Smooth Approximation of $L_\infty$-Norm for Multi-View Geometry

Yuchao Dai, Hongdong Li, Mingyi He, Chunhua Shen

Abstract—Recently the $L_\infty$-norm optimization has been introduced to multi-view geometry to achieve global optimality. It is solved through solving a sequence of SOCP (second order cone programming) feasibility problem which needs sophisticated solvers and time consuming. This paper presents an efficient smooth approximation for $L_\infty$-norm optimization in multi-view geometry using log-sum-exp functions. We have proven that the proposed approximation is pseudo-convex with the property of uniform convergence. This allows us to solve the problem using gradient based algorithms such as gradient descent to overcome the non-differentiable property of $L_\infty$ norm. Experiments on both synthetic and real image sequence have shown that the proposed algorithm achieves high precision and also significantly speeds up the implementation.

Keywords—log-sum-exp; smooth approximation; $L_\infty$ norm

I. INTRODUCTION

A recent trend in multi-view computer vision research is to seek global optimality for structure and motion computation (see [1] for an overview). However this is a hard non-convex problem in most cases. Multiple local minima exist for the problems in geometric computer vision under $L_2$-norm (for example up to 3 local minima exist for two-view triangulation, up to 24 local minima exist for three-view triangulation and for higher numbers of views, the number of local minima grows cubically). Globally optimal estimates have only been achieved for some small scale problems under $L_2$ norm.

The $L_\infty$ framework shares advantages with both $L_2$ method and the linear algorithms. Globally optimal solutions are guaranteed with a geometrically meaningful cost function and a reasonable computational cost. Thus $L_\infty$ optimality has been recently applied to many geometric problems including: multi-view triangulation, camera resection (uncalibrated case), homography estimation, structure and motion recovery with known camera orientations [2], outlier removing [3], camera pose (calibrated case), two-view relative motion [4], etc. Recently the $L_\infty$ norm has also been applied to multi-camera system [5]. In all of these cases, a single minimum for $L_\infty$ norm exists and it may be easily found. The bisection algorithm using a sequence of SOCP feasibility tests has been viewed as the standard approach for geometric $L_\infty$ optimization problems.

As bisection with SOCP feasibility test is time consuming, there has been work on speeding up the implementation. Seo and Hartley [6] adopted an incremental strategy testing only part of the measurements, and proposed to use linear programming with $L_\infty$ reprojection error as approximation for SOCP. Agarwal et al. [7] introduced fractional programming theory into the community and showed that the $L_\infty$ problems in geometric optimization are instance of a convex-concave generalized fractional programming. They proposed five algorithms to solve the problem using local Newton method thus decreasing the number of feasibility tests. Li [8] applied LP-type problem theory to geometric optimization problem based on the fact that $L_\infty$ norm minimum is determined by only a small subset of all the constraints. In this paper, we take a different view of the problem and present an efficient smooth approximation of $L_\infty$ to speed up the implementation and provide theoretical proof.

Mathematically, minmax problem can be solved using polynomials or exponential approximation. Li and Fang [9] used an entropic regularization procedure to provide smooth approximation of the minmax problem that uniformly converges to the minmax function. Sheu et al. [10] proposed an iterative entropic regularization method using uniform error estimate. The idea of approximation of a non-convex function using smooth function have been applied in sparse decomposition. Mohimani et al. [11] presented a fast approach based on smoothed $L_0$ norm.

In this paper, we present an efficient smooth approximation algorithm for $L_\infty$ optimization in multi-view geometry. We start considering the $L_\infty$ norm as the extreme case of $p$ norm when $p$ approximates $\infty$. Replacing the polynomial based formulation with exponent function, we obtain the log-sum-exp approximation form of the original $L_\infty$ problem. Further more we have proven that the proposed function is pseudo-convex thus only global minimum exists. Under this framework, we transform the non-differentiable $L_\infty$ problem to a differentiable problem which could be solved efficiently using simple iterative convex optimization (such as Newton or Gauss-Newton method). This makes the implementation easy and avoid using SOCP. Experimental results on both synthetic and real images sequence show that the proposed method can approximate the $L_\infty$-norm with small tolerance,
and also works fast.

The remainder of the paper is organized as follows: the $L_\infty$-norm and its solution through SOCP is reviewed in Section 2, in Section 3 we present the proposed algorithm in detail with theoretical analysis. Section 4 illustrates the experimental results on both synthetic and real images. Finally, Section 5 concludes the paper with discussion and future direction.

II. $L_\infty$-NORM AND SOCP

In this section, we briefly review the problem of $L_\infty$ optimization and its solution through SOCP. Some concepts from optimization are given which will be needed in the following sections.

A. $L_\infty$ framework

In the sense of $L_\infty$-norm, we are seeking to minimize the maximal residual. Problems in geometric computer vision that we consider in this paper can be written in the following form:

$$\min_{x} \max_{i} \frac{\|A_i x + b_i\|_2}{c_i^T x + d_i} \quad \text{subject to } c_i^T x + d_i > 0 \quad (1)$$

where $x$ is the variable determined by the specific problem. The above model is equivalent to the following model:

$$\min_{x} \max_{i} \frac{\|A_i x + b_i\|_2^2}{(c_i^T x + d_i)^2} \quad \text{subject to } c_i^T x + d_i > 0 \quad (2)$$

We are interested in the model (2) because of its pseudo-convex property to be analyzed below.

Taking the multi-view triangulation problem as an example in which we are seeking the optimal 3D point $x$ giving the minimal maximal residual (reprojection error), we have:

$$h_i(x) = \sqrt{((P_i^T - x_i P_i^i)^T x)^2 + ((P_i^T - y_i P_i^i)^T x)^2}$$

where $(x_i, y_i)$ is the image point and $P_i$ is the camera matrix and $P_i^T x > 0$. It is easily verified that the above model has the form of equation (1).

Consider the individual function $h_i(x) = \frac{\|A_i x + b_i\|_2}{c_i^T x + d_i}$ as a component of a vector, then the problem can be thought as minimizing the $L_\infty$-norm of this vector.

The $L_\infty$-norm problem has only one local minimum which is still the global minimum. However the problem with $L_\infty$ norm is that it is not everywhere differentiable thus we can not solve it using gradient based algorithms.

B. Second Order Cone Programming

It is easily verified that the problem (1) can be transformed, by introducing an additional auxiliary variable $\gamma$, into an equivalent problem of the following form

$$\min_{\gamma, x} \gamma \quad \text{subject to} \|A_i x + b_i\|_2 - \gamma (c_i^T x + d_i) \leq 0 \quad \text{and } \gamma > 0 \quad (3)$$

For any fixed value of $\gamma$, the set of points satisfying the constraint $\|A_i x + b_i\|_2 - \gamma (c_i^T x + d_i) \leq 0$ is a convex set. Thus (3) is an SOCP problem which we may easily solve.

Kahl and Hartley [2] proposed to use improved bisection solving sequence of feasibility problems. This improved bisection algorithm is shown as Algorithm 1.

Require: Initial interval $[\gamma_l, \gamma_u]$ known to contain the optimal value of $\gamma^*$ and tolerance $\epsilon > 0$.

1: repeat
2: $\gamma := (\gamma_l + \gamma_u)/2$.
3: Solve the SOCP feasibility problem to get $x$.
4: if feasible then
5: $\gamma_u := \max_i \|A_i x + b_i\|_2$.
6: else
7: $\gamma_l := \gamma$.
8: end if
9: until $\gamma_u - \gamma_l \leq \epsilon$.

Algorithm 1: Improved Bisection Algorithm

C. Pseudo-convex function

Definition A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be pseudo-convex if for all $x, y \in \text{dom}f$

$$\nabla f(x)(y - x) \geq 0 \Rightarrow f(x) \leq f(y)$$

Lemma II.1. [12] Suppose $f$ is pseudo-convex, then $\nabla f(x) = 0$ if and only if $f(y) \geq f(x)$ for all $y$.

Thus for a pseudo-convex function any stationary point is a global minimum. This is a useful property since it ensures that the gradient does not vanish anywhere except at the optimum, making it possible to apply gradient decent algorithm.

Lemma II.2. [13] If $w : S \to \mathbb{R}$ is concave then $f(x) = \frac{w(x)}{v(x)}$ is pseudo-convex on $S = \{x | v(x) > 0\}$.

As $\omega(x) = \frac{\|A_i x + b_i\|_2}{(c_i^T x + d_i)}$ is convex and $v(x) = c_i^T x + d_i$ is concave, from the above lemma, we can easily verify the following lemma:

Lemma II.3. $h_i^2(x) = \frac{\|A_i x + b_i\|_2^2}{(c_i^T x + d_i)^2}$ is pseudo-convex on the set $\{x | c_i^T x + d_i > 0\}$.

III. SMOOTH APPROXIMATION OF $L_\infty$-NORM

The main difficulty in minimizing the $L_\infty$-norm lies in that it is quasi-convex however not differentiable. Therefore it can not be solved using gradient based method.

Mathematically, $L_\infty$ optimality or equivalently minmax optimality can be solved using polynomial or exponential approximation. The main benefit of smooth approximation is that it transforms the non-differentiable function to differentiable function thus a local gradient method can be applied.
easily. The differentiable approximation function must have a parameter to determine the quality of the approximation.

A. The method of “sum of powers”

The vector norm \( \|x\|_p \) for \( p = 1, 2, \ldots \) is defined as
\[
\|x\|_p = \left( \sum_i |x_i|^p \right)^{1/p}
\]
(4)

One can also extend this to \( p = \infty \) via
\[
\|x\|_\infty = \max\{|x_1|, |x_2|, \ldots, |x_n|\}
\]
(5)

which in fact is the limit of the \( p \) norms for finite \( p \).

This implies that the \( L_\infty \)-norm minimization can be approximated using sum of powers by \( \min_x (\sum_i |f_i(x)|^p)^{1/p} \).

For large \( p \), we will have a good approximation for \( L_\infty \) norm minimization. The problem with “sum of powers” is that there is no parameter we can adjust to control the quality of the approximation and its approximation performance improves slowly with the increase of \( p \) thus time-consuming.

B. Log-Sum-Exp approximation

Instead of using the “sum of powers” approximation for \( L_\infty \) norm, we extend the idea to exponential function which is more smooth and with excellent approximation ability. Define the following function:
\[
f(X, \sigma) = \sigma \ln \left\{ \sum_{i=1}^n \exp \left( \frac{h_i^2(x)}{\sigma} \right) \right\}
\]
(6)

Compared with equation (4), equation (6) is smoother and the introduction of parameter \( \sigma \) controls the approximation performance. A small \( \sigma \) makes the equation approach \( L_\infty \) fast and achieve small approximation error. The smaller the \( \sigma \), the better approximation we will obtain.

The log-sum-exp approximation (6) is a continuous and everywhere differentiable function. Hence, through introducing the log-sum-exp approximation for \( L_\infty \)-norm, we can solve the \( L_\infty \) problem using differentiable method. Before presenting the detail algorithm, we need to prove the approximation performance and pseudo-convexity of the approximation function.

C. Theoretical Analysis for Uniform Convergence

In this part, we will give theoretical analysis for the proposed log-sum-exp approximation of \( L_\infty \) norm. First, we have to prove that when \( \sigma \) tends to 0, we will approach \( L_\infty \) norm as close as possible which means our approximation function converge to the objective function. Second, we have to show that the proposed approximation (6) is pseudo-convex when \( \sigma \to 0 \).

Let us consider the equivalent form of the log-sum-exp approximation (6) as:
\[
\min_X \sum_{i=1}^n \exp \left( \frac{h_i^2(x)}{\sigma} \right)
\]
(7)

where function \( h_i(x) \) is smooth and positive such that \( h_i(x) \geq 0 \).

For each \( i = 1, \ldots, n \), we have \( h_i^2(x)/\sigma \leq \max_i h_i^2(x)/\sigma \) and \( \exp(h_i^2(x)/\sigma) \leq \exp(\max_i h_i^2(x)/\sigma) \).

Thus for the sum of the \( n \) elements, we have
\[
\max_i \exp(h_i^2(x)/\sigma) \leq \sum_{i=1}^n \exp(h_i^2(x)/\sigma) \leq n \max_i \exp(h_i^2(x)/\sigma)
\]

Also, taking logarithm function \( \ln \) and multiplying positive constant \( \sigma \) do not change the monotonicity, thus we have:
\[
\sigma \ln(\exp(\max_i h_i^2(x)/\sigma))) \leq \sigma \ln(\sum_{i=1}^n \exp(h_i^2(x)/\sigma)) \leq \sigma \ln(n \exp(\max_i h_i^2(x)/\sigma)))
\]

At the same time based on the property of logarithm operation we have the following equation:
\[
\max_i h_i^2(x) \leq \sigma \ln(\sum_{i=1}^n \exp(h_i^2(x)/\sigma)) \leq \sigma n + \max_i h_i^2(x)
\]
(8)

In the above equation, \( \sigma \ln n \) has a linear relationship with \( \sigma \). So with the decrease of \( \sigma \), we will have:
\[
\sigma \to 0, \sigma \ln(\sum_{i=1}^n \exp(h_i^2(x)/\sigma)) \to \max_i h_i^2(x)
\]
(9)

The above formula leads us a way to minimize \( \max_i h_i^2(x) \) via its smooth approximation \( \sigma \ln(\sum_{i=1}^n \exp(h_i^2(x)/\sigma)) \). And the approximation error is \( \sigma \ln n \) which has linear relation with \( \sigma \). This admits a well-known uniform error estimate.

D. Analysis of Pseudo-Convexity

In multi-view geometry problem, \( L_\infty \) norm is a quasi-convex function [2]. However it is not pseudo-convex as it is not differentiable everywhere.

Our approximation function (6) is differentiable. To apply local gradient algorithm, we must prove that there is only one local minimum which is also the global minimum in the feasible set. As the definition of pseudo-convex and lemma II.2 imply, if a function is pseudo-convex, then its gradient does not vanish anywhere except at the optimum, rendering it possible to solve using gradient based algorithm. In this part, we will prove that with \( \sigma \to 0 \), our approximation function (6) is pseudo-convex.

**Theorem III.1.** As \( \sigma \) tends to 0, \( f(x, \sigma) \) is a pseudo-convex and also quasi-convex function on the convex set defined by \( v_1(x) > 0 \).
Proof: First function (6) is everywhere differentiable and its gradient is obtained as:

$$\nabla f(x) = \frac{\sum_{i=1}^{n}(\exp(h_i^2(x)/\sigma))\nabla h_i^2(x)}{\sum_{i=1}^{n}\exp(h_i^2(x)/\sigma)}$$

Suppose we have unique maximum at any point and suppose that $\nabla f(y)(x - y) \geq 0$, which is

$$\sum_{i=1}^{n}(\exp(h_i^2(y)/\sigma))\nabla h_i^2(y) \geq 0$$

As $\sum_{i=1}^{n}\exp(h_i^2(y)/\sigma)$ is positive, multiple the above inequality with this constant will not change the inequality, thus we have

$$\sum_{i=1}^{n}\exp(h_i^2(y) - \max_i h_i^2(y))\nabla h_i^2(y)(x - y) \geq 0$$

With $\sigma \rightarrow 0$, the left side of the above inequality tends to $\nabla h_i^2(y)(x - y)$ where $k = \arg\max_i h_i(y)$. Thus we obtain the following inequality:

$$\nabla h_k^2(y)(x - y) \geq 0 \quad (10)$$

We prove that $f(x) \geq f(y)$ by contradiction. Suppose we have $f(x) < f(y)$ which is:

$$\sigma \ln(\sum_{i=1}^{n}\exp(h_i^2(x)/\sigma)) < \sigma \ln(\sum_{i=1}^{n}\exp(h_i^2(y)/\sigma)) \quad (11)$$

As $\sigma > 0$ and logarithm operation does not change the inequality, we obtain

$$\sum_{i=1}^{n}\exp(h_i^2(x)/\sigma) < \sum_{i=1}^{n}\exp(h_i^2(y)/\sigma) \quad (12)$$

Divide the above equation by $\exp(\max_i h_i^2(y)/\sigma)$ which is the $L_\infty$ norm at $y$, we obtain

$$\sum_{i=1}^{n}\exp\left(\frac{h_i^2(x) - \max_i h_i^2(y)}{\sigma}\right) < \sum_{i=1}^{n}\exp\left(\frac{h_i^2(y) - \max_i h_i^2(y)}{\sigma}\right) \quad (13)$$

With $\sigma \rightarrow 0$, the right hand side of the above inequality approximate to 1. If we have some index $k$ such that $h_k(x) > \max_i h_i(y)$, then the left hand side will approximate $+\infty$ thus contradict to the above inequality. So we have

$$\forall i, h_i^2(x) \leq \max_i h_i^2(x) < \max_i h_i^2(y) \quad (14)$$

Especially when $k = \arg\max_i h_i(y)$, we have $h_k^2(x) < h_k^2(y)$. Meanwhile, according to the definition of pseudo-convex we have $\nabla h_k^2(y)(x - y) < 0$ on the convex set defined by $g_k(x) > 0$. This is inconsistent with (10) that $\nabla h_k(y)^2(x - y) \geq 0$ for any $x, y$.

So we obtain $f(x) \geq f(y)$. Thus we confirm that when $\sigma \rightarrow 0$, $f(x, \sigma)$ is a pseudo-convex function.

As a pseudo-convex function is necessarily quasi-convex function, $f(x, \sigma)$ is pseudo-convex and also quasi-convex.

$\blacksquare$

E. An example

Before presenting the detail algorithm, we use an example to show such smooth approximation. A synthetic example of three view triangulation using 1D cameras was discussed in [2] to show the difference between $L_2$ optimization and $L_\infty$ optimization. This three-view triangulation problem has three local $L_2$ minima, lying in front of all three cameras. In this example, all points lie in the plane $z = 0$, thus the problem is simplified to a 2D triangulation problem.

Let $P_0$ represents the camera matrix of 1D camera

$$P_0 = \begin{pmatrix} -3 & 1 & -8 \\ -1 & -3 & -6 \end{pmatrix}$$

The center of this camera is at point $(-3, -1, 1)^T$. The other two cameras are obtained by rotations of angles $\pm 120^\circ$. It is shown that three $L_2$ local minima exist threelfold symmetry and the $L_\infty$ minimum exists at the origin with the minimum 5/3.

Applying our approximation to this problem, we obtain the contour for the smooth approximation function on feasible set. In Figure 1 we show that the contour plot under various smooth parameters as $\sigma = 5, 2, 0.9, 0.75, 0.1, 0.02$. From the figure we see that when $\sigma$ is large, there are multiple minima in the feasible set. With the decrease of $\sigma$, this function approximates $L_\infty$-norm. When $\sigma$ is smaller than some threshold there is only one minimum. In Figure 2 we demonstrate the minimum of approximate $L_\infty$ and $L_\infty$ as a function of $\sigma$. With the decrease of $\sigma$, the minimum of approximate $L_\infty$ approaches $L_\infty$ linearly. This observation validates our analysis that the approximation error is a linear function of $\sigma$.

F. Algorithm for Solving the Smooth $L_\infty$ Approximation

In the above sections, we have proven that when $\sigma \rightarrow 0$ (6) is pseudo-convex in the feasible set thus local gradient based algorithm can be applied.
From the above example, we have seen that when \(\sigma\) is below some threshold, one may get one local minimum (also the global minimum). One direct thought is to choose a sufficient small \(\sigma\) then solving the differentiable function through a gradient decent algorithm. However this parameter is problem-dependent and may not be obtained a priori. Also giving a too small \(\sigma\) and an initial value which may be far away from the actual minimum will make the gradient descent algorithm spending long time finding the minimum.

Our idea to overcome the problem is to use a decreasing sequence of the smooth parameter \(\sigma\). We decrease the value of \(\sigma\) gradually: for each value of \(\sigma_{i+1}\) gradient based algorithm is applied to minimize \(f(x, \sigma_{i+1})\), and the previous position of this gradient algorithm is the minimizer of \(f(x, \sigma_i)\) obtained for the previous value of \(\sigma_i\). As the above example has illustrated, when \(\sigma\) is large there may be multiple minima for \(f(x, \sigma)\), but no worry being trapped in local minimum as with the decrease of \(\sigma\) the number of minima will decrease and finally only one minimum exists in the feasible set. That means starting from an initial position in the feasible set, with the decrease of \(\sigma\), though may be different pathes but we will arrive at same destination--the global minima as shown by Theorem III.1 has proven.

Based on the main idea above, the final algorithm is given below:

1) Initialization:
   a) Let \(x_0\) be the linear solution obtained by direct linear transformation;
   b) Choose a suitable decreasing sequence for \(\sigma, [\sigma_0, ..., \sigma_m]\), for example \(\sigma_k = 0.1\sigma_{k-1}\).

2) while \(\|x_t - x_{t-1}\| > \epsilon\) do:
   a) Set \(\sigma = \sigma_k\);
   b) Minimizing the function \(f(x, \sigma) = \sigma \log(\sum_{i=1}^{n} \exp(h_i^2(x)/\sigma))\) through the gradient descent algorithm, starting from the previous solution \(x_{t-1}\) to obtain the current minima position \(x_t\) followed by verifying whether the position is in the feasible set.

G. Problems can be solved

As our algorithm aims at speeding up the \(L_\infty\) minimization problems, so the problems that can be solved through SOCP can be solved using smooth approximation for \(L_\infty\) including the problems ranging from multi-view triangulation, homography, camera resection to structure and motion with known rotations. Here we discuss the multi-view triangulation problem as an example however the same principle can be directly applied to other problems.

IV. Experimental Validation

In order to test the proposed algorithm, we have conducted experiments on both synthetic data and real images to show the performance and the efficiency of the algorithm implementation. A comparison with \(L_\infty\) using improved bisection

Figure 1. Contour plot of the approximate \(L_\infty\) under different parameter of \(\sigma\). (a) \(\sigma = 5\); (b) \(\sigma = 2\); (c) \(\sigma = 0.9\); (d) \(\sigma = 0.75\); (e) \(\sigma = 0.1\); (f) \(\sigma = 0.02\).
SOCP\(^1\) is presented to show the efficiency and performance. We use the relative percent error to quantize the effect of approximation which is defined as
\[
\delta = \frac{|e_{app} - e|}{e} \times 100\% 
\]
where \(e_{app}\) denotes the approximate \(L_\infty\) minimum of our algorithm and \(e\) denotes the \(L_\infty\) minimum. Another factor is the efficiency factor which is defined as the ratio of time consuming between improved bisection with SOCP and our smooth approximation. All the experiments are conducted using Matlab 7.4.0 without any optimization on a machine with Core 2 Quad, 2.4GHz processor and 4GB of memory under Windows XP operating system.

### A. Synthetic Data

We demonstrate various aspects of our algorithm such as scalability and running time using extensive synthetic data. These data are generated by creating random 3D points within the cube \([-1, 1]^3\) and then projecting to the images. The image coordinates are corrupted with iid Gaussian noise with different levels of variance. In all the graphs, the average of 200 trials are plotted.

In the first group of experiments, we test the algorithm on various number of views range from 2 to 300. From the Table I and Figure 3(a), we see that for most of the cases, our proposed algorithm reaches relative error below 1\% while speed up the implementation with a factor of more than 5 in the problem of multi-view triangulation.

To illustrate the performance of the algorithm under different levels of noise, we conduct experiments on various levels of noise on 20-view triangulation problem. The result is shown in Figure 3(b). The figure shows the approximation algorithm has achieved less than 1\% error under noise level from 0.005 to 0.04 pixels and there is no directly deterministic relation between noise level and approximation error.

![Figure 3](image1.png)

**Figure 3.** Performance of Approximate \(L_\infty\) on Synthetic Data.(a) Relative Error VS View Numbers; (b) Relative error VS noise level.

### B. Real Image

In order to test the proposed algorithm on real case, we have used two publicly available sequences.\(^2\) The “corridor" sequence consists of 11 images all of size \(512 \times 512\). There are 104 points correspondence visible in the images.

The other image set is a turnable sequence of a dinosaur, containing 36 images(of size \(576 \times 768\)) and, in total, 328 images correspondences with lots of occlusions. In the experiments, the number of views and points has been artificially varied to test the performance in different settings. The results for multi-view triangulation are given in Table II where we have conducted triangulation problem on various number of views. From the table, we learn that for these real image case we could speed up the algorithm implementation with a factor more than 10 while the approximation percent error is usually less than 1\%.

![Figure 4](image2.png)

**Figure 4.** Corridor and Dinosaur

![Figure 5](image3.png)

**Figure 5.** The Notre Dame dataset consists of 595 images of this cathedral

Finally we apply the algorithm on NotreDame data set \(^{[14]}\) to show its performance in large scale triangulation application. This data set contains 277887 3D points, 595 cameras and 1087113 image points, and involves triangulation problems ranging from 2 up to over 100 images. We have tested on 20% of the 277887 points triangulation. The distribution of the speed up factor and approximation error are shown in Figure 6 where the average approxi-

\(^1\)http://www.maths.lth.se/mathmatiklth/personal/fredrik/download.html
\(^2\)Available at http://www.robots.ox.ac.uk/˜vgg/data.html
Table I

<table>
<thead>
<tr>
<th>Views</th>
<th>$L_\infty$ Time(s)</th>
<th>Approximate $L_\infty$ Time(s)</th>
<th>Speed Up Factor</th>
<th>Relative Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.4152</td>
<td>0.0137</td>
<td>30.3714</td>
<td>3.6651</td>
</tr>
<tr>
<td>3</td>
<td>0.4776</td>
<td>0.0195</td>
<td>24.4520</td>
<td>1.7834</td>
</tr>
<tr>
<td>5</td>
<td>0.5192</td>
<td>0.0229</td>
<td>22.6826</td>
<td>0.6969</td>
</tr>
<tr>
<td>10</td>
<td>0.5481</td>
<td>0.0302</td>
<td>18.1292</td>
<td>0.5736</td>
</tr>
<tr>
<td>20</td>
<td>0.5839</td>
<td>0.0345</td>
<td>16.9095</td>
<td>0.4836</td>
</tr>
<tr>
<td>50</td>
<td>0.6546</td>
<td>0.0545</td>
<td>12.0043</td>
<td>0.4265</td>
</tr>
<tr>
<td>100</td>
<td>0.7560</td>
<td>0.0888</td>
<td>8.5110</td>
<td>0.3485</td>
</tr>
<tr>
<td>200</td>
<td>0.9502</td>
<td>0.1486</td>
<td>6.3948</td>
<td>0.3389</td>
</tr>
<tr>
<td>300</td>
<td>1.1583</td>
<td>0.2219</td>
<td>5.2204</td>
<td>0.3120</td>
</tr>
</tbody>
</table>

Table II

<table>
<thead>
<tr>
<th>Views</th>
<th>$L_\infty$ Time(s)</th>
<th>Approximate $L_\infty$ Time(s)</th>
<th>Speed Up Factor</th>
<th>Relative Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corridor:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.5133</td>
<td>0.0417</td>
<td>12.2975</td>
<td>1.0208</td>
</tr>
<tr>
<td>5</td>
<td>0.5558</td>
<td>0.0508</td>
<td>10.9326</td>
<td>0.2289</td>
</tr>
<tr>
<td>7</td>
<td>0.5712</td>
<td>0.0515</td>
<td>11.0931</td>
<td>0.3219</td>
</tr>
<tr>
<td>9</td>
<td>0.5919</td>
<td>0.0531</td>
<td>11.1523</td>
<td>0.1697</td>
</tr>
<tr>
<td>11</td>
<td>0.5972</td>
<td>0.0601</td>
<td>9.9375</td>
<td>0.1303</td>
</tr>
<tr>
<td>Dinosaur:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.3563</td>
<td>0.0125</td>
<td>28.5</td>
<td>0.0687</td>
</tr>
<tr>
<td>3</td>
<td>0.4656</td>
<td>0.0313</td>
<td>14.9</td>
<td>0.0789</td>
</tr>
<tr>
<td>7</td>
<td>0.5703</td>
<td>0.0401</td>
<td>14.218</td>
<td>0.0894</td>
</tr>
<tr>
<td>8</td>
<td>0.5766</td>
<td>0.0381</td>
<td>15.122</td>
<td>0.1017</td>
</tr>
<tr>
<td>9</td>
<td>0.6004</td>
<td>0.0420</td>
<td>14.2832</td>
<td>0.0984</td>
</tr>
<tr>
<td>10</td>
<td>0.6003</td>
<td>0.0404</td>
<td>14.8710</td>
<td>0.0759</td>
</tr>
<tr>
<td>11</td>
<td>0.5993</td>
<td>0.0446</td>
<td>13.4250</td>
<td>0.0864</td>
</tr>
<tr>
<td>12</td>
<td>0.6116</td>
<td>0.0458</td>
<td>13.3659</td>
<td>0.0764</td>
</tr>
<tr>
<td>13</td>
<td>0.6042</td>
<td>0.0521</td>
<td>11.60</td>
<td>0.0638</td>
</tr>
<tr>
<td>14</td>
<td>0.6094</td>
<td>0.0547</td>
<td>11.1429</td>
<td>0.0306</td>
</tr>
<tr>
<td>21</td>
<td>0.6875</td>
<td>0.0469</td>
<td>14.6667</td>
<td>0.0633</td>
</tr>
</tbody>
</table>

Figure 6. Experimental result on Notre Dame. (a) Histogram of speed up factor; (b) Histogram of approximation error(%).

V. CONCLUSIONS

In this paper we have proposed an efficient smooth approximation for the $L_\infty$-norm using log-sum-exp function. Theoretical analysis is given and experimental results on both synthetic and real image data have shown that the proposed algorithm achieves high precision approximation while speed up the implementation. In this paper, we show the application of the proposed algorithm on multiple view triangulation, however the same principle can be directly applied to other $L_\infty$ problems such as camera resection, homography estimate, structure and motion with known rotation etc.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China under key project number 60736007. Yuchao Dai would like to thank the China Scholarship Council Prof. Richard Hartley for his immeasurable discussion. 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


