2013 IEEE Conference on Computer Vision and Pattern Recognition

A Practical Rank-Constrained Eight-Point Algorithm for Fundamental Matrix Estimation

Yinqiang Zheng Shigeki Sugimoto Masatoshi Okutomi
Department of Mechanical and Control Engineering, Tokyo Institute of Technology
{zheng,shige}@ok.ctrl.titech.ac.jp mxo@ctrl.titech.ac.jp

Abstract

Due to its simplicity, the eight-point algorithm has been widely used in fundamental matrix estimation. Unfortunately, the rank-2 constraint of a fundamental matrix is enforced via a posterior rank correction step, thus leading to non-optimal solutions to the original problem. To address this drawback, existing algorithms need to solve either a very high order polynomial or a sequence of convex relaxation problems, both of which are computationally ineffective and numerically unstable. In this work, we present a new rank-2 constrained eight-point algorithm, which directly incorporates the rank-2 constraint in the minimization process. To avoid singularities, we propose to solve seven subproblems and retrieve their globally optimal solutions by using tailored polynomial system solvers. Our proposed method is noniterative, computationally efficient and numerically stable. Experiment results have verified its superiority over existing algebraic error based algorithms in terms of accuracy, as well as its advantages when used to initialize geometric error based algorithms.

1. Introduction

Fundamental matrix estimation from eight or more point correspondences is a classical and important problem in multiview geometry analysis [11], with widespread applications in camera calibration, 3D reconstruction, motion segmentation and so on. To properly account for the epipolar geometry, a fundamental matrix should be of rank-2, an essential but extremely challenging nonconvex constraint that every fundamental matrix estimation algorithm has to cope with.

1.1. Related Works

Ever since the pioneering work by Longuet-Higgins [17], a great variety of algorithms, taking into consideration the rank constraint via either posterior rank correction or interior rank-2 parametrization, have been proposed in the past few decades, see [1, 18, 23] for overall reviews. Among them, there is a category of robust estimation methods, like RANSAC [7] and MLESAC [22], that seek to estimate the fundamental matrix in the presence of outliers, *i.e*. mismatches. Although robustness to outliers is certainly of great interest, in this work, we concentrate on the other important category of methods that aim at estimating an accurate fundamental matrix from noisy and redundant correspondences after outliers, if any, have been properly removed.

Under the assumption of independently and identically distributed (i.i.d.) Gaussian noise, the best criterion in the sense of maximum likelihood estimation (MLE) is to minimize the reprojection error, also known as the gold standard [11]. Some noticeable algorithms, like [2], have been proposed for this regard, however, it is still widely believed that reprojection error minimization is a challenging task. It is even nontrivial to evaluate the minimum reprojection error compatible with a known fundamental matrix, *i.e*. the optimal two-view triangulation problem [10]. Therefore, a common practice is to minimize instead the first order approximation to the reprojection error, namely, the Sampson error, which is relatively easy to evaluate and minimize. Zheng *et al*. [24] designed a branch-and-contract algorithm to retrieve its global optimum, whose computational burden is, unfortunately, quite heavy. A more practical and frequently adopted compromise is to use some local optimization based methods, such as the constrained Levenberg-Marquardt (CLM) method [14] and the extended fundamental numerical scheme (EFNS) [13], usually initialized by some algebraic error based algorithms that will be reviewed below.

Due to their simplicity, algebraic error based algorithms have attracted a lot of attention in fundamental matrix estimation. Longuet-Higgins [17] proposed the first eight-point algorithm to estimate the (calibrated therein) fundamental matrix, which boils down to a simple eigenvalue factorization problem. Hartley [8] successfully defended this eight-point algorithm by introducing data normalization, leading to the well-known normalized eight-point algorithm. In [6],

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.203

Chojnacki and Brooks disclosed the underlying mechanism of Hartley's normalization from the viewpoint of statistical bias correction. Following this thread of idea, Kanatani *et al*. [12] recently revitalized the renormalization technique, referred to as hyper-renormalization, and proposed to correct the statistical bias further iteratively. Although this hyper-renormalization technique is shown to be effective in ellipse fitting [12], we have found that it fails to work as expected for fundamental matrix estimation. The main reason lies in that the rank-2 constraint of a fundamental matrix is ignored in the process of iterative bias correction. As in the normalized eight-point algorithm [8], one has to posteriorly correct the estimated fundamental matrix to be of rank-2, which leads to non-optimal solutions.

The importance of directly handling the rank-2 constraint has been justified via theoretical analysis in [20] and empirical results in [9, 19]. However, it is very hard to directly incorporate the highly nonconvex rank-2 constraint in the minimization process. In [9], Hartley proposed a projection technique to estimate the epipole in an iterative way, which suffers from the risk of getting trapped into local minimum. As shown in [19], to find the globally optimal solution, it is necessary to solve a 1,728th-order polynomial, which is surely impractical at all. Chesi *et al*. [5] proposed a sum-of-square (SOS) convex relaxation method to minimize the algebraic error with rank-2 constraint, in which one has to adopt bisection and solve a sequence of semidefinite programming (SDP) problems of fixed size. This iterative algorithm is time-consuming, and admits no guarantee of global optimality. As will be shown in the experiment section, it usually offers a quite roughly approximate solution due to finite accuracy of existing off-the-shelf SDP solvers[1]. More seriously, it suffers from some singularities caused by improper parametrization[2], thus inapplicable to some otherwise well-posed camera configurations (provided nondegenerate parametrization), such as pure camera translation along the horizontal axis. In a very recent online manuscript, Bugarin *et al*. [3] avoided these singularities by using the determinant constraint and advocated moment relaxation to solve the resulting polynomial optimization problem. The cost is to solving a hierarchy of SDP relaxation problems of increasing size, whose computational burden is even higher than the SOS relaxation in [5]. In addition, the global optimality certificate is to check whether the rank of the moment matrix is one, again a numerically sensitive operation.

1.2. Overview of Our Work

In this work, we propose a new rank-2 constrained eight-point algorithm for fundamental matrix estimation. Similar to [3,5], it is based on the algebraic error, and copes with the rank-2 constraint directly. By carefully investigating the linear dependence between the three columns of a fundamental matrix, we solve seven subproblems so as to avoid improper singularities and keep the resulting optimization problems tractable. According to the problem structure of their first-order optimality conditions, we use univariate/multivariate polynomial system solving techniques to find the guaranteed globally optimal solution without iterations. Especially, for high order multivariate systems, we customize a novel generalized eigenvalue solver, which successfully conquers numerical instabilities that would upset an automatically constructed Gröbner basis solver [15] and a polynomial eigenvalue solver with linearization [16].

Based on extensive experiment results, we have found that our rank-constrained eight-point algorithm is much more accurate than the normalized eight-point algorithm [8], especially when the number of correspondences is less than 40, or the noise level is higher than 1 pixel. Actually, in the majority of the test cases, its accuracy is sufficiently close to that of CLM [14], a popular Sampson error based iterative method. In addition, when using the solution from our rank-constrained eight-point algorithm to initialize CLM, it is possible to avoid some poor local minimum and improve the worst-case performance. When compared with the state-of-the-art SOS convex relaxation algorithm [5], our proposed method is noniterative, computationally efficient and numerically stable.

The remaining of this work is organized as follows. In Sec.2, we present the mathematical formulations of the rank-constrained eight-point algorithm, including the subproblems and the simplification process. Sec.3 shows the details of solving polynomial systems, with emphasis on the generalized eigenvalue factorization method for multivariate systems. We show extensive experiment results in Sec.4 and conclude this work in Sec.5.

2. Mathematical Formulations

2.1. Preliminaries

Throughout this work, we use uppercase letters (*e.g*., X), lowercase letters in bold (*e.g*., $\mathbf{x}$) and plain lowercase letters (*e.g*., x) to denote matrices, column vectors and scalars, respectively. The uppercase letter T is reserved for matrix or vector transpose.

Given a set of point correspondences in homogeneous coordinates $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}, i = 1, 2, \cdots, n, n \geq 8,$[3] between two uncalibrated perspective views of a rigid scene, the ob-

[1] The bisection path is determined by the sign of the maximum eigenvalue in Equation 15 of [5]. Due to finite accuracy of existing SDP solvers, it is numerically unstable to determine the sign. Even making one mistake in this process, the whole bisection path might be changed.

[2] We note that, by following our idea below, this problem could be alleviated by solving different subproblems. However, it is impractical due to the aforementioned inefficiency and instability of the SOS solution.

[3] The minimal case is with 7 correspondences, which is trivial to solve.

jective is to estimate a 3×3 fundamental matrix F satisfying the epipolar constraints

$$\mathbf{x}_i'^T F \mathbf{x}_i = 0, i = 1, 2, \cdots, n. \tag{1}$$

Due to certain measurement noise, the epipolar constraints in Eq.(1) could not be completely satisfied in general. Therefore, to estimate F, it is common to minimize the algebraic error defined by

$$\min_F \quad \sum_{i=1}^{n} (\mathbf{x}_i'^T F \mathbf{x}_i)^2. \tag{2}$$

Because of the epipolar geometry constraint that all epipolar lines must intersect at a point, *i.e.* the epipole, a fundamental matrix should be of rank-2. In order to avoid the drawback of posterior rank-2 correction in [8, 12], it is mandatory to incorporate the rank constraint in the minimization process. To this end, we can add the determinant constraint [3, 24], which leads to a challenging polynomial optimization problem. On the contrary, it is possible to parameterize the fundamental matrix properly such that the rank-2 constraint is interiorly satisfied. In this work, we adopt the right epipole parametrization, such that $F\mathbf{e} = \mathbf{0}$. Specifically, we denote the epipole $\mathbf{e} = \begin{bmatrix} x, & y, & z \end{bmatrix}^T$, and obtain the following linear dependency constraint

$$x \begin{bmatrix} f_1 \\ f_4 \\ f_7 \end{bmatrix} + y \begin{bmatrix} f_2 \\ f_5 \\ f_8 \end{bmatrix} + z \begin{bmatrix} f_3 \\ f_6 \\ f_9 \end{bmatrix} = \mathbf{0}. \tag{3}$$

It is straightforward to note that the variables x, y, z are subject to scale and sign ambiguities. Chesi *et al.* [5] proposed to rescale z to 1 so as to eliminate these ambiguities. However, this rescaling is singular when $z = 0$, which corresponds to some well-posed camera motions, like camera pure translation along the horizontal axis. Therefore, when using this epipole parametrization, special attention must be paid in order to avoid any possibility of additional singularities.

In addition, to avoid the trivial all-zero solution of F in Eq.(2) and Eq.(3), we need a scale constraint. Again, simply setting f_9 to 1 would cause additional singularities, as in [5].

In the following, we present seven subproblems so as to avoid improper singularities and keep the resulting optimization problems tractable.

2.2. Subproblems

2.2.1 Case-1: $f_3 = f_6 = f_9 = 0$

When $f_3 = f_6 = f_9 = 0$, the fundamental matrix F is naturally rank deficient. To avoid the all-zero solution, we add the unit-norm constraint $\sum_{k=1}^{9} f_k^2 = 1$, leading to the first subproblem (*SP-1*)

$$\begin{aligned} \min_F \quad & \sum_{i=1}^{n} (\mathbf{x}_i'^T F \mathbf{x}_i)^2, \\ s.t., \quad & \sum_{k=1}^{9} f_k^2 = 1, f_3 = f_6 = f_9 = 0. \end{aligned} \tag{4}$$

By analyzing its first order optimality condition, it is trivial to solve the optimization problem in Eq.(4) for (*SP-1*) via eigenvalue factorization.

2.2.2 Case-2: $f_3 \neq 0$

When $f_3 \neq 0$, we can rescale F such that $f_3 = 1$ to avoid the all-zero solution. Now we consider the following three possibilities:

(i) $x \neq 0$: Based on the scale and sign ambiguities of x, y, z, we can rescale x such that $x = 1$. The optimization problem corresponding to the second subproblem (*SP-2*) reads

$$\begin{aligned} \min_{F,y,z} \quad & \sum_{i=1}^{n} (\mathbf{x}_i'^T F \mathbf{x}_i)^2, \\ s.t., \quad & \begin{bmatrix} f_1 \\ f_4 \\ f_7 \end{bmatrix} + y \begin{bmatrix} f_2 \\ f_5 \\ f_8 \end{bmatrix} + z \begin{bmatrix} f_3 \\ f_6 \\ f_9 \end{bmatrix} = \mathbf{0}, f_3 = 1. \end{aligned} \tag{5}$$

Now, we denote $\mathbf{f} = \begin{bmatrix} f_1, f_2, f_4, \cdots, f_9 \end{bmatrix}^T$, Eq.(5) can be rewritten into the following problem in matrix form

$$\begin{aligned} \min_{\mathbf{f},y,z} \quad & \|M\mathbf{f} - \mathbf{b}\|_2^2, \\ s.t., \quad & N(y,z)\mathbf{f} = \mathbf{c}(y,z), \end{aligned} \tag{6}$$

in which M is a known $n \times 8$ matrix and $\mathbf{b}$ a known n-D column vector constructed from $\mathbf{x}_i$ and $\mathbf{x}_i'$, $i = 1, 2, \cdots, n$. $N(y,z)$ is a 3×8 matrix linearly parameterized by y, z, and $\mathbf{c}(y,z)$ a 8-D column vector. For simplicity, we abbreviate $N(y,z)$ and $\mathbf{c}(y,z)$ as N and $\mathbf{c}$, respectively.

Let us observe that Eq.(6) is a linear least square (LLS) problem with separable nonlinear equality constraints. In other words, assuming known y, z, we can obtain $\mathbf{f}$ by solving a LLS problem with linear equality constraints. Specifically, by introducing the Lagrange multiplier $\boldsymbol{\lambda}$, we can obtain $\mathbf{f}$ as follows

$$\mathbf{f} = (M^T M)^{-1}(M^T\mathbf{b} + \frac{1}{2}N^T\boldsymbol{\lambda}). \tag{7}$$

Considering that $N\mathbf{f} = \mathbf{c}$, we can solve $\boldsymbol{\lambda}$ by

$$\boldsymbol{\lambda} = 2G^{-1}\mathbf{c} - 2G^{-1}N(M^T M)^{-1}M^T\mathbf{b}, \tag{8}$$

where $G = N(M^T M)^{-1}N^T$.

After plugging **f** and $\boldsymbol{\lambda}$ into Eq.(6), the original optimization problem can be simplified into the following equivalent

$$\begin{aligned}\min_{y,z} \quad & \mathbf{v}^T G^{-1}\mathbf{v} + s \\ & = \frac{\mathbf{v}^T adj(G)\mathbf{v} + det(G)s}{det(G)} = \frac{p(y,z)}{q(y,z)},\end{aligned} \tag{9}$$

where $\mathbf{v} = \mathbf{c} - N(M^TM)^{-1}M^T\mathbf{b}$, $s = \mathbf{b}^T\mathbf{b} - \mathbf{b}^TM(M^TM)^{-1}M^T\mathbf{b}$, $det(G)$ represents the determinant of G, and $adj(G)$ the adjoint matrix of G.

The denominator $q(y,z)$ is a 6-order positive bivariate polynomial with respect to y and z, since G is positive definite. The nominator $p(y,z)$ is a 6-order nonnegative bivariate polynomial with respect to y and z, due to the facts that the objective function is nonnegative and $q(y,z) > 0$. $p(y,z) = 0$ corresponds to the ideal noise-free situation, which is unlikely to happen in case of general $n \geq 8$ noisy correspondences.

We defer the minimization of the fractional programming problem in Eq.(9) to Sec.3.2.

(ii) $x = 0, y \neq 0$: Based on the scale and sign ambiguities of x, y, z, we can rescale y such that $y = 1$. Now the optimization problem corresponding to the third subproblem (*SP-3*) is formulated as

$$\begin{aligned}\min_{F} \quad & \sum_{i=1}^{n}(\mathbf{x}_i'^T F\mathbf{x}_i)^2, \\ s.t., \quad & \begin{bmatrix} f_2 & f_5 & f_8\end{bmatrix}^T + z\begin{bmatrix} f_3 & f_6 & f_9\end{bmatrix}^T = \mathbf{0}^T, \\ & f_3 = 1.\end{aligned} \tag{10}$$

Similar to the aforementioned operations, given z, we can project out **f** in Eq.(10). Finally, the constrained optimization problem in Eq.(10) can be simplified (details omitted) into the following unconstrained fractional programming

$$\min_{z} \quad \frac{\tilde{p}(z)}{\tilde{q}(z)}, \tag{11}$$

in which $\tilde{q}(z)$ is a 4th-order univariate polynomial, while $\tilde{p}(z)$ a 6th-order univariate polynomial with respect to z. The details of minimizing Eq.(11) is deferred to Sec.3.1.

(iii) $x = 0, y = 0, z \neq 0$: According to Eq.(3), $f_3 = f_6 = f_9 = 0$, which contradicts against the assumption that $f_3 \neq 0$. Therefore, this possibility will not occur.

2.2.3 Case-3: $f_6 \neq 0$

When $f_6 \neq 0$, we can rescale F such that $f_6 = 1$ to avoid the all-zero solution. Similar to Case-2, we can obtain the fourth subproblem (*SP-4*) corresponding to the possibility $x \neq 0$, and the fifth subproblem (*SP-5*) corresponding to the possibility $x = 0, y \neq 0$. We omit the details of derivation to avoid repeating. Although the coefficients might be different, the polynomial formation of (*SP-4*) is the same as that in (*SP-2*). This holds for (*SP-5*) and (*SP-3*).

2.2.4 Case-4: $f_9 \neq 0$

Similarly, when $f_9 \neq 0$, we can rescale F such that $f_9 = 1$ to avoid the all-zero solution. Similar to Case-2, we can obtain the sixth subproblem (*SP-6*) corresponding to the possibility $x \neq 0$, and the seventh subproblem (*SP-7*) corresponding to the possibility $x = 0, y \neq 0$. The derivation details are omitted. Again, the polynomial formation of (*SP-6*) is the same as that in (*SP-2*) and (*SP-4*). This holds for (*SP-5*), (*SP-3*) and (*SP-7*).

2.3. Choosing the best solution

As will be shown in Sec.3, for each subproblem, we can find all its stationary points by solving the polynomial system arising from its first-order optimality condition. We choose the real stationary point with the smallest objective value as its globally optimal solution. As a consequence, there are at most seven solutions after solving all seven subproblems. In our implementation, we choose the one with the smallest *Sampson error* as our final solution.

3. Solving Polynomial Systems

To find the globally optimal solution of the fractional problem, we derive its first-order optimality condition, and identify all the stationary points.

3.1. Univariate Polynomial

For (*SP-3*), its first-order optimality condition reads

$$\frac{d\tilde{p}(z)}{dz}\tilde{q}(z) - \tilde{p}(z)\frac{d\tilde{q}(z)}{dz} = 0, \tag{12}$$

which is a 9th-order univariate polynomial. We calculate the eigenvalues of its companion matrix, and choose the real eigenvalue with the smallest objective value in Eq.(10) as the solution for (*SP-3*). The same method can be used to solve (*SP-5*) and (*SP-7*).

3.2. Multivariate Polynomials

The first-order optimality condition corresponding to (*SP-2*) is composed of the following two 11-order bivariate polynomials

$$\begin{aligned}\frac{\partial p(y,z)}{\partial y}q(y,z) - p(y,z)\frac{\partial q(y,z)}{\partial y} = 0, \\ \frac{\partial p(y,z)}{\partial z}q(y,z) - p(y,z)\frac{\partial q(y,z)}{\partial z} = 0.\end{aligned} \tag{13}$$

For multivariate polynomial systems, it is common to use the Gröbner basis technique. Noticeably, Kukelova *et al.* [15] developed an automatic program to build the Gröbner basis based polynomial system solver. Unfortunately, although there are only two variables in Eq.(13), the order is a bit too high. When using the automatic generator in [15], the size of the elimination template is 108×218, while that of the action matrix is of 110×110. Through extensive tests, we have found that the numerical stability of this automatically generated Gröbner basis solver is poor. The reason might lie in the relatively large size of the elimination template.

The other possible choice is the latest polynomial eigenvalue factorization technique [16], which can be further reduced to the general eigenvalue factorization problem using linearization. However, for our problems, the linearization operation is not generally applicable, since y or z might be zero. In addition, it drastically deteriorates the condition number and causes numerical instability, since the order of the polynomial system in Eq.(13) is relatively high.

To avoid the aforementioned numerical problems, we first introduce an auxiliary variable δ, such that

$$\min_{y,z,\delta} \quad \delta, \quad s.t., p(y,z) = \delta q(y,z). \tag{14}$$

At first sight, introducing δ might give rise to a more difficult constrained optimization problem. Actually, it helps avoid any explicit elimination and makes possible a stable polynomial system solver via generalized eigenvalue factorization.

After introducing the Lagrange multiplier ρ, we derive the first-order optimality condition of Eq.(14) as follows

$$\begin{aligned} &1 - \rho q(y,z) = 0, \\ &\rho(\frac{\partial p(y,z)}{\partial y} - \delta \frac{\partial q(y,z)}{\partial y}) = 0, \\ &\rho(\frac{\partial p(y,z)}{\partial z} - \delta \frac{\partial q(y,z)}{\partial z}) = 0, \\ &p(y,z) - \delta q(y,z) = 0. \end{aligned} \tag{15}$$

Let us note that $\rho = 1/q(y,z)$ and $q(y,z) > 0$, then $\rho > 0$. As a consequence, the first-order optimality condition in Eq.(15) can be simplified into

$$\begin{aligned} &\frac{\partial p(y,z)}{\partial y} = \delta \frac{\partial q(y,z)}{\partial y}, \\ &\frac{\partial p(y,z)}{\partial z} = \delta \frac{\partial q(y,z)}{\partial z}, \\ &p(y,z) = \delta q(y,z), \end{aligned} \tag{16}$$

which is composed of three polynomial equations with respect to three variables y, z, δ.

A key observation is that δ is linear in Eq.(16). Inspired by the hidden variable technique in [16], to solve Eq.(16), we can directly set δ as the hidden variable, resulting in a generalized eigenvalue factorization problem without the necessity of linearization.

Recall that both $p(y,z)$ and $q(y,z)$ are 6th-order polynomials. Let $\mathbf{u}$ denote a 28-D column vector containing all the monomials of y and z up to 6th-order. The three equations in Eq.(16) can be organized into

$$C_0\mathbf{u} = \delta C_1 \mathbf{u}. \tag{17}$$

Since there are only three equations in Eq.(16), we have to generate more equations by multiplying some monomials at both sides of the three equations. We have found that it is necessary to multiply all 45 monomials (up to 8th-order) at both sides of the third equation in Eq.(16), and all 55 monomials (up to 9th-order) at both sides of the first and second equation in Eq.(16). It leads to an expanded polynomial system

$$\tilde{C}_0\tilde{\mathbf{u}} = \delta \tilde{C}_1 \tilde{\mathbf{u}}, \tag{18}$$

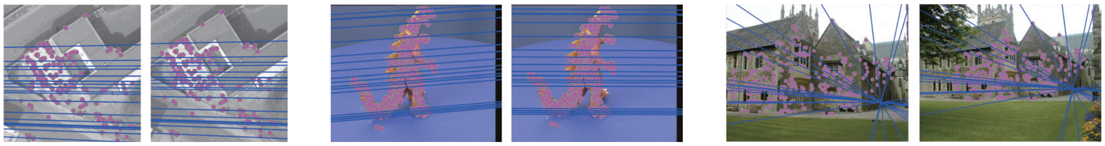
in which $\tilde{\mathbf{u}}$ is a 120-D column vector including all 120 monomials up to 14th-order, *i.e.* $\tilde{\mathbf{u}} = \left[1, y, z, y^2, yz, z^2, \cdots, z^{14}\right]^T$, and $\tilde{C}_0$ and $\tilde{C}_1$ are 155×120 coefficient matrices by reorganizing C_0 and C_1, properly and respectively.

Ideally, we should construct $\hat{C}_0$ by choosing 120 rows of $\tilde{C}_0$, such that $\hat{C}_0$ has the smallest condition number. In our implementation, we simply select a prescribed subset of rows from $\tilde{C}_0$ and $\tilde{C}_1$ to construct $\hat{C}_0$ and $\hat{C}_1$, respectively. As will be verified by experiments, it is of sufficient numerical stability. By applying generalized eigenvalue factorization on the 120×120 matrix pair $(\hat{C}_0, \hat{C}_1)$, we can find all the eigenvectors $\tilde{\mathbf{u}}$, from which y, z can be obtained after normalizing the first element of $\tilde{\mathbf{u}}$ to 1. Note that the generalized eigen-factorization overlooks the correlations among the elements in $\tilde{\mathbf{u}}$. Therefore, there might be some dummy solutions that violate the original polynomial system in Eq.(16). To find the globally optimal solution, we evaluate the objective value for each real solution pair (y, z), and choose the one with the smallest objective value (*e.g.*, the objective value in Eq.(9) for (*SP-2*)) as the optimal solution.

By using the same generalized eigenvalue solver, we can also solve the polynomial system from (*SP-4*) and (*SP-6*). Since their formations are the same, we record the track of constructing $\tilde{C}_0$ and $\tilde{C}_1$, and use the same solver for all these three subproblems.

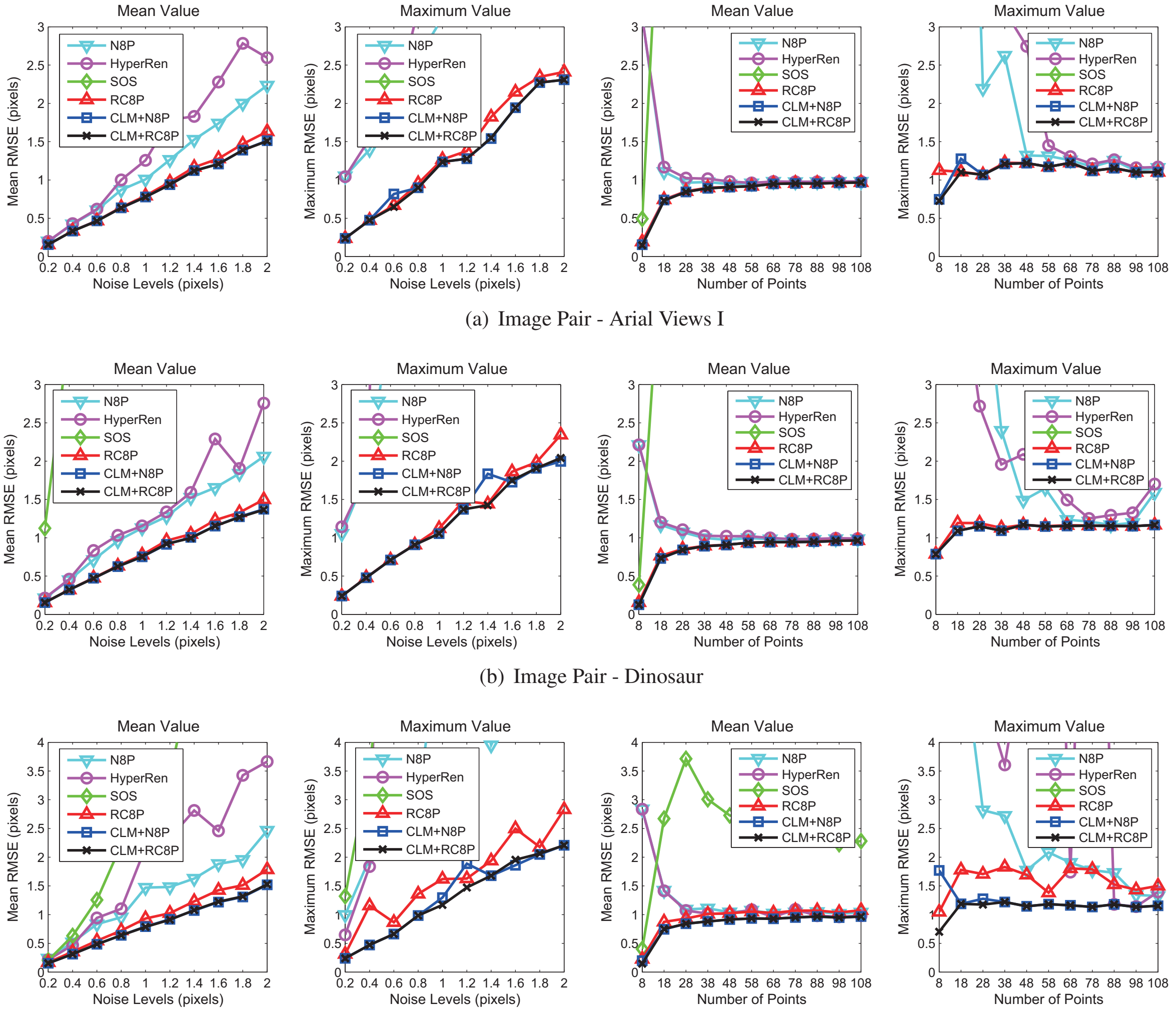
4. Experiment Results

In this section, we evaluate the performance of our proposed rank-constrained eight-point algorithm, denoted by **RC8P** in short, and compare it with the state-of-the-art algorithms, including the popular normalized eight-point algorithm (**N8P**) [8] and the latest iterative hyper-renormalization algorithm [12], denoted by **HyperRen**. In both algorithms, the rank-2 constraint is imposed via posterior rank-2 correction. The SOS-based convex relaxation by Chesi [5] is also considered, which is represented by



(a) Image Pair - Arial Views I (b) Image Pair - Dinosaur (c) Image Pair - Merton College III

Figure 1. Image pairs used to synthesize data. The pairs in (a), (b) and (c) are from the Aerial Views I, Dinosaur and Merton College III sequence, respectively. In each pair, the corrected noise-free feature points are shown in magenta, while 20 epipolar lines in blue.



(a) Image Pair - Arial Views I

(b) Image Pair - Dinosaur

(c) Image Pair - Merton College III

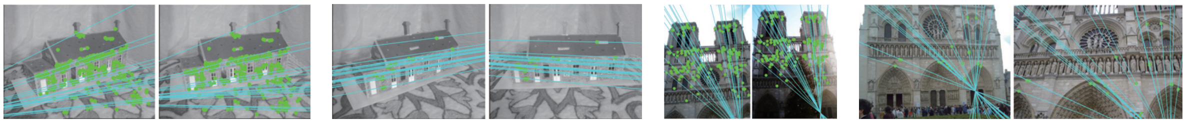
Figure 2. Experiment results using synthetic data. Note that SOS lies almost outside the visible region of every subfigure.

SOS. We also use the CLM algorithm [14] to minimize the Sampson error, which is initialized by **N8P** (thus named as **CLM+N8P**) and our **RC8P** (**CLM+RC8P**), respectively.

As for the evaluation criterion, we use the root mean square error (RMSE) on the basis of the Sampson distance, which is defined as $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\frac{(\mathbf{x}_i'^T F\mathbf{x}_i)^2}{(\tilde{\mathbf{e}})_1^2+(\tilde{\mathbf{e}})_2^2+(\hat{\mathbf{e}})_1^2+(\hat{\mathbf{e}})_2^2}}$, where $\tilde{\mathbf{e}} = F\mathbf{x}_i$ and $\hat{\mathbf{e}} = F^T\mathbf{x}_i'$. In addition, $(\tilde{\mathbf{e}})_k^2$ and $(\hat{\mathbf{e}})_k^2, k = 1, 2$, denote the square of the k-th element of $\tilde{\mathbf{e}}$ and $\hat{\mathbf{e}}$, respectively. It is widely recognized that the Sampson distance error is sufficiently close to, but much easier to evaluate than, the reprojection error. We normalize the data before estimation by following Hartley's normalization [8], while evaluate the RMSE after denormaliation.

4.1. Synthetic Data

To make sure that the synthetic data reflect real-world camera motions, we try to synthesize data on the basis of real images. Given an image pair, we first build tentative matches by matching SIFT points and remove potential outliers by using RANSAC together with the minimal 7-point



(a) Image Pair ♯1 (b) Image Pair ♯2 (c) Image Pair ♯3 (d) Image Pair ♯4

Figure 3. Image pairs used in experiments. In each pair, feature points are shown in green, while 20 epipolar lines estimated from **RC8P** are in cyan. The two pairs in (a) and (b) are from the House sequence, while the two pairs in (c) and (d) from the Notre Dame Cathedral community image collection.

Table 1. RMSE of competing algorithms for the four image pairs in Fig.3. The number of inliers is shown in the square bracket.

	RMSE (pixles)					RMSE (pixles)			
Method	♯1 [276]	♯2 [22]	♯3 [195]	♯4 [20]	Method	♯1 [276]	♯2 [22]	♯3 [195]	♯4 [20]
N8P	0.8168	5.8340	0.4216	1.6139	RC8P	0.7978	0.1121	0.3996	0.6824
HyperRen	0.8156	6.1521	0.4037	1.2739	CLM+N8P	0.7918	0.1296	0.3744	0.7243
SOS	9.7009	0.1205	1.5124	0.7616	CLM+RC8P	0.7918	0.1119	0.3744	0.5445

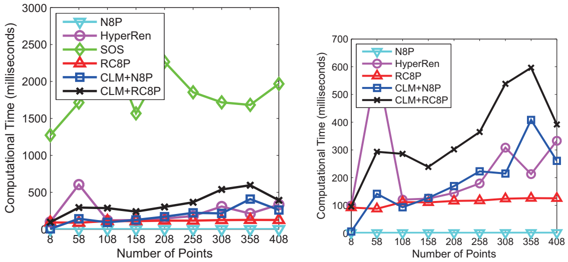


Figure 4. Average running time in milliseconds with respect to varying number of correspondences. A close-up is at the right side.

solver. Then, we estimate the fundamental matrix by using **CLM+RC8P**, and correct the correspondences to be noise-free via optimal triangulation [10]. These noise-free correspondences are used as test data.

In our experiments, we use three image pairs from the Oxford multiview geometry datasets[4]. Specifically, the pairs in Fig.1(a), Fig.1(b) and Fig.1(c) are from the Arial Views I, Dinosaur and Merton College III sequence, representing different motion types such as (near) translation, turntable motion and backward-forward motion.

We first investigate the performance of all competing algorithms with respect to varying noise levels. The number of point correspondence is fixed to be 20. We add zero mean Gaussian noise, with varying deviation σ in the range of [0.2, 2] pixels, onto the randomly selected (20) noise-free correspondences. At each noise level, we repeat 100 times, and report the mean and maximum value of RMSE in the 1st and 2nd column of Fig.2, respectively.

The performance with respect to varying number of correspondences n is also investigated. The noise level here is frozen to 1 pixel. We vary the n from 8 to 108. At each n, we run all algorithms 100 times using randomly selected n correspondences. The mean and maximum value of RMSE are shown in the 3rd and 4th column of Fig.2, respectively.

From Fig.2, we can see **N8P** is inaccurate, especially when the noise level is higher than 1 pixel, or the number of point correspondence is less than 40. The iterative bias correction technique in **HyperRen** fails to improve the accuracy, since the statistical analysis ignores the rank-2 constraint. The **SOS** algorithm provides poor estimation results due to its numerical instability as well as its risk of inaccuracy when the camera motion approaches its singularities. Our proposed **RC8P** algorithm is definitely better than **N8P**, **HyperRen** and **SOS**, and is even sufficiently close to CLM that minimizes the Sampson error. When higher accuracy is desirable, one can initialize CLM by using **RC8P**. Although **CLM+RC8P** and **CLM+N8P** have similar performance on the average, we can observe from the maximum value curves that the maximum RMSE of **CLM+N8P** is sometimes clearly larger than that of **CLM+RC8P**, which indicates that initializing CLM by using **RC8P** successfully contributes to avoiding some poor local minimum with large RMSE.

The computational efficiency is also of great interest. All competing algorithms are implemented in MATLAB, and run on a laptop with 2.8GHz CPU and 4GB RAM. We use SeDuMi [21] to solve the SDP relaxation problems in **SOS**. We report the average running time in milliseconds over 100 independent trials in Fig.4. Although our proposed **RC8P** is slower than **N8P**, it is much faster than the iterative **SOS** method. Taking about 100 milliseconds, **RC8P** is believed to be fast enough for most practical applications.

4.2. Real Images

We have also conducted numerous experiments using real images. After building tentative matches by matching SIFT points, we remove potential outliers through RANSAC. Then, we estimate the fundamental matrix by

[4] http://www.robots.ox.ac.uk/˜vgg/data/data-mview.html

the competing algorithms. Here, we only show the results in Table 1 for the four image pairs in Fig.3, including two pairs from the House sequence and the other two from the Notre Dame Cathedral community image collection[5]. From Table 1, we can observe again that our **RC8P** algorithm has better accuracy than **N8P**, **HyperRen** and **SOS**. When seeding CLM by **RC8P**, it is possible to avoid some poor solutions, *e.g.*, for pair ♯2 and pair ♯4, as shown in Table 1.

5. Conclusions

We have presented a new eight-point algorithm, in which the rank-2 constraint of a fundamental matrix is directly enforced in the process of minimization. Unlike the state-of-the-art sum-of-square relaxation method, our proposed algorithm does not suffer from additional singularities caused by improper parametrization. Although we have to handle seven subproblems, the proposed algorithm is the first practical eight-point algorithm with rank-2 constraint, when considering that it is noniterative, fast and capable of retrieving the global optimum in each subproblem. Numerical experiments, using both synthetic data and real images, have demonstrated its superiority over the popular normalized eight-point algorithm with posterior rank-2 correction as well as the convex relaxation method. The MATLAB source code and a detailed technical report are available at `https://sites.google.com/site/yinqiangzheng/`.

We have noted that the most time-consuming operation in our proposed algorithm is the generalized eigenvalue factorization of three 120×120 matrix pairs. In fact, we are only interested in the real eigenvalues (and their corresponding eigenvectors) in a narrow interval, rather than a full factorization. It is expected that our algorithm could be significantly accelerated by using the characteristic polynomial based technique in [4]. We leave the exploration of this acceleration technique as our future work.

References

[1] X. Armangue and J. Salvi. Overall view regarding fundamental matrix estimation. *Image and Vision Computing*, 21:205–220, 2003. 1

[2] A. Bartoli and P. Sturm. Nonlinear estimation of fundamental matrix with minimal parameters. *IEEE TPAMI*, 26(3):426–432, 2004. 1

[3] F. Bugarin, A. Bartoli, D. Henrion, J. B. Lasserre, J. J. Orteu, and T. Sentenac. Rank-constrained fundamental matrix estimation by polynomial global optimization versus the eight-point algorithm. *HAL:hal-00723015*, 2012. 2, 3

[4] M. Bujnak, Z. Kukelova, and T. Pajdla. Making minimal solvers fast. *In Proc. CVPR*, pages 1506–1513, 2012. 8

[5] G. Chesi, A. Garulli, A. Vicino, and R. Cipolla. Estimating the fundamental matrix via constrained least-squares: a convex approach. *IEEE TPAMI*, 24(3):397–401, 2002. 2, 3, 5

[6] W. Chojnacki and M. Brooks. Revisiting Hartley's normalized eight-point algorithm. *IEEE TPAMI*, 25(9):1172–1177, 2003. 1

[7] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24(6):381–395, 1981. 1

[8] R. Hartley. In defense of the eight-point algorithm. *IEEE TPAMI*, 19(6):580–593, 1997. 1, 2, 3, 5, 6

[9] R. Hartley. Minimizing algebraic error in geometric estimation problems. *In Proc. ICCV*, pages 469–476, 1998. 2

[10] R. Hartley and P. Sturm. Triangulation. *CVIU*, 68(2):146–157, 1997. 1, 7

[11] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, second edition, 2004. 1

[12] K. Kanatani, A. Al-Sharadqah, N. Chernov, and Y. Sugaya. Renormalization returns: Hyper-renormalization and its applications. *In Proc. ECCV*, pages 384–397, 2012. 2, 3, 5

[13] K. Kanatani and Y. Sugaya. Extended FNS for constrained parameter estimation. *In Proc. 10th Meeting on Image Recognition and Understanding*, 2007. 1

[14] K. Kanatani and Y. Sugaya. High accuracy computation of rank-constrained fundamental matrix. *In Proc. BMVC*, 2007. 1, 2, 6

[15] Z. Kukelova, M. Bujnak, and T. Pajdla. Automatic generator of minimal problem solvers. *In Proc. ECCV*, pages 302–315, 2008. 2, 4

[16] Z. Kukelova, M. Bujnak, and T. Pajdla. Polynomial eigenvlaue solutions to minimal problems in computer vision. *IEEE TPAMI*, 34(7):1381–1393, 2012. 2, 5

[17] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981. 1

[18] Q.-T. Luong and O. Faugeras. The fundamental matrix: theory, algorithms, and stability analysis. *IJCV*, 17(1):43–75, 1996. 1

[19] T. Migita and T. Shakunaga. One-dimensional search for reliable epipole estimation. *In Proc. Advances in Image and Video Technology*, pages 1215–1224, 2006. 2

[20] T. Migita and T. Shakunaga. Evaluation of epipole estimation methods with/without rank-2 constraint across algebraic/geometric error functions. *In Proc. CVPR*, 2007. 2

[21] J. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11-12:625–653, 1999. 7

[22] P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *CVIU*, 78(1):138–156, 2000. 1

[23] Z. Zhang. Determining the epipolar geometry and its uncertainty: a review. *IJCV*, 27(2):161–195, 1998. 1

[24] Y. Zheng, S. Sugimoto, and M. Okutomi. A branch and contract algorithm for globally optimal fundamental matrix estimation. *In Proc. CVPR*, pages 2953–2960, 2011. 1, 3

[5] http://phototour.cs.washington.edu/