Five-Point Motion Estimation Made Easy

Hongdong Li and Richard Hartley

RSISE, The Australian National University.
Canberra Research Labs, National ICT Australia.
Email: firstname.lastname@anu.edu.au

Abstract

Estimating relative camera motion from two views is a classical problem in computer vision. The minimal case for such problem is the so-called five-point-problem, for which the state-of-the-art solution is Nistér's algorithm [9]. *However, due to the heuristic and* ad hoc *nature of the procedures it applies, to implement it is not so easy for non-expert users. This paper provides a much easier algorithm based on* hidden variable resultant *technique. Instead of eliminating the unknown variables one by one (i.e., sequentially) using the Gaussian method as in* [9], *our algorithm eliminates many unknowns all at once. Moreover, in the equation solving stage, instead of back-substituting and solve all the unknowns sequentially, we compute the minimal singular vector of the coefficient matrix, by which all the unknown parameters can be estimated simultaneously. Experiments on both simulation and real images have validated the new algorithm.*

1. Introduction

This paper studies the classical problem of estimating relative camera motion from two views. Particularly, we are interested in the *minimal case problem*. That is, to estimate the rigid motion from minimal *five* corresponding points of two views. Since the relative geometry between two view is faithfully described by an essential matrix E, which is an real 3 by 3 homogeneous matrix, the task is therefore equivalent to estimating the essential matrix from five points.

The classical way of estimating E is using the eight-point algorithm [1]. Because it is a linear algorithm and by which the obtained accuracy is reasonably good it is widely adopted as a benchmark algorithm. Considering this, why do we need a five-point algorithm? The justifications to this lie in both theoretic and practical aspects. In theory, the significance of having a minimal-case solver is quite obvious, which enable us a deeper understanding to the vision problem itself. In practice, a five-point algorithm also offers many benefits in reality. As demonstrated in [11], (1)this five-point algorithm suffers fewer types of "critical surface". For example, a arbitrary plane is not dangerous for the five-point algorithm; (2)when a 5-pt algorithm is used as a hypothesis-generator for RANSAC, its computational efficiency is much higher than 8pt algorithm; (3)somewhat surprisingly, the accuracy of the 5pt estimation is also higher than 8pt algorithm. This is because the minimal solver has better exploited all available geometric constraints of the problem.

In [3], we proposed a very simple resultant-based algorithm to solve the six-point focal-length problem [4], which has proven to be quite successful. In that paper we argue that that was *not* an individual success, but rather generally applicable. This paper aims at substantializing such argument. It will show that by using the resultant technique the five-point problem can also be solved easily and elegantly.

2. Some Historic Reviews

It is well known that from five points only one can estimate the relative motion between two calibrated views due to Kruppa, Demazure [5], Maybank and Faugeras [6], Hayde and Sparr [7] etc. [2]. However, despite the theoretical progresses, there was *no* practical algorithmic implementation of the theory until very recently [8] [9].

The state-of-the-art technique is Nistér's five-point algorithm proposed in [9] or [10], which is based on a modified Gaussian-Jordan elimination procedure. His algorithm is founded on Philip's previous work but made significantly improvements. However, the technique he applied, especially the elimination sequences he adopted, is quite *ad hoc* and heuristic. For example, different sequences of elimination may lead to different computer programs. A cautious reader may notice that even in Nister's two versions of implementations (cf.[9] and [10]) there are differences. Stèwinius later revised this algorithm by porting the problem into the $\mathbb{Z}_p$ domain [11]. The Gröbner basis technique is used to find suitable elimination sequences. This method is interesting, promising and quite general. However, their algorithm is not easy to re-implement, mainly because of the complicate and special Gröbner technique it uses.

This paper provides a (hidden-variable) resultant-based. It is so simple that is almost self-explained. Compared with existing algorithms, our new algorithm is easier (to implement) and more efficient (in computation). Non-expert user can apply it with comfort. Rather than eliminating irrelevant unknowns one by one (i.e., sequentially), our algorithm eliminates all the irrelevant unknowns all at once. Furthermore, in the equation solving stage, we show that by computing the minimal singular vector of the coefficient matrix, all the solutions can be found in one-shot.

3. Five-point Motion Estimation

We assume the reader is familiar with camera calibration and epipolar geometry (or, is referred to [1]). To save space we simply list some fundamental results without explanation.

Consider a camera, with constant intrinsic matrix K, observing a static scene. Two corresponding image points $\mathbf{m}$ and $\mathbf{m}'$ are then related by a fundamental matrix F:

$$\mathbf{m}^{'\mathbf{T}}\mathsf{F}\mathbf{m} = 0. \tag{1}$$

A valid F must satisfy the following *cubic* singularity condition:

$$\det(\mathsf{F}) = 0. \tag{2}$$

If the camera is fully-calibrated, then the fundamental matrix is reduced to an *essential matrix*, denoted by E, and the relationship becomes:

$$\mathsf{K}^{-\mathbf{T}}\mathsf{E}\mathsf{K}^{-1} = \mathsf{F}. \tag{3}$$

Since an essential matrix E is a faithful representation of the motion (translation and rotation, up to a scale), it has only five DOFs. Consequently, to be a valid essential matrix E, it must further satisfy two more constraints, which are characterized by the following result:

$$2\mathsf{E}\mathsf{E}^{\mathbf{T}}\mathsf{E} - tr(\mathsf{E}\mathsf{E}^{\mathbf{T}})\mathsf{E} = 0. \tag{4}$$

This actually gives nine equations in the elements of E, but only two of them are algebraically independent. Given five corresponding points, there are five epipolar equations eq.(1), plus the above nine equations and the singularity condition eq.(2), one therefore has enough equations to estimate the essential matrix.

4. Overview of Nistér's 5pt Algorithm

Nistér's 5pt Algorithm (based on [9]) proceeds as follows.

1. Writing down the epipolar equation eq.(1) for the five points, one can get a null-space representation $\mathsf{F} = x\mathsf{F}_0 + y\mathsf{F}_1 + z\mathsf{F}_2 + w\mathsf{F}_3$, where $\mathsf{F}_i, i = 0, 1, 2, 3$ are the null-space bases. Using the fact that F is homogeneous, without loss of generality, let $w = 1$.
2. Using the nine equations of eqs.(4), form a 9×20 coefficient matrix corresponding to a monomial vector:

$$[x^3, y^3, x^2y, xy^2, x^2z, x^2, y^2z, y^2, xyz, xy, \\ xz^2, xz, x, yz^2, yz, y, z^3, z^2, z, 1]. \tag{5}$$

   Then apply Gaussian-Jordan elimination to the 9×20 matrix, reduce it to an upper triangle form.
3. Use some *ad hoc* procedures to extract the determinants of two 4×4 matrices, followed by a second stage of elimination. Finally a 10-th degree univariate polynomial is obtained. Solving it, one then obtains 10 solutions of z.
4. Back-substituting those real roots, one can solve other unknowns one by one.
5. Recover the essential matrix, and extract the corresponding motion vectors of rotation and translation ([14]).

5. Derivation of Our New 5pt Algorithm

5.1 Hidden Variable Resultant

Our algorithm is based on the *hidden variable* technique, which is probably the best known resultant technique for algebraic elimination, and very easy to implement. The purpose of this technique is to eliminate variables from a multivariate polynomial equation system. Its basic idea is as follows.

Given a system of M homogeneous polynomial equations in N variables, say, $p_i(x_1, x_2, ..., x_N) = 0$, for $i = 1, 2, ..., M$. If we treat one of the unknowns (for example, x_1) as a *parameter* (that is, we *hide* the variable x_1), then by some simple algebra we can re-write the equation system as a matrix equation: $\mathsf{C}(x_1)\mathbf{X} = 0$, where the coefficient matrix C will depend on the *hidden variable* x_1, and the $\mathbf{X}$ is a vector space consisting of the homogeneous monomial terms of all other N-1 variables (say, $x_2, x_3, \cdots, x_N$). If the number of equations equals the number of monomial terms in the vector $\mathbf{X}$ (i.e. the matrix C is square), then the equation system will have non-trivial solutions *if and only if* $\det(C(x_1)) = 0$.

By such procedures, one thus eliminates N-1 variables all at once.

5.2 Algebraic Derivation

Notice eq.(4) and eq.(2)again. They are all cubic in x, y, z. For a moment let us treat the unknown z as a parameter (i.e., a *hidden variable*), and collect an

coefficients matrix C with respect to the other **two** variables x, y. The monomials we obtain span a vector space of:

$$\mathbf{X} = [x^3, y^3, x^2y, xy^2, x^2, y^2, xy, x, y, 1]^{\mathbf{T}}. \qquad (6)$$

Note its dimensionality is only ten.

Combining these nine equations with the singularity condition (eq.2), we now have totally ten equations in the above ten-dimensional monomial vector. Then we have a generalized linear matrix equation:

$$\mathsf{C}(z)\mathbf{X}(x, y) = 0.$$

Note that we have explicitly included the dependent variable z in the coefficient matrix and the monomial vector.

Recall that this matric equation will have non-trivial solutions *if and only if* the **determinant** of the coefficient matrix vanishes. That is:

$$\det(\mathsf{C}(z)) = 0. \qquad (7)$$

This determinant is better known as a *hidden-variable resultant*, which is an univariate polynomial of the hidden variable.

Inspecting the hidden-variable-resultant closely and carefully, one will (excitingly) find that terms whose degree is greater than ten have been precisely cancelled-out, and a precisely tenth-degree polynomial is left. As a result, we will obtain at most ten solutions to the five-point problem. This result accords precisely with previous result, but we achieve this via an easier and more straightforward way.

There are many methods to solve the univariate resultant equation, for example, the companion matrix method, or Sturm's bracketing method. In our experiments we simply use the first one, for it is easy to implement.

6. Solve Other Unknowns using SVD

Once z is computed by solving the resultant equation, the other two unknowns are usually solved by back-substituting the solved z. Notice that this back-substitution needs to be done multiple times, as in general we will have multiple roots of z. This approach is the one adopted by the previous work [10]. However, due to the multiplicities of each unknowns, the computational efficiency is quite low. By contrast, the hidden-variable technique used by this paper also suggests a much easier and more efficient way of solving other unknowns.

We notice that the monomial vector of eq.(6) has exhausted all the bi-variate combinations of variables x and y up to order three. Therefore, one can find the solutions of x and y simply as the right null-space of $\mathsf{C}(z)$. The null-space can be effectively computed through a numerical SVD decomposition. By contrast, the back-substitution approach used in [10][9] is much more involved.

Once all x, y, z have been solved, one can find the essential matrix E. And the motion parameters can be extracted easily using method reported elsewhere [1].

6.1 Algorithm Outlines

1. Write down the five epipolar equations of the five points.
2. Compute the null-space of the essential matrix.
3. Compute the symbolic determinant of the coefficient matrix $\mathsf{C}(z)$. Solving the determinant equation, one then find the solution z.
4. Back-substitute the real roots of z. Compute the null-space of $\mathsf{C}(z)$ and extract x, y from the null-space.
5. Recover the essential matrix, and extract the motion vectors.

7. Experiment Validation

7.1 Synthetic Data

We generate synthetic image and essential matrix using Torr's Matlab SFM Toolbox [15]. To resemble the real situation, the synthetic image size is set to be 512×512. Camera motions between two views are randomly drawn from a uniform distribution. No special attention has been paid to avoid the degenerate motion. The 3D points are located in general positions (though can be on a single plane).

In section 5.2, we have shown theoretically the determinant of $\mathsf{C}(z)$ is a 10-th degree polynomial in z. Now Our first experiment is used to validate this result.

From five corresponding points, after applying the proposed five-point algorithm, we obtain for example the following 10-th degree determinant equation:

$$\begin{gathered}\det(C(z)) = -.2996e^{-5}z^{10} - .3233e^{-4}z^{9} \\ +.9819e^{-3}z^{8} - .1547e^{-2}z^{7} + .4625e^{-3}z^{6} \\ +.1496e^{-4}z^{5} + .9100e^{-4}z^{4} - .1060e^{-2}z^{3} \\ +.3477e^{-3}z^{2} + .8115e^{-3}z - .3587e^{-5}\end{gathered}$$

Solving this equation using the companion matrix method, we then obtain ten complex roots.

To measure the estimation precision, we adopt the formula of [10], which is

$$\epsilon_{\mathsf{E}} = \min_i \min \left(\left\| \frac{\hat{\mathsf{E}}_i}{\|\hat{\mathsf{E}}_i\|} - \frac{\mathsf{E}_i}{\|\mathsf{E}_i\|} \right\|, \left\| \frac{\hat{\mathsf{E}}_i}{\|\hat{\mathsf{E}}_i\|} + \frac{\mathsf{E}_i}{\|\mathsf{E}_i\|} \right\| \right) \quad (8)$$

Fig-1 display the $\log_{10}$ error (using eq.(8)) distribution results. The top row figures are results by the 8pt linear algorithm (with Hartley's normalization and rank(2)modification) for noise-free case (in the left) and for $noise = 1.0 pixels$ case (in the right). The bottom row are our results. Both are the averages of 100 independent tests. We also compare the estimation errors under different noise conditions by linear algorithm and our algorithm, as shown

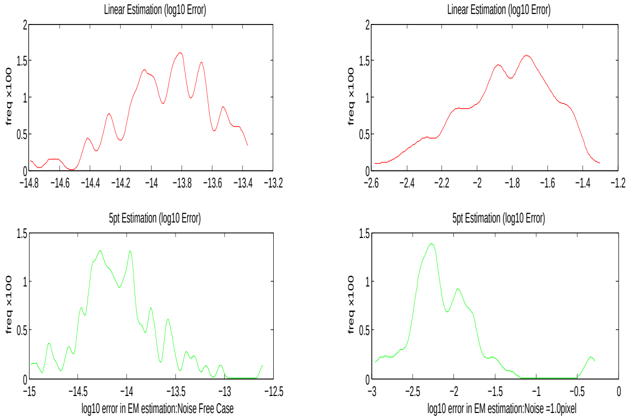

Figure 1. log10 error distribution (Left: noise free case; Right: noise=1.0 pixels; Top row: result by linear algorithm; Bottom: by the proposed 5pt algorithm).

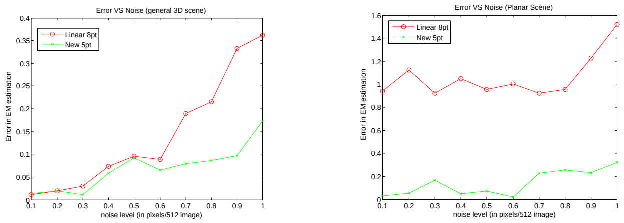

Figure 2. estimation error vs. noise level. (Left: a general 3D scene; Right: a planar scene; Red curve: by linear algorithm; Green curve: by the proposed 5pt algorithm)

in fig-2-Left. The results are the average of 100 times independent experiments. It is clear in both experiments that our algorithm is more accurate than the linear algorithm. Finally, we verify that the five-point algorithm works well for planar scene. we synthesize a single plane and test our algorithm again. A result is shown in fig-2-Right.We further compare the numerical performance of our new algorithm with Nistér's original algorithm, but have not found significant difference.

7.2 Tests on Real Images

We test our algorithm on some standard real images with known calibration information (see fig-3, courtesy of Oxford VGG and INRIA). Good results are obtained by our algorithm. Fig-3 shows the estimated epipolar lines superimposing on the images. In these experiments known camera calibration information is assumed. To use the 5pt algorithm more effectively, it is highly recommended to combine it with the RANSAC scheme [9].

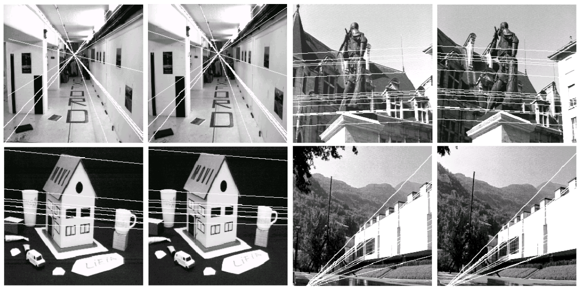

Figure 3. Some standard test images with estimated epipolar lines (using the 5-pt algorithm) superimposed on.

8. Conclusion

We have proposed a new algorithm for solving the five-point motion estimation problem. This algorithm follows very simple hidden-variable resultant idea, and very easy to implement. We believe that the simplicity suggests more deeper understandings to the essential-matrix. In addition, we wish the new five-point algorithm, combining with the powerful RANSAC scheme, will become a practical tool in structure and motion computation.

ACK: NICTA is funded through the Australian Government's Backing Australia's Ability Initiative, in part through the ARC.

References

[1] R.Hartley, A.Zisserman,Mutiview Geometry in computer vision, 2nd Edition, Cambridge University Press, 2004.

[2] R.I.Hartley, Estimation of Relative Camera Positions for Uncalibrated Cameras, In Proc. 2nd ECCV, 1992.

[3] Hongdong Li, A simpe solution to the six-point focal-elngth problem, In proc. ECCV-2006, May,2006.

[4] H.Stewénius, et al, A minimal solution for relative pose with unknown focal length, In Proc.CVPR-2005, 2005.

[5] M. Demazure, Sur Deux Problemes de Reconstruction, Technical Report No 882, INRIA, France, 1988.

[6] Faugeras, S. Maybank, Motion from Point Matches: Multiplicity of Solutions, IJCV, 4(3):225-246, 1990.

[7] A. Heyden, G. Sparr, Reconstruction from Calibrated Cameras - a New Proof of the Kruppa-Demazure Theorem, JMIV,vol-10:1-20, 1999.

[8] B. Triggs, Routines for Relative Pose of Two Calibrated Cameras from 5 Points, INRIA-Technical Report,INRIA, France, 2000.

[9] D.Nistér, An efficient solution to the five-point relative pose problem, in Proc. IEEE-CVPR-2003, 2003.

[10] D.Nistér, An efficient solution to the five-point relative pose problem, IEEE-T-PAMI, 26(6), 2004.

[11] David Nister and Henrik Stewenius, Using Algebraic Geometry for Solving Polynomial Problems in Computer Vision, short course at ICCV-2005.

[12] D.Cox, J.Little and D.O'shea, *Using Algebraic Geometry*, 2nd Edition, Springer, 2005.

[13] Long Quan, Zhong-Dan Lan, Linear N-Point Camera Pose Determination. IEEE PAMI-21(8): 774-780, 1999.

[14] Yi Ma, S. Soatto, J. Kosecka, and S. Sastry,An Invitation to 3-D Vision: From Images to Geometric Models, Springer-Verlag,2003.

[15] P.H.S. Torr, and D.W. Murray, The Development and Comparison of Robust Methods for Estimating the Fundamental Matrix, IJCV, pp.271300, v-24,1997.