Pseudoconvex Proximal Splitting for L_∞ Problems in Multiview Geometry

Anders Eriksson
School of Computer Science
The University of Adelaide
anders.eriksson@adelaide.edu.au

Mats Isaksson
Centre for Intelligent Systems Research
Deakin University
mats.isaksson@gmail.com

Abstract

In this paper we study optimization methods for minimizing large-scale pseudoconvex L_∞ problems in multiview geometry. We present a novel algorithm for solving this class of problem based on proximal splitting methods. We provide a brief derivation of the proposed method along with a general convergence analysis. The resulting meta-algorithm requires very little effort in terms of implementation and instead makes use of existing advanced solvers for non-linear optimization. Preliminary experiments on a number of real image datasets indicate that the proposed method experimentally matches or outperforms current state-of-the-art solvers for this class of problems.

1. Introduction

In this paper we revisit the problem of pseudoconvex optimization problems in multiview geometry, with emphasis on solving large scale problems. It has recently been proven [17, 18] that certain types of geometric vision problems are in fact pseudoconvex when formulated using the L_∞ norm. This result can be used to eliminate the occurrence of locally optimal solutions to the reconstruction problem at hand, a common concern in most traditional, least-squares formulations. An additional application of the L_∞ formulation is also in its use in decting and removing outliers [16].

A number of different methods for efficiently solving the resulting L_∞ optimization problem have been proposed in recent years. With a few exceptions, most of these approaches are based on formulating the problem as a parameterised optimization problem. The solution to the original problem is then obtained by solving a sequence of convex feasibility problems. This class of solvers include the bisection approach of [10], the improved bisection algorithm of [17] and Gugat's algorithm for generalized fractional program presented in [1].

Alternative approaches for L_∞ minimization include the use of interior point solvers as suggested in [17] and the reduction scheme of [13]. The former however, was found to have issues with numerical instability and the latter is most suitable for low-dimensional problems. We are also aware of work on interior point methods developed specifically for solving the fractional optimization problems that appear in these formulations [6]. Even though such efforts have shown to obtain a certain amount of computational speedup, they do require the development of specialized and dedicated software packages. For the same reason as given by [1] we chose not cover efforts in this direction in this work but instead only focus on methods that are less cumbersome to implement and that can take advantage of existing mathematical programming tools.

A common property of all these methods are nonetheless that they all require separate and dedicated solvers. And more often than not, the solutions obtained through L_∞ optimization are typically passed on and refined in a least-squares formulation using bundle adjustment [25]. The present paper focuses on the problem of efficiently solving large-scale pseudoconvex L_∞ problems and one of the main contributions of this work is the proposal of a meta-algorithm, based on ideas from proximal splitting methods [5], that rely on existing least squares solvers[1]. As a consequence, in this framework no dedicated solver for the L_∞ problem is required and hence transitions over to an L_2 formulation can be done seamlessly.

We present this method in a general view for solving L_∞ problems in geometric vision along with some simple convergence proofs and strong optimality guarantees on the solutions. In addition, the numerical results we obtained by comparing our method against the state-of-the-art method of [1], indicate that this algorithm is highly competitive and appears to offer observable speedups in all of the experiments conducted.

This paper is organized as follows. In the next section, a brief review of L_∞ optimization is given, along with an introduction to Bregman iterations and a discussion on its connection to proximal splitting methods. This is then followed by a section containing the derivation of the proposed

[1] In this paper we used the publicly available package of [14]

algorithm along with a convergence analysis. Section 4 states the actual algorithm and discusses some practical issues in regards to its implementation. Finally, experimental results are given in section 5 followed by a concluding discussion.

2. Preliminaries

First we give some prerequisites needed for the derivation of our proposed algorithm. This includes a brief review of the use of the L_∞-norm in multiview geometry as well as an introduction to Bregman iterative algorithms.

2.1. The L_∞ Problem Formulation

The multiview geometry problems considered under this framework are ones where the residuals can be expressed as quotients of affine functions. If we let $r_i : \mathcal{C}_i \mapsto \mathbb{R}_+$ denote the L_p-norm of the i-th residual, then for this class of problems we can write

$$r_i(x) = ||(\frac{a_{i1}^T x + b_{i1}}{c_i^T x + d_i} - u_{1i},\ \frac{a_{i2}^T x + b_{i2}}{c_i^T x + d_i} - u_{2i})||_p, \quad i = 1, ..., N. \tag{1}$$

Here $a_{i1}, a_{i2}, c_i \in \mathbb{R}^n$ and $b_{i1}, b_{i2}, d_i, u_{1i}, u_{2i} \in \mathbb{R}$ and the set $\mathcal{C}_i = \{x \in \mathbb{R}^n | c_i^T x + d_i > 0\}$ represents the requirement that the reconstructed points lie in front of the cameras.

If we take N-view triangulation as an example, where the aim is that given N camera matrices, denoted $P_i \in \mathbb{R}^{3\times4}$, and as many image measurements $u_i \in \mathbb{R}^2$, to recover the coordinates of the corresponding 3D point. Here $x \in \mathbb{R}^3$ would represent the real world location of the point and $[a_{i1}^T\ b_{i1}]$, $[a_{i2}^T\ b_{i2}]$ and $[c_i^T\ d_i]$ would correspond to each of the rows of camera matrix i.

It has been shown that a large number of multiview problems, apart from triangulation, can be written in the above form. This includes camera resectioning, homography estimation as well as certain structure from motion estimation problems, see [12, 11].

The interesting property of the formulation (1) is that each residual r_i is a quasiconvex function of x. Quasiconvexity is not preserved under the L_2-norm, but it is under the L_∞ norm. This means that issues of local minima, which typically needs to be addressed when using the traditional L_2 norm, can be avoided and guarantees of global optimality can actually be achieved when instead using the latter norm. Hence, any local minimizer of the problem

$$\min_{x \in \mathcal{C}} \max_i r_i(x) \tag{2}$$

is also a global minimizer. For a further discussion on the different convexity properties of L_∞ problems we refer to [18].

These properties hold, for any choice of $p \geq 1$ but will result in slightly different formulations of the particular subproblems that needs to be solved. Setting $p = 2$ leads to the SOCP formulation used in [11] and $p = 1$ or $p = \infty$ to the LP formulations of [20] and [16] respectively.

To simplify notation it will prove convenient to introduce the following perspective mapping $\Pi : \mathbb{R}^n \mapsto \mathbb{R}^{N\times2}$, defined as

$$\Pi(x) = \left[\ \Pi_1^T \quad ... \quad \Pi_N^T\ \right]^T,\ i = 1, ..., N \tag{3}$$

$$\Pi_i(x) = \left[\ \frac{a_{i1}^T x + b_{i1}}{c_i^T x + d_i}, \quad \frac{a_{i2}^T x + b_{i2}}{c_i^T x + d_i}\ \right]. \tag{4}$$

This allows us to write (2) as

$$\min_{x \in \mathcal{C}} ||\Pi(x) - u||_{p,\infty}, \tag{5}$$

where $|| \cdot ||_{p,\infty}$ denotes the mixed matrix norm

$$||A||_{p,\infty} = \max_i ||a_i||_p, \tag{6}$$

and a_i is the i-th row of matrix A.

2.2. Bregman Methods

Bregman methods were initially introduced as a method for finding extreme points of convex functionals [4]. In this work we will follow the approach of [8] and apply it to the problem of nonconvex constrained minimization. Let us assume we wish to solve,

$$\min_u F(u), \tag{7a}$$

$$s.t.\ Au = b. \tag{7b}$$

This problem can be transformed into an unconstrained problem with the introduction of a penalty function $\Psi : \mathbb{R}^n \mapsto \mathbb{R}_+$. This function is positive when the constraints are violated and zero otherwise.

$$\min_u F(u) + \rho\Psi(Au - b) \tag{8}$$

By letting $\rho \to \infty$ it can be shown that the solution of (8) approaches that of (7). Unfortunately this also leads to very ill-conditioned problems, typically with numerical instabilities as a consequence. It is this ill-conditioning that Bregman methods aim to avoid by instead minimizing the Bregman distance.

Definition 2.1 *The Bregman distance $D_\varphi^{(p)}$ associated with a convex function φ is given by*

$$D_\varphi^{(p)}(u, v) = \varphi(u) - \varphi(v) - < p, u - v >, \tag{9}$$

with $p \in \partial\varphi(v)$.

Using this notion of a Bregman distance we arrive at the *Split Bregman* algorithm of [8] by replacing the objective function F in (8) by its associated Bregman distance, resulting in the following iterations

$$u^{k+1} = \arg\min_u \; D_F^{(p^k)}(u, u^k) + \rho\Psi(Au - b), \tag{10}$$

where $p^k \in \partial F(u^k)$. It can then be shown that, under very mild assumptions and for an arbitrary initial value u^0 and penalty parameter $\rho > 0$, the sequence $\{u^k\}_{k\in\mathbb{N}}$ converges to a global minimizer of (7).

Now the above subproblems are no longer ill-conditioned but they can nevertheless be costly to solve. However, if the objective function happens to be separable, i.e. on the form $E(u, d) = g(u) + \Phi(d)$, the iterations (10) can be decomposed further. The idea is that this decoupling of the entering variables will result in simpler subproblems and thus improve computational efficiency. This leads to the *Alternating Split Bregman* method, also proposed by [8],

$$u^{k+1} = \arg\min_u \; D_E^{(p^k)}(u, d^k, u^k, d^k) + \rho\Psi(A\left[\begin{smallmatrix} u \\ d^k \end{smallmatrix}\right] - b), \tag{11}$$

$$d^{k+1} = \arg\min_d \; D_E^{(p^k)}(u^{k+1}, d, u^k, d^k) + \rho\Psi(A\left[\begin{smallmatrix} u^{k+1} \\ d \end{smallmatrix}\right] - b), \tag{12}$$

where now $p^k \in \partial E(u^k, d^k)$. The convergence properties of this algorithm has shown to be similar in many aspects to that of the split Bregman metod.

It was later established that there are strong connections between all these variations of the Bregman method and a number of already existing algorithms. For instance, if we let $\Phi = \frac{1}{2}||\cdot||_2^2$ then both the Split Bregman and alternating Split Bregman are special instances of the classical proximal splitting methods in convex analysis. The Split Bregman algorithm can be identified with the augmented Lagrangian algorithm [2] which in turn can be interpreted as applying Douglas-Rachford [7] splitting to its corresponding dual problem. Similarly the Alternating Split Bregman algorithm can be shown to coincide with the alternating direction method of multipliers [2]. This connection was established in [21, 22] and greatly helps to improve the understanding of the Split Bregman methods as well as providing clarification into their convergence properties.

3. Bregman Iterations and Pseudoconvex Optimization

We now describe our formulation of a proximal splitting method that is particularly effective for globally solving pseudoconvex minimization problems on the form (5).

As a starting point for this presentation we chose the Bregman iteration approach. However, as indicated by the discussion in the previous section and the work of [21, 22] chosing any of the other interpretations of this approach will without doubt lead to an identical, or very similar, algorithm. In fact, for those readers who have an intimate knowledge of these methods, the resulting formulation will probably appear familiar. The derivation closely follows that of [21] but does, due to the nonconvexity, deviate from that work.

Let $E : \mathcal{C} \times \mathcal{C} \mapsto \mathbb{R}_+$ be the function $E(U, V) = ||U||_{p,\infty}$, with $\mathcal{C} = \bigcup \mathcal{C}_i$, a convex subset of $\mathbb{R}^{2\times N}$. As the penalty function $\Psi : \mathbb{R}^{2\times N} \mapsto \mathbb{R}_+$ we use $\Psi(U) = \frac{1}{2}||U||_F^2$. The associated Bregman distance $D_E^{(p)}([T, x], [T^k, x^k])$ for our iterations then becomes

$$D_E^{(p)}([T, x], [T^k, x^k]) = ||T||_{p,\infty} - ||T^k||_{p,\infty} - < p_T^{kT}, T - T^k > - < p_x^{kT}, x - x^k >, \tag{13}$$

$$\begin{bmatrix} p_T^k \\ p_x^k \end{bmatrix} \in \partial E(T^k, x^k). \tag{14}$$

The Split Bregman iterations are obtained by solving

$$\{T^{k+1}, x^{k+1}\} = \arg\min_{T,x\in\mathcal{C}} \; D_f^{(p^k)}(T, x, T^k, x^k) + \frac{\rho}{2}||\Pi(x) - u - T||_F^2. \tag{15}$$

Now Π is no longer a linear operator, nor is it convex, but we can still proceed in a similar fashion to that of previous work. Using the fact that (15) implies

$$\partial_T E|_{T=T^k} - p_T^k - \rho\left(\Pi(x^k) - u - T^k\right) \ni 0 \tag{16}$$

$$-p_x^k - \rho\nabla\Pi(x^k)^T\left(\Pi(x^k) - u - T^k\right) = 0 \tag{17}$$

since $\partial_x E = 0$. Then letting

$$p_T^{k+1} = p_T^k + \rho\left(\Pi(x^k) - u - T^k\right) \tag{18}$$

$$p_x^k = -\rho\nabla\Pi(x^k)^T\left(\Pi(x^k) - u - T^k\right) \tag{19}$$

we have that $\left[\begin{smallmatrix} p_T^k \\ p_x^k \end{smallmatrix}\right] \in \partial E(T^k, x^k)$, satisfying (14). Inserting (18) and (19) into (15) we then obtain the following equivalent iterations

$$\{T^{k+1}, x^{k+1}\} = \arg\min_{T,x\in\mathcal{C}} \; ||T||_{p,\infty} + \frac{\rho}{2}||b^k + \Pi(x) - u - T||_F^2. \tag{20}$$

where for convenience of notation we have introduced $p_T^k = \rho b^k$. From an augmented Lagrangian point of view, b^k could be interpreted as an approximation of the scaled Lagrangian multipliers associated with the equality constraint $T = \Pi(x) - u$.

Finally, by splitting the variables x and T and, thus only approximately solving the subproblem (15) at each iteration we arrive at our splitting algorithm.

$$x^{k+1} = \arg\min_{x \in \mathcal{C}} \frac{\rho}{2} ||b^k + \Pi(x) - u - T^k||_F^2. \quad (21)$$

$$T^{k+1} = \arg\min_{T} ||T||_{p,\infty} + \frac{\rho}{2}||b^k + \Pi(x^{k+1}) - u - T||_F^2, \quad (22)$$

$$b^{k+1} = b^k + \left(\Pi(x^{k+1}) - u - T^{k+1}\right). \quad (23)$$

Note first that the (22) is a convex problem in T, in fact it is the proximity operator of the $||\cdot||_{p,\infty}$-norm. In addition, the subproblem (21) is not convex. That means we could potentially have issues with local minimas. However, next we will show that for our specific original problem (5), this is interestingly enough not an issue.

3.1. Convergence Analysis

Unfortunately proximal splitting methods are not directly applicable to problems that are not convex. This nonconvexity introduces significant obstructions to the theoretical analysis. For general nonconvex problems, proofs of convergence to a local minima is extremely difficult for most problem formulations. Consequently the majority of convergence results for nonconvex problems are typically weak. In this section we will nonetheless establish some of the convergence properties of the proposed algorithm.

We start off by showing the boundedness of the sequence of scaled Lagrangian multipliers b^k.

Lemma 3.1 *The sequence $\{b^k\}_{k\in\mathbb{N}}$ generated by (21)- (23) is bounded.*

Proof 3.1 *Since the subdifferentials of the convex functions $h(z) = ||z||_\infty$ and $g(x) = ||x||_p$ are both bounded sets, it follows from subgradient calculus, see [3], and the non-increasing property of h on $\mathbb{R}_+$ that the subdifferential of the composition $\partial(h \circ g) = \partial(h(g(x_1), ..., g(x_N))) = \partial(|| \left[x_1, \dots x_N \right]^T ||_{p,\infty})$ must also be bounded. The lemma follows from the fact that $p_T^k \in \partial_T E(T^k) = \partial(h \circ g)(T^k)$ and $\rho b^k = p_T^k$ for $\rho > 0$.*

Next we state a weak convergence lemma regarding our method, with the proof is omitted.

Lemma 3.2 *If we let $\rho \to \infty$ as in (21)-(23) then $\{b^k\}_{k\in\mathbb{N}} \to b^*$.*

This lemma is weak in the sense that for convergence to be guaranteed ρ needs to approach ∞. It also means that instances of ill-conditioning can not be ruled out, despite our best efforts to eliminate them through the use of the Bregman distance. However, empirically we have found that the proposed algorithm indeed onverges for even moderately large values of ρ. Proving under what conditions such a proposition is true in general is highly desirable and is currently also at the top of our list of future work.

Next we show that if our algorithm converges it does so to a *global* solution of our original problem. This despite the fact our iterations involve subproblems that are neither convex nor quasiconvex.

Theorem 3.1 *If the sequence $\{b^k\}_{k\in\mathbb{N}}$ defined as in (23) converges to some b^*, then the sequences $\{x^k\}_{k\in\mathbb{N}}$ converges to x^*, a global minimizer of (5).*

Proof 3.2 *First, from (21) and (22) we have that $\{T^k\}_{k\in\mathbb{N}}$ and $\{x^k\}_{k\in\mathbb{N}}$ must converge to local minimizers T^* and x^*[2]. Then from (21) and (22) it must hold that*

$$0 = \rho(\nabla_x \Pi(x^*))^T (b^* + \Pi(x^*) - u - T^*), \quad (24)$$

$$0 \in \partial ||T^*||_{p,\infty} - \rho(b^* + \Pi(x^*) - u - T^*). \quad (25)$$

Our initial assumption on $\{b^k\}_{k\in\mathbb{N}}$ implies through (23) that

$$(\Pi(x^k) - u - T^k) \to 0. \quad (26)$$

Inserting this back into (24)-(25) and simplifying we obtain

$$0 \in (\nabla_x \Pi)^T \partial ||\Pi(x^*) - u||_{p,\infty}. \quad (27)$$

Identifying the above expression as a necessary condition for optimality associated with our original problem (5) it follows that x^* is a stationary point for this function. Then by pseudoconvexity [18] we have that any such stationary point must also be a global minimizer of (5) and the lemma follows.

4. A Proximal Splitting Algorithm for L_∞ Minimization

Here we present a summary (algorithm 1) of the proposed method derived in the previous section, along with discussions on some of the practical aspects of this algorithm.

4.1. Initialization

Since the convergence of the proposed algorithm is not dependent on initialization, due to pseudoconvexity, any choice of x^0, b^0, T^0 will in theory suffice. However, the better the initialization is the faster convergence will be achieved. In this work we initialize x^0 using standard algebraic methods of [9], and set $T^0 = \Pi(x^0) - u$ and $b^0 = 0$.

[2] For the sake of brevity we have here assumed that the solution of (15) is unique. Extending this proof to include instances when this does not necessarily hold is a straightforward exercise but requires slightly lengthier notation.

Algorithm 1 The proposed proximal splitting algorithm for L_∞ minimization.

Input: u, ρ^k
Initialize: x^0, b^0, T^0
repeat

- Find (approximate) solution to the subproblem in x

$$x^{k+1} = \arg\min_{x \in \mathcal{C}} ||b^k + \Pi(x) - u - T^k||_F^2 \tag{28}$$

- Evaluate the proximity operator

$$T^{k+1} = \arg\min_{T} ||T||_{p,\infty} + \frac{\rho}{2}||b^k + \Pi(x^{k+1}) - u - T||_F^2 \tag{29}$$

- Update b

$$b^{k+1} = b^k + \left(\Pi(x^{k+1}) - u - T^{k+1}\right) \tag{30}$$

- Update ρ (31)

until convergence

4.2. The subproblem in x

As the resulting subproblem in x (28), can be identified as a least squares minimization of the reprojection error, but now with respect to the modified image points

$$\hat{u} = (u + T^k - b^k). \tag{32}$$

This subproblem can consequently be solved using existing bundle adjustment solvers. In addition, since the our main problem (20) is solved in alternation, starting with x, the obtained x^{k+1} will, unless we are nearing convergence, not be a stationary point of (20) at iteration k. It may therefore suffice to only solve this subproblem approximately. In fact, in our current implementation we terminate after a single bundle adjustment iteration. The constraint $x \in \mathcal{C}$ is handled by simply defining $\Pi(x) = \infty$ if $x \notin \mathcal{C}$.

4.3. The subproblem in T

In order to solve the second step of algorithm 1, (29), we will use a result of [23]. Here we need to solve problems on the form

$$\min_{X \in \mathbb{R}^{n\times m}} \frac{1}{\rho}||X||_{p,\infty} + \frac{1}{2}||X - A||_F^2. \tag{33}$$

By Moreau's decomposition theorem [19] we have that the dual problem of (33) is given by

$$\max_{Z \in \mathbb{R}^{n\times m}} -\frac{1}{2}||Z||_F^2 + < Z, A >, \quad \text{s.t. } ||Z||_{q,1} \leq \frac{1}{\rho}, \tag{34}$$

where $||\cdot||_q$ is the dual norm of $||\cdot||_p$[3]. It was shown in [23] that that solving (34) is equivalent to finding the root of the following monotonically decreasing function $g : \mathbb{R} \mapsto \mathbb{R}$

$$g(\theta) = \frac{1}{\rho} - Z(\theta), \tag{35}$$

$$Z(\theta) = \arg\min_{Z \in \mathbb{R}^{n\times m}} -\frac{1}{2}||Z - A||_F^2 + \theta||Z||_{q,1}. \tag{36}$$

The solution to the original primal problem (33) is then obtained through $X^* = A - Z(\theta^*)$.

Solving (22) thus only involves finding the root of a one-dimensional function. This can be efficiently carried out by employing standard root finding algorithms, such as quadratic interpolation. However, what ultimately makes solving the subproblem (22) so efficient is the fact that for the choices of p that we consider in this paper, the resulting proximity operators in (36) have very simple closed form solution. For instance, for $p = 1$ the solution is given by soft-thresholding,

$$Z_{ij}(\theta) = \text{sign}(A_{ij}) \max(|A_{ij}| - \theta, 0). \tag{37}$$

For $p = 2$ the solution is obtained by

$$Z_{ij}(\theta) = \max(1 - \frac{\theta}{||A||_F}, 0)A. \tag{38}$$

A similar and equally efficient, but slightly more intricate expression also exists for $p = \infty$. For this choice of p the solution is obtained by again invoking the Moreau decomposition theorem and evaluating the resulting dual problems. We refer the interested reader to [23, 5] for the details.

4.4. The Penalty Parameter ρ

As mentioned in the previous section, we currently have no theoretical result prohibiting ρ from approaching ∞ in

[3] Where q is conjugate to p, $(\frac{1}{p} + \frac{1}{q} = 1)$.

order to achieve convergence. However, we have empirically observed that algorithm 1 indeed converges for even moderate values of ρ. A standard extension these types of methods is to modify this penalty parameter ρ during the progress of the algorithm. The simplest possible such scheme is possibly,

$$\rho^{k+1} = \eta \rho^k, \tag{39}$$

where typical choices of these parameters in this work were $\rho^0 = 10^{-3}$ and $\eta = 1.01$.

4.5. Necessary Conditions for Optimality and Stopping Criteria

The necessary conditions for optimality of (20) are given as its stationary points

$$\rho \nabla \Pi(x^{k+1})^T \left(b^k + \Pi(x^{k+1}) - u - T^{k+1}\right) = 0, \tag{40}$$
$$\partial ||T^{k+1}||_{p,\infty} - \rho \left(b^k + \Pi(x^{k+1}) - u - T^{k+1}\right) \ni 0. \tag{41}$$

We have, by construction, that (41) is satisfied for every T^k. Rewriting (40) gives us

$$0 = \rho \nabla \Pi(x^{k+1})^T \left(b^k + \Pi(x^{k+1}) - u - T^k\right) = \tag{42}$$
$$= \rho \nabla \Pi(x^{k+1})^T \left(b^k + \Pi(x^{k+1}) - u - T^{k+1}\right) + \rho \nabla \Pi(x^{k+1})^T \left(T^{k+1} - T^k\right). \tag{43}$$

Hence, if the second part of (43) holds then (40) is also satisfied. Then with the addition of the requirement that $b^k \to b^*$, see section 4.5, we can write the necessary condition for the convergence of (21)-(23) to a local minima as

$$\rho \nabla \Pi(x^k)^T \left(T^k - T^{k-1}\right) = 0, \tag{44}$$
$$b^k - b^{k-1} = 0. \tag{45}$$

These conditions then suggest that using the norms of (44)-(45) would be a sensible choice for a stopping criteria. In this work algorithm 1 was terminated when the L_2 norm of both these expressions were smaller than $\varepsilon = 10^{-3}$.

Finally, we would like to point out an interesting observation, or interpretation, regarding the proposed algorithm. This observation perhaps only becomes apparent when considering the implementation of this method. In doing so, one basically only need to add three lines of code (29), (30) and (31) to the main loop of any existing bundle adjustment package. So essentially what we are doing is deceiving this nonlinear bundle solver into unknowingly minimizing the reprojection residuals under a very different norm to the L_2-norm it was designed for. And we do so by simply moving the measurements around in a very specific manner. We are using a solver intended for finding local minima of smooth, nonconvex problems to instead obtain global optima of a nonsmooth and pseudoconvex problem.

5. Experiments

This section contains an experimental evaluation of our proposed method. As we are primarily interested in large scale L_∞ optimization we limited our experiments to SfM with known camera rotation. We performed experiments on six different, publicly available datasets. These are listed in table 1. The dinosaur and corridor sequences are available from the Oxford visual geometry group. The house, cathedral, UWO and Alcatraz datasets are available from [15].

All methods were implemented in Matlab on a 3.20 GHz desktop computer with 8 GB of RAM. The MOSEK[4] optimization package was used as LP solver, SeDuMi [24] as the SOCP solver and the SBA package of [14] for solving the bundle adjustment subproblems that arises in (28).

In the following experiments we compare the running times achieved by our method and the state-of-the-art approach as described in [1] for different choices of $p \in \{1, 2, \infty\}$. To provide a baseline we also include the results from the standard bisection methods of [10]. It is not the intention of this work to compare and argue the merits of different choice of norms used in L_∞ formulations of multiview geometry problems but rather to show the efficiency and scalability that comes from splitting a complex problem into subproblems that can then be solved separately and efficiently.

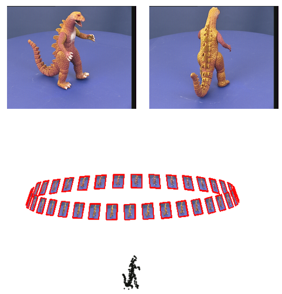
The results are given in table 1, as the average computation time over 100 separate runs. Execution times were not reported on some instances where convergence was not achieved within a reasonable time frame, due to numerical or memory issues of the associated solvers. This typically occurred for large-scale SOCP subproblem, an issue also reported in [1].

These results do seem to indicate that the proposed algorithm matches or outperforms state-of-the-art solvers for this class of problems. It should also be noted that the execution times of our algorithm closely match those reported in [6]. However, since comparisons between different algorithms as carried out here heavily depend on the different level of effort put into their implementation, the programming language used and what solvers were employed, these results can only be seen as indicative. Future versions of this manuscript will contain a more detailed and extensive experimental evaluation.

6. Conclusions

In this paper we have proposed a proximal point formulation for large-scale pseudoconvex L_∞ optimization. The resulting meta-algorithm is both computationally efficient as well as very uncomplicated to implement. The experimental evaluation clearly demonstrates the competitive performance of this algorithm as it appears to be able to offer

[4] Available from http://www.mosek.com.

		 Dinosaur	 Corridor	 House
Observations		2663	4035	35321
Points		328	737	12475
Images		36	11	12
$\|\| \cdot \|\|_{2,\infty}$	Bisection	27.5	39.6	-
	Gugat	9.1	11.6	-
	Proximal	**1.8**	**3.8**	**45.1**
$\|\| \cdot \|\|_{1,\infty}$	Bisection	11.5	18.3	172.0
	Gugat	6.8	8.8	61.7
	Proximal	**1.8**	**3.9**	**44.7**
$\|\| \cdot \|\|_{\infty,\infty}$	Bisection	8.3	16.9	144.2
	Gugat	2.8	6.8	49.1
	Proximal	**1.8**	**3.8**	**44.3**

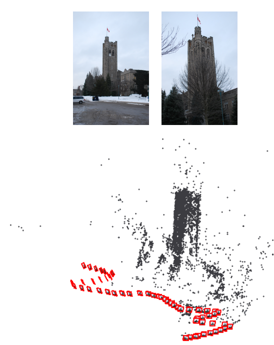
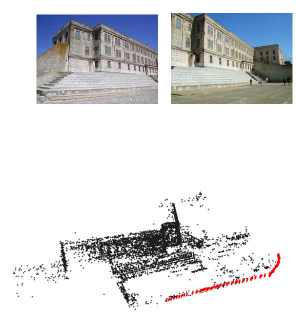
		 Cathedral	 UWO	 Alcatraz
Observations		45,553	27,309	68,615
Points		16,961	8,880	23,674
Images		17	57	67
$\|\| \cdot \|\|_{2,\infty}$	Bisection	-	-	-
	Gugat	-	-	-
	Proximal	**31.4**	**30.0**	**117.1**
$\|\| \cdot \|\|_{1,\infty}$	Bisection	181.3	158.2	459.3
	Gugat	63.1	52.1	129.1
	Proximal	**31.2**	**29.5**	**115.3**
$\|\| \cdot \|\|_{\infty,\infty}$	Bisection	146.0	122.3	404.0
	Gugat	50.1	37.1	118.0
	Proximal	**31.8**	**29.8**	**112.3**

Table 1. Performance of various methods on 6 different image sequences. We report the execution time in seconds averaged over 100 runs.

significant speedups over the current state-of-the-art.

As an additional advantage, and perhaps the main contribution of this method, it does not require a separate dedicated numerical solver from the corresponding least-squares formulation. Allowing it to take advantage of the computational efficiency of existing state-of-the-art software packages for non-linear minimization as well as to seamlessly transition between different residual norms.

At this point, several avenues for future work remain open. The most challenging is perhaps in strengthening the proof of convergence given in section 4.5. Empirically we have observed that the proposed algorithm does indeed converge even for moderate choices of ρ. Proving this is currently our main focus.

Acknowledgements

This research was supported under the Australian Research Councils Discovery Early Career Researcher Award project DE130101775.

References

[1] S. Agarwal, N. Snavely, and S. M. Seitz. Fast algorithms for l infinity problems in multiview geometry. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. IEEE Computer Society, 2008. 1, 6

[2] D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods (Optimization and Neural Computation Series)*. Athena Scientific, 1 edition, 1996. 3

[3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004. 4

[4] L. Bregman. The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex optimization. *USSR Computational Mathematics and Mathematical Physics*, 7:200–217, 1967. 2

[5] P. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In H. Bauschke, R. Burachik, P. Combettes, V. Elser, D. Luke, and H. Wolkowicz, editors, *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer, 2011. 1, 5

[6] Z. Dai, Y. Wu, F. Zhang, and H. Wang. A novel fast method for l problems in multiview geometry. In *European Conference on Computer Vision*, volume 7576 of *Lecture Notes in Computer Science*, pages 116–129. Springer, 2012. 1, 6

[7] J. Eckstein and D. P. Bertsekas. On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators. *Math. Program.*, 55(3):293–318, June 1992. 3

[8] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Science*, 2(2):323–343, Apr. 2009. 2, 3

[9] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 4

[10] F. Kahl. Multiple view geometry and the l-infinity norm. In *International Conference on Computer Vision*, 2005. 1, 6

[11] F. Kahl and R. Hartley. Multiple view geometry under the l-infinity norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1603–1617, 2008. 2

[12] Q. Ke and T. Kanade. Quasiconvex optimization for robust geometric reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1834–1847, 2007. 2

[13] H. Li. Efficient reduction of l-infinity geometry problems. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2695–2702. IEEE, 2009. 1

[14] M. A. Lourakis and A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Transactions Mathematical Software*, 36(1):1–30, 2009. 1, 6

[15] C. Olsson and O. Enqvist. Stable structure from motion for unordered image collections. In *Scandinavian Conference on Image Analysis, SCIA 2011*, 2011. 6

[16] C. Olsson, A. Eriksson, and R. Hartley. Outlier removal using duality. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2010. 1, 2

[17] C. Olsson, A. Eriksson, and F. Kahl. Efficient optimization for l-infinity problems using pseudoconvexity. In *International Conference on Computer Vision*, 2007. 1

[18] C. Olsson and F. Kahl. Generalized convexity in multiple view geometry. *Journal of Mathematical Imaging and Vision*, 2010. 1, 2, 4

[19] R. T. Rockafellar. *Convex Analysis (Princeton Landmarks in Mathematics and Physics)*. Princeton University Press, Dec. 1996. 5

[20] Y. Seo and R. I. Hartley. A fast method to minimize l error norm for geometric vision problems. *International Conference on Computer Vision*, pages 1–8, 2007. 2

[21] S. Setzer. Split bregman algorithm, douglas-rachford splitting and frame shrinkage. *Scale space and variational methods in computer vision*, pages 464–476, 2009. 3

[22] S. Setzer. Operator splittings, bregman methods and frame shrinkage in image processing. *International Journal of Computer Vision*, 92(3):265–280, 2011. 3

[23] S. Sra. Fast projections onto mixed-norm balls with applications. *Data Mining and Knowledge Discovery*, 25(2):358–377, 2012. 5

[24] J. F. Sturm. Using sedumi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12:625–653, 1999. 6

[25] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment a modern synthesis. 1883:298–372, 2000. 1