Robust Maximum Likelihood Estimation by Sparse Bundle Adjustment using the L_1 Norm

Zhijun Dai[1] Fengjun Zhang[1] Hongan Wang[1,2]
Intelligence Engineering Lab[1] State Key Lab of Computer Science[2]
Institute of Software, Chinese Academy of Sciences
zhijun@iscas.ac.cn {zfj,wha}@iel.iscas.ac.cn

Abstract

Sparse bundle adjustment is widely used in many computer vision applications. In this paper, we propose a method for performing bundle adjustments using the L_1 norm. After linearizing the mapping function in bundle adjustment on its first order, the kernel step is to compute the L_1 norm equations. Considering the sparsity of the Jacobian matrix in linearizing, we find two practical methods to solve the L_1 norm equations. The first one is an interior-point method, which transfer the original problem to a problem of solving a sequence of L_2 norm equations, and the second one is a decomposition method which uses the differentiability of linear programs and represents the optimal updating of parameters of 3D points by the updating variables of camera parameters. The experiments show that the method performs better for both synthetically generated and real data sets in the presence of outliers or Laplacian noise compared with the L_2 norm bundle adjustment, and the method is efficient among the state of the art L_1 minimization methods.

1. Introduction

Optimization using the L_1 norm is an increasingly important research area. We consider a nonlinear optimization problem

$$\min_P ||f(P) - u||, \tag{1}$$

where f is a nonlinear map $\mathbb{R}^N \to \mathbb{R}^M$ which transforms a set of parameters $P \in \mathbb{R}^N$ to a set of estimated measurements in the form of a vector $\bar{u} \in \mathbb{R}^M$.

In general, $|| \cdot ||$ can be any vector norm. The L_2 norm minimization corresponds to Maximum Likelihood (ML) estimate of P on the condition of the measurement noise following a Gaussian distribution.

The Gaussian model of noise distribution is appeared almost everywhere. However, different noise models may be equally justified. The chosen of noise models is application specific. This paper deals with the L_1 norm minimization which corresponds to ML estimate of P for the Laplacian distribution of the measurement noise.

In multiple view geometry [9], bundle adjustment [19] is employed as a final step to optimize camera parameters C and 3D reconstructed points U simultaneously through minimizing a nonlinear objective function which could be represented in the form of (1) in L_2 norm. Here P is (C, U), and measurements are observed image coordinates. Bundle adjustment is successful on the nonlinear L_2 norm optimization. We find that the processing flow of the bundle adjustment could also be incorporated into the nonlinear L_1 norm optimization.

2. Problem background

Bundle adjustment has been extensively researched for several decades using the L_2 norm. There is an outstanding survey [19] for this research. In recent years, some strategies for scaling up the bundle adjustment using the L_2 norm have been proposed [10, 1]. The objective function of this norm has a differentiable structure and then the fast converging Newton-like method is applicable for the optimization. Currently sparse bundle adjustment software for the L_2 norm, such as [13], is widely used. Also, there are many standard tools on the non-linear L_2 norm minimization.

Unfortunately, this is not the case with the L_1 norm despite it has been extensively researched in a wide range of fields including compressive sensing [5] and sparse coding [20]. The non-linear L_1 norm minimization is a tough problem due to the non-differentiability and non-convex. To address the non-differentiability, differentiable functions similar to the L_1 norm has been proposed such as the Huber-norm [12]. The Huber-norm is useful to approximate the L_1 norm gradually by reducing its width of quadratic part, but it could not simplify the L_1 minimization problem in nonlinear case. Furthermore, Iteratively Re-weighted Least-Squares (IRLS) algorithm has been proposed for computing any kind of norm iteratively, but the success of IRLS depends on the initial assigned weights.

In [15], they have proposed a method to do bundle adjustment using the L_∞ norm. One of the main drawbacks of the L_∞ norm is that the solution could be heavily influenced by outliers.

Exploring the current state of knowledge of bundle adjustment, there is no method on the bundle adjustment using the L_1 norm. Compared with the L_2 norm, the L_1 norm is robust to outliers and could get more accurate estimations corresponding to Laplacian distribution of measurement noise. Although there exists a variety of approaches to remove outliers, such as the classical RANSAC algorithm, these approaches usually need to preset some parameters to define the outlier. In this paper, we propose a method to make the L_1 sparse bundle adjustment problem tractable.

For the L_1 norm minimization, one of the most related works is the research of low-rank matrix approximations. Eriksson and Hengel have proposed a robust low-rank matrix approximations using the L_1 norm [7]. In this problem, low-rank matrices u and v are coupled bilinearly, and the optimum v could be parameterized by u (or parameterize the optimum u by v), fix u to get v is a *linear programming* problem and the authors gave a method to compute the partial derivative $\partial u/\partial v$. This method could be directly applied to the affine camera model. But for the projective camera model, the camera parameters C and 3D locations U are coupled in a non-linear form (and also not bilinear). Even for two cameras, the optimum U is parameterized by a root of six degrees of polynomial [8] (47 degrees for 3-view [18]). Considering the complex coupling between C and U, we linearize f in equation (1) and we parameterize the linearized form by both C and U.

After the linearization, the original problem is converted to a sequence of L_1 norm approximation problems. Fortunately, there are some standard algorithms [3, 2] to solve the L_1 norm approximation problem.

Also, the excellent work on finding a global optimal solution using L_1 or L_2 norm [11] needs mentioning. By formulating Lagrangian dual and applying *branch and bound* algorithm, the approach could get a provably global optimum. However, this approach currently are only used for solving triangulation or camera resection problem, and they are usually used for small problems in practice because of time complexity. We still lack globally optimal methods for both C and U. Thus there is an increasing research trend to find robust and efficient methods to get local optima.

2.1. Notation

All of the variables used in this letter are clearly defined or could be understood from the context. In the following, $\cdot^T$ denotes matrix transpose; Δ or δ with a subscript denotes an updating variable corresponding to the variable in the subscript position; $\mathrm{diag}(v)$ is a sparse matrix with diagonal entries of v; the symbol $\preceq$ denotes *vector inequality*.

3. Bundle adjustment using the L_1 norm

Given an initial value P_0, we want to find Δ_P to minimize

$$||f(P_0 + \Delta_P) - u||_1. \tag{2}$$

By linearizing $f(P_0 + \Delta_P)$ at the point P_0 through first order Taylor expansion, (2) is changed to

$$||J_f(P_0)\Delta_P + r_0||_1, \tag{3}$$

where $J_f(P_0)$ is the Jacobian matrix of f, and $r_0 = f(P_0) - u$. Next we propose to use two methods for solving (3). One is an interior point method which is efficient by taking the advantage of the sparse structure of the matrix J_f in subsection 3.1. Another one is a decomposition method in subsection 3.2.

3.1. Interior point method

An equivalent formulation of minimizing (3) is

$$\min_{\Delta_P} \quad 1^T s \tag{4}$$

$$\text{s.t.} \quad \underbrace{\begin{bmatrix} J_f(P_0) & -I \\ -J_f(P_0) & -I \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \Delta_P \\ s \end{bmatrix}}_{x} \preceq \underbrace{\begin{bmatrix} -r_0 \\ r_0 \end{bmatrix}}_{b}. \tag{5}$$

Interested readers for this derivation is referred to the chapter 6 in [3].

In the interior point method, (4)-(5) is minimized through consecutively minimizing a log-barrier function

$$\phi(x) = t(1^T s) - \sum_{i=1}^{2N} \log(b_i - A_i x_i), \tag{6}$$

where A_i or b_i means the ith row of the matrix A or b. The scalar t is used to weight the original objective function (4), and when $t \to \infty$ minimizing (6) equals (4)-(5). This objective function $\phi(x)$ could be minimized using Newton's method. A locally optimal step of the method is computed by solving

$$\begin{bmatrix} J_f^T D_1 J_f & J_f^T D_2 \\ D_2 J_f & D_1 \end{bmatrix} \begin{bmatrix} \delta_{\Delta_P} \\ \delta_s \end{bmatrix} = \begin{bmatrix} -J_f^T(c_1 - c_2) \\ -t1 + c_1 + c_2 \end{bmatrix}, \tag{7}$$

where $D_1 = \mathrm{diag}(d_1)$, $D_2 = \mathrm{diag}(d_2)$. The ith entry of c_1, c_2, d_1, or d_2 is computed as follows: $c_{1i} = 1/(s_i - (J_f\Delta_{Pi} + r_{0i}))$, $c_{2i} = 1/(s_i + (J_f\Delta_{Pi} + r_{0i}))$, $d_{1i} = c_{1i}^2 + c_{2i}^2$, and $d_{2i} = c_{2i}^2 - c_{1i}^2$. By using the block elimination technique, the main computational cost of (7) is reduced to solve

$$J_f^T \mathrm{diag}(d) J_f \delta_{\Delta_P} = -J_f^T g, \tag{8}$$

where d and g are vectors could be efficiently solved. In L_2 case of the sparse bundle adjustment, the main computational cost is on solving the L_2 norm equations which have a similar form of (8). Thus we could solve (8) efficiently by taking the advantage of the sparse banded structure of the matrix $J_f^T \operatorname{diag}(d) J_f$.

3.2. Decomposition method

In this subsection, we propose a method to minimize (3) by directly taking the advantage of the sparse structure of J_f.

Before moving on, we need to show the differentiation property of linear programming of standard form:

$$\min_{x\in\mathbb{R}^n} \quad c^T x \tag{9}$$
$$\text{s.t.} \quad Ax = b \tag{10}$$
$$x \succeq 0 \tag{11}$$

From [14] and [7], we have the following two theorems.

Theorem 3.1 *Given a linear program in canonical form such as (9)-(11), then if the problem is feasible there always exists an optimal basic solution.*

Theorem 3.2 *Let B be a unique optimal basis for (9)-(11), with minimizer x^* partitioned as $x^* = \begin{bmatrix} x_B^* \\ x_N^* \end{bmatrix}$. Where x_B^* is the optimal basic solution and x_N^* the optimal non-basic solution.*

Reordering the columns of A if necessary, there is a partition of A such that $A = [BN]$, and N are the columns of A associated with the non-basic variables N. Then x^* is differentiable at b with the partial derivatives given by

$$\frac{\partial x_B^*}{\partial b} = B^{-1} \tag{12}$$
$$\frac{\partial x_N^*}{\partial b} = 0. \tag{13}$$

Alternated linear programming (ALP) [12] is applicable to (3). For fixed Δ_C, optimize Δ_U independently and then fix Δ_U to optimize Δ_C. During optimizing Δ_U, the points are independent with each other, thus we could optimizing the location updating Δ_{U_i} of each point separately, and we could also optimize the updating Δ_{C_i} of each camera parameter separately for fixed Δ_U. Using ALP in this problem could convert the original large LP problem to a bunch of small LP problems. Each small problem could be solved efficiently, but it has the problem of slow convergence.

To speed up the convergence, we re-formulate (3) as

$$\min_{\Delta_C} \left\| J_f(P_0) \begin{bmatrix} \Delta_C \\ \Delta_U^*(\Delta_C) \end{bmatrix} + r_0 \right\|_1, \tag{14}$$

where

$$\Delta_U^*(\Delta_C) = \arg\min_{\Delta_U} \left\| J_f(P_0) \begin{bmatrix} \Delta_C \\ \Delta_U \end{bmatrix} + r_0 \right\|_1. \tag{15}$$

Let $\varphi(\Delta_C) = J_f(P_0)[\Delta_C^T, \Delta_U^*(\Delta_C)^T]^T$. To linearizing $\varphi(\Delta_C)$ we need to solve $\partial\Delta_U^*/\partial\Delta_C$. By the sparse structure of J_f, Δ_{U_i}s are independent with each other for fixed Δ_C, then we write the relation between Δ_{U_i} and Δ_C in (15) as

$$\Delta_{U_i}^*(\Delta_C) = \arg\min_{\Delta_{U_i}} \left\| [A_{C_i} A_{U_i}] \begin{bmatrix} \Delta_C \\ \Delta_{U_i} \end{bmatrix} + r_{U_i} \right\|_1, \tag{16}$$

where the entries of A_{C_i}, A_{U_i} corresponds to the entries of J_f related with Δ_C and Δ_{U_i} respectively and r_{U_i} is selected from r_0. Without losing generality, we assume that 3D point U_i are represented by 3 coordinates, the number of visible measurements for U_i under the imaging of all cameras is m, and the total number of camera parameters is n. Then we have $A_{C_i} \in \mathbb{R}^{m\times n}$, $A_{U_i} \in \mathbb{R}^{m\times 3}$

Let $\Delta_{U_i} = w^+ - w^-$, then we formulate (16) as a standard LP problem

$$\min_{w^+,w^-,r,s} \quad \begin{bmatrix} 0 & 0 & 1^T & 0 \end{bmatrix} \begin{bmatrix} w^+ \\ w^- \\ r \\ s \end{bmatrix} \tag{17}$$

$$\text{s.t.} \quad \underbrace{\begin{bmatrix} A_{U_i} & -A_{U_i} & -I & I \\ -A_{U_i} & A_{U_i} & -I & \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} w^+ \\ w^- \\ r \\ s \end{bmatrix}}_{x}$$
$$= \underbrace{\begin{bmatrix} -r_{U_i} - A_{C_i}\Delta_C \\ r_{U_i} + A_{C_i}\Delta_C \end{bmatrix}}_{b(\Delta_C)} \tag{18}$$
$$w^+, w^-, r, s \succeq 0 \tag{19}$$
$$w^+, w^- \in \mathbb{R}^3, r \in \mathbb{R}^m, s \in \mathbb{R}^{2m}. \tag{20}$$

Using simplex method, we get the basic optimal solution $x^* = [w^{*+T}, w^{*-T}, r^{*T}, t^{*T}]^T$. Then under the unique assumption of the solution, we use theorem 3.2 to get the differentiation on the basic variables

$$\frac{\partial x_B^*}{\partial \Delta_C} = \frac{\partial x_B^*}{\partial b}\frac{\partial b}{\partial \Delta_C} = B^{-1} \begin{bmatrix} -A_{C_i} \\ A_{C_i} \end{bmatrix}, \tag{21}$$

and the differentiation of non-basic variables is 0,

$$\frac{\partial x_N^*}{\partial \Delta_C} = 0. \tag{22}$$

Combining (21) and (22) we get

$$\frac{\partial \Delta_{U_i}^*(\Delta_C)}{\partial \Delta_C} = \frac{\partial w^{*+}}{\partial \Delta_C} - \frac{\partial w^{*-}}{\partial \Delta_C}. \tag{23}$$

Let

$$\varphi_i(\Delta_C) = [A_{C_i} A_{U_i}] \begin{bmatrix} \Delta_C \\ \Delta_{U_i} \end{bmatrix}, \tag{24}$$

then we compute Jacobian matrix of φ_i

$$J_{\varphi_i}(\Delta_C) = A_{C_i} + A_{U_i} \frac{\partial \Delta^*_{U_i}(\Delta_C)}{\partial \Delta_C}. \tag{25}$$

Use z to denote the total number of points U_i. Let $J_\varphi = [J^T_{\varphi_1}, ..., J^T_{\varphi_z}]^T$, then the updating δ_{Δ_C} on Δ_C is computed by

$$\min_{\delta_{\Delta_C}} \left\| J_\varphi(\Delta_C)\delta_{\Delta_C} + \begin{bmatrix} \varphi_1(\Delta_C) + r_{U_1} \\ \vdots \\ \varphi_z(\Delta_C) + r_{U_z} \end{bmatrix} \right\|_1. \tag{26}$$

M is the total number of measurements, and $J_\varphi \in \mathbb{R}^{M \times m}$. For a typical 3D reconstruction problem, M is usually very large, and using simplex method to solve (26) is very costly. Considering the interior-point algorithm, from (8) we know that the main computational steps are solving

$$J^T_\varphi \text{diag}(d) J_\varphi \delta_{\delta_{\Delta_C}} = -J^T_\varphi g, \tag{27}$$

where $J^T_\varphi \text{diag}(d) J_\varphi$ is a $m \times m$ symmetry matrix. If we have $m \ll M$ which is common for 3D reconstruction with less than hundreds of images, we could solve (27) efficiently.

3.3. Updating strategy

After solving the L_1 norm equations, now we are ready to discuss the strategy for updating P to reduce the L_1 norm error in (2). In [7], they proposed L_1-Wiberg algorithm which uses a trust region method. In their formulation for trust region, additional norm constraint is added. The constraint makes the original problem become more complex. We do not use that strategy due to we cannot find efficient solutions for the new formulation.

Let

$$\psi(P_0 + \Delta_P) = ||J_f(P_0)\Delta_P + r_0||_1. \tag{28}$$

Assume we find a Δ^*_P which makes

$$\psi(P_0 + \Delta^*_P) < \psi(P_0), \tag{29}$$

then according to the convexity of ψ, we have

$$\begin{aligned} \psi(P_0 + \alpha\Delta^*_P) &= \psi(\alpha(P_0 + \Delta^*_P) + (1-\alpha)P_0) \\ &\leq \alpha\psi(P_0 + \Delta^*_P) + (1-\alpha)\psi(P_0) \\ &< \alpha\psi(P_0) + (1-\alpha)\psi(P_0) \\ &= \psi(P_0) \qquad (30) \\ \text{s.t.} \quad & 0 < \alpha < 1, \qquad (31) \end{aligned}$$

we could choose α as small as possible, and $\psi(P_0 + \alpha\Delta^*_P) < \psi(P_0)$ always holds, thus the vector Δ^*_P is a descent direction of the function ψ at point P_0. The function ψ is a locally linear approximation of (2) at P_0. Thus we could always find a small enough α^* to satisfy

$$||f(P_0 + \alpha^*\Delta^*_P) - u||_1 < ||f(P_0) - u||_1. \tag{32}$$

By this well-behavior property, we finally propose our back-line tracking strategy on the updating of P in algorithm 1. For the step Δ_P that we solved from (3), we use a greedy line search method to get a step to minimize (2). If the step Δ_P could minimize (2), then we accept the update and continue for the next iteration, otherwise we step back on the line from $P_0 + \Delta_P$ to P_0 until we get a point that has less L_1 norm residual for (2).

Input: u, P_0, $\eta > 0$, and $\lambda > 1$
Output: P
$k \leftarrow 0$;
while *not converge* **do**
  Compute the Jacobian $J_f(P_k)$;
  Get Δ_{P_k} by solving the problem (3);
  $r \leftarrow f(P_k + \Delta_{P_k}) - u$;
  while $||r||_1 \geq ||r_k||_1$ **do**
    $\Delta_{P_k} \leftarrow \frac{\Delta_{P_k}}{\lambda}$;
    if $||\Delta_{P_k}||_1 < \eta$ **then**
      $converge \leftarrow True$;
      break the current loop;
    end
    $r \leftarrow f(P_k + \Delta_{P_k}) - u$;
  end
  $P_{k+1} \leftarrow P_k + \Delta_{P_k}$;
  $r_{k+1} \leftarrow r$;
  $k \leftarrow k + 1$;
end
return P_k;
/* In our current implementation, we set $\lambda = 2$, $\eta = 10^{-6}$. */

Algorithm 1: L_1 SBA (Sparse Bundle Adjustment) algorithm.

4. Experiments

In this section we report some experiments carried out to evaluate our method.

Currently, the algorithm is implemented with Matlab. Our current implementation on interior-point method is based on the ℓ_1 − MAGIC [4].

In our implementation on the bundle adjustment, we assume that the intrinsic matrix K of the cameras is calibrated and the same in all views. We simultaneously adjust

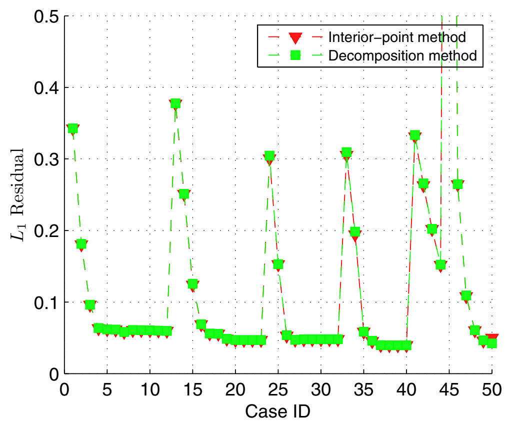


Figure 1. The residual curve of interior-point method and decomposition method for 50 cases in L_1 SBA.

six extrinsic parameters of each camera and 3D locations of points. For *Gauge freedom*, the first camera location and orientation are fixed, and the first coordinate of the first point is fixed as a scale of the reconstruction. In comparing with the result of the L_2 norm of bundle adjustment, we use the Levenberg-Marquardt (LM) algorithm [16].

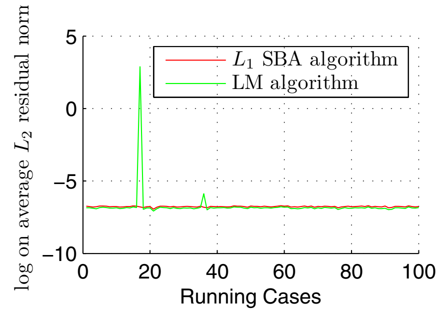
4.1. L_1 norm equations

To solve problem (3), we have proposed to use interior-point method or decomposition method to solve it. First we check whether the decomposition method will converge to the global optimum for convex problem (3). In the running of our methods, we choose 50 cases of Jacobian matrix and r_0 in the steps of our algorithm 1 and compare the residuals for the solutions of interior-point method and decomposition method. We plot the residual curve in figure 1. It shows that the two curves are almost coincide. This demonstrates the convergence property of our algorithm.

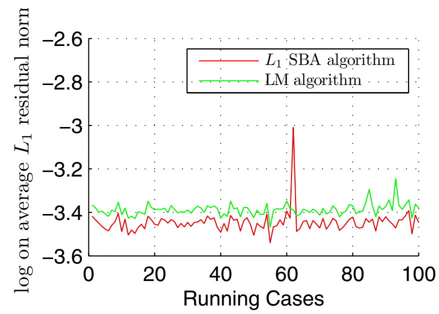
For the runtime of the two methods, our current implementation for the interior point method is much more efficient than the decomposition method, and about 10 times faster. In the following experiment, we run our L_1 SBA algorithm by using interior-point method to solve the linearized L_1 norm equations.

4.2. Simulated experiments

We randomly generate m views and n 3D points, where m is randomly chosen from 10 to 20, and n is randomly chosen from 100 to 300. The 3D points are located within a cube in front of all m views. For projected measurements, we randomly choose 30% of them as missing data. Initial estimates are given by adding random Gaussian noise on parameters which is used for generating projective measurements.



(a) Gaussian noise.



(b) Laplacian noise.

Figure 2. The log residual norm curve for L_1 SBA algorithm and LM algorithm.

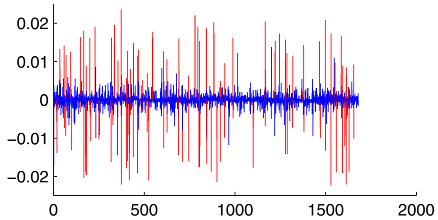
4.2.1 Gaussian noise

We have done experiments on 100 randomly generated data set with Gaussian noise on the measurements. The standard deviation of the noise is 0.025. The L_1 SBA algorithm achieves the comparable performance with LM bundle adjustment algorithm. Figure 2(a) shows the log function on the average L_2 residual norm.

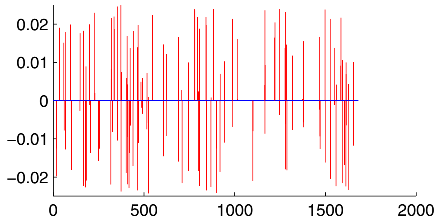
4.2.2 Laplacian noise

Similar experiments on 100 randomly generated dataset with Laplacian noise are also carried out. For the Laplacian distribution $f(x|\mu, b) = \frac{1}{2b}\exp\left(-\frac{|x-\mu|}{b}\right)$, we choose $b = 0.025$. The L_1 SBA algorithm achieves better performance compared with LM bundle adjustment algorithm. Figure 2(b) shows the log function on the average L_1 residual norm.

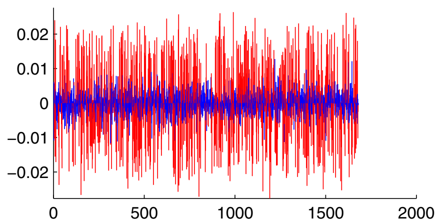
Note there are spikes in the two figures. Sometimes, the algorithms may converge to a bad local optimum, and this may happen for both algorithms. We have observed that the L_1 SBA algorithm and LM bundle adjustment algorithm have almost equal chance for converging to a bad local optimum.



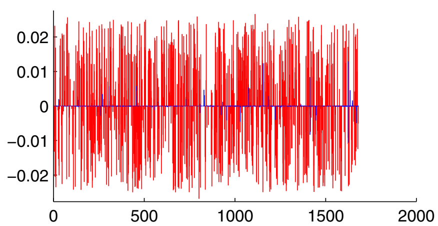
(a) LM algorithm with 5% outliers.



(b) L_1 SBA algorithm with 5% outliers.



(c) LM algorithm with 30% outliers.



(d) L_1 SBA algorithm with 30% outliers.

Figure 3. Residual graph. Outliers are marked with red color.

4.2.3 Outliers

We have carried out the experiment on the robustness against outliers. Here we show 2 noise-free running cases in which we added outliers 5% and 30% respectively. The outliers are drawn from Gaussian distribution with a large deviation. Figure 3 shows the residual graph, where the outliers are marked with red color. This experiment proved that L_1 SBA is much more robust against outliers, while in the L_2 norm case, the error is evenly distributed among inliers.

4.3. Experiments on real data sets

We report our experiment on two real scene data sets, the Cathedral of Lund and a House on Mårtenstroget in Lund which are public data sets from [17]. The cathedral data set consists of 17 cameras and 16,961 3D points projected into 46,045 image points, and the house data set consists of 12 cameras and 12,475 3D points projected into 35,470 image points. Figure 4 is an overview of the data sets.

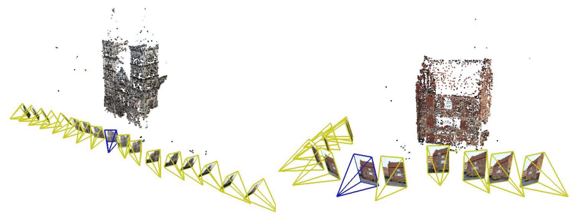


Figure 4. Image data set overview. The left scene shows the Cathedral data set and the right one shows the House data set

Initial translation estimates are obtained by L_∞ norm minimization for known rotation problem. For each iteration step of the algorithms, we record the L_1 norm residual and L_2 norm residual in each step, and plot the $\log$ residual downing curve for the result. Figure 5(a) and 5(b) are L_1 and L_2 norm residual curves for Cathedral data set.

From the figure 5(a) and 5(b), we could see that there is a tradeoff between L_1 and L_2 reprojection errors.

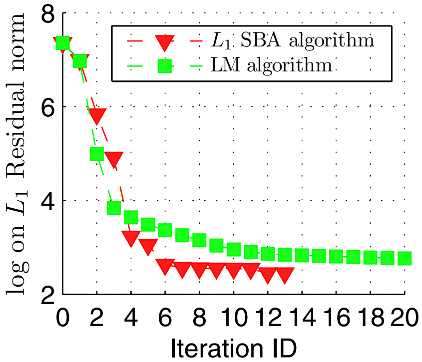
Figure 8(a) and 8(b) show the residual curves of the L_1 norm and L_2 norm for House data set. In this data set, we achieve better result and faster convergence rate both on the L_1 norm error and the L_2 norm error. This data set has more outliers than the previous data set, thus the better performance is achieved due to the robustness of the L_1 norm.

4.4. Running time

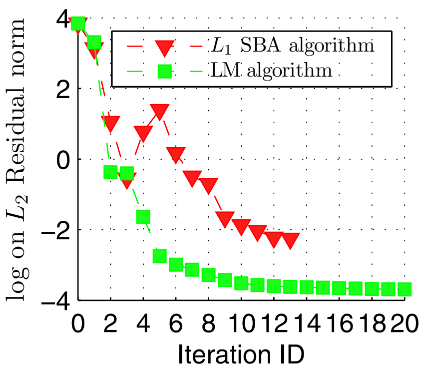
We use the standard Dino data set which is from `http://www.robots.ox.ac.uk/~vgg/data.html`. Figure 7 is an overview of the dataset. It consists of 36 images and 4986 3D points projected into 16432 image points.

We have carried out 100 different running cases using a commodity computer. Uniform random noise is added into the initial camera parameters and 3D points for each running case. Figure 8 shows the L_1 norm residuals and the L_2 norm residuals in a typical running case.

The average running time for the 100 cases is 125.8 seconds, and the average running time for an iteration is 12.1 seconds. Figure 9 shows the histogram of iterations of the 100 cases on this data set for L_1 SBA. The running time of our current implementation on L_1 SBA does not beat the efficiency of L_2 SBA. However, in the field of L_1 norm based optimization, our algorithm is significant efficient compared with the state of the art L_1-Wiberg algorithm [7]. The algorithm takes 17 minutes 44 seconds on a small Dino data set with 319 tracked points, while our algorithm takes only about 2 minutes for the data set which is ten times larger.

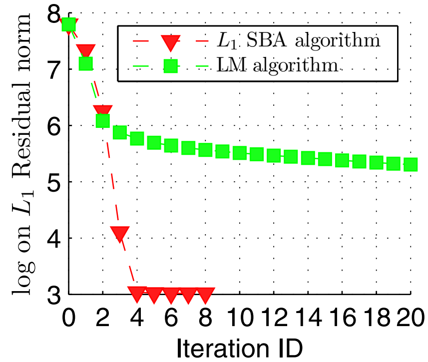


(a) L_1 norm of the residual.

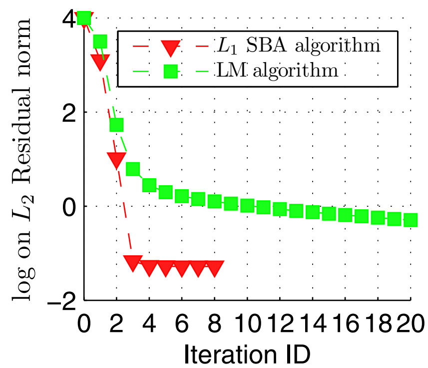


(b) L_2 norm of the residual.

Figure 5. The L_1 log residual norm curve of iterations for L_1 SBA algorithm and LM algorithm running on the Cathedral data set.



(a) L_1 norm of the residual.



(b) L_2 norm of the residual.

Figure 6. The L_1 log residual norm curve of iterations for L_1 SBA algorithm and LM algorithm running on the House dataset.

Here L_1 SBA could be applied to low rank matrix approximation directly. This is the power after L_1 *meets with bundle adjustment*.

5. Conclusion

We presented a new method for sparse bundle adjustment using the L_1 norm and demonstrated the effectiveness of the method in both simulated and real-world data applications. Future work in the area will focus on the strategies to speed up the L_1 SBA algorithms. There are following potential ways to do

- Parallel implementation for the decomposition method.
- From (30), we could know that any non-zero solution Δ_P^* to decrease ψ corresponding to a locally descent direction of the original objective function (1). Thus we could research on the fast sub-optimal solver for L_1 norm approximation.
- Find faster algorithms when the solution is sparse [6]. With the steps converging to a local minimum, the number of parameters that need to be updated may be small and the solution is sparse.

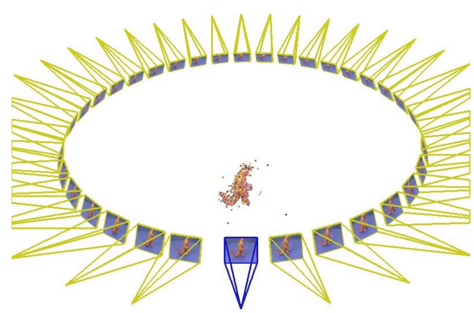
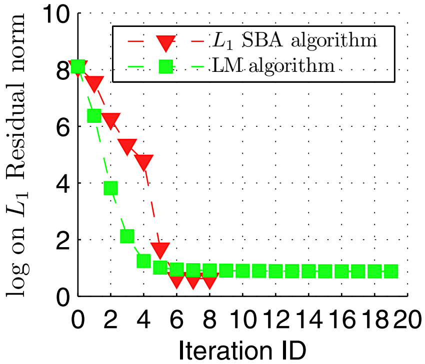


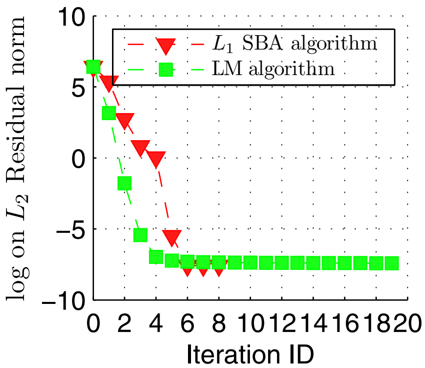
Figure 7. Dino data set overview.

Acknowledgments

We thank the valuable suggestions from Yihong Wu. We also thank the insightful comments from anonymous reviewers. This work was supported by the 973 program under



(a) L_1 norm of the residual.



(b) L_2 norm of the residual.

Figure 8. The L_1 log residual norm curve of iterations for L_1 SBA algorithm and LM algorithm running on the Dino dataset.

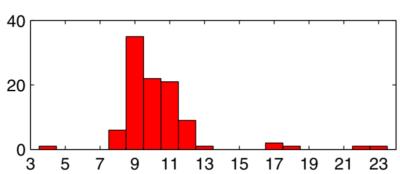


Figure 9. The histogram of iterations of 100 random running cases on the dino data set

grant No. 2009CB320804 and the National Science Foundation of China (61173059, 61135003).

References

[1] S. Agarwal, N. Snavely, S. Seitz, and R. Szeliski. Bundle adjustment in the large. In *ECCV*, pages 29–42, 2010. 1

[2] A. Bhusnurmath and C. Taylor. Graph Cuts via ℓ_1 Norm Minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10), 2008. 2

[3] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge Univ Pr, 2004. 2

[4] E. Candes and J. Romberg. ℓ_1-magic: Recovery of sparse signals via convex programming. *URL: www.acm.caltech.edu/l1magic/*. 4

[5] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006. 1

[6] D. Donoho and Y. Tsaig. Fast solution of ℓ_1-norm minimization problems when the solution may be sparse. *IEEE Transactions on Information Theory*, 2008. 7

[7] A. Eriksson and A. van den Hengel. Efficient Computation of Robust Low-Rank Matrix Approximations in the Presence of Missing Data using the L_1 Norm. In *CVPR*, pages 771–778, 2010. 2, 3, 4, 6

[8] R. Hartley and P. Sturm. Triangulation. In *Computer Analysis of Images and Patterns*, pages 190–197. Springer, 1995. 2

[9] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge Univ Pr, 2003. 1

[10] Y. Jeong, D. Nister, D. Steedly, R. Szeliski, and I. Kweon. Pushing the Envelope of Modern Methods for Bundle Adjustment. In *CVPR*, pages 1474–1481. IEEE, 2010. 1

[11] F. Kahl, S. Agarwal, M. Chandraker, D. Kriegman, and S. Belongie. Practical global optimization for multiview geometry. *International Journal of Computer Vision*, 79(3):271–284, 2008. 2

[12] Q. Ke and T. Kanade. Robust L_1 Norm Factorization in the Presence of Outliers and Missing Data by Alternative Convex Programming. In *CVPR*, volume 1, pages 739–746. IEEE, 2005. 1, 3

[13] M. A. Lourakis and A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software*, 36(1):1–30, 2009. 1

[14] D. Luenberger and Y. Ye. *Linear and nonlinear programming*. Springer Verlag, 2008. 3

[15] K. Mitra and R. Chellappa. A Scalable Projective Bundle Adjustment Algorithm using the L_∞ Norm. In *ICVGIP*, pages 79–86. IEEE, 2008. 2

[16] J. More. The Levenberg-Marquardt algorithm: implementation and theory. *Numerical analysis*, pages 105–116, 1978. 5

[17] C. Olsson, A. Eriksson, and R. Hartley. Outlier removal using duality. In *CVPR*, pages 1450–1457. IEEE, 2010. 6

[18] H. Stewenius, F. Schaffalitzky, and D. Nister. How hard is 3-view triangulation really? In *ICCV*, volume 1, pages 686–693. IEEE, 2005. 2

[19] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustmentła modern synthesis. *Vision algorithms: theory and practice*, pages 153–177, 2000. 1

[20] W. Vinje and J. Gallant. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, 287(5456):1273, 2000. 1