

Fast and Robust Relative Pose Estimation for Forward and Sideways Motions

Johan Hedborg and Michael Felsberg
 Department of Electrical Engineering
 Linköping University, Sweden
 Email: hedborg@isy.liu.se

Abstract—We presented a new method for calculating the relative pose between two calibrated cameras. The 5-point method has been proved in numerous publications to be state-of-the-art in terms of accuracy, however for the important forward motion case it is significantly less accurate than the 8-point method. This paper present a novel method to calculate the relative pose which combines the accuracy of the 5-point method with the accuracy of the 8-point method when having a forwards motion. A secondary aspect of our method is the simplicity. Estimation the ego motion from 5 points can be done in real-time, but the implementation is complex and takes quite a bit of effort. Our approach is simpler and still has the potential to run in a real-time system.

I. INTRODUCTION

This paper studies the problem of positioning cameras and estimating structure from multiple views, known as Structure and Motion (SaM). The paper focuses on one part of SaM, namely the problem of position one camera relative another. When the internal structure of the camera/cameras is unknown the relative pose estimation problem is usually solved by the well known 7-point method [1]. Solving the relative pose problem is frequently used in many application types.

A related and equally important case is for calibrated cameras where the intrinsic camera parameters are known. In some cases they are estimated before from the images, but more common is when a special camera calibration pattern is used in a separate calibration step. In order to calibrate a camera it needs to have fixed optics and be available for calibration, typically the case when a camera is mounted on a robot or a vehicle. Having a calibrated camera makes the problem of determining the relative pose can be further constrained and thus more robust. Instead of needing 7 point correspondences between two images we only need 5 for solving the problem. Using 5 point correspondences to solve the pose estimation problem has been shown in numerous papers [2], [3], [4] to be significantly more robust then using other less constrained solutions i.e. 7-point or 8-point methods.

There are however drawbacks with the 5-point method. Foremost its sensitive to forward motion, camera motion of this kind are common to certain applications, i.e. autonomous navigation where a robot or a vehicle has a forward looking camera. This has been observed in several publications [3], [4], [2], they all show that for the forward motion case the 8-point method significantly out performances the 5-point method. Compared to the

7-point or 8-point it has significant larger compute complexity and is nontrivial to implement it in real-time. In this paper we aim to solve some of these issues.

II. PREVIOUS WORK

There are several proposed solutions for the 5-point problem ranging from Faugersa early work [5] to more recent work [6], [4], but the current state-of-the-art is published by David Nistér [3]. In [3] Nistér shows how to solve the 5-point problem for 2 or 3 views well within video real-time.

We propose an iterative method that handles the forward case as good as the 8-point method and the sideways case as good as the 5-point method. This is done seamlessly without switching between two cases, which is of importance because of the natural occurring continuum between the straight forward case and the sideways motion.

The implementation effort of the current state-of-the-art method is high. We have developed a method with less complexity and thus simpler to implement. An important aspect of the current state-of-the-art method is the compute complexity of the solver. In order to make a new method relevant we need it to reach similar execution speed.

To summarize this in 3 goals, ordered with respect to importance:

- 1) To handle the forward case better than the 5-point method, without any method switching.
- 2) Make a method that is simpler to implement.
- 3) Make the computational speed on pair with the state-of-the-art method [3].

III. FIVE POINT PROBLEM

The Epipolar constraint on corresponding points between two views are as follows

$$\mathbf{p}'^T \mathbf{E} \mathbf{p}. \quad (1)$$

Where $\mathbf{p} = [p_1, p_2, p_3]^T$ is a point in one image and $\mathbf{p}' = [p'_1, p'_2, p'_3]^T$ is the corresponding point in the second image. This can notated as a dot product between two vectors

$$\mathbf{q}^T \cdot \tilde{\mathbf{E}}, \text{ where} \quad (2)$$

$$\mathbf{q} = [p_1 p'_1 \ p_2 p'_1 \ p_3 p'_1 \ p_1 p'_2 \ p_2 p'_2 \ p_3 p'_2 \ p_1 p'_3 \ p_2 p'_3 \ p_3 p'_3]^T, \\ \tilde{\mathbf{E}} = [e_{11} \ e_{12} \ e_{13} \ e_{21} \ e_{22} \ e_{23} \ e_{31} \ e_{32} \ e_{33}]^T.$$

Given five points in one image $\mathbf{p}_1 = [x_1, y_1, 1]^T, \dots, \mathbf{p}_5 = [x_5, y_5, 1]^T$ and their corresponding points in a second image $\mathbf{p}'_1 = [x'_1, y'_1, 1]^T, \dots, \mathbf{p}'_5 = [x'_5, y'_5, 1]^T$. It is possible to construct 5 equations and form a linear equation system. The solution for the 5-point problem lies in the 4-dimensional right null-space of the equation system

$$\mathbf{A}_{5 \times 9} = \begin{pmatrix} \mathbf{q}_1^T \cdot \tilde{\mathbf{E}} \\ \vdots \\ \mathbf{q}_5^T \cdot \tilde{\mathbf{E}} \end{pmatrix} = \mathbf{0}, \quad (3)$$

and can be written as

$$\mathbf{E}^* = a\mathbf{X} + b\mathbf{Y} + c\mathbf{Z} + d\mathbf{W}. \quad (4)$$

There are additional conditions that has to be satisfied in order for $\mathbf{E}$ to be a true essential matrix:

$$\mathbf{E}\mathbf{E}^T\mathbf{E} - \frac{1}{2}\text{trace}(\mathbf{E}\mathbf{E}^T)\mathbf{E} = \mathbf{0}.$$

By further derivation one of the result is a constraint on the three singular values, the two biggest singular values should be equal and the last should be 0 $[\alpha, \alpha, 0]^T$.

A. The proposed method

Our approach is to solve the relative pose problem by an iterative solver, which is initiated with the result from a linear system using more than 5-points. The idea is to find the 5-point solution by iterating between two conditions, the singular value condition (rang 2 constraint) and that the solution should lie in the 4-dimensional linear subspace (4).

The method can be divided in to four steps, where the two first are the initial steps and the last two are the iteration steps. For the initiation solution the 8-point method is used, which uses 8 point correspondences resulting in a linear equation system and is very fast to solve.

$$\mathbf{A}_{8 \times 9} = \begin{pmatrix} \mathbf{q}_1^T \cdot \mathbf{E}_0 \\ \vdots \\ \mathbf{q}_8^T \cdot \mathbf{E}_0 \end{pmatrix} = \mathbf{0}. \quad (5)$$

The null-space to this equation system is an 8 dimensional vector that corresponds to our initial estimate $\mathbf{E}_0$ of the Essential matrix. An observation worth mentioning is that the Essential matrix has 9 elements in the matrix but only 8 degrees of freedom and is invariant to multiplication with an arbitrary scalar, and that is why we only need 8 points to uniquely determine the Essential matrix.

Next we pick 5 point correspondences out of the 8 and solve

$$\mathbf{A}_{5 \times 9} = \begin{pmatrix} \mathbf{q}_1^T \cdot \tilde{\mathbf{E}} \\ \vdots \\ \mathbf{q}_5^T \cdot \tilde{\mathbf{E}} \end{pmatrix} = \mathbf{0} \quad (6)$$

which results in the 4-dimensional subspace

$$\tilde{\mathbf{E}} = a\mathbf{X} + b\mathbf{Y} + c\mathbf{Z} + d\mathbf{W}$$

spanned by the four 3x3 matrices $\mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{W}$. This subspace contains the solution for the 5-point problem.

Here begins the iterative part of the algorithm (step 3), we start by enforce the rang 2 constraint on $\mathbf{E}_k$. This is achieved by a singular value decomposition of $\mathbf{E}_k$ where the new estimate of the Essential matrix is constructed by setting the singular values to $[1, 1, 0]^T$.

$$\mathbf{E}_k = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

$$\mathbf{E}_k = \mathbf{U}\mathbf{S}^*\mathbf{V}^T, \mathbf{S}^* = [1, 1, 0]^T$$

The result from the previous step $\mathbf{E}_k$ is then used to solve a, b, c, d from the over determined linear equation system (7). The nine elements $e_{11}, \dots, e_{33}$ in the vector on the right hand side are the elements of the matrix $\mathbf{E}_k$.

$$\begin{pmatrix} x_{11} & y_{11} & z_{12} & w_{12} \\ x_{12} & y_{12} & z_{12} & w_{12} \\ \vdots & \vdots & \vdots & \vdots \\ x_{33} & y_{33} & z_{12} & w_{33} \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} e_{11} \\ e_{12} \\ \vdots \\ e_{33} \end{pmatrix} \quad (7)$$

The solution a, b, c, d is then used to get the next estimate of the Essential matrix

$$\mathbf{E}_{k+1} = a\mathbf{X} + b\mathbf{Y} + c\mathbf{Z} + d\mathbf{W}.$$

Exit the iteration loop if the solution has converged to a desired precision, or if the upper iteration limit has been reached, else go to step 3.

IV. EFFICIENT IMPLEMENTATION

There are three matrix operations involved in the proposed method. First a solver for the basis of the null-space of a matrix, this is applied to both an 8x9 matrix and an 5x9 matrix. This is usually done with a Singular Value Decomposition (SVD), but it is more efficient (and accurate enough) to use an ordinary Gauss-Jordan elimination.

The second operation is a SVD of a 3x3 matrix. When the size of the matrix is fixed at 3x3 the singular values can be solved by solving a third grade polynomial, and this can be efficiently implemented in numerous ways [7], [8].

The last operation that is needed is a linear least square (LLS) solver to solve the 9x4 overdetermined equation system. The LLS solution can partly be solved before entering the iterative loop.

V. RESULT

A. Experiment

We have tested the algorithm on a synthetic data set, which consists of 250 randomly positioned world points viewed by a camera moving both sideways and forward. Gaussian noise ($N(0, \sigma)$) was added to the projected image points. The noise is varied between 0 and 1 standard deviation, where 1 standard deviation is equivalent to one pixel in an image with the size 1024x1024. The horizontal view-angle is set to 45°. The proposed method is compared to both the 5-point method and the 8-point method. All three were ran within a RANSAC [9] loop,

200 iterations were the solution that is closest to ground-truth is used. This was ran for 100 different world point configurations.

Figure 1 show results from sideways motion and figure 2 is equal but for forward motion. On the x-axis are the different noise levels, and on the y-axis are the relative position error. This clearly shows the differences with respect to accuracy between the 5-point and the 8-point method. It also shows that the proposed method performs equally to the 5-point in the sideways case and it performs equal to the 8-point,

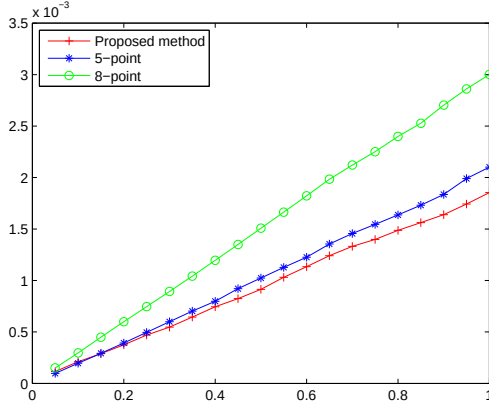


Fig. 1. Accuracy for forward case.

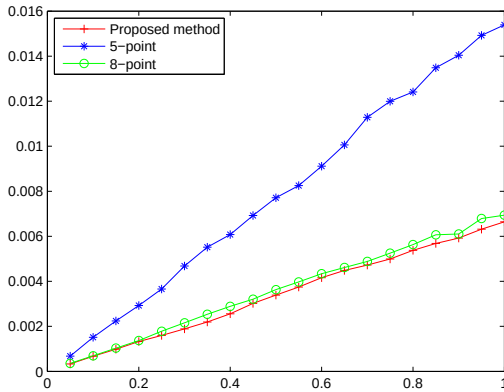


Fig. 2. Accuracy for sideways motion.

B. Compare to the minimal case

A valid argument against the proposed method is that it is not the minimal case for solving the pose estimation, giving it an disadvantage when running the solver in a RANSAC loop.

On first glance it looks like the 5-point method needs three less point correspondences to estimate a model within a RANSAC loop, this is however not entirely true. The 5-point solution results in up to 10 different solutions and to solve this ambiguity we need more information. There are several solutions to this problem,

Nister proposes in [3] to use an additional point making the count 6 points. The extra point can also be used for a preemptive cancellation before continuing with more expensive model evaluation [10].

If desired it would be possible to create an initial guess, for the proposed method, by using the 7 point method instead of the currently used 8 point method. Reducing the difference on needed points from three to one, and in terms of extra iterations for RANSAC. If the model assumption is to have either inliers or outliers (which is not an entirely valid assumption [11]) the number of extra iterations can be determined, see V-B.

RANSAC, around 200 iterations			
Inlier percentage	50%	60%	70%
Additional iterations	76%	3.4%	~ 0.0%

VI. CONCLUSION

We have presented a new method for calculating the relative pose between two calibrated cameras. Its biggest strength is the stable performance over several types of camera motions such as forward and sideways motions. This strength has been shown to be continues over varying noise levels. Another strength of the method is its simplicity, and it is relative straight-forward to implement with standard mathematical operations, and with little tweaking the performance is on pair with the current state-of-the-art method.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 215078, DIPLECS.

REFERENCES

- [1] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2003.
- [2] V. Rodehorst, M. Heinrichs, and O. Hellwich, "Evaluation of relative pose estimation methods for multi-camera setups," 2008, p. B3b: 135 ff.
- [3] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE TPAMI*, vol. 6, no. 26, pp. 756–770, June 2004.
- [4] H. Stewénus, C. Engels, and D. Nistér, "An efficient minimal solution for infinitesimal camera motion," in *Accepted for CVPR*, Zurich, Switzerland, Jun. 2007.
- [5] O. Faugeras and S. Maybank, "Motion from point matches: Multiplicity of solutions," vol. 4, no. 3, pp. 225–246, 1990.
- [6] H. Li and R. I. Hartley, "Five-point motion estimation made easy," in *ICPR (1)*, 2006, pp. 630–633.
- [7] G. Farneback, "Polynomial expansion for orientation and motion estimation," Ph.D. dissertation, Linköping University, Sweden, SE-581 83 Linköping, Sweden, 2002, dissertation No 790, ISBN 91-7373-475-6.
- [8] V. P. Vertechy, Rocco Castelli, *A Fast and Accurate Method for the Singular Value Decomposition of 3x3 Matrices*. Kluwer Academics Publishers, 2004, ch. methods in kinematics.
- [9] M. Fischler and R. 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. 381–395, 1981.
- [10] D. Nistér, "Preemptive RANSAC for live structure and motion estimation," *Machine Vision and Applications*, vol. 16, no. 5, pp. 321–329, December 2005.
- [11] O. Chum, J. Matas, and J. Kittler, "Locally optimized ransac," 2003, pp. 236–243.