Outlier Removal Using Duality

Carl Olsson
Centre for Mathematical Sciences
Lund University, Sweden

Anders Eriksson
School of Computer Science
University of Adelaide, Australia

Richard Hartley
Australian National University
and NICTA, Australia

Abstract

In this paper we consider the problem of outlier removal for large scale multiview reconstruction problems. An efficient and very popular method for this task is RANSAC. However, as RANSAC only works on a subset of the images, mismatches in longer point tracks may go undetected. To deal with this problem we would like to have, as a post processing step to RANSAC, a method that works on the entire (or a larger) part of the sequence.

In this paper we consider two algorithms for doing this. The first one is related to a method by Sim & Hartley where a quasiconvex problem is solved repeatedly and the error residuals with the largest error is removed. Instead of solving a quasiconvex problem in each step we show that it is enough to solve a single LP or SOCP which yields a significant speedup. Using duality we show that the same theoretical result holds for our method. The second algorithm is a faster version of the first, and it is related to the popular method of L_1-optimization. While it is faster and works very well in practice, there is no theoretical guarantee of success. We show that these two methods are related through duality, and evaluate the methods on a number of data sets with promising results. [1]

1. Outliers in Multiple View Geometry

Geometric reconstruction is a core problem in computer vision. Typically, feature points are first extracted from the images and then matched to corresponding points in other images. The corresponding points are used to estimate the geometry. Given correct correspondences, accurate methods to estimate the geometry often exist. However, automatic matching usually results in a number of mismatches which gives rise to outliers in the data, often degrading the accuracy of the solution.

In this paper we develop methods for detecting these mismatches before computing the final reconstruction. The predominant approach for doing this is the well known method of RANSAC. However, as RANSAC only works on a subset of the images, mismatches in longer point tracks may go undetected. In order to address this problem, Sim and Hartley [16] gave an algorithm that considers the entire image sequence. In this paper we present an algorithm similar to their approach. We show that in each step we only need to solve a single LP which results in a significant speedup compared to [16].

In an effort to further increase the speed up we also consider a second algorithm, related to the popular L_1-relaxation approach. It has been shown [19, 7, 5] in the context of inverse linear problems that regularization with the L_1 norm often provides solutions that are close to the sparsest solution. Unfortunately whether this strategy is successful or not depends on the data. Strong apriori bounds can only be given for data drawn from special statistical distributions.

We relate the these two methods through duality, and our experimental results show that they both yield a significant drop in reprojection error.

1.1. Framework and Related Work

In multiview geometry the object is to estimate the structure of the scene and the camera motion form image projections. For example, let $z^i = (z_1^i, z_2^i)$ be a measurement in one of the images and Z be be its corresponding 3D-point. If the camera matrix is $P = [Q\ t]$ then the (squared) reprojection error can be written

$$R_i(Z,Q,t) = \left|\left|\left(z_1^i - \frac{Q_1 Z + t_1}{Q_3 Z + t_3}, z_2^i - \frac{Q_2 Z + t_2}{Q_3 Z + t_3}\right)\right|\right|^2, \tag{1}$$

where Q_j and t_j denotes the j'th row of Q and t respectively. In the general framework we consider the (squared) error residuals are of the form

$$R_i(\mathbf{x}) = \frac{(\mathbf{u}_i^T \mathbf{x} + \tilde{u}_i)^2 + (\mathbf{v}_i^T \mathbf{x} + \tilde{v}_i)^2}{(\mathbf{w}_i^T \mathbf{x} + \tilde{w}_i)^2}, \tag{2}$$

[1] Code available at http://www.maths.lth.se/matematiklth/personal/calle/

where $R_i : \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{w}_i^T\mathbf{x} + \tilde{w}_i > 0\} \mapsto \mathbb{R}_+$, $i = 1, ..., m$ and m is the number of measurements. The condition $\mathbf{w}_i^T\mathbf{x} + \tilde{w}_i > 0$ reflects the fact that the viewed 3D points should be located in front of the camera.

If either Z or Q is known then (1) can be expressed in the form of (2). Triangulation (Q and t are kept fixed) is an example of where the residuals are of this form. In this paper we will allow both the 3D-points and the positions of the cameras to vary simultaneously (that is, the parameters x in (2) will be the 3D-points and the camera translations). This is called the known rotation problem (see [10]).

Formally we state the outlier removal problem as finding a set of residuals I (preferably as large as possible) such that there is a solution $\mathbf{x}$ with all errors $R_i(\mathbf{x}) \leq \delta$. The constant δ depends on the particular noise model. In the case of Gaussian noise it is typically chosen to be 2 or 3 times the standard deviation of the noise.

Once the outliers have been removed, typically, one wants to solve either $\min_{\mathbf{x}\in C} \sum_i R_i(\mathbf{x})$, that is, the least squares solution, or

$$\min_{\mathbf{x}\in C} \max_i R_i(\mathbf{x}), \tag{3}$$

where $C = \{\mathbf{x} \mid \mathbf{w}_i^T\mathbf{x} + \tilde{w}_i > 0, \ \forall i\}$. The reason for minimizing the maximum residual error is that this formulation has some useful convexity properties (see [10]). This makes it possible to solve efficiently and reliably using methods from convex optimization. However in this work we are mainly interested in the outlier problem.

The work that is perhaps the closest to ours is that of Sim and Hartley [16]. In each step of their algorithm they solve the problem (3). If the the optimal value is larger than δ they discard all the measurements for which the error residuals attain the maximal error at the optimum (the support set), and solve (3) again. In general this procedure is not guaranteed to work, however, their main result is that for problems with residuals of the form (2) there will be at least one outlier among those measurements that attain the largest error. In each step of the algorithm we might, if we are unlucky, discard only one outlier while the rest are inliers. It was shown in [14] that the number of residuals that attain the largest value is in general less than $n + 1$ where n is the dimension of $\mathbf{x}$. Hence in order not to run out of inliers the ratio of inliers to outliers has to be roughly $n : 1$. In each step we also have to solve the min-max problem (3), using bisection (that is, a sequence of LP or SOCP). Although this is relatively fast we shall see that this is not necessary. We will refer to this algorithm as iterated L_∞ since the program (3) is solved repeatedly.

The second approach is related to L_1 approximation [19, 7, 5]. In the context of multiple view geometry Ke and Kanade [11] used a related approach. However, in their work they assume that the number of outliers is less than k, then formulate an integer program which is relaxed to a LP and used for checking if there is a solution where at least $m - k$ reprojection errors are less than or equal to δ. The linear program is then used in a bisection scheme over the parameter δ. If instead the number of outliers are determined by a fixed parameter δ, only a single LP needs to be solved. This formulation was considered in [15], however no theoretical results were given. In this paper we consider the same approach as [15]. Using duality we are able to show that the two algorithms are in fact related. In [6] an approach very close to that of [15] is considered. Conditions similar to those given by [5] where presented for the multiview geometry case. Unfortunately, checking whether these conditions are fulfilled turns out to be a very difficult problem in itself.

2. A Duality Approach

Before we consider the full outlier detection problem we will, for simplicity, consider a version that can be solved using linear programming. In this case the error residuals have the form

$$R_i(\mathbf{x}) = \frac{(\mathbf{u}_i^T\mathbf{x} + \tilde{u}_i)^2}{(\mathbf{w}_i^T\mathbf{x} + \tilde{w}_i)^2}. \tag{4}$$

Errors of this form occur for example in 1D-vision (see [2]), but also in 2D-vision, if instead of using the Euclidian distance ($|\mathbf{x} - \mathbf{y}|_2 = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$) we use the coordinate-wise max distance ($|\mathbf{x} - \mathbf{y}|_\infty = \max(|x_1 - y_1|, |x_2 - y_2|)$).

We start by selecting an outlier threshold δ, which will remain fixed throughout the outlier detection procedure. Since $\mathbf{w}_i^T\mathbf{x} + \tilde{w}_i > 0$ we see that there exists a solution without outliers if and only if there exists $\mathbf{x}$ satisfying

$$|\mathbf{u}_i^T\mathbf{x} + \tilde{u}_i| \leq \sqrt{\delta}(\mathbf{w}_i^T\mathbf{x} + \tilde{w}_i), \ \forall i. \tag{5}$$

This constraint can be realized as two linear constraints for each i. Therefore we may write this in compact matrix form as a feasibility problem: does there exist $\mathbf{x}$ such that $A\mathbf{x} \leq \mathbf{b}$, where A is a $2m \times n$ matrix and $\mathbf{b}$ is a $2m \times 1$ vector, both depending on the outlier threshold, δ. This question may be answered by solving the linear programming problem

$$\min_{\mathbf{x},s} \quad s \tag{6}$$
$$\text{s.t.} \quad A\mathbf{x} \leq \mathbf{b} + \mathbb{1}s, \tag{7}$$

where $\mathbb{1}$ is a vector of all ones. There is an outlier-free solution $\mathbf{x}$ if and only if there is a solution to (7) with $s \leq 0$.

2.1. Using the Dual Problem for Outlier Removal

The dual (see [4]) of the LP problem (7) is given by

$$\max_{\mathbf{y}} \quad -\mathbf{b}^T\mathbf{y} \tag{8}$$
$$\text{s.t.} \quad A^T\mathbf{y} = 0 \tag{9}$$
$$\mathbb{1}^T\mathbf{y} = 1 \tag{10}$$
$$\mathbf{y} \geq 0. \tag{11}$$

Since the primal problem is always feasible, the optimal value of the primal problem (6)-(7) and the dual program (8)-(11) are the same. Hence we may conclude that the solution to (8)-(11) satisfies $-\mathbf{b}^T\mathbf{y} > 0$ precisely when the solution to the problem (6)-(7) is positive; i.e. when there exist outliers among the set of residuals.

What makes the dual appropriate for outlier removal is the observation that the dual variables $\mathbf{y}$ can be used for finding the outliers. To see this suppose that $\mathbf{y}$ is a solution of (8)-(11) with objective value $-\mathbf{b}^T\mathbf{y} > 0$. For the nonzero elements of $\mathbf{y}$ we have

$$\sum_{i;y_i \neq 0} y_i(b_i - A_i\mathbf{x}) = \mathbf{y}^T(\mathbf{b} - A\mathbf{x}) = \mathbf{b}^T\mathbf{y} < 0 \qquad (12)$$

Here b_i and A_i denote the i-th entry of $\mathbf{b}$ and row of A respectively. Now, since $y_i \geq 0$ we may conclude that there must be at least one $A_i\mathbf{x} > b_i$. Therefore no inlier set (and in particular a maximal inlier set) can contain all of the residuals i for which y_i is non-zero, at least one is an outlier.

Finally, to show that $\mathbf{y}$ will only have $n+1$ nonzero variables we recall a result from linear programming.

Lemma 1. [3] Let M be an $l \times k$ matrix where $l < k$. Then the extreme points of the simplex $\{\mathbf{y} \mid M\mathbf{y} = \mathbf{r},\ \mathbf{y} \geq 0\}$ have at most l nonzero entries.

See [3] for a proof. For our problem the matrix M will be

$$M = \begin{pmatrix} A^T \\ \mathbb{1}^T \end{pmatrix} \qquad (13)$$

which has size $(n+1) \times m$ and therefore the extreme points will have at most $n+1$ nonzero components. Since the optimal value is always attained in an extreme point the solution $\mathbf{y}$ will only have $n+1$ nonzero variables.

In summary the algorithm for outlier removal will be

Algorithm 1: Outlier Removal by Duality.

input : The set of all measurements I
output: A set of measurements for which $\exists \mathbf{x} \in \mathbb{R}^n$ with $R_i(\mathbf{x}) \leq \delta$.

repeat
    Construct A and $\mathbf{b}$ from I ;
    Compute $\mathbf{y}$ by solving (8)-(11) ;
    if $-\mathbf{b}^T\mathbf{y} > 0$ **then**
        Remove all measurements from I for which $y_i > 0$;
    end
until $-\mathbf{b}^T\mathbf{y} \leq 0$;

Note that when we are working with 2d-image points each measurement gives four linear constraints. Hence if any one of the dual variables corresponding to these four constraints are non-zero we will remove all of them.

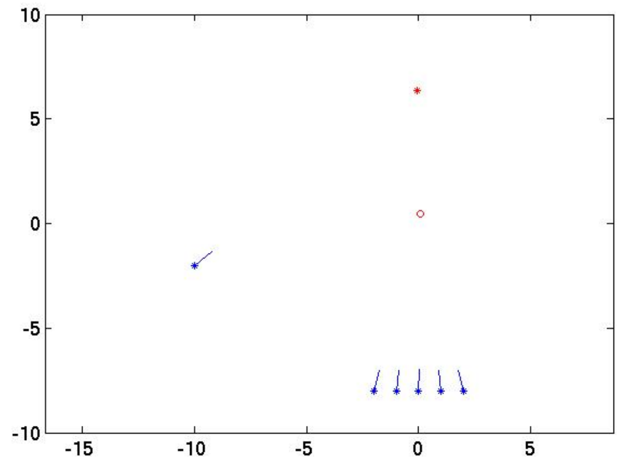

Figure 1. *An example where the L_1 method only removes inliers. *-solution from the L_1-method, o-solution from Algorithm 1*

2.2. Outlier Removal by L_1-minimization

When using the feasibility program (6)-(7) we add a single slack variable $\mathbf{s}$ and solve the feasibility problem repeatedly. It would however, be desirable to solve everything in a single convex program. Therefore instead of adding a single slack variable we add one for each residual and obtain the program

$$\min_{\mathbf{x},\mathbf{s}} \quad \mathbb{1}^T\mathbf{s} \qquad (14)$$
$$\text{s.t.} \quad A\mathbf{x} \leq \mathbf{b} + \mathbf{s} \qquad (15)$$
$$\mathbf{s} \geq 0, \qquad (16)$$

where $\mathbf{s}$ is a vector containing the slack variables. We will refer to this approach as the L_1 method as it is the L_1 norm of the vector $\mathbf{s}$ that is minimized. Ideally we would like to minimize the number of nonzero values in $\mathbf{s}$ however as this is a non-convex function this is not feasible. Instead we minimize the L_1 norm of $\mathbf{s}$. Although this results in a fast algorithm there are no guarantees of optimality. For example, it can be shown that the formulation (14)-(16) is not scale invariant. Choosing a different scale in one of the cameras is equivalent to reweighting the corresponding s_i which may cause problems.

Figure 1 shows a 1D-triangulation example where only inliers are discarded. Here we have placed 6 calibrated cameras viewing a 2D-point at the origin. To the 5 cameras placed on the line $y = -8$ we added normalized noise with standard deviation 1 degree. And to the 6th camera we added an error of 30 degrees to represent a mismatch. In this case the L_1-method removes 4 of the inlier cameras but not the outlier. Algorithm 1 removes the outlier and two inliers as it should ($n = 2$).

Using duality we may see that the L_1 method is connected to Algorithm 1. The dual problem of (14)-(16) is

$$\max_{\mathbf{y}} \quad -\mathbf{b}^T\mathbf{y} \tag{17}$$
$$\text{s.t.} \quad A^T\mathbf{y} = 0 \tag{18}$$
$$0 \leq \mathbf{y} \leq 1. \tag{19}$$

This program is very similar to (8)-(11). The only difference is that in (17)-(19) we do not require that the components of $\mathbf{y}$ should sum to one. Suppose that at each iteration of Algorithm 1 we obtain a dual solution $\mathbf{y}^k$ to (8)-(11). And let S_k be the set of residuals that are removed after iteration k. As we have seen the $\mathbf{y}^k$ identifies a set S_k of residuals for which there is no solution $\mathbf{x}$ with reprojection errors less than δ. (We may assume that all $\mathbf{y}^k$ have the same size by adding zeros corresponding to residuals that have been removed before iteration k.) Now it is easy to see that since $\mathbf{y}^k$ is feasible in (8)-(11) it is also feasible in (17)-(19) with $\mathbf{b}^T\mathbf{y} < 0$ (since there where outliers left at iteration k). For a feasible (primal) solution $(\mathbf{x}, \mathbf{s})$ of (14)-(16) we have

$$\sum_{i; y_i^k \neq 0} y_i^k (b_i + s_i - A_i\mathbf{x}) = \mathbf{b}^T\mathbf{y}^k + \sum_{i; y_i^k \neq 0} y_i^k s_i. \tag{20}$$

And since the sum on the left hand side is positive we get

$$\sum_{i; y_i^k \neq 0} y_i^k s_i \geq -\mathbf{b}^T\mathbf{y} > 0. \tag{21}$$

Since both y_i^k and s_i are positive we may conclude that at least one of the s_i will be nonzero. That is, the program (14)-(16) will always remove at least one residual from every set S_k.

Algorithm 2: Outlier Removal by L_1 minimization.

input : The set of all measurements I
output: A set of measurements for which $\exists \mathbf{x} \in \mathbb{R}^n$ with $R_i(\mathbf{x}) \leq \delta$.

```
begin
    Construct A and b ;
    Solve the program (14)-(16).;
    Remove the residuals for which s_i > 0.;
end
```

3. An SOCP version

Next, we consider residuals of the form in (2). In terms of a fixed inlier threshold δ, we write

$$A_i = \begin{pmatrix} \mathbf{u}_i^T \\ \mathbf{v}_i^T \end{pmatrix}, \quad \mathbf{b}_i = \begin{pmatrix} \tilde{u}_i \\ \tilde{v}_i \end{pmatrix},$$

$\mathbf{c}_i = \delta\mathbf{w}_i$ and $d_i = \delta\tilde{w}_i$. The data is outlier-free in terms of residuals (2) if and only if there exists an $\mathbf{x}$ such that

$$||A_i\mathbf{x} + \mathbf{b}_i||_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i, \ \forall i. \tag{22}$$

Similarly to the LP case, this can be tested by solving the second-order cone program (SOCP)

$$\min_{s,\mathbf{x}} \quad s \tag{23}$$
$$\text{s.t.} \quad ||A_i\mathbf{x} + \mathbf{b}_i||_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i + s, \ \forall i \tag{24}$$

The data is outlier-free if and only if the solution to this problem satisfies $s \leq 0$.

3.1. Using the Dual Variables for Outlier Removal

The dual of problem (23)–(24) is

$$\max_{\mathbf{y}_1,\ldots,\mathbf{y}_m, z_1,\ldots,z_m} -\sum_{i=1}^{m} \left(\mathbf{b}_i^T\mathbf{y}_i + d_i z_i\right) \tag{25}$$

$$\text{s.t.} \quad \sum_i (A_i^T\mathbf{y}_i + \mathbf{c}_i z_i) = 0 \tag{26}$$

$$||\mathbf{y}_i||_2 \leq z_i, \ \forall i = 1, \ldots, m \tag{27}$$

$$\sum_{i=1}^{m} z_i = 1. \tag{28}$$

where each $\mathbf{y}_i$ is a 2-vector.

Now let $(\mathbf{y}_i, \mathbf{z})$ be the optimal solution of the dual problem (25)–(28) and assume that there exists some $\mathbf{x}$ that satisfies the constraints (22) for all i such that z_i is nonzero. Then for all such i, we have

$$0 \leq ||A_i\mathbf{x} + \mathbf{b}_i||_2 \leq \mathbf{c}_i^T\mathbf{x} + d_i \tag{29}$$

and

$$0 \leq ||\mathbf{y}_i||_2 \leq z_i. \tag{30}$$

The Cauchy-Schwartz inequality then gives

$$\sum_{i; z_i \neq 0} \left(\mathbf{y}_i^T(A_i\mathbf{x} + \mathbf{b}_i) + z_i(\mathbf{c}_i^T\mathbf{x} + d_i)\right) \geq 0. \tag{31}$$

The sum on the left may be extended to all $i = 1, \ldots, m$, since the other terms for which $z_i = 0$ will vanish, because of (27). Furthermore,

$$\sum_{i=1}^{m} \mathbf{y}_i^T(A_i\mathbf{x} + \mathbf{b}_i) + z_i(\mathbf{c}_i^T\mathbf{x} + d_i) = \sum_{i=1}^{m} \mathbf{b}_i^T\mathbf{y}_i + d_i z_i.$$

because of (26). Hence the optimal solution satisfies $\sum_{i=1}^{m} \left(\mathbf{b}_i^T\mathbf{y}_i + d_i z_i\right) \geq 0$. If on the other hand the optimal solution satisfies $\sum_{i=1}^{m} \left(\mathbf{b}_i^T\mathbf{y}_i + d_i z_i\right) < 0$, we conclude that there is no $\mathbf{x}$ that satisfies these constraints, and hence the set $\{i \mid z_i \neq 0\}$ contains at least one outlier.

Since there is no duality gap for SOCP problems (see [4]), the converse is also true, namely that if the solution of the dual problem satisfies $\sum_i \left(\mathbf{b}_i^T\mathbf{y}_i + d_i z_i\right) \geq 0$, there is an $\mathbf{x}$ satisfying the constraints (22), and hence the data is outlier-free.

Figure 2. *Left: A pair of images of a road warning sign with candidate matches found using SIFT descriptors. Middle: Resulting homography estimated using the dual SOCP approach. Right: Histogram of the number of measurements removed at each iteration using LP and SOCP formulations.*

	Alg. 1 LP	Alg. 2 LP	Alg. 1 SOCP	Alg. 2 SOCP
Residuals removed	212.98	303.67	213.93	318.32
Remaining residuals	447.02	356.33	446.07	341.68
Number of LP/SOCP solved	28.96	1	27.22	1
Execution time (s)	5.04	0.34	9.79	0.47

Table 1. *Synthetic problem. Averaging results from* 100 *repetitions using the four proposed formulations.*

3.2. The Number of Removed Residuals

We wish to establish the size of the set $\{i \mid z_i \neq 0\}$ in an optimal solution to the dual problem (25)-(28). Substituting $\mathbf{y}_i = \mathbf{w}_i z_i$ we obtain the equivalent problem

$$\begin{aligned} \max_{\mathbf{w}_i, z_i} \quad & -\sum_{i=1}^{m} \left(\mathbf{b}_i^T \mathbf{w}_i + d_i\right) z_i \\ \text{s.t.} \quad & \sum_i (A_i^T \mathbf{w}_i + \mathbf{c}_i) z_i = 0 \\ & ||\mathbf{w}_i||_2 \leq 1, \ \forall i \\ & z_i \geq 0, \ \forall i \ ; \ \sum_i z_i = 1. \end{aligned}$$

Clearly, the value of $\mathbf{z}$ appearing in an optimal solution of this problem is the same as for problem (25)-(28). Let $(\mathbf{w}_i^*, z_i^*); i = 1, \ldots, m$ be an optimal solution for the new problem. Substituting the values of $\mathbf{w}_i^*$ in (25)-(28), leads to a minimization problem in the remaining variables $\mathbf{z} = (z_1, \ldots, z_m)$ only, namely

$$\begin{aligned} \mathbf{z}^* \quad = \quad & \text{argmax}_{\mathbf{z}} -\sum_{i=1}^{m} \left(\mathbf{b}_i^T \mathbf{w}_i^* + d_i\right) z_i \\ \text{s.t.} \quad & \sum_i (A_i^T \mathbf{w}_i^* + \mathbf{c}_i) z_i = 0 \\ & z_i \geq 0, \ \forall i \ ; \ \sum_i z_i = 1 \end{aligned}$$

This is an LP program of the form given in Lemma 1, with $l = n + 1$ linear constraints. It follows that $z_i = 0$ for at most $n + 1$ values of i.

3.3. Outlier Removal by L_1-minimization

As in the LP case we can replace the single slack variable in (23)-(24) with one slack variable s_i for each residual. If we let $(\mathbf{y}^k, \mathbf{z}^k)$ be the sequence of dual solutions and $S_k = \{i \mid z_i^k \neq 0\}$ then the solution s will have at least one non-zero component in each set S_k. The proof is similar to the LP case and we will therefore omit it.

4. Experiments

In this section we describe a number of experiments in order to evaluate the proposed methods. Our main application of interest is large scale 3D-reconstruction. Here we use the known-rotation formulation for removing outliers. As these problems are very large we use the LP formulation since LPs are generally faster than SOCPs.

Before proceeding to the main application we verify the theory in Sections 2.1 and 3.1 on a smaller problem, namely homography estimation. To solve the LP and SOCP problems we use the freely available SeDuMi [18].

4.1. Homography Estimation

In order to compare the different methods, as well as to validate the theoretical results we first consider the homography estimation problem illustrated in figure 2. The goal is to find the planar part of the scene (the road sign) by estimating the dominating homography. SIFT descriptors were used according to [12], to find candidate matches across images. Note that here the outliers originate from mismatches but also from correctly matched points in the background. This resulted in 114 potential matches in both images. Without removing any points, fitting a homography using bundle adjustment results in a rms error of 18.70.

When running Algorithm 1, with $\delta = 5$ pixels, a sequence of 5 second order programs had to be solved and the method removed 35 points. The the rms-error (after bundle adjustment) was reduced to .37 pixels. The resulting homography can be seen in the middle of Figure 2.

We also generated 100 synthetic instances of this problem. First we randomly placed 600 points uniformly distributed within the unit box and on the plane $z = 0$. Pairs of cameras were created by randomly placing them on the unit sphere directed at the origin. The principal point and focal length were chosen so that the projection of the 3D-points typically result in an image of size about 100×100. Noise with standard deviation 2 pixels was then added, as well as a number of randomly generated outliers (10% of the points), uniformly distributed over the approximate size of each image. Figure 2 (right) shows a histogram of the number of points removed at each iteration by Algorithm 2, using both the LP and SOCP formulation. For homography estimation we have that $n = 8$ and therefore no more than 9 residuals will be removed in each iteration. Table 1 shows the averaged results of the proposed methods.

4.2. Structure and Motion Estimation

In this section we will present experiments on our main application of interest, namely full 3D-reconstruction. We will compare the two approaches: RANSAC followed by bundle adjustment versus RANSAC followed by additional outlier removal and bundle adjustment.

We will follow the approach of [13], where the additional outlier removal step is achieved using optimization over the structure and translational part of the cameras. Note that the rotations used when solving the known rotation problem are the initial estimates that are obtained from the RANSAC step. Even though these are not the optimal ones our experiments show that the reprojection error drops considerably when using this alternative.

First we consider the well known dinosaur sequence shown in Figure 3. The data set is available from
`http://www.robots.ox.ac.uk/~vgg/data.html`.
It contains projections of 4983 points in 36 cameras. Solving the known rotation problem (with the cameras provided) reveals that there is no solution where all the error residuals are less than roughly 20 pixels (where the cameras have size 720×576 pixels). We ran the algorithms from Section 2 using the known rotation formulation. For comparison we also ran the iterated L_∞ algorithm from [16]. In [16] the L_∞-problem is solved using bisection however in [1] it is shown that the most efficient method is the so called Gugat's algorithm [8]. Therefore we also ran Sim and Hartley's algorithm with this method.

When running the various algorithms there was a significant drop in the rms-errors indicating that the removed residuals are really outliers. When using all the measurements the rms-error was 1.281 pixels. After running the L_1-method we obtained an rms-error of 0.2831 pixels even though we have only removed 179 out of 16432 image points. In the reconstruction in Figure 3 one can see that there are still 3D-points that do not belong to the surface of the dinosaur. Even though they are probably mismatches they have low reprojection errors and hence cannot be discarded in this way. Table 2 shows the results of the various algorithms. In this example the L_1-version is faster and also removes fewer residuals than the others. Note that this is in contrast to the results for the homography estimation problem. We believe this due to the fact that the number of outliers is smaller here.

Note that the iterated L_∞-method does not give the same result when we use bisection and Gugat's algorithm. This is because the solution of the L_∞-problem is in general not unique. There are a number of 3D-points that are free to move in a small area in 3D-space without attaining the maximal error. Since the implementations differ between the two methods they often give different solutions, with the same maximal error.

In terms of execution times Gugat's method is much more efficient than solving the bisection problem. Note that in each step a good upper bound is readily avaliable from the previous iteration, which makes Gugat's method converge very fast, typically in less than 5 iterations. Still, solving a single linear program in each iteration, which our dual method does, proves more efficient than solving the quasiconvex problem in each iteration.

Next we present two more experiments on real data. In both cases we have used SIFT descriptors [12] to generate point correspondences. We then used RANSAC with the 8-point solver [9] to discard outliers and determine orientations between pairs of cameras. The set of inliers was chosen to be all points with a reprojection error less than 5 pixels, where the size of the images are 2592×3872 pixels. We also tried the five-point solver [17] with similar results.

The first sequence is a number of images of a house (see Figure 4). The sequence consists of 12 images and 12475 3D-points. Solving the known rotation problem after running RANSAC shows that there is no solution where all errors are less than 140 pixels. Running bundle adjustment without removal of any outliers gives an rms-error of 2.677 pixels. Running the L_1 method reduces this error to 0.6094 pixels by removing 149 of the 35470 image points. Table 3 shows the rest of the results of the various algorithms.

In our final experiment we use a number of images of a cathedral (see Figure 5). The sequence consists of 17 images and 16961 3D-points. Solving the known rotation problem reveals that there is no solution where all errors are less than 215 pixels, and the rms-error after bundle adjustment is 3.181 pixels. Running the L_1 method reduces this error to 0.8191 pixels by removing 492 of the 46045 image

	Iterated L_∞ (bisection)	Iterated L_∞ (Gugat's)	Alg. 1	Alg. 2
Residuals removed	706	853	783	179
Remaining residuals	15726	15579	15649	16253
Rms error (pixels)	0.3956	0.3221	0.2553	0.2831
Number of LP solved	332	102	18	1
Execution time (s)	3889	821	231	24

Table 2. *Results on the Dino experiment. The data set contains 36 cameras and 4983 3D-points which are visible in at least 2 images. The total number of projections (error residuals) is 16432. The rms-error for the solution with all residual included is 1.281 pixels.*

	Iterated L_∞ (bisection)	Iterated L_∞ (Gugat's)	Alg. 1	Alg. 2
Residuals removed	608	546	512	149
Remaining residuals	34862	34924	34958	35321
Rms error (pixels)	1.0003	0.8732	0.5854	0.6094
Number of LP solved	312	84	12	1
Execution time (s)	11627	2142	335	222

Table 3. *Results on the House experiment. The data set contains 12 cameras and 12475 3D-points which are visible in at least 2 images. The total number of projections (error residuals) is 35470. The rms-error for the solution with all residual included is 2.6774 pixels.*

	Iterated L_∞ (bisection)	Iterated L_∞ (Gugat's)	Alg. 1	Alg. 2
Residuals removed	1746	1486	1493	492
Remaining residuals	44300	44559	44552	45553
Rms error (pixels)	1.1932	1.0988	0.7758	0.8191
Number of LP solved	395	120	26	1
Execution time (s)	28818	5966	1566	450

Table 4. *Results on the Cathedral experiment. The data set contains 17 cameras and 16961 3D-points which are visible in at least 2 images. The total number of projections (error residuals) is 46045. The rms-error for the solution with all residual included is 3.1812 pixels.*

points. Table 4 shows the rest of the results of the various algorithms.

Acknowledgements

This work has been funded by the European Research Council (GlobalVision grant no. 209480), the Swedish Research Council (grant no. 2007-6476), the Swedish Foundation for Strategic Research (SSF) through the programme Future Research Leaders and the Australian Research Council's Discovery Projects funding scheme (project DP0988439).

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

[1] S. Agarwal, N. Snavely, and S. Seitz. Fast algorithms for L_∞ problems in multiview geometry. In *Conf. Computer Vision and Pattern Recognition*, Anchorage, USA, 2008.

[2] K. Åström, O. Enqvist, C. Olsson, and F. Kahl. An l-infinity approach to structure and motion problems in 1d-vision. In *Int. Conf. Computer Vision*, Rio de Janeiro, Brazil, 2007.

[3] M. Bazaraa, H. Sherali, and C. Shetty. *Nonlinear Programming, Theory and Algorithms.* Wiley, 1993.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] E. J. Candes and T. Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

[6] A. Dalalyan and R. Keriven. L1-penalized robust estimation for a class of inverse problems arising in multiview geometry. In *Annual Conference on Neural Information Processing Systems*, Vancouver, Canada, 2009.

[7] D. L. Donoho. For most large underdetermined systems of equations, the minimal l1-norm near-solution approximates the sparsest near-solution. *Communications on pure and applied mathematics*, 59(7):907–934, 2006.

[8] M. Gugat. A fast algorithm for a class of generalized fractional programs. *Managment Sci.*, 42:1493–1499, 1996.

[9] R. Hartley. In defence of the 8-point algorithm. In *Int. Conf. Computer Vision*, pages 1064–1070, Boston, USA, 1995.

[10] F. Kahl and R. Hartley. Multiple view geometry under the L_∞-norm. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 30(9):1603–1617, 2008.

[11] Q. Ke and T. Kanade. Quasiconvex optimization for robust geometric reconstruction. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 29(10):1834–1847, 2007.

[12] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal Computer Vision*, 2004.

[13] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *Conf. Computer Vision and Pattern Recognition*, Minneapolis, USA, 2007.

[14] C. Olsson, O. Enqvist, and F. Kahl. A polynomial-time bound for matching and registration with ouliers. In *Conf. Computer Vision and Pattern Recognition*, Anchorage, USA, 2008.

Figure 3. *Two images from the dinosaur sequence, and the resulting reconstruction.*

Figure 4. *Two images from the house sequence, and the resulting reconstruction.*

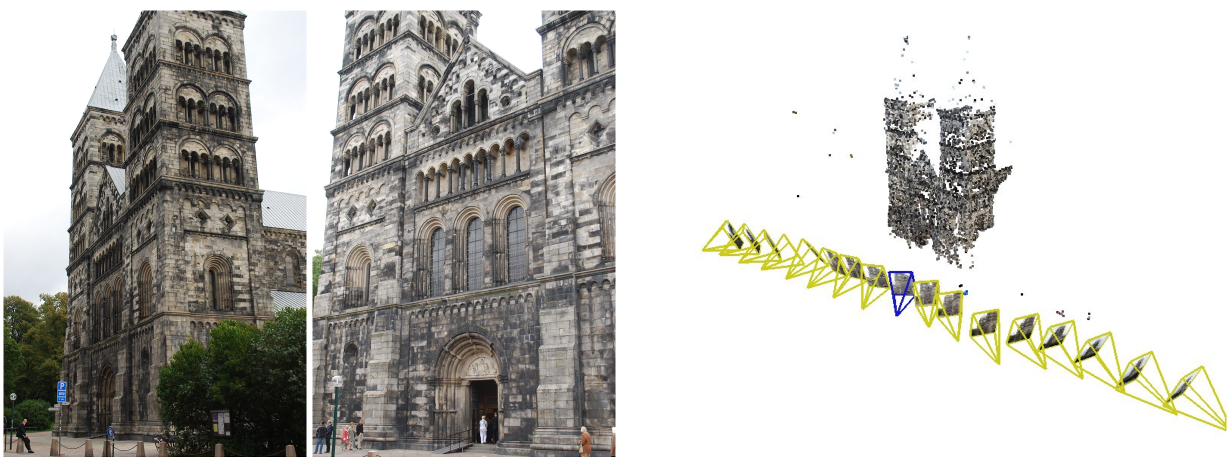

Figure 5. *Two images from the church sequence, and the resulting reconstruction.*

[15] Y. Seo, H. Lee, and S. W. Lee. Outlier removal by convex optimization for l-infinity approaches. In *PSIVT '09: Pacific Rim Symposium on Advances in Image and Video Technology*, 2009.

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

[17] H. Stewénius, C. Engels, and D. Nistér. Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 60:284–294, 2006.

[18] J. F. Sturm. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12:625–653, 1999.

[19] J. Tropp. Just relax: convex programming methods for identifying sparse signals in noise. *Information Theory, IEEE Transactions on*, 52(3):1030–1051, March 2006.