau^2)$. Consequently, the least lower bounds for the set $\mathcal{H}$ are the matrices $\mathrm{diag}(1-1/\lambda, \ldots, 1-1/\lambda, (3-4\lambda)\tau^2)$ with $\lambda > 0$, which is the conclusion of Theorem 4.

If $\lambda < 1$, these lower bounds are not very useful, since the eigenvalues $1-1/\lambda$ are negative. Our strategy in this paper is to select the value of λ that leads to the largest absolute value of the ratio of the largest to smallest eigenvalues of this matrix. Hence, we seek to maximize $(1-1/\lambda)/(4\lambda-3)\tau^2$ with $\lambda > 1$. The maximum occurs when $\lambda = 3/2$, and the corresponding optimal bound is $\mathrm{diag}(1/3, 1/3, -3\tau^2)$. This is the value that is used in (7).

Proof of theorem. We now start the proof of Theorem 5, which will be completed through a sequence of lemmas. The connection of ellipsoids with lower bounds of sets of matrices is given by the following lemma.

Lemma 3 $\mathtt{G} \succeq \mathtt{S}(\mathbf{v})$ *if and only if* $\mathtt{G}$ *is of the form (29) and* $\mathbf{v} \in C(\mathtt{G})$*. Consequently,* $\mathtt{G}$ *is an upper bound for a set of matrices* $\mathtt{S}(\mathbf{v})$*,* $\mathbf{v} \in V$ *if and only if* $C(\mathtt{G})$ *contains the set* V*.*

Proof First, we prove the *if* part of this lemma. Writing $\mathtt{G}$ as in (29), choose $\mathtt{K}$ such that $\mathtt{K}^2 = \mathtt{A}$ with $\mathtt{K}$ symmetric. If $\mathbf{v} \in C(\mathtt{G})$, then we can write $\mathbf{v} - \mathbf{r} = \mathtt{K}\mathbf{w}$ with $\mathbf{w}^\top\mathbf{w} \leq q$. Substituting for $\mathbf{v} - \mathbf{r}$ in $\mathtt{G} - \mathtt{S}(\mathbf{v})$ we see

$$\begin{aligned} \mathtt{G} - \mathtt{S}(\mathbf{v}) &= \begin{bmatrix} \mathtt{K}^2 & -\mathtt{K}\mathbf{w} \\ -\mathbf{w}^\top\mathtt{K} & q \end{bmatrix} \\ &= \begin{bmatrix} \mathtt{K} & 0 \\ -\mathbf{w}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathtt{I} & \mathbf{0} \\ \mathbf{0}^\top & q - \mathbf{w}^\top\mathbf{w} \end{bmatrix} \begin{bmatrix} \mathtt{K} & -\mathbf{w} \\ \mathbf{0}^\top & 1 \end{bmatrix}. \end{aligned}$$

Therefore $\mathbf{x}^\top(\mathtt{G} - \mathtt{S}(\mathbf{v}))\mathbf{x} \geq 0$ for any $\mathbf{x}$, and it follows that $\mathtt{G} - \mathtt{S}(\mathbf{v}) \succeq 0$.

For the converse, assume that $\mathtt{G} - \mathtt{S}(\mathbf{v}) \succeq 0$. Then $\mathtt{G}$ can be written as in (29) where $\mathtt{A} \succeq 0$ and $q \geq 0$, since any principal minor of a positive semi-definite matrix must be semi-definite.

If $\mathbf{v}-\mathbf{r}$ does not lie in the span of the columns of $\mathtt{K}$, then there exists a vector $\mathbf{x}$ such that $\mathtt{K}\mathbf{x}=0$ and of suitable length such that $(\mathbf{v}-\mathbf{r})^\top\mathbf{x}>q$. Writing $\hat{\mathbf{x}}=(\mathbf{x}^\top,1)^\top$, we see that

$$\hat{\mathbf{x}}^\top(\mathtt{G}-\mathtt{S}(\mathbf{v}))\hat{\mathbf{x}}=q-(\mathbf{v}-\mathbf{r})^\top\mathbf{x}<0,$$

so $\mathtt{G}-\mathtt{S}(\mathbf{v})$ can not be positive semi-definite.

Consequently, $\mathbf{v}-\mathbf{r}$ must lie in $\mathrm{Sp}(\mathtt{K})$ and we may write $\mathbf{v}-\mathbf{r}=\mathtt{K}\mathbf{w}$. Select the $\mathbf{w}$ of minimum norm that satisfies this equation. This is given by $\mathbf{w}=\mathtt{K}^+(\mathbf{v}-\mathbf{r})$. Then

$$\mathbf{w}=\mathtt{K}^+(\mathbf{v}-\mathbf{r})=\mathtt{K}(\mathtt{K}^+\mathtt{K}^+(\mathbf{v}-\mathbf{r}))=\mathtt{K}\mathbf{x}$$

where $\mathbf{x}$ is defined by this formula. Let $\hat{\mathbf{x}}=(\mathbf{x}^\top,1)^\top$. Then, $\begin{bmatrix}\mathtt{K} & -\mathbf{w}\\ \mathbf{0}^\top & 1\end{bmatrix}\hat{\mathbf{x}}=(\mathbf{0}^\top,1)^\top$ and it follows that $\hat{\mathbf{x}}^\top(\mathtt{G}-\mathtt{S}(\mathbf{v}))\hat{\mathbf{x}}=q-\mathbf{w}^\top\mathbf{w}$. Since by assumption $\mathtt{G}-\mathtt{S}(\mathbf{v})\succeq 0$, it follows that $\|\mathbf{w}\|^2\leq q$, and so $\mathbf{v}=\mathtt{K}\mathbf{w}+\mathbf{r}$ is in the ellipsoid $C(\mathtt{G})$, and the proof is complete.

This identifies the upper bounds for a set of matrices $\mathtt{S}(\mathbf{v})$. We need to identify the least upper bounds. This will be done by the following lemma.

Lemma 4 *Let $\mathtt{G}$ and $\mathtt{G}'$ be matrices as in (29). If $\mathtt{G}'\prec\mathtt{G}$ then $C(\mathtt{G}')\subset C(\mathtt{G})$. Conversely, if $C(\mathtt{G}')$ is a proper subset of $C(\mathtt{G})$ then there exists a number λ such that $\mathtt{G}'_\lambda\prec\mathtt{G}$.*

Note that the ellipsoids represented by $\mathtt{G}'_\lambda$ are the same for all $\lambda\neq 0$. It is clearly not true that $\mathtt{G}'_\lambda\preceq\mathtt{G}$ for all λ, since by making λ large or small enough, this condition will surely be broken.

A corollary of this lemma is

Lemma 5 *Let V be a bounded set of points in $\mathbb{R}^n$. Matrix $\mathtt{G}$ is a least upper bound for matrices $\mathtt{S}(\mathbf{v})$, $\mathbf{v}\in V$ if and only if $\mathtt{G}$ is of the form (29) and $C(\mathtt{G})$ is a minimal ellipsoid containing the set V.*

Proof of the first part of Lemma 4 is easy. Let $\mathbf{v}$ be a point in $C(\mathtt{G}')$. Then, by Lemma 3, $\mathtt{G}'\succeq\mathtt{S}(\mathbf{v})$. If $\mathtt{G}\succ\mathtt{G}'$ then $\mathtt{G}\succeq\mathtt{S}(\mathbf{v})$. Again by Lemma 3, this implies that $\mathbf{v}\in C(\mathtt{G})$. Hence $C(\mathtt{G}')\subset C(\mathtt{G})$.

Now, we turn to the converse statement, which is considerably more difficult. First, note that it is sufficient to prove this result in the case where the matrices $\mathtt{A}$ and $\mathtt{A}'$ belonging to $\mathtt{G}$ and $\mathtt{G}'$ are invertible. If this is not the case, then the ellipsoids they define are degenerate, lying in a lower-dimensional subspace. By an appropriate affine transformation of the matrices, we can reduce them to matrices of lower dimension after eliminating rows and columns of zeros, and subsequently assume that the matrices are invertible.

We simplify this problem a little by simplifying the form of the matrices. The first observation is that we can assume that one of the two vectors $\mathbf{r}$ and $\mathbf{r}'$ is zero. Thus, we can assume that $\mathbf{r}'$ is replaced by $\mathbf{r}'-\mathbf{r}$, and $\mathbf{r}$ is replaced by the zero vector.

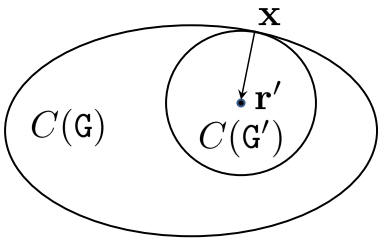


Fig. 5 *A result about ellipsoids. If a sphere is contained in an ellipsoid, then there exists an ellipsoid represenetd by a matrix of the form (32) that containes the centre of the sphere.*

The second simplification is to apply an affine transformation to reduce to the case where $\mathtt{A}'=\mathtt{I}$. There exists a matrix $\mathtt{P}$ such that $\mathtt{P}^\top\mathtt{A}'\mathtt{P}=\mathtt{I}$. Essentially, this affine transformation rescales non-isometrically so that the second ellipsoid becomes a spherical ball. We may also assume that $q=q'=1$.

It is now easily verified that if we prove Lemma 4 in the special case where $\mathtt{A}'$ is the identity and $\mathbf{r}=\mathbf{0}$, then it will hold generally. Thus, given

$$\mathtt{G}=\begin{bmatrix}\mathtt{A} & \mathbf{0}\\ \mathbf{0}^\top & 1\end{bmatrix}\quad;\quad\mathtt{G}'=\begin{bmatrix}\mathtt{I} & \mathbf{r}'\\ \mathbf{r}'^\top & 1\end{bmatrix}$$

our task is to show that

$$\mathtt{G}-\mathtt{G}'_\lambda=\begin{bmatrix}\mathtt{A}-\mathtt{I}/\lambda & -\mathbf{r}'\\ -\mathbf{r}'^\top & 1-\lambda\end{bmatrix}\succ\mathtt{0}\tag{32}$$

for some λ, under the assumption that the ellipsoid $C(\mathtt{G})$ contains the sphere $C(\mathtt{G}')$.

However, this is equivalent, according to Lemma 3 to showing that $\mathtt{A}-\mathtt{I}/\lambda\succeq\mathtt{0}$, and $1-\lambda\geq 0$, and that the ellipsoid $\mathbf{x}^\top(\mathtt{A}-\mathtt{I}/\lambda)^+\mathbf{x}\leq(1-\lambda)$ contains the point $\mathbf{r}'$. This is shown in Fig 5.

In the worst case, we may assume that the sphere $C(\mathtt{G}')$ is touching the ellipsoid $C(\mathtt{G})$ tangentially at some point, which we denote by $\mathbf{y}$. The inward-pointing normal to the ellipsoid at $\mathbf{y}$ is obtained by differentiating $\mathbf{y}^\top\mathtt{A}^{-1}\mathbf{y}$. It is $-2\mathtt{A}^{-1}\mathbf{y}$ and the unit vector is $-\mathtt{A}^{-1}\mathbf{y}/\lambda$ where we define $\lambda=\|\mathtt{A}^{-1}\mathbf{y}\|$. Therefore, the centre of the unit sphere touching the ellipsoid at point $\mathbf{y}$ is given by

$$\mathbf{r}'=\mathbf{y}-\mathtt{A}^{-1}\mathbf{y}/\lambda=(\mathtt{A}-\mathtt{I}/\lambda)\mathtt{A}^{-1}\mathbf{y}\tag{33}$$

where we define $\lambda=\|\mathtt{A}^{-1}\mathbf{y}\|$. This shows that $\mathbf{r}'\in\mathrm{Sp}(\mathtt{A}-\mathtt{I}/\lambda)$. We need to show that $\mathbf{r}'$ lies in the ellipsoid $\mathbf{x}^\top(\mathtt{A}-\mathtt{I}/\lambda)^+\mathbf{x}\leq(1-\lambda)$. We compute

$$\begin{aligned}\mathbf{r}'^\top(\mathtt{A}-\mathtt{I}/\lambda)^+\mathbf{r}'&=\mathbf{y}^\top\mathtt{A}^{-1}(\mathtt{A}-\mathtt{I}/\lambda)\mathtt{A}^{-1}\mathbf{y}\\&=\mathbf{y}^\top\mathtt{A}^{-1}\mathbf{y}-\mathbf{y}^\top\mathtt{A}^{-2}\mathbf{y}/\lambda\\&=1-\lambda\end{aligned}\tag{34}$$

as required.

We also need to show that the matrix $\mathtt{A} - \mathtt{I}/\lambda$ is positive semi-definite. This will be true if and only if $1/\lambda \leq \lambda_{\min}(\mathtt{A})$, the smallest eigenvalue of $\mathtt{A}$. Equivalently, our task is to show that

$$\lambda \geq \lambda_{\max}(\mathtt{A}^{-1}) \tag{35}$$

under the assumption that the unit sphere touching the ellipsoid $\mathbf{x}^\top \mathtt{A}^{-1}\mathbf{x} = 1$ at the point $\mathbf{y}$ lies entirely inside the ellipsoid. Now, let $\mathbf{u}$ be the eigenvector of $\mathtt{A}^{-1}$ such that $\mathtt{A}^{-1}\mathbf{u} = \lambda_{\max}\mathbf{u}$, and suppose that the condition (35) is violated. With the centre $\mathbf{r}'$ of the sphere given by (33) we verify that

$$\begin{aligned}\mathbf{u}^\top\mathbf{r}' &= \mathbf{u}^\top\mathbf{y} - \mathbf{u}^\top\mathtt{A}^{-1}\mathbf{y}/\lambda = \mathbf{u}^\top\mathbf{y}\left(1 - \lambda_{\max}(\mathtt{A}^{-1})/\lambda\right) \\ &= c\,\mathbf{u}^\top\mathbf{y}\end{aligned}$$

with $c < 0$. This means that $\mathbf{r}'$ and $\mathbf{y}$ are on opposite sides of the principal plane of the ellipsoid defined by points $\mathbf{x}$ such that $\mathbf{u}^\top\mathbf{x} = 0$. We argue that this is impossible.

Let $\mathbf{y}'$ be the point symmetrically opposite to $\mathbf{y}$, reflected across this plane. This point also lies on the boundary of the ellipsoid $C(\mathtt{G})$. Since $\mathbf{y}'$ is on the same side of the plane as $\mathbf{r}'$, we claim that $\mathbf{r}'$ is closer to $\mathbf{y}'$ than it is to $\mathbf{y}$. This means that $\mathbf{y}'$ lies closer than unit distance from $\mathbf{r}'$ and so the unit ball centred at $\mathbf{r}'$ is not contained completely inside the ellipsoid. This contradiction indicates that (35) must be true, and hence $\mathtt{A} - \mathtt{I}/\lambda$ is positive semi-definite, as required. This being so, it follows directly from (34) that also $(1 - \lambda) \geq 0$, as required.

This completes the proof of Lemma 4. The truth of Theorem 5 follows from Lemmas 2 and 5.

Remarks on the lower bound matrix. It was assumed throughout most of this paper that the errors in any particular are bounded within a circle of radius $\tau_{\max}$. This allows an easy derivation of a good maximal lower bound for the Hessians. The justification for this is that the region $\mathcal{D}$, formed as an intersection of cones, must project to points within the stated error bound. However, in reality the projection of the region $\mathcal{D}$ in each image may be significantly smaller, and not circular in shape. In this case, we could think of using the more general form of the maximal lower bound Hessians for each image given by Theorem 5. This would be expected to give better results, at the cost of greater computation. The improvement may, however, not be worth the extra computational effort.

7 Conclusion

The tests described here are extremely effective at verifying convexity, and hence global optimality of a local minimum for various multiple view geometry problems. The convexity test for the primary bound seems to do almost as well as α-bound. As the second bound is harder to compute, it is often advantegous to try the simple bound first. Experimental evaluations on several data sets showed the practical efficacy of the tests.

Failure of verifying optimality is quite rare. In the case of triangulation, the success rate is almost 100% and verification time is about 0.5ms per point. The main reason for failure is that the viewing rays are being parallel and the domain of interest becomes elongated, as illustrated in Fig 1. For larger problems with many residuals, the success rate decreases, and it is possible only to verify subproblems. We have been able to verify optimality for reconstructions with more than 1500 image points and 20 cameras.

This work has shown how to deal with a large class of multiple view geometry problems under the L_2 cost function. On the other hand, many geometric problems involve optimizations including rotations, for example, the estimation of two-view relative motion. Such problems cannot be handled with the presented techniques and are left as a challenge for future work.

Acknowledgement

This research was supported by (i) NICTA, a research centre funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy, (ii) the Australian Research Council through the ICT Centre of Excellence program, (iii) the Swedish Research Council (grant no. 2007-6476), (iv) the Swedish Foundation for Strategic Research (SSF) through the programmes Future Research Leaders and Wearable Visual Information Systems, and (v) the European Research Council (GlobalVision grant no. 209480).

References

1. R. Hartley and F. Kahl. Optimal algorithms in multiview geometry. In *Asian Conf. Computer Vision*, Tokyo, Japan, 2007.
2. R. Hartley and F. Schaffalitzky. L_∞ minimization in geometric reconstruction problems. In *Conf. Computer Vision and Pattern Recognition*, volume I, pages 504–509, Washington DC, USA, 2004.
3. R. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157, 1997.
4. R. I. Hartley. Chirality. *Int. Journal Computer Vision*, 26(1):41–61, 1998.
5. R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. Second Edition.
6. F. Kahl, S. Agarwal, M. K. Chandraker, D. J. Kriegman, and S. Belongie. Practical global optimization for multiview geometry. *Int. Journal Computer Vision*, 79(3):271–284, 2008.
7. F. Kahl and R. Hartley. Multiple view geometry under the L_∞-norm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(9):1603–1617, 2008.

8. Q. Ke and T. Kanade. Quasiconvex optimization for robust geometric reconstruction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 29(10):1834–1847, 2007.
9. H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.
10. F. Lu and R. Hartley. A fast optimal algorithm for L_2 triangulation. In *Asian Conf. Computer Vision*, volume 2, pages 279–288, November 2007.
11. D. Nistér, R. Hartley, and H. Stewénius. Using galois theory to prove structure from motion algorithms are optimal. In *Conf. Computer Vision and Pattern Recognition*, Minneapolis, USA, 2007.
12. D. Nistér, F. Kahl, and H. Stewénius. Structure from motion with missing data is NP-hard. In *Int. Conf. Computer Vision*, Rio de Janeiro, Brazil, 2007.
13. C. Olsson, M. Byröd, and F. Kahl. Globally optimal least squares solutions for quasiconvex 1d vision problems. In *Scandinavian Conf. on Image Analysis*, Oslo, Norway, 2009.
14. C. Rother and S. Carlsson. Linear multi view reconstruction and camera recovery using a reference plane. *Int. Journal Computer Vision*, 49(2/3):117–141, 2002.
15. N. Snavely, S.M Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3d. *ACM SIGGRAPH*, 25(3):835–846, 2006.
16. H. Stewenius, F. Schaffalitzky, and D. Nister. How hard is 3-view triangulation really? In *Proc. International Conference on Computer Vision*, pages 686 – 693, 2005.
17. A. Vedaldi, G. Guidi, and S. Soatto. Moving forward in structure from motion. In *Conf. Computer Vision and Pattern Recognition*, Minneapolis, USA, 2007.