An L_∞ Approach to Structure and Motion Problems in 1D-Vision

Kalle Åström, Olof Enquist, Carl Olsson, Fredrik Kahl
Centre for Mathematical Sciences, Lund University, Sweden
P.O. Box 118, SE-221 00 Lund, Sweden
(kalle,olofe,calle,fredrik)@maths.lth.se

Richard Hartley
Department of Systems Engineering, RSISE, Australian National University and NICTA
ACT 0200, AUSTRALIA
Richard.Hartley@anu.edu.au

Abstract

The structure and motion problem of multiple one-dimensional projections of a two-dimensional environment is studied. One-dimensional cameras have proven useful in several different applications, most prominently for autonomous guided vehicles, but also in ordinary vision for analysing planar motion and the projection of lines. Previous results on one-dimensional vision are limited to classifying and solving minimal cases, bundle adjustment for finding local minima to the structure and motion problem and linear algorithms based on algebraic cost functions.

In this paper, we present a method for finding the global minimum to the structure and motion problem using the max norm of reprojection errors. We show how the optimal solution can be computed efficiently using simple linear programming techniques. The algorithms have been tested on a variety of different scenarios, both real and synthetic, with good performance. In addition, we show how to solve the multiview triangulation problem, the camera pose problem and how to dualize the algorithm in the Carlsson duality sense, all within the same framework.

1. Introduction

Understanding of one-dimensional cameras is important in several applications. In [14] it was shown that the structure and motion problem using line features in the special case of affine cameras can be reduced to the structure and motion problem for points in one dimension less, i.e. one-dimensional cameras.

Another area of application is vision for planar motion. It has been shown that ordinary vision (two-dimensional cameras) can be reduced to that of one-dimensional cameras if the motion is planar, i.e. if the camera is rotating and translating in one specific plane only, cf. [5]. In another paper the planar motion is used for auto-calibration [1]. A typical example is the case where a camera is mounted on a vehicle that moves on a flat plane or a flat road, or a fixed camera viewing an object moving on a plane, e.g. in traffic scenarios.

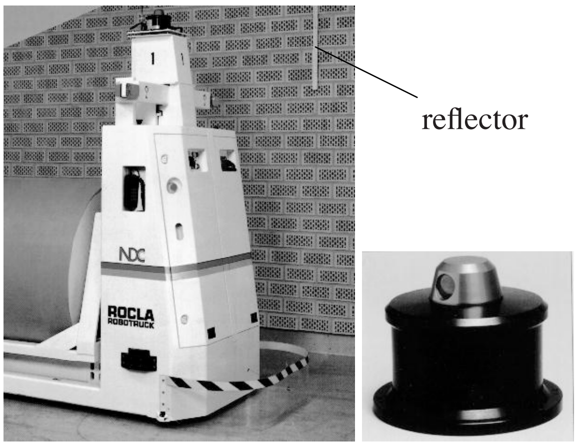

Figure 1. Left: A laser guided vehicle. Right: A laser scanner or angle meter.

A third motivation is that of autonomous guided vehicles, which are important components for factory automation. The navigation system uses strips of reflector tape, which are put on walls or objects along the route of the vehicle, cf. [7]. The laser scanner measures the direction from the vehicle to the beacons, but not the distance. This is the information used to calculate the position of the vehicle.

One of the key problems here is the structure and motion problem, also called simultaneous localisation and mapping (SLAM). This is the procedure of obtaining a map of the unknown positions of the beacons using images at unknown positions and orientations. This is usually done off-line, when the system is installed and then occasionally if there are changes in the environment. High accuracy is needed, since the precision of the map is critical for the performance of the navigation routines. In this article we present a method to find the globally optimal solution to this structure and motion problem.

Previous results concerning 1D projections of 2D include solving minimal cases without [13, 14, 2, 4] and with [12] missing data, autocalibration [5] critical configurations [3] and structure and motion systems in general [11].

The paper is organized as follows. In Section 2 we review the method L_∞ optimization. In Section 3 a brief introduction to the geometry of the problem is given. Section 4 discusses the problems of resection and intersection showing that they can be solved efficiently. An optimization method for the structure and motion problem is presented in Section 5 along with the required theoretic results. Finally, Section 6 presents some experiments illustrating the performance of the optimization method.

2. L_∞ optimization

Many geometrical problems can be formulated as optimization problems. Consider for instance the n-view triangulation problem with 2D-cameras. Statistically optimal estimates can be found by minimizing some error measure between the image data and the reprojected data. The usual choice of objective function is the L_2-norm of the reprojection errors, since this is the statistically optimal choice assuming independent Gaussian noise of equal and isotropic variance. Since closed form solutions are rarely available for these problems, they are often solved by iterative algorithms. The problem with this approach is that these methods often depend on good initialization to avoid local minima.

To resolve this problem L_∞ optimization was introduced in [6]. The idea is to minimize the maximal reprojection error instead of the L_2-norm. In [8], [9] it was shown that the optimization problems obtained for a number of multiview-geometry problems using the L_∞-norm are examples of quasiconvex problems. A function f is called quasiconvex if its sublevel sets $S_\mu(f) = \{x; f(x) \leq \mu\}$ are convex. The reason for using the L_∞-norm when dealing with quasiconvex functions is that quasiconvexity is preserved under the max operation. That is, if f_i are quasiconvex functions then $f(x) = \max_i f_i(x)$ is also a quasiconvex function. It was shown in [8], [9] that for a number of multiview geometry problems the (squared) reprojection errors are quasiconvex, and therefore the problem of minimizing the maximum reprojection error is a quasiconvex problem.

A useful property of quasiconvex functions is that checking whether there is an x such that $f(x) \leq \mu$ is a convex feasibility problem and can usually be solved efficiently. This gives a natural algorithm for minimizing a quasiconvex function. Suppose we have bounds μ_h and μ_l such that $\mu_l \leq \min_x f(x) \leq \mu_h$ then a bisection algorithm for solving $\min_x f(x)$ is

1. $\mu = \frac{\mu_h + \mu_l}{2}$.
2. If there exists x fulfilling $f(x) \leq \mu$ then $\mu_h = \mu$. Otherwise $\mu_l = \mu$.
3. If $\mu_h - \mu_l > \epsilon$ return to 1. Otherwise quit.

Although the L_∞-norm is not statistically optimal it has been shown to give almost as good solutions as the L_2-norm ([8], [9]). The only real weakness is that it is sensitive to outliers, but solutions to this problem has been presented in [15, 10].

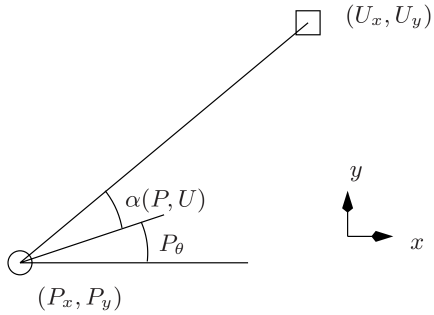

Figure 2. The figure illustrates the measured angle α as a function of scanner position (P_x, P_y), scanner orientation P_θ and beacon position (U_x, U_y).

3. Preliminaries

In one-dimensional vision only the bearing, α, of the beam from the camera to the object can be observed. In case of one-dimensional cameras this is measured using a laser scanner, and if the one-dimensional problem comes from ordinary vision, the bearing is calculated from higher-dimensional data. Only the bearing relative a fixed direction of the camera is measured so if the orientation of the camera is unknown, it has to be estimated as well.

We introduce an object coordinate system which will be held fixed with respect to the scene. The measured bearing of an object defined above, depends on the position of the object point (U_x, U_y) and the position (P_x, P_y) and orientation P_θ of the camera.

$$\alpha(P, U) = \arg(U_x - P_x + i(U_y - P_y)) - P_\theta \,, \quad (1)$$

where $\arg$ is the complex argument (the angle of the vector $(U_x - P_x, U_y - P_y)$ relative to the positive x-axis). The vector (P_x, P_y, P_θ) is called the **camera state**.

The same equation can be rephrased as

$$\lambda \underbrace{\begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix}}_{\mathbf{u}} = \underbrace{\begin{bmatrix} a & b & c \\ -b & a & d \end{bmatrix}}_{\mathbf{P}} \underbrace{\begin{bmatrix} U_x \\ U_y \\ 1 \end{bmatrix}}_{\mathbf{U}}, \tag{2}$$

where there is a one-to-one mapping between camera matrices $\mathbf{P}$ with variables (a, b, c, d) and camera states $P = (P_x, P_y, P_\theta)$. We will freely switch between these two representations and will use non boldface α, $P = (P_x, P_y, P_\theta)$ and $U = (U_x, U_y)$ for first representation and variables and boldface $\mathbf{u}$, $\mathbf{P}$ and $\mathbf{U}$ to denote image and object points in homogeneous coordinates and camera matrices.

Note that all bearings should be considered modulo 2π. Addition and subtraction are defined in the normal ways. When calculating absolute values the bearings should be represented with angles between $-\pi$ and π. Also note that two solutions (U_J, P_I) and $(\tilde{U}_J, \tilde{P}_I)$ to a problem are considered equal if they are related by a similarity transformation.

In this paper we are mostly interested in overdetermined problems, i.e. problems in which we have more measurements $\tilde{\alpha}$ than degrees of freedom. For such problems the equations $\alpha(P_I, U_J) = \tilde{\alpha}$ cannot be satisfied exactly. Instead we are forced to solve a minimization problem. Motivated by the previous section we choose to minimize the L_∞ norm of the error.

4. Intersection and resection

Before moving on to the structure and motion problem, which is the main subject of this paper, we consider the simpler problems of intersection and resection.

Consider a number of cameras seeing the same object. If the position and orientation of the cameras is known, the object is to determine the position of the object. This is called the intersection problem.

Problem 4.1. *Given bearings $\tilde{\alpha}_1, \ldots, \tilde{\alpha}_m$ from m different positions $P_1, \ldots, P_m$ the L_∞* **intersection problem** *is to find reconstructed point U such that*

$$f_i^\infty(U) = \max_I |\alpha(P_I, U) - \tilde{\alpha}_I| \tag{3}$$

is minimal.

If instead, the positions of a number of objects are known, the goal is to determine the position and orientation of the camera seeing those objects. This is the resection problem.

Problem 4.2. *Given n bearings $\tilde{\alpha}_1, \ldots, \tilde{\alpha}_n$ and points $U_1, \ldots, U_n$ the L_∞* **resection problem** *is to find the camera state P such that*

$$f_r^\infty(P) = \max_J |\alpha(P, U_J) - \tilde{\alpha}_J| \tag{4}$$

is minimal.

These two problems are in a sense easy to solve. We shall see that both of them can be formulated as quasiconvex problems.

Lemma 4.1. *The function*

$$f_i^\infty(U) = \max_I |\alpha(P_I, U) - \tilde{\alpha}_I|, \tag{5}$$

is quasiconvex on the set $\{\mathbf{U}; \mathbf{u}_I \cdot \mathbf{P}_I \mathbf{U} > 0,\ \forall I\}$, and the function

$$f_r^\infty(P) = \max_J |\alpha(P, U_J) - \tilde{\alpha}_J|. \tag{6}$$

is quasiconvex on the set $\{\mathbf{P}; \mathbf{u}_J \cdot \mathbf{P}\mathbf{U}_J > 0,\ \forall J\}$.

Proof. For given $\mathbf{U}$,$\mathbf{P}$ and corresponding $\mathbf{u}$ we have

$$\frac{\mathbf{u} \times \mathbf{PU}}{\mathbf{u} \cdot \mathbf{PU}} = \frac{|\mathbf{u}||(\mathbf{PU})| \sin(\alpha - \tilde{\alpha})}{|\mathbf{u}||(\mathbf{PU})| \cos(\alpha - \tilde{\alpha})} = \tan(\alpha - \tilde{\alpha}). \tag{7}$$

Here $\mathbf{a} \times \mathbf{b}$ denotes the scalar $a_1 b_2 - a_2 b_1$. Since $\mathbf{u} \cdot \mathbf{PU} > 0$, checking whether $|\alpha - \tilde{\alpha}| \leq \Delta$ is equivalent to

$$|\mathbf{u} \times \mathbf{PU}| \leq \tan(\Delta)(\mathbf{u} \cdot \mathbf{PU}) \tag{8}$$

In the intersection case $\mathbf{u}$ and $\mathbf{P}$ are known and in the resection case $\mathbf{u}$ and $\mathbf{U}$ are known. Therefore in both cases (8) constitute two linear equations. Hence the sublevel sets $\{\mathbf{U}; f^\infty(U) \leq \Delta\}$ and $\{\mathbf{P}; f^\infty(P) \leq \Delta\}$ are polyhedrons and thereby convex. ∎

Note that if we use the bisection algorithm, this result also tells us that the feasibility problems can be stated as linear programs.

5. Structure and motion

In the next problem we will assume that neither the positions of the objects or the positions and orientations of the cameras are known. However, we will still assume that the correspondence problem is solved, i.e. that it is known which measured bearings correspond to the same object. If the problem is is deduced from ordinary vision this correspondence can be decided using features in the two-dimensional image. In case of one-dimensional cameras the correspondence can be estimated with a RANSAC-type algorithm.

Problem 5.1. *Given n bearings from m different positions $\tilde{\alpha}_{I,J}$, $I = 1, \ldots, m$, $J = 1, \ldots, n$ the* $\boldsymbol{L_\infty}$ **structure and motion problem** *is to find the solution $z = (P_1, \ldots, P_m, U_1, \ldots, U_n)$ containing the the camera matrices $P_1, \ldots, P_m$ and the reconstructed points $U_1, \ldots, U_n$ such that*

$$f^\infty(z) = \max_{(I,J)} |\alpha(P_I, U_J) - \tilde{\alpha}_{I,J}|, \tag{9}$$

is minimal.

Unfortunately this problem does not have the same nice properties as the intersection and resection problems of the previous section. The reason is that when both P and U are unknown (8) are in general not convex conditions. Nonetheless, quasiconvexity will play an important role for this problem as well.

The basic idea of our optimization scheme is to first consider optimization with fixed camera orientations, and then use branch and bound over the space of possible orientations. A problem here is that, especially with many cameras, the set of possible orientations is large. A method to reduce this set, using linear conditions on the orientations is presented in Section 5.3.

5.1. Optimization with fixed orientations

It is useful to divide the parameter space $z = (P, U)$ into one part that correspond to the position of the cameras and points z_p and one part that correspond to the orientation of the cameras $z_\theta = (\theta_1, \ldots, \theta_m)$.

Definition 5.1. We define a function $d(z_\theta)$ as

$$d(z_\theta) = \min_{z_p} f^\infty(z_\theta, z_p). \tag{10}$$

Lemma 5.1. *The problem of verifying if*

$$d(z_\theta) = \min_{z_p} f^\infty(z) \leq \Delta \tag{11}$$

for $\Delta < \pi/2$ is a linear programming feasibility problem (and thus, the minimization over z_p is a quasiconvex problem).

Proof. If the orientations are fixed we can without loss of generality assume that orientations have been corrected for. Let

$$\tilde{\mathbf{u}} = \begin{bmatrix}\cos(\tilde{\alpha}) \\ \sin(\tilde{\alpha})\end{bmatrix}$$

be the measured angle represented as a normal vector u and let $\mathbf{U}_p = \begin{bmatrix}U_x & U_y\end{bmatrix}^T$ and $\mathbf{P}_p = \begin{bmatrix}P_x & P_y\end{bmatrix}^T$. Now

$$\frac{\mathbf{u} \times (\mathbf{U}_p - \mathbf{P}_p)}{\mathbf{u} \cdot (\mathbf{U}_p - \mathbf{P}_p)} = \frac{|\mathbf{u}||(\mathbf{U}_p - \mathbf{P}_p)| \sin(\alpha - \tilde{\alpha})}{|\mathbf{u}||(\mathbf{U}_p - \mathbf{P}_p)| \cos(\alpha - \tilde{\alpha})} = \tan(\alpha - \tilde{\alpha}).$$

The constraint that

$$|\alpha - \tilde{\alpha}| \leq \Delta$$

is equivalent to

$$|\mathbf{u} \times (\mathbf{U}_p - \mathbf{P}_p)| \leq \tan(\Delta)(\mathbf{u} \cdot (\mathbf{U}_p - \mathbf{P}_p)),$$

which constitutes two linear constraints in the unknowns. ∎

This means that we can use linear programming to determine if the minimal max norm is less than some certain bound Δ and using bisection, we can get a good estimation of the minimal max norm.

To get better convergence we modify the normal bisection algorithm slightly. The idea is to seek a solution to the problem in Lemma 5.1, that lies in the interior of the feasible space. For such a solution the max norm of the reprojection errors might be smaller than the current Δ, say $\Delta^\star$. Then one knows that the minimal max norm, $d(z_\theta)$ must be smaller than this $\Delta^\star$. To find such an interior solution we introduce a new variable k and try to maximize k under the constraints

$$|\mathbf{u} \times (\mathbf{U}_p - \mathbf{P}_p)| + k \leq \tan(\Delta)(\mathbf{u} \cdot (\mathbf{U}_p - \mathbf{P}_p)).$$

We can now present an algorithm for finding the minimal max norm for fixed orientations z_θ.

1. Check if there is a feasible solution with all reprojected errors less than $\pi/2$. This corresponds to $\tan(\Delta) = \infty$ in the equations above. This can be solved by a simpler linear programming feasibility test. Use only $(\mathbf{u} \cdot (\mathbf{U}_p - \mathbf{P}_p)) > 0$. If this is feasible continue, otherwise return $d_{min} > \pi/2$.
2. Let $\mu_l = 0$ and $\mu_h = \pi/2$ be lower and upper bounds on the minimal max error norm.
3. Set $\mu = (\mu_h + \mu_l)/2$. Test if $d(z_\theta) \leq \mu$. If this is feasible calculate $\mu^\star = f^\infty(z^\star)$ for the feasible solution $z^\star$ and set $\mu_h = \mu^\star$. Otherwise set $\mu_l = \mu$.
4. Iterate step 3 until $\mu_h - \mu_l$ is below a predefined threshold.

An example of how this function $d(z_\theta)$ may look like is shown in Figure 4.

5.2. Branch and bound over orientations

To get further, we need an idea of how the minimal maximum norm $d(z_\theta)$ depends on the camera orientations in z_θ. This is given by the following lemma.

Lemma 5.2. *The function $d(z_\theta)$ satisfies*

$$|d(z_\theta) - d(\bar{z}_\theta)| \leq |z_\theta - \bar{z}_\theta|_\infty, \tag{12}$$

which implies that it is Lipschitz continuous.

Proof. Recalling (1), we note that

$$\alpha_{I,J}(z) = \beta_{I,J}(z_p) - \theta_I. \tag{13}$$

We let z_p^* be the optimal camera and point positions corresponding to z_θ, so that $d(z_\theta) = f^\infty(z_\theta, z_p^*)$. Similarly, we define $\bar{z}_p^*$. Then

$$f^\infty(\bar{z}_\theta, z_p^*) - f^\infty(z_\theta, z_p^*) =$$

$$= \max_{(I,J)} |\alpha_{I,J}(\bar{z}_\theta, z_p^*) - \tilde{\alpha}_{I,J}| - \max_{(I,J)} |\alpha_{I,J}(z_\theta, z_p^*) - \tilde{\alpha}_{I,J}| \leq$$
$$\leq \max_{(I,J)} |\alpha_{I,J}(\bar{z}_\theta, z_p^*) - \alpha_{I,J}(z_\theta, z_p^*)| \leq$$
$$\leq \max_{I} |\theta_I - \bar{\theta}_I| = |z_\theta - \bar{z}_\theta|_\infty.$$

But

$$f^\infty(\bar{z}_\theta, \bar{z}_p^*) = \min_{\bar{z}_p} f^\infty(\bar{z}_\theta, \bar{z}_p) \leq f^\infty(\bar{z}_\theta, z_p^*)$$

so

$$f^\infty(\bar{z}_\theta, \bar{z}_p^*) - f^\infty(z_\theta, z_p^*) \leq |z_\theta - \bar{z}_\theta|_\infty.$$

After letting z and w switch places and repeating the argument, we can conclude

$$|d(z_\theta) - d(\bar{z}_\theta)| = |f^\infty(z_\theta, z_p^*) - f^\infty(\bar{z}_\theta, \bar{z}_p^*)| \leq |z_\theta - \bar{z}_\theta|_\infty.$$

■

Using the fact that the function $d(z_\theta)$ can be calculated and that it is never steeper than one, we will show how to solve globally for structure and motion. For the three view problem, this is done as follows.

First a candidate z_θ for the global minima is found with error d_{opt}. Then a quad-tree search is performed for the parameter space $z_\theta \in [0, 2\pi]^2$. At each level a square with center z_{mid} and width w is studied. The square cannot contain any points z with lower error function if $d(z_{mid}) > d_{opt} + w$, because of Lemma 5.2. This can be tested with a single linear programming feasibility test.

In fact it is sufficient to study the feasibility problem of the errors between the measured angles and the reprojected angles less than $d_{opt} + w$ for view 2 and 3 and less than d_{opt} for view 1.

Note that the algorithms work equally well for problems with missing data.

5.3. Linear conditions on z_θ

A problem with the branch and bound approach presented in the previous section is that when many cameras are used the set of possible orientations is large. To reduce this set, we consider the pairwise intersection of beams. This gives us linear constraints on the camera orientations.

Consider two cameras and a point which is visible in both cameras. Let β_j be the bearing of the beam from camera j to the point (in a global coordinate frame). For the two beams to intersect at the point we get a linear condition on the relative positions of the cameras (see Figure 3). We formulate this in the following lemma.

Lemma 5.3. *The bearings β_1 and β_2 of one point in two different cameras gives a condition on the bearing $\gamma_{1,2}$ of the beam from camera 1 to camera 2.*

$$\gamma_{1,2} \in [\beta_2 - \pi, \beta_1] \quad \text{if} \quad (\beta_2 - \beta_1) \in [0, \pi]$$
$$\gamma_{1,2} \in [\beta_1, \beta_2 - \pi] \quad \text{if} \quad (\beta_2 - \beta_1) \in [-\pi, 0]$$

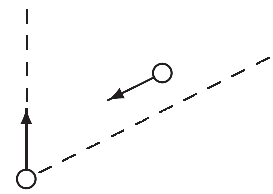

Figure 3. For the beams from the two cameras (circles) to intersect, the right camera has to lie between the dashed lines.

For each triplet of two cameras and one point, this lemma gives us a condition of the type

$$\gamma_{j,k} \in [\alpha_{j,m} + \theta_j - \pi, \alpha_{k,m} + \theta_m]$$

If we have N image points, all visible in all cameras, this gives us N linear constraints on each angle $\gamma_{j,k}$. For the problem to have a solution with zero error the intersection of these intervals must be non-empty. This implicitly defines a condition on the camera orientations. Since all equations are linear, finding the set of possible camera orientations is a linear problem that can be solved analytically. For the orientations where intersection is not possible, the same calculations give us the minimal error such that all beams intersect pairwise. Also note that since each inequality only involves two cameras, the complexity of the calculations increases only linearly with the number of cameras. In Figure 8 an illustration of the conditions on the camera orientations is given.

6. Experiments

Illustration of a typical three view problem

Study the problem of 3 views of 7 points with measured angles (in radians)

$$\alpha = \begin{pmatrix} 3.1 & -1.9 & -0.1 & -1.9 & -1.3 & 1.7 & -0.4 \\ -2.2 & -1.3 & -0.2 & -1.3 & -1 & 1.9 & -0.4 \\ 2.6 & -2.9 & -1.4 & -2.9 & -2.6 & 1.4 & -1.7 \end{pmatrix},$$

where rows denot different views I and columns denote different points J.

Initial estimates of the minimal solution shows that $d_{opt} \leq 0.05$. The first step of the quadtree (with a centre point at (π, π) and width 2π) is feasible for the bound $0.05 + \pi$. In the next two steps of the algorithm there are four and 16 regions respectively. None of these can be outruled. At the next level 60 out of 64 squares of width $\pi/4$ can be outruled.

We summarize the first 10 steps of the algorithm by describing (i) the number n_{sq} of feasible squares there are left at each level and (ii) how much area A out of the total area $A_{tot} = (2\pi)^2$ do these squares represent.

After 10 steps of the algorithm the optimal solution is bounded by $5.94 \leq \mathbf{P}_{2,\theta} \leq 5.98$ and $0.74 \leq \mathbf{P}_{3,\theta} \leq 0.86$.

step	n_{sq}	$\log(A/A_{tot})$
1	4	0
2	16	0
3	4	-1.2041
4	8	-1.5051
5	20	-1.7093
6	28	-2.1652
7	40	-2.6124
8	68	-2.984
9	104	-3.4015
10	92	-4.0568

In Figure 4 a plot of $d(z_\theta)$ for this problem is shown. An illustration of the convergence of the optimization is given in Figure 5.

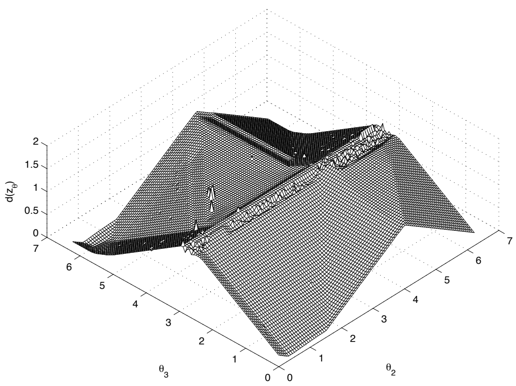

Figure 4. In the figure is shown the function $d_z(\theta)$ as a function of (θ_2, θ_3)-parameter space while keeping θ_1 fixed, for the data in the example with 3 views of seven points. Notice that the function is periodic so for this particular example there is only one local minimum.

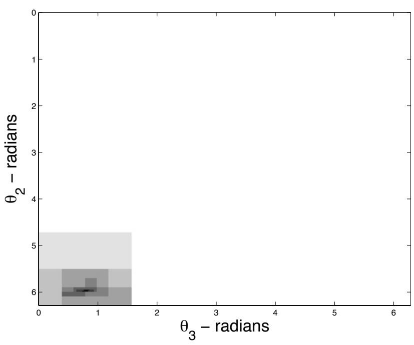

Figure 5. The quadtree map of the goal function in the (θ_2, θ_3)-parameter space where white corresponds to regions that are discarded early and darker areas correspond to regions that are kept longer in the quadtree branch and bound algorithm.

Performance on synthetic data

To illustrate the convergence of the optimization method, a number of synthetic examples were examined. Figure 6 shows how the feasible area decreases with each step of the algorithm and Figure 7 shows illustrations of typical, random, synthetic three-view examples with varying number of points. In certain cases there may be several local optima and even in underconstrained cases (meaning less equations than unknowns) one can locate the global minimum to a small region of parameter space due to all positive depth constraints.

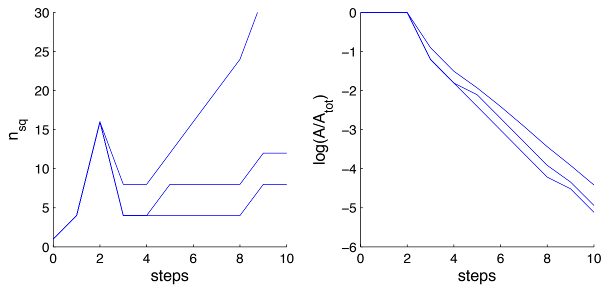

Figure 6. To illustrate the convergence of the algorithm, 100 random examples with 3 cameras and 20 points were constructed. The plots show the median and the first and ninth decile of the data. In the left plot the number of squares left after each step is shown and the right plot shows how much of the total area these squares constitutes.

Another underdetermined problem is showed in Figure 8. It shows clearly how the solution curve is cut off by the linear conditions of Section 5.3.

Hockey rink data with Cremona dual

It is possible to convert every structure and motion problem with m images of n points into a dual problem of $n-1$ images of $m+1$ points. We illustrate this with a subset of the data from a real set of measurements performed at a ice hockey rink. The set contains 70 images of 14 points. Here we studied a subset of 37 images of 4 points. Its dual consists of 3 images of 38 points. The global optimum to the L_∞ structure and motion problem is calculated for this set. The solution is shown in Figure 9. By forming the primal solution from this solution we get the solution for the original problem of 37 views of 4 points, also shown in Figure 9.

Hockey rink data

By combining optimal structure and motion with optimal resection and intersection it is possible to solve for many cameras and views. We illustrate this with the data from a real set of measurements performed at a ice hockey rink in 1991. The set contains 70 images of 14 points. The result is shown in Figure 10.

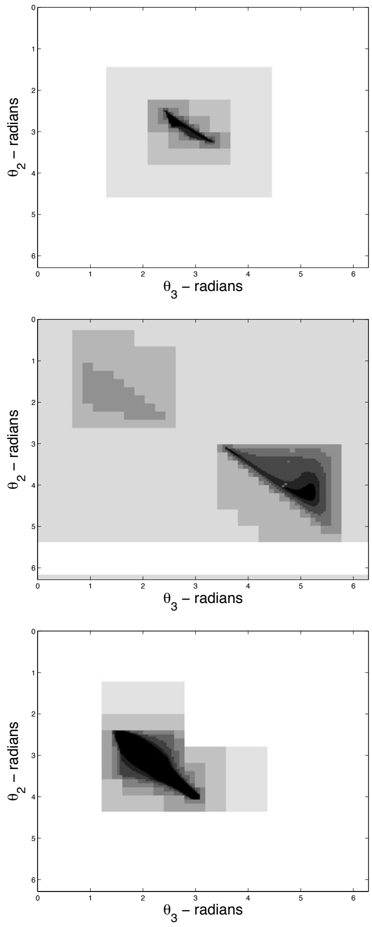

Figure 7. Illustrations of the evolution of the quadtree map. See caption of Figure 5 for explanation. Top: 4 points, 3 cameras (underconstrained case). Middle: 5 points, 3 cameras with two local optima. Bottom: 6 points, 3 cameras (overconstrained case). Note that even though the solution is underconstrained with 4 points, one can locate the global optimum in a small region of parameters space.

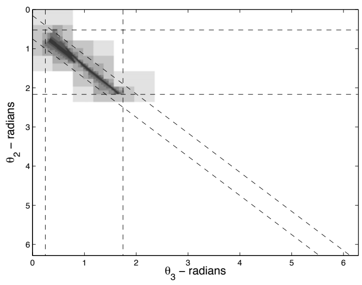

Figure 8. Illustration of the linear conditions of Section 5.3. Each pair of dashed lines shows a condition on the orientations of the cameras. Note how the solution curve is cut off by the linear conditions. (3 cameras and 4 points.)

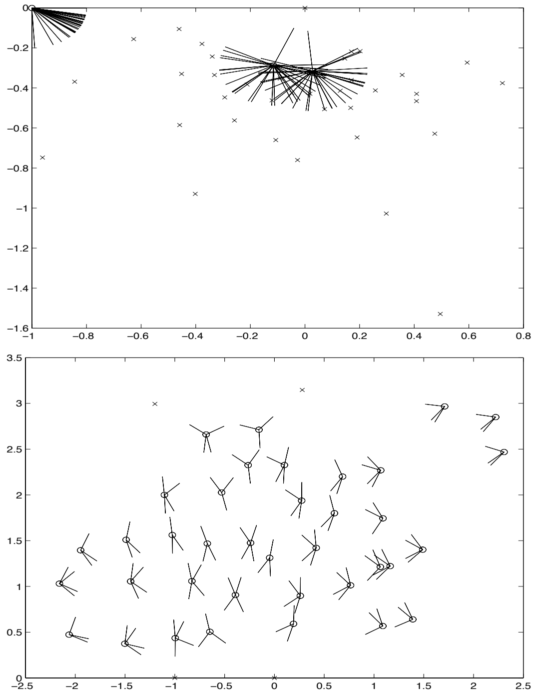

Figure 9. The global optimum to the structure and motion problem for the dual problem (top) and the primal problem (bottom)

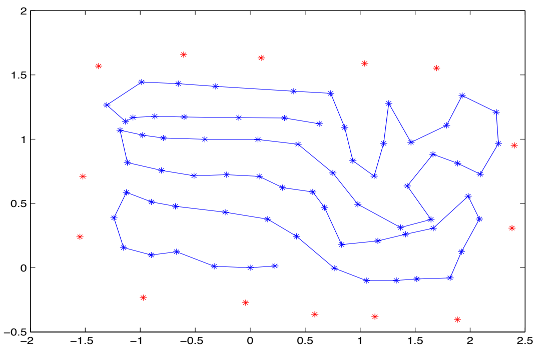

Figure 10. Calculated structure and motion for the icehockey experiment.

7. Conclusions

In this paper we have studied the problem of finding global minima to the structure and motion problem (SLAM, surveying) for one-dimensional retina cameras using the max-norm on reprojected angular errors. We have shown how the problem for known orientation can be reduced to a series of linear programming feasibility tests. We have also shown that the objective function as a function of orientation variables has slope less than one. This important observation gives us a way to search orientation space for the optimal solution, resulting in a globally optimal algorithm with good empirical performance.

Acknowledgments

This work has been supported by the European Commission's Sixth Framework Programme under grant no. 011838 as part of the Integrated Project SMErobot^{TM}, Swedish Foundation for Strategic Research (SSF) through the programme Vision in Cognitive Systems II (VISCOS II), Swedish Research Council through grants no. 2004-4579 'Image-Based Localisation and Recognition of Scenes' and no. 2005-3230 'Geometry of multi-camera systems'. Richard Hartley is with ANU and NICTA. NICTA is a research centre funded by the Australian Government's Department of Communications, Information Technology and the Arts and the Australian Research Council, through Backing Australia's Ability and the ICT Research Centre of Excellence programs

References

[1] M. Armstrong, A. Zisserman, and R. Hartley. Self-calibration from image triplets. In *Proc. 4th European Conf. on Computer Vision, Cambridge, UK*, pages 3–16. Springer-Verlag, 1996.

[2] K. Åström, A. Heyden, F. Kahl, and M. Oskarsson. Structure and motion from lines under affine projections. In *Proc. 7th Int. Conf. on Computer Vision, Kerkyra, Greece*, pages 285–292, 1999.

[3] K. Åström and F. Kahl. Ambiguous configurations for the 1d structure and motion problem. *Journal of Mathematical Imaging and Vision*, 18(2):191–203, 2003.

[4] K. Åström and M. Oskarsson. Solutions and ambiguities of the structure and motion problem for 1d retinal vision. *Journal of Mathematical Imaging and Vision*, 12(2):121–135, 2000.

[5] O. D. Faugeras, L. Quan, and P. Sturm. Self-calibration of a 1d projective camera and its application to the self-calibration of a 2d projective camera. In *Proc. 5th European Conf. on Computer Vision, Freiburg, Germany*, pages 36–52. Springer-Verlag, 1998.

[6] R. Hartley and F. Schaffalitzky. L_∞ minimization in geometric reconstruction problems. In *Proc. Conf. Computer Vision and Pattern Recognition, Washington DC*, pages 504–509, Washington DC, USA, 2004.

[7] K. Hyyppä. Optical navigation system using passive identical beacons. In L. O. Hertzberger and F. C. A. Groen, editors, *Intelligent Autonomous Systems, An International Conference, Amsterdam, The Netherlands, 8-11 December 1986*, pages 737–741. North-Holland, 1987.

[8] F. Kahl. Multiple view geometry and the L_∞-norm. In *International Conference on Computer Vision*, pages 1002–1009, Beijing, China, 2005.

[9] Q. Ke and T. Kanade. Quasiconvex optimization for robust geometric reconstruction. In *International Conference on Computer Vision*, pages 986 – 993, Beijing, China, 2005.

[10] H. Li. A practical algorithm for l_∞ triangulation with outliers. In *Proc. Conf. Computer Vision and Pattern Recognition*, Minneapolis, USA, 2007.

[11] M. Oskarsson and K. Åström. Automatic geometric reasoning in structure and motion estimation. *Pattern Recognition Letters*, 21(13-14):1105–1113, 2000.

[12] M. Oskarsson, K. Åström, and N. C. Overgaard. The minimal structure and motion problems with missing data for 1d retina vision. *Journal of Mathematical Imaging and Vision*, 26(3):327–343, 2006.

[13] L. Quan. Uncalibrated 1d projective camera and 3D affine reconstruction of lines. In *Proc. CVPR*, pages 60 – 65, 1997.

[14] L. Quan and T. Kanade. Affine structure from line correspondences with uncalibrated affine cameras. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(8):834–845, August 1997.

[15] K. Sim and R. Hartley. Removing outliers using the L_∞-norm. In *Proc. Conf. Computer Vision and Pattern Recognition*, pages 485–492, New York City, USA, 2006.