

Revisiting Trifocal Tensor Estimation using Lines

Yubin Kuang, Magnus Oskarsson, Kalle Åström
 Centre for Mathematical Sciences
 Lund University, Sweden
 {yubin,magnuso,kalle}@maths.lth.se

Abstract—In this paper, we revisit the problem of estimating the trifocal tensor from image line measurements. With measurements of corresponding lines in three views, a linear method [1] requiring 13 lines was developed to estimate the trifocal tensor from which projective reconstruction of the scene is made possible. By further utilizing the nonlinear constraints on the trifocal tensor, we propose several new linear solvers that require fewer number of lines (10,11,12) than the previous linear method. We use methods based on algebraic geometry to incorporate the non-linear constraints in the estimation. We demonstrate the performance of the proposed solvers on synthetic data. We also test the solvers on real images and obtain promising results.

I. INTRODUCTION

As one of the key problems in computer vision, 3D reconstruction methods have been successfully applied to projective reconstruction within an uncalibrated framework [2]. In this paper we focus on projective reconstruction using line features. We assume no information on the camera motion or camera calibration and only image data is given. From this we seek projective solutions for the structure of the scene and the motion of the cameras. The motivation for using lines to get initial estimates, is that they often can be measured with much higher accuracy than points can. This is due to the fact that a line can be estimated using a large number of extracted points in the image. The accuracy of the initial solutions will then benefit from this. Our goal is to develop solvers that require fewer or near minimal number of lines which is beneficial for robust estimation schemes such as RANSAC or LMS, [3].

For three views and a projective reconstruction, the minimum number of points is six [4], [5], and the minimum number of lines is nine. There are no known algorithms for nine lines, but there are for minimal combinations of lines and points, cf. [6].

Linear algorithms have been developed for over-constrained solutions of at least 13 lines [7], and for combinations of lines and points [1]. Non-linear maximum likelihood estimators have also been developed for these over-constrained cases [8]. However, there has been little work on minimal cases for lines. Over-constrained solutions have also been developed for other camera models for lines, including calibrated cameras [9], [10], [11], [12], [13], and affine cameras [14], [15], [16].

In this paper, we focus on solving for the trifocal tensor using lines for three-view projective reconstruction. For over-constrained cases, by exploring the non-linear constraints on the trifocal tensors, we reduce the number of the required lines. We evaluate the performance of the proposed solvers on both synthetic and real images.

To ease the analysis and derivation of the solvers, we will assume in the following that the lines are in general positions.

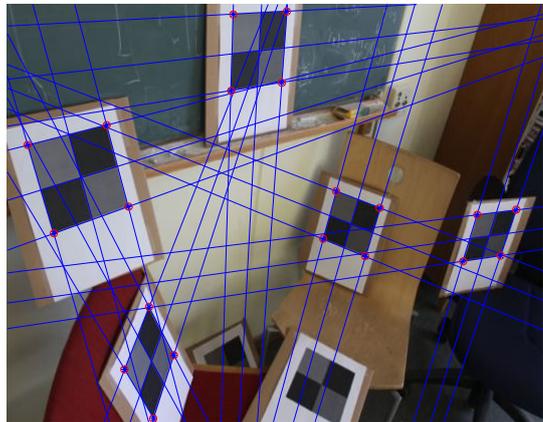


Fig. 1. An image of cardboards with detected lines (blue lines) and points (red circles).

The results will not hold for critical configurations, which do exist. For lines cf. [17], [18], and for points, see [19].

II. TRIFOCAL TENSORS WITH n LINES

In this paper we have pursued two lines of research in an attempt to provide solution algorithms for the problem. In this section we describe the first strategy, i.e. that of parametrizing the problem in terms of the trifocal tensor, [20].

For three uncalibrated cameras $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3\}$, the trifocal tensor is given by

$$T_i^{j,k}$$

The tensor has one covariant index i corresponding to the first image and two contra variant indices (j, k) corresponding to images 2 and 3. The tensor has 27 elements. With the scale of the tensor fixed ($T_3^{3,3} = 1$), it has 26 degrees of freedom. However, not all $3 \times 3 \times 3$ tensors, corresponding to the trifocal tensor of a camera triplet. There are several necessary and sufficient constraints [20] such that a trifocal tensor has only 18 degrees of freedom. There are two types of constraints. First each linear combination of the slices of T should have rank 2, i.e.

$$\det\left(\sum_i \lambda_i T_i^{\cdot\cdot\cdot}\right) = 0, \quad \forall \lambda. \quad (1)$$

This can be expressed as 10 constraints of degree 3 in T . This means that there is a null vector N_i for each slice so that $T_i^{\cdot\cdot\cdot} N_i = 0$. The second type of constraint is

$$\det([N_1 \mid N_2 \mid N_3]) = 0. \quad (2)$$

The null vectors can be expressed as degree two polynomials in T , so this second type of constraint is of degree 6. One

way to make polynomial constraints of this is to form the null vector as the cross product of two vectors within a slice. There are three ways of doing this, i.e.

$$N_1 = T_i^1 \times T_i^2, N_2 = T_i^1 \times T_i^3, N_3 = T_i^2 \times T_i^3.$$

If the slice T_i has full rank, then the first type of constraint is not fulfilled. However, for tensors with T_i the vectors formed above, will either be zero or if non-zero will coincide with the null-space. At least one of the three constructions above will provide the null space. Thus by forming all combinations of constraints

$$\det([N_1 \mid N_2 \mid N_3]) = 0$$

for all ways of forming N_1 , N_2 and N_3 , one will ensure that the last necessary and sufficient constraints are enforced. In total this gives 27 constraints of degree 6.

In this paper, we consider estimating the trifocal tensor estimation using image line correspondences. For each 3D line L , the corresponding image projections in the three views (a, b, c) give two linear constraints on the trifocal tensor T . This can be expressed in the following way. First, we select two points u and v on the line a in the first image. Then we enumerate the coordinates u^i, v^i, b_j, c_k . the constraints are then

$$\sum_i \sum_j \sum_k T_i^{j,k} u^i b_j c_k = 0, \quad \sum_i \sum_j \sum_k T_i^{j,k} v^i b_j c_k = 0. \quad (3)$$

In the following, we describe a novel scheme that utilizes combination of linear and nonlinear constraints to derive new linear solvers.

A. Solution Strategy with 12 Lines

With n lines we obtain $2n$ linear constraints using the line correspondences in (3). For $n \geq 13$, the trifocal tensor can be estimated linearly [1]. For $9 < n < 13$, the nonlinear constraints on a trifocal tensor [20] are needed to resolve the remaining $26 - 2n$ degrees of freedom. For the over-determined case where $n = 12$, there are 24 linear constraints and 2 degrees of freedom for the possible trifocal tensors. This means that we can eliminate 24 unknowns in T linearly using the constraints in (3). It is thus possible to parametrize the tensor T linearly using two unknowns $\mathbf{z} = (z_1, z_2)$. To solve for these 2 unknowns, we can utilize the 37 polynomial equations in the six unknowns with the non-linear constraints (1) and (2). In our experiments, it turns out that it is sufficient to use the constraints (1) only, which gives 10 cubic equations in 10 monomials. These equations can be expressed as the product of a coefficient matrix \mathbf{C} and a monomial vector as follows:

$$\begin{bmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,10} \\ c_{2,1} & c_{2,2} & \dots & c_{2,10} \\ \vdots & \vdots & \ddots & \vdots \\ c_{10,1} & c_{10,2} & \dots & c_{10,10} \end{bmatrix} \begin{bmatrix} z_1^3 \\ z_1^2 z_2 \\ z_1 z_2^2 \\ z_2^3 \\ z_1 z_2 \\ z_2^2 \\ z_1 \\ z_2 \\ 1 \end{bmatrix} = 0. \quad (4)$$

If we ignore the constraints between the monomials in the monomial vector, we can treat this as a linear system. Since the problem of determining the trifocal tensor using 12 lines is overdetermined, there is in general a unique null-vector to \mathbf{C} for the noise-free case. The elements of this null vector are the evaluation of the monomial vector at the solution \mathbf{z} . To extract the solution of z_1 and z_2 , we first find the null vector of \mathbf{C} (with e.g. SVD) i.e. the singular vector corresponding to the smallest singular value. Then we normalize the vector such that the last element (corresponding to the constant term) is 1. The values of z_1 and z_2 can then be extracted directly from the normalized vector. From the solution of z_1, z_2 , we can calculate the trifocal tensor $T = T(\mathbf{z})$ using linear substitution. The calculation of the three camera matrices \mathbf{P}_1 , \mathbf{P}_2 and \mathbf{P}_3 from T is straightforward [2]. From this, it is also straightforward to estimate the i^{th} line linearly as the singular vector corresponding to the smallest singular value of the matrix:

$$\mathbf{M}_i = [P_1^T l_{1i} \quad P_2^T l_{2i} \quad P_3^T l_{3i}].$$

Finally one may either perform bundle adjustment over the 3D line parameters only, in order to evaluate the quality of the estimate of the trifocal tensor, or perform bundle adjustment of the whole structure from motion setup, [21].

B. Solution Strategy with 11 Lines

For 11 line correspondences, we can obtain 22 linear constraints and the trifocal tensor can be parametrized linearly using 4 unknowns $\mathbf{z} = (z_1, \dots, z_4)$. With the constraints (1), we obtain 10 cubic equations in 35 monomials. To be able to formulate this as a linear system, an expansion step for the equation system is needed. This is also usually done in solving polynomial equations. We multiply these 10 equations with a set of low-order monomials such that the highest degree of the resulting equations is five. Further lowering the degree of resulting polynomials does not yield a solvable case. On the other hand, increasing the degree does not improve the numerical stability of the solver significantly and gives slower solvers. We find that it is a good trade-off to obtain 150 polynomial equations in 126 monomials. As for the 12-line case, we can rewrite the polynomial system as a product of a sparse coefficient matrix of size 150×126 . Similarly, we can find a unique null vector to this coefficient matrix which is the evaluation of the corresponding monomial vector at the solution.

C. Solution Strategy with 10 Lines

For 10 line correspondences, we can obtain 20 linear constraints and the trifocal tensor can be parametrized linearly using 6 unknowns $\mathbf{z} = (z_1, \dots, z_6)$. This yields a set of 10 equations in 84 monomials. Similarly, to extract the solutions of \mathbf{z} , we first need to expand the 10 cubic equations by multiplying low-order monomials. Specifically, we multiply each of the 10 cubic equations with a set of monomials such that the highest degree of the resulting equations is up to seven. It turns out that there are 210 such monomials.

This generates in total 2100 equations in 1716 monomials as the elimination template. We can see that this is a much larger linear system to solve and it is the minimal set of equations that yields a relative stable solution in our experiments.

In our experiments, further incorporating the higher-degree equations in (2) introduces even more unnecessary monomials in the polynomial systems. Thus we see that using only the equations in (1) gives a smaller elimination template.

In both the 11-line and the 10-line cases, it is straightforward to determine the camera matrices and 3D line parameters as described for the 12 line case.

III. EXPERIMENTAL RESULTS

A. Synthetic Experiments

To verify the numerical stability and noise sensitivity of the proposed solvers, we evaluate them on synthetic data. We first simulate the 3D lines in space by generating random 3D points in a cube of width 500 centered at the origin. Then we place 3 cameras with a distance to the origin of approximately 1000. The focal length of the cameras are chosen randomly and approximately 1000.

1) *Noise-Free Cases*: To study the numerical stability of our proposed solvers, we first test our solvers on noise-free data. Since all the solvers are overdetermined, normalization is essential for numerical stability [22]. For all the experiments, we normalize image lines such that the endpoints for all the lines in each image are of zero-mean and standard deviation 1. In our experiments, this normalization improves the numerical stability greatly. In Figure 2 (Left), we can see that the solvers for 11 and 12 lines are fairly stable with numerical behavior similar to the 13-line solver using only the linear constraints. On the other hand, the solver for the 10 lines case is not as stable as the other ones. It is probably related to the step of decomposition on a large matrix. The numerical stability of all the solvers can be improved by optimizing the set of monomials to multiply with the original set of equations. As for the speed of the solvers, both the 12 and 11 solvers run very fast, 5 *ms* and 16 *ms*, respectively on a Macbook Air with 8GHz RAM and a 1.8 GHz i5 core. Thus those two solvers are well-suited for RANSAC iterations. On the other hand, due to the large linear equation systems (2100×1716), the 10-line solver performs significantly slower. Depending on the method to find the null vector/smallest singular vector, the 10-line solver takes around 1.2 second using QR factorization, and 5 seconds using SVD. SVD performs slightly more stable in our experiments than sparse QR factorization. For all the experimental results, we have used SVD to compute the null vector.

2) *Sensitivity to Noise*: In this section, we test the solvers on data with varying levels of noise. For each noise level, we perturbed the positions of endpoints for the image lines with Gaussian noise. It is shown in the Figure 2 that all the solvers are fairly sensitive to the existence of noise. For instance, with a noise-level of 1 pixel, all the solvers provide solutions with fairly large reprojection errors. We evaluate the quality of these solutions by using them as initial solutions for the nonlinear optimization. In the non-linear bundle adjustment step, we minimize the reprojection errors of the lines, i.e. the sum of square distances of endpoints of measurement image lines to the reprojected lines. In the next section, we demonstrate that for real data, the solutions from different solvers actually provide reasonable initial solutions for the non-linear minimization.

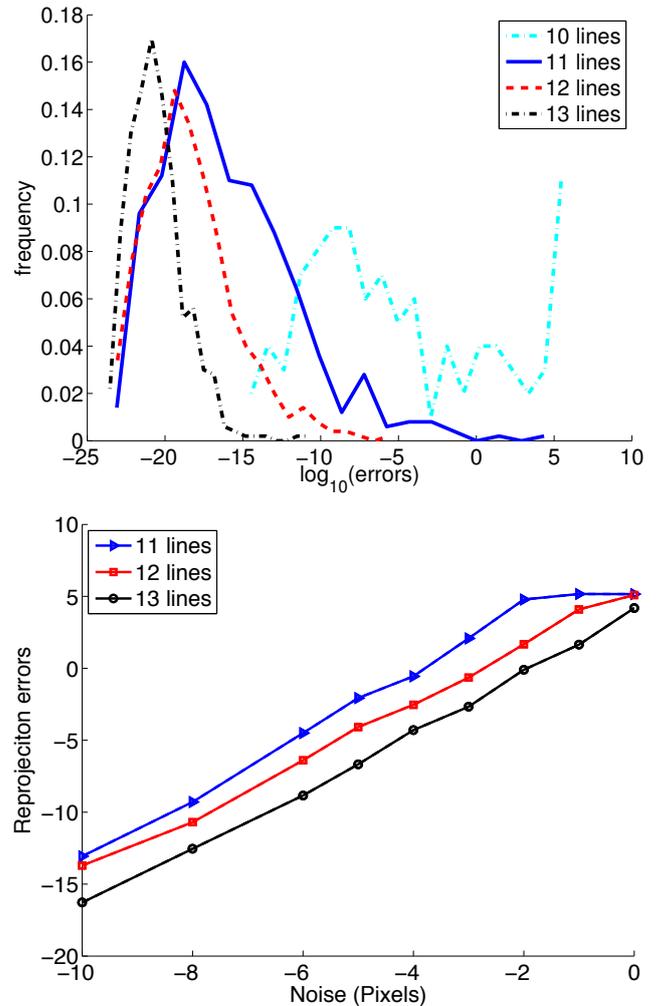


Fig. 2. Performance of solvers on synthetic data. *Left*: histogram of the reprojection errors for noise-free data over 500 runs; *Right*: medians of the reprojection errors with varying noise levels over 500 runs without bundle adjustment (the errors for the 10-line solver is not shown due to its large errors).

B. Real Experiments

To test the different solvers on real images, we took 3 images of seven cardboards placed in a non-planar configuration using a standard Canon EOS 50D camera. Each cardboard is attached with a pattern with dark and light squares for the ease of line detection. The lines were estimated by sub-pixel edge-detection, cf. [23]. This makes it possible to both estimate edge position and edge position uncertainty. Lines as well as the uncertainty in their parameters were then obtained by fitting the lines to the estimated edge points. The automatic line detection algorithm detected 6 lines for each of the card boards.

Nr. of Lines	10	11	12	13
Reproj. Errors (bundle)	3.90	1.11	1.53	2.37

TABLE I. AVERAGE REPROJECTION ERRORS (IN PIXELS) OF IMAGE LINES AFTER BUNDLE ADJUSTMENT OVER BOTH THE STRUCTURE AND MOTION FOR DIFFERENT SOLVERS ON 3 IMAGES.

The detected lines and images are illustrated in Figure 3. In

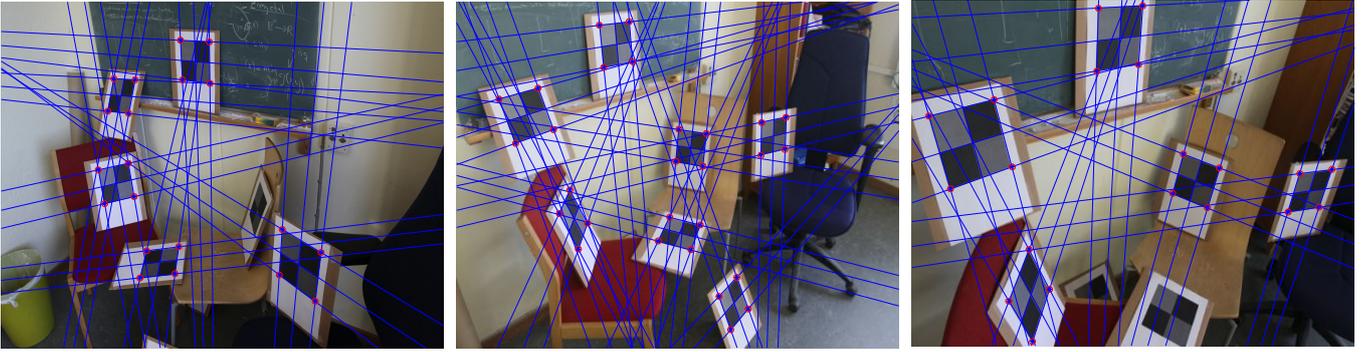


Fig. 3. Images of cardboards with detected lines (blue lines) and points (red circles).

this experiments, we choose image lines that are visible in all 3 images. We then solve for the trifocal tensor using different solvers. For each of the trifocal tensor recovered, we calculate the 3 projection matrices and then reconstruct the 3D lines. With the solutions from different solvers as initialization for bundle adjustment with a fixed number (1000) of iterations, the reprojection errors are summarized in Table 1. We can see all the solvers provide reasonable initial solutions for the bundle adjustment.

IV. CONCLUSIONS

In this paper, we present several solvers for estimating trifocal tensors using fewer number of lines than previous linear method [7]. We investigated the numerical stability of the different solvers on synthetic data. Two of the proposed solvers (11 and 12 lines) are as stable as the 13-line solver in [7], while the 10-line solver suffers from numerical instability. We also observed that all the solvers including the 13-line solver are very sensitive to the presence of noise. We verified the applicability of the solvers in providing initial solutions for bundle adjustment. It is shown in the real experiments that all proposed solvers provide good initial solutions comparable to the 13-line solvers, which requires larger number of lines. As future work, we will investigate the schemes for improving the numerical stability and noise sensitivity of the solvers (especially of the 10-line solver). We are also working on better parameterization for the minimal case of 9 lines such that one can derive efficient and stable polynomial solvers e.g. Gröbner basis solvers for this problem.

REFERENCES

- [1] R. Hartley, "Lines and points in three views and the trifocal tensor," *Int. Journal of Computer Vision*, vol. 22, no. 2, pp. 125–140, March 1997.
- [2] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [3] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–95, 1981.
- [4] A. Heyden, "Geometry and algebra of multiple projective transformations," Ph.D. dissertation, Lund Institute of Technology, Sweden, 1995.
- [5] L. Quan, "Invariants of six points and projective reconstruction from three uncalibrated images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 1, pp. 34–46, 1995.
- [6] M. Oskarsson, A. Zisserman, and K. Åström, "Minimal projective reconstruction for combinations of points and lines in three views," *Image and Vision Computing*, vol. 22, no. 10, pp. 777–785, 2004.
- [7] R. Hartley, "Projective reconstruction from line correspondences," in *Proc. Conf. Computer Vision and Pattern Recognition*. IEEE Computer Society Press, 1994, pp. 903–907.
- [8] P. Torr and A. Zisserman, "Robust parameterization and computation of the trifocal tensor," *Image and Vision Computing*, vol. 15, no. 8, pp. 591–605, 1997.
- [9] R. Holt and A. Netravali, "Uniqueness of solutions to structure and motion from combinations of point and line correspondences," *J. of visual communication and image representation*, vol. 7, no. 2, pp. 126–136, 1996.
- [10] M. E. Spetsakis and J. Aloimonos, "Structure from motion using line correspondences," *Int. Journal of Computer Vision*, vol. 4, no. 3, pp. 171–183, 1990.
- [11] T. Vieville and O. D. Faugeras, "Feed-forward recovery of motion and structure from a sequence of 2d-lines matches," in *Proc. 3rd Int. Conf. on Computer Vision, Osaka, Japan, 1990*, pp. 517–520.
- [12] J. Weng, T. Huang, and N. Ahuja, "Motion and structure from line correspondences: Closed-form solution, uniqueness, and optimization," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 3, 1992.
- [13] J. Weng, Y. Liu, T. Huang, and N. Ahuja, "Estimating motion/structure from line correspondences: A robust linear algorithm and uniqueness theorems," in *Proc. Conf. Computer Vision and Pattern Recognition, 1988*, pp. 387–392.
- [14] K. Åström, A. Heyden, F. Kahl, and M. Oskarsson, "Structure and motion from lines under affine projections," in *Proc. 7th Int. Conf. on Computer Vision, Kerkyra, Greece, 1999*, pp. 285–292.
- [15] R. Holt and A. Netravali, "Motion and structure from line correspondences under orthographic projection," *Int. J. of Imaging Systems and Technology*, vol. 8, no. 3, pp. 301–312, 1997.
- [16] L. Quan and T. Kanade, "Affine structure from line correspondences with uncalibrated affine cameras," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 8, pp. 834–845, August 1997.
- [17] T. Buchanan, "Critical sets for 3d reconstruction using lines," in *Proc. 2nd European Conf. on Computer Vision, Santa Margherita Ligure, Italy*, G. Sandini, Ed. Springer-Verlag, 1992, pp. 730–738.
- [18] S. Maybank, "The critical line congruences for reconstruction from three images," *Applicable Algebra in Engineering, Communication and Computing*, vol. 6, no. 2, pp. 89–113, 1995.
- [19] F. Kahl, R. Hartley, and K. Åström, "Critical configurations for n-view projective reconstruction," in *Proc. Conf. Computer Vision and Pattern Recognition, Hawaii, USA, 2001*.
- [20] A. Heyden, "Tensorial properties of multilinear constraints," *Mathematical Methods in the Applied Sciences*, vol. 23, pp. 169–202, 2000.
- [21] K. Åström, "Using combinations of points, lines and conics to estimate structure and motion," in *Proc. Symposium on Image Analysis, Uppsala, Sweden, 1998*.
- [22] R. I. Hartley, "In defense of the eight-point algorithm," *Pattern Analysis*

and Machine Intelligence, IEEE Transactions on, vol. 19, no. 6, pp. 580–593, 1997.

- [23] K. Åström and A. Heyden, “Stochastic analysis of scale-space smoothing,” in *13th International Conference on Pattern Recognition*, 1996, pp. 305–309.