A factorization algorithm for trifocal tensor estimation

L. Wang[1,2]
[1]School of EI&CE, Beijing University of Technology, Beijing 100124, China
wangliang@bjut.edu.cn

F.C. Wu[2]
[2]NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
fcwu@nlpr.ia.ac.cn

Abstract

Trifocal tensor encapsulates the geometric constraints between three views. It plays an important role in computer vision. However elements in measurement matrix of existing linear trifocal tensor estimation algorithms are products of several measurement data, which can amplify measurement error. The factorization algorithm of trifocal tensor estimation is presented, which can overcome this shortcoming. To overcome the deficiency that the extended solution space of factorization algorithm may result in an unstable solution, some modifications are adopted. Synthetic and real image data experiments show that the proposed algorithm is accurate and robust. This provides a new way different from normalized linear algorithm to improve performance of linear algorithm.

1. Introduction

Trifocal tensor encapsulates the geometric constraints between three views [1, 2, 3]. It plays an important role in computer vision. Owing to its wide applicability [4, 5, 6], the estimation of trifocal tensor has received considerable attention. Generally speaking, there exist two kinds of methods: the linear and the nonlinear iterative. The former have simple execution [6, 7]. The latter usually yield better results, but they are numerically more challenging [8, 9].

There are mainly two linear algorithms: the DLT (Direct Linear Transform) algorithm and normalized DLT algorithm. With a minor measurement error the DLT algorithm's result may significantly diverge from the correct one. Such a deficiency is mainly from two sources: (1) the poor condition of the DLT's linear equations [10]. (2) Measurement errors amplification. Since some elements in the DLT's measurement matrix are products of several measurement data, error amplification inevitably occurs.

Hartley proposed the normalized DLT algorithm [10] to overcome the first shortcoming by improving the condition number of DLT's linear equations. We proposed a factorization algorithm to overcome the second deficiency [11], and successfully applied it to fundamental matrix estimation.

The factorization algorithm factorizes measurement matrix into several factor matrices whose each element is either a measurement datum or a constant. Then construct new measurement matrix with introduced auxiliary variables. Finally solve corresponding linear least squares problem to gain the estimation. The factorization algorithm is linear. Most importantly, it does not involve amplification of measurement error, thus can effectively boost the robustness of the estimation. This shows a new way different from normalized DLT algorithm [6] to improve performance of linear trifocal tensor estimation algorithm.

This paper presents a factorization algorithm for trifocal tensor estimation. Experiments with synthetic and real image data show that it can consistently outperform the DLT algorithm, and is comparable with the normalized DLT algorithm. This shows the validity of the factorization algorithm.

This paper is organized as follows. In Section 2 the factorization algorithm for trifocal tensor estimation are given in detail. Experiments with synthetic data and real images are reported in Section 3. Some conclusions are in Section 4.

2. Trifocal tensor estimation

2.1. DLT algorithm

Corresponding image points in three images satisfy trilinear relations, which are encapsulated in the trifocal tensor. The trifocal tensor, $\mathbf{T}_i^{qr}$, is a $3\times3\times3$

homogenous tensor. If three 3×4 camera matrices are $\mathbf{P}=[\mathbf{I}\,|\,\mathbf{0}]$, $\mathbf{P}'=[a_q^i]$ and $\mathbf{P}''=[b_q^i]$, the trifocal tensor can be expressed as: $\mathbf{T}_i^{qr}=a_i^q b_4^r - a_4^q b_i^r$.

Let $\mathbf{X}=\begin{bmatrix}x\\y\\1\end{bmatrix}\leftrightarrow\mathbf{X}'=\begin{bmatrix}x'\\y'\\1\end{bmatrix}\leftrightarrow\mathbf{X}''=\begin{bmatrix}x''\\y''\\1\end{bmatrix}$ be a point correspondence across three views. Then, independent trilinear equations between three views can be expressed as the following vector-matrix form [6]:

$$\mathrm{A}_{4\times 27}\mathbf{t}_{27\times 1}=0_{4\times 1}, \tag{1}$$

where: $\mathbf{A}_{4\times 27}=\begin{bmatrix} x\mathbf{C}_1 & y\mathbf{C}_1 & \mathbf{C}_1\\ x\mathbf{C}_2 & y\mathbf{C}_2 & \mathbf{C}_2\\ x\mathbf{C}_3 & y\mathbf{C}_3 & \mathbf{C}_3\\ x\mathbf{C}_4 & y\mathbf{C}_4 & \mathbf{C}_4\end{bmatrix}$,

$\mathbf{C}_1=[-1\ \ 0\ \ x''\ \ 0\ \ 0\ \ 0\ \ x'\ \ 0\ \ -x'x'']$,

$\mathbf{C}_2=[0\ \ -1\ \ y''\ \ 0\ \ 0\ \ 0\ \ 0\ \ x'\ \ -x'y'']$,

$\mathbf{C}_3=[0\ \ 0\ \ 0\ \ -1\ \ 0\ \ x''\ \ y'\ \ 0\ \ -y'x'']$,

$\mathbf{C}_4=[0\ \ 0\ \ 0\ \ 0\ \ -1\ \ y''\ \ 0\ \ y'\ \ -y'y'']$,

$\mathbf{t}_{27\times 1}=[\mathbf{T}_1^{11},\mathbf{T}_1^{12},\mathbf{T}_1^{13},\mathbf{T}_1^{21},\mathbf{T}_1^{22},\mathbf{T}_1^{23},\mathbf{T}_1^{31},\mathbf{T}_1^{32},\mathbf{T}_1^{33},\ldots,\mathbf{T}_3^{33}]^T$.

Hence, given n point correspondences across three views, we can obtain 4n linear equations with respect to the trifocal tensor:

$$\mathbf{A}_{4n\times 27}\mathbf{t}_{27\times 1}=\mathbf{0}_{4n\times 1} \tag{2}$$

The DLT algorithm is to find the least squares solution of the linear equations (2) with the constraint $\|\mathbf{t}\|=1$.

2.2. Factorization algorithm

With factorization algorithm, the DLT's measurement matrix should be decomposed as product of factor matrices whose each element is a measurement datum or a constant [11]. Firstly let's recall some results of tensor algebra.

The tensor product of matrices $\mathbf{A}\in\mathbf{R}^{n\times m}$ and $\mathbf{B}\in\mathbf{R}^{p\times q}$ is defined as $\mathbf{A}\otimes\mathbf{B}=[a_{ij}\mathbf{B}]_{np\times mq}$.

If $\mathbf{A}\in\mathbf{R}^{1\times p}$, $\mathbf{B}_j\in\mathbf{R}^{1\times q}$, $j=1,2,\ldots,m$, then

$$\begin{bmatrix}\mathbf{A}\otimes\mathbf{B}_1\\ \mathbf{A}\otimes\mathbf{B}_2\\ \vdots\\ \mathbf{A}\otimes\mathbf{B}_m\end{bmatrix}=(\mathbf{I}_m\otimes\mathbf{A})\begin{bmatrix}\mathbf{I}_p\otimes\mathbf{B}_1\\ \mathbf{I}_p\otimes\mathbf{B}_2\\ \vdots\\ \mathbf{I}_p\otimes\mathbf{B}_m\end{bmatrix}, \tag{3}$$

$$\begin{bmatrix}\mathbf{I}_n\otimes\mathbf{A}\otimes\mathbf{B}_1\\ \mathbf{I}_n\otimes\mathbf{A}\otimes\mathbf{B}_2\\ \vdots\\ \mathbf{I}_n\otimes\mathbf{A}\otimes\mathbf{B}_m\end{bmatrix}=(\mathbf{I}_m\otimes\mathbf{I}_n\otimes\mathbf{A})\begin{bmatrix}\mathbf{I}_{np}\otimes\mathbf{B}_1\\ \mathbf{I}_{np}\otimes\mathbf{B}_2\\ \vdots\\ \mathbf{I}_{np}\otimes\mathbf{B}_m\end{bmatrix}. \tag{4}$$

Where I_k is the unit matrix of order k.

Let $\mathbf{U}'=\begin{bmatrix}1\\0\\-x'\end{bmatrix}$, $\mathbf{V}'=\begin{bmatrix}0\\1\\-y'\end{bmatrix}$, $\mathbf{U}''=\begin{bmatrix}-1\\0\\x''\end{bmatrix}$, $\mathbf{V}''=\begin{bmatrix}0\\-1\\y''\end{bmatrix}$.

Evidently, $\mathbf{U}'$ and $\mathbf{V}'$ are the vertical line and the horizontal line passing through $\mathbf{X}'$ in the second view, respectively; $\mathbf{U}''$ and $\mathbf{V}''$ are those passing through $\mathbf{X}''$ in the third view, respectively.

It is not difficult to see that

$\mathbf{C}_1=\mathbf{U}'^T\otimes\mathbf{U}''^T$, $\mathbf{C}_2=\mathbf{U}'^T\otimes\mathbf{V}''^T$,

$\mathbf{C}_3=\mathbf{V}'^T\otimes\mathbf{U}''^T$, $\mathbf{C}_4=\mathbf{V}'^T\otimes\mathbf{V}''^T$.

Thus, by the equation (3), the equation (1) can be written as

$$\mathbf{M}_{4\times 27}=(\mathbf{I}_4\otimes\mathbf{X}^T)\begin{bmatrix}\mathbf{I}_3\otimes\mathbf{U}'^T\otimes\mathbf{U}''^T\\ \mathbf{I}_3\otimes\mathbf{U}'^T\otimes\mathbf{V}''^T\\ \mathbf{I}_3\otimes\mathbf{V}'^T\otimes\mathbf{U}''^T\\ \mathbf{I}_3\otimes\mathbf{V}'^T\otimes\mathbf{V}''^T\end{bmatrix}.$$

By the equation (4), we have

$$\begin{bmatrix}\mathbf{I}_3\otimes\mathbf{U}'^T\otimes\mathbf{U}''^T\\ \mathbf{I}_3\otimes\mathbf{U}'^T\otimes\mathbf{V}''^T\end{bmatrix}=(\mathbf{I}_6\otimes\mathbf{U}'^T)\begin{bmatrix}\mathbf{I}_9\otimes\mathbf{U}''^T\\ \mathbf{I}_9\otimes\mathbf{V}''^T\end{bmatrix},$$

$$\begin{bmatrix}\mathbf{I}_3\otimes\mathbf{V}'^T\otimes\mathbf{U}''^T\\ \mathbf{I}_3\otimes\mathbf{V}'^T\otimes\mathbf{V}''^T\end{bmatrix}=(\mathbf{I}_6\otimes\mathbf{V}'^T)\begin{bmatrix}\mathbf{I}_9\otimes\mathbf{U}''^T\\ \mathbf{I}_9\otimes\mathbf{V}''^T\end{bmatrix}.$$

Thus we have

$$\mathbf{M}_{4\times 27}=(\mathbf{I}_4\otimes\mathbf{X}^T)\begin{bmatrix}\mathbf{I}_6\otimes\mathbf{U}'^T\\ \mathbf{I}_6\otimes\mathbf{V}'^T\end{bmatrix}\begin{bmatrix}\mathbf{I}_9\otimes\mathbf{U}''^T\\ \mathbf{I}_9\otimes\mathbf{V}''^T\end{bmatrix}. \tag{5}$$

Hence, given n point correspondences across three views, the DLT's measurement matrix $\mathbf{A}_{4n\times 27}$ can be decomposed as:

$$\mathrm{M}_{4\mathrm{n}\times 27}=\mathbf{P}_{4n\times 12n}\mathbf{Q}_{12n\times 18n}\mathbf{L}_{18n\times 27}, \tag{6}$$

where $\mathbf{P}_{4n\times 12n}=diag(\mathbf{I}_4\otimes\mathbf{X}_1^T,\ldots,\mathbf{I}_4\otimes\mathbf{X}_n^T)$,

$\mathbf{Q}_{12n\times 18n}=diag\left(\begin{bmatrix}\mathbf{I}_6\otimes\mathbf{U}_1'^T\\ \mathbf{I}_6\otimes\mathbf{V}_1'^T\end{bmatrix},\ldots,\begin{bmatrix}\mathbf{I}_6\otimes\mathbf{U}_n'^T\\ \mathbf{I}_6\otimes\mathbf{V}_n'^T\end{bmatrix}\right)$,

$\mathbf{L}_{18n\times 27}=\begin{bmatrix}\begin{bmatrix}\mathbf{I}_9\otimes\mathbf{U}_1''^T\\ \mathbf{I}_9\otimes\mathbf{V}_1''^T\end{bmatrix}\\ \vdots\\ \begin{bmatrix}\mathbf{I}_9\otimes\mathbf{U}_n''^T\\ \mathbf{I}_9\otimes\mathbf{V}_n''^T\end{bmatrix}\end{bmatrix}$.

Each element of $\mathbf{P}_{4n\times 12n}$, $\mathbf{Q}_{12n\times 18n}$ and $\mathbf{L}_{18n\times 27}$ is a measurement datum or a constant, and $\mathbf{P}_{4n\times 12n}$ ($\mathbf{Q}_{12n\times 18n}$, $\mathbf{L}_{18n\times 27}$) is only depend on the measurement data in the first view (second view, third view).

We introduce auxiliary variables $\mathbf{h}=\mathbf{L}_{18n\times 27}\mathbf{t}$ and $\mathbf{l}=\mathbf{Q}_{12n\times 18n}\mathbf{h}$, and construct a new measurement matrix:

$$\tilde{\mathbf{M}}_{34n\times(30n+27)}=\begin{bmatrix}\mathbf{L}_{18n\times 27} & -\mathbf{I}_{18n\times 18n} & \\ & \mathbf{Q}_{12n\times 18n} & -\mathbf{I}_{12n\times 12n}\\ & & \mathbf{P}_{4n\times 12n}\end{bmatrix}. \tag{7}$$

The factorization algorithm is to find the least squares solution of the following linear equation with the constraint $\| \tilde{\mathbf{t}} \| = 1$:

$$\tilde{\mathbf{M}}_{34n\times(30n+27)}\tilde{\mathbf{t}} = 0\,, \tag{8}$$

where $\tilde{\mathbf{t}} = [\mathbf{t}^T \quad \mathbf{h}^T \quad \mathbf{l}^T]^T$. Hence, by singular value decomposing of $\tilde{\mathbf{M}}_{34n\times(30n+27)}$, the trifocal tensor $\mathbf{T}_i^{qr}$ can be obtained up to a scale.

2.3 Some modifications

The coefficient matrix (7) of equations (8) obtained by factorization algorithm is a $34n\times(30n+27)$ sparse high-dimension matrix. Solving equations (8) with SVD (Singular Value Decomposition) requires employing huge vector space, which needs huge computation and may result in unstable result. So some modifications are adopted.

Equations (8) can be expressed as:

$$\begin{cases} \mathbf{L}_{18n\times27}\mathbf{t} - \mathbf{h} = 0 \\ \mathbf{Q}_{12n\times18n}\mathbf{h} - \mathbf{l} = 0 \\ \mathbf{P}_{4n\times12n}\mathbf{l} = 0 \end{cases} \tag{9}$$

Because $rank(\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}) = 27$, $\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}$ can be decomposed as with SVD:

$$\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27} = \mathbf{U}^T_{12n\times12n}\begin{bmatrix} \mathbf{D}_{27\times27} \\ 0_{(12n-27)\times27} \end{bmatrix}\mathbf{V}_{27\times27}\,.$$

So the former two equations of (9) can be expressed as:

$$\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}\mathbf{t} = \mathbf{l} \Leftrightarrow \begin{bmatrix} \mathbf{D}_{27\times27} \\ 0_{(12n-27)\times27} \end{bmatrix}\mathbf{V}_{27\times27}\mathbf{t} = \mathbf{U}_{12n\times12n}\mathbf{l}$$

$$\Leftrightarrow \begin{cases} \tilde{\mathbf{U}}_{27\times12n}\mathbf{l} - \mathbf{D}_{27\times27}\mathbf{V}_{27\times27}\mathbf{t} = 0 \\ \tilde{\mathbf{U}}_{(12n-27)\times12n}\mathbf{l} = 0 \end{cases}$$

where $\begin{bmatrix} \tilde{\mathbf{U}}_{27\times12n} \\ \tilde{\mathbf{U}}_{(12n-27)\times12n} \end{bmatrix} = \mathbf{U}_{12n\times12n}$. With $\tilde{\mathbf{U}}_{(12n-27)\times12n}\mathbf{l} = 0$ and $\tilde{\mathbf{U}}_{(12n-27)\times12n}\tilde{\mathbf{U}}^T_{27\times12n} = 0_{(12n-27)\times27}$, we have $\mathbf{l} = \tilde{\mathbf{U}}^T_{27\times12n}\mathbf{c}$, $\mathbf{c} \in \mathbf{R}^{27}$. Equation (9) can be rewritten as:

$$\begin{cases} \mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}\mathbf{t} = \mathbf{l} \\ \mathbf{P}_{4n\times12n}\mathbf{l} = 0 \end{cases} \Leftrightarrow \begin{cases} \mathbf{c} - \mathbf{D}_{27\times27}\mathbf{V}_{27\times27}\mathbf{t} = 0 \\ \mathbf{P}_{4n\times12n}\tilde{\mathbf{U}}^T_{27\times12n}\mathbf{c} = 0 \end{cases}$$

$$\Leftrightarrow \begin{bmatrix} \mathbf{D}_{27\times27}\mathbf{V}_{27\times27} & -\mathbf{I}_{27\times27} \\ 0 & \mathbf{P}_{4n\times12n}\tilde{\mathbf{U}}^T_{27\times12n} \end{bmatrix}\begin{bmatrix} \mathbf{t} \\ \mathbf{c} \end{bmatrix} = 0 \tag{10}$$

Then,

$$\mathbf{t} = (\mathbf{D}_{27\times27}\mathbf{V}_{27\times27})^{-1}\mathbf{v}1\,, \tag{11}$$

where $\mathbf{v}1$ is the singular value of $\mathbf{P}_{4n\times12n}\tilde{\mathbf{U}}^T_{27\times12n}$. So we can obtain $\mathbf{t}$ by SVD of $\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}$ and $\mathbf{P}_{4n\times12n}\tilde{\mathbf{U}}^T_{27\times12n}$, whose dimension is much less than $34n\times(30n+27)$. The computation is dramatically reduced.

The factorization algorithm of trifocal tensor has the following three main steps:

1) The measurement matrix $\mathbf{M}_{4n\times27}$ of DLT algorithm is decomposed as: $\mathbf{M}_{4n\times27} = \mathbf{P}_{4n\times12n}\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}$. Here each element of matrix $\mathbf{P}_{4n\times12n}$, $\mathbf{Q}_{12n\times18n}$ and $\mathbf{L}_{18n\times27}$ is a measurement datum or a constant, and is only depend on the measurement data of one view.

2) Decompose $\mathbf{Q}_{12n\times18n}\mathbf{L}_{18n\times27}$ and $\mathbf{P}_{4n\times12n}\tilde{\mathbf{U}}^T_{27\times12n}$ with SVD, construct new equations (10).

3) The estimated trifocal tensor can be obtained by equation (11) with SVD.

3. Experimental Evaluation

In this section, the factorization algorithm (FA) is compared with the DLT algorithm (DLT) and normalized DLT algorithm (NDLT). Here the FA works without normalization. Two measures are used to evaluate the performance of three algorithms: (1) the mean of distances from measured image points to epipolar lines corresponding to the estimated trifocal tensor (MD_1). (2) the mean of distances between measured image points and the reprojections computed with the estimated trifocal tensor (MD_2).

3.1. Simulation

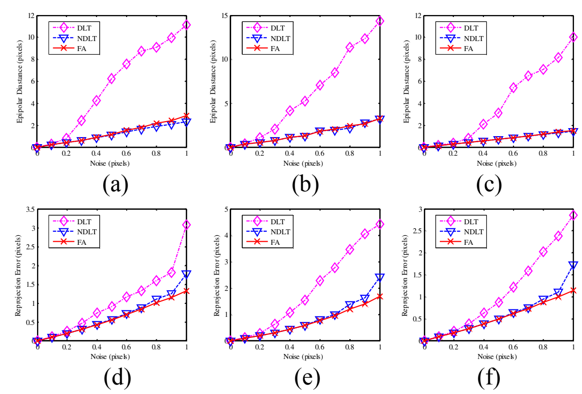

Fig.1 Experiment results with synthetic data

Synthetic data of reported simulation are generated as follows. Three cameras have the same intrinsic parameters: $(f_u, f_v, s, u_0, v_0) = (1000, 1000, 0, 512, 512)$, and image size is of 1024×1024. The world coordinate system coincides with that of the first camera. Space points are randomly generated in the cube of $[-40, 40]\times[-40, 40]\times[100, 180]$. The other two cameras' translation vectors and rotation matrices are generated randomly, which insure that the image of space points in the two image planes are within the square of $[0, 1024]\times[0, 1024]$. Gaussian noises with 0 mean and σ standard deviation are added to image points. Estimate the trifocal tensor from these point correspondences with three methods. On each noise level, 100 independent trials are performed and errors with the estimated trifocal tensor are computed. MD_1 of each view are shown in Fig.1 (a), (b) and (c). MD_2 of each view are shown as Fig. 1 (d), (e) and (f). We can see that the FA and the NDLT perform comparably. And the FA can consistently outperform the DLT.

3.2. Real image data experiment

The reported real image experiment is performed on the first three frames of the Corridor sequence from VGG, Oxford University [12]. There are 298 matches among three frames. Randomly choose a certain number of points from these matches to estimate the trifocal tensor with three algorithms. The number of randomly chosen points is 10, 12, 14, 16, 18 and 20 in turn. On each point number level, 100 independent trials are performed, and two measures mentioned above are computed over each turn. MD_1 curves of three algorithms for each frame are shown in Fig. 2(a), (b) and (c). And MD_2 curves are shown in Fig. 2(d), (e) and (f). We can see that the FA and the NDLT is comparable, and the former even outperforms the latter. The FA can consistently outperform the DLT.

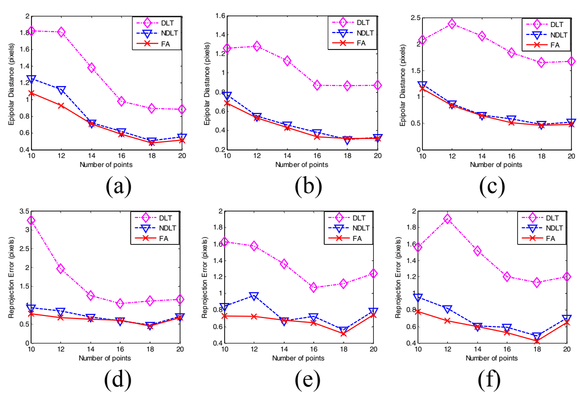

Fig.2 Experiment results with real image data

4. Conclusions

This paper presents a factorization algorithm to estimate trifocal tensor. To overcome the deficiency that the extended solution space of the original factorization algorithm may result in an unstable solution, some modifications are adopted. The proposed method reduces amplification of measurement error, and can boost effectively the robustness of the estimation. The results of synthetic data and real image data experiments show that the factorization algorithm can consistently outperform the DLT algorithm. And the performance of the factorization algorithm is comparable with that of the normalized DLT algorithm. Although not statistically optimal in the MLE sense, the proposed factorization algorithm provides results of satisfactory accuracy and is a completely linear algorithm, which can provide a start point of an iterative method. And more importantly, this method provides a new way to improve performance of linear algorithm different from the normalized DLT algorithm.

Acknowledgements

This work was supported by National Natural Science Foundation of China under grant No. 60575019.

References

[1] R. Hartley. Projective reconstruction from line correspondence. *In Proc. IEEE Conference on Computer Vision and Pattern Recognition,* 1994.

[2] A. Shashua and M. Werman. On the trilinear tensor of three perspective views and its underlying geometry. *In Proc. 5 th International Conference on Computer Vision,* Boston, 1995.

[3] A. Shashua. Trilinearity in visual recognition by alignment. *In Proc. 3rd European Conference on Computer Vision,* 479-484, 1994.

[4] M. Armstrong. *Self-calibration from Image Sequences.* PhD thesis, University of Oxford, England, 1996.

[5] P. A. Beardsley, P. H. S. Torr, and A. Zisserman. 3D model acquisition from extended image sequences. *In Proc. 4th European Conference on Computer Vision, LNCS 1065,* Cambridge, pages 683-695, 1996.

[6] R. Hartley, A. Zisserman. *Multiple view geometry in computer vision,* Cambridge university press, Cambridge, UK, 2000.

[7] R. Hartley. Lines and points in three views and the trifocal tensor. *Int. Journal of Computer Vision,* 22(2): 125–140, 1997.

[8] P. H. S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. *Image and Vision Computing,* 15(8): 591–605, 1997.

[9] O. Faugeras and T. Papadopoulo. A nonlinear method for estimating the projective geometry of 3 views. *In Proc. 6th Int. Conf. on Computer Vision,* 477–484, 1998.

[10] R. Hartley, In defense of the eight-point algorithm, *IEEE-T. PAMI*, 19(6): 580-593, 1997.

[11] F.C.Wu, Z.Y. Hu, and F.Q. Duan. 8-point algorithm revisited: factorized 8-point algorithm. In: *Proc. of the 10th International Conference on Computer Vision*, 488-494, 2005.

[12] http://www.robots.ox.ac.uk/~vgg/data