

Research Article

Relative Pose Estimation Algorithm with Gyroscope Sensor

Shanshan Wei,^{1,2} Zhiqiang He,² and Wei Xie³

¹School of Computer Science and Technology, Beihang University, Beijing 100191, China

²Lenovo Group, Ecosystem & Cloud Services Business Group, Beijing 100085, China

³Lenovo Corporate R&D, SoC Center, Beijing 100085, China

Correspondence should be addressed to Shanshan Wei; wswss11986@qq.com

Received 26 July 2016; Accepted 27 October 2016

Academic Editor: Jose R. Martinez-de Dios

Copyright © 2016 Shanshan Wei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a novel vision and inertial fusion algorithm S^2fM (Simplified Structure from Motion) for camera relative pose estimation. Different from current existing algorithms, our algorithm estimates rotation parameter and translation parameter separately. S^2fM employs gyroscopes to estimate camera rotation parameter, which is later fused with the image data to estimate camera translation parameter. Our contributions are in two aspects. (1) Under the circumstance that no inertial sensor can estimate accurately enough translation parameter, we propose a translation estimation algorithm by fusing gyroscope sensor and image data. (2) Our S^2fM algorithm is efficient and suitable for smart devices. Experimental results validate efficiency of the proposed S^2fM algorithm.

1. Introduction

Camera relative pose estimation (CPE) is the estimation of camera extrinsic parameters, that is, camera 3D rotation parameter and 3D translation parameter. It is one of the key issues in computer vision and is widely applied in 3D scene reconstruction, augment reality, panorama, and digital video stabilization solutions.

The traditional solutions to CPE problem are based on image processing technique, that is, the visual methods. These solutions usually first extract feature correspondences between frame pairs and then model CPE problem as linear equations under the epipolar geometry constraint. In such a way CPE problem is transformed into an optimal solution problem. Hartley [1] proved the feasibility of using 8 pairs of feature correspondences to handle CPE problem and proposed 8-point (8 pt) algorithm to solve CPE problem for uncalibrated cameras. After that, in order to find simpler solutions, researchers proposed 7 pt algorithm [2], 6 pt algorithm [3, 4], and 5 pt algorithm [4–6]. These are mature traditional solutions based on image processing technique with accurate estimation results but complex calculation and slow computing speed. With the fast growing employment of MEMS sensors in smart devices, inertial-based solutions

for CPE problem have been tried recently. These solutions [7, 8] usually first perform CPE by visual and inertial methods individually and then adopt data filter to fuse the two results in order to obtain a more reliable estimation result. These two individual algorithms complement each other and improve the robustness of CPE. The disadvantage of this solution lies in the fact that it needs additional fusing time of the two results and thus reduces the efficiency.

It can be seen from the above analysis that visual solutions for CPE are mature and accurate but with complex computation and inertial solutions have been tried but without satisfactory results. Considering that rotation can be estimated fast and accurately by gyroscopes but there are no proper sensors to estimate translation accurately enough for CPE application; this paper proposes a visual and inertial fusion solution S^2fM (Simplified Structure from Motion). S^2fM divides the CPE problem into two parts: the rotation estimation part and the translation estimation part. It first employs gyroscope sensor to estimate the rotational information and then fuses the estimated rotational information with image data to estimate camera translation. Our solution relies on both gyroscope sensor data and image data but there might be time delay between them, so a calibration algorithm is necessary to align the gyroscope data and image data.

The camera focal length is also estimated in the calibration algorithm, which further simplifies the visual algorithm for translation estimation. Since the calibration needs to be done only once for each device, the main cost of our solution lies in the visual algorithm stage, which has been simplified to deal with only 3 feature pairs. Our main contributions are in two aspects. (1) Under the circumstance that no inertial sensor can estimate accurately translation parameter, we propose a translation estimation algorithm fusing gyroscope sensor and image data. (2) Our S^2fM algorithm is efficient and suitable for smart devices.

The rest of the paper is organized as follows. Section 2 reviews the related works. Section 3 describes the proposed solution. Section 4 presents the experimental results, and Section 5 draws the conclusion.

2. Related Work

Generally, CPE solutions can be classified into two major groups.

The first group of solutions are the traditional solutions. These solutions model CPE problem as linear estimation problem based on image feature correspondences under two-view geometry (mostly adopted) or multiview geometry [2]. A fundamental matrix will be determined by the feature correspondences, which can then be decomposed to give relative camera orientation and translation. Thus, CPE problem is transformed into the fundamental matrix estimation and decomposition problem. The fundamental matrix decomposition problem is called the minimal problem in computer vision, whose solutions are divided into two categories: one for calibrated camera and the other for uncalibrated camera. The essential issue in minimal problem is that how many correspondence points the solution needs at least. For uncalibrated camera, the solutions include 8 pt (point) algorithm, 7 pt algorithm, and 6 pt algorithm. For calibrated camera, the solution is 5 pt algorithm, since the relative pose parameter number is 5, that is, 3 for rotation and 2 for translation (up to an unknown scale factor). Hartley proved the validity of 8 pt algorithm [1] in 1997, in which the correspondence problem is supposed to have been solved. After 8 pt algorithm, in order to find simpler algorithm for uncalibrated camera, researchers tried to add constraints to the formulated equations and proposed 7 pt and 6 pt algorithms. In 2003, Hartley and Zisserman proposed 7 pt algorithm [2], which added the constraint that fundamental matrix and essential matrix are singular matrices. In 2005, Stewénius et al. proposed 6 pt algorithm [3] and in 2012 Kukulova et al. proposed a 6 pt algorithm based on polynomial eigenvalue [4]. 5 pt algorithms for calibrated camera include Nistér's 5 pt algorithm [5] in 2004, Li and Hartley's 5 pt algorithm [6] in 2006, and Kukulova's polynomial-eigenvalue-based 5 pt algorithm [4] in 2012. These widely used traditional visual solutions rely on image correspondence points, which may contain error and noise. RANSAC [9] method is usually introduced to reduce those error and noise. Brückner et al. [10] compared these traditional solutions above. The advantage of these traditional solutions is that they can generate accurate results but the disadvantage is that they are complex in computing: the

more correspondent points the algorithm needs, the slower its computing speed is.

Another group of solutions for CPE are the inertial-based solutions, which were not proposed until the MEMS sensors were accurate enough. In 2008, Gaida et al. [7] introduced a multisensor framework that combines gyroscopes, accelerometers, and magnetometers as a unit to estimate camera pose. Then a visual method is adopted to estimate camera pose too. Finally extended Kalman filter is adopted to fuse their results to obtain the final pose. One disadvantage of using accelerometers for translation estimation is that translation measurements from accelerometers are significantly less accurate than orientation measurements [11–13]. This is because gyroscope data need to be integrated only once to obtain the camera's orientation but accelerometer data need to be integrated twice to obtain the camera's translation, which will introduce too much noise that will affect significantly the accuracy. Miyano et al. [8] proposed an inertial and visual combination solution. It uses acceleration and a magnetic sensor to roughly estimate a camera pose and then searches the accurate pose by matching a captured image with a set of reference images. Corke et al. [14] made a survey on inertial and vision fusion solutions. These fusion solutions usually first perform CPE with separate inertial-based and visual-based solutions, generating respective results, and then fuse them by data filters. This is cooperation between inertial and visual methods. Its advantage lies in the robustness because the two methods can complement each other. Its disadvantage is the slow computing speed because of the fusion process.

This paper proposes an inertial and visual fusion solution called S^2fM for CPE. Different from existing fusion solutions which fuse inertial data and visual data in a cooperation manner, our solution fuses them in a division manner: it divides CPE problem into a rotation part and a translation part. Our solution first estimates camera rotation by gyroscopes and then uses it as known parameter in the visual method to estimate camera translation. Since the reliability and efficiency of gyroscopes for rotation estimation have been proven [7, 8, 11–14], they can significantly simplify the visual solution for camera translation estimation. As we will derive in the next section, only 3 pairs of correspondence points are needed for translation estimation. Different from Hartley and Nistér, who made great efforts to find the solution of the established equation sets, our focus is on proposing an inertial and visual fusion solution to solve CPE problem efficiently under the circumstance that no inertial sensor can estimate accurately enough translation parameter.

3. Proposed Solution

This section describes our proposed solution which is under the pinhole camera model and consists of three steps: camera and gyroscope calibration, estimation of camera rotation, and estimation of camera translation.

3.1. Camera and Gyroscope Calibration. Our solution first calibrates the camera and gyroscope and the calibrating contents are as follows:

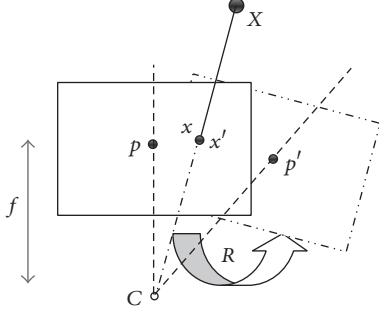


FIGURE 1: Camera 3D rotation motion model.

- (1) Gyroscope noise processing
- (2) Camera focal length f calibration (in pixel unit)
- (3) The delay t_d between the gyroscope and frame sample timestamps

3.1.1. Gyroscope Noise Processing. Raw MEMS gyroscope data need to be processed to remove zero-drift and random noise. We take the general statistical method to remove zero-drift. Put the device in static position for a period of time to get statistics of zero-drift and subtract it from the source data to obtain a series of stable, zero-expectation, and normally distributed random noise. Those random noises are then modeled through time sequence method and depressed by Kalman filter to give usable gyroscope data.

3.1.2. Calibrating Algorithm. After noise processing, the gyroscope data can be used for the calibrating operation. The purpose of our calibrating algorithm is to calibrate the parameters t_d (delay between the gyroscope and frame sample timestamps) and f (camera focal length). We take a similar calibrating algorithm as in Miyano et al. [8] under camera rotation model (as Figure 1 shows) but with an optimized objective function.

As Figure 1 shows, a camera moves under the rotation model with its optical center unchanged [15]. A point X in world coordinate and its projected image coordinate x have the following mapping relation:

$$\begin{aligned} x &= KX, \\ X &= \lambda K^{-1}x. \end{aligned} \quad (1)$$

And we have the following:

- (1) K is the camera intrinsic:

$$K = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

where f is camera focal length to be recovered and (u_0, v_0) is the projected point of the optical center, which is $(0, 0)$ in the proposed solution, since we set the image center as the optical center.

(2) λ is an unknown scaling factor, since the translation can only be determined up to scale in CPE.

Under this rotation model, the relationship between image points in a pair of frames for two different camera orientations (as Figure 1 shows) can be derived. For a scene point X the projected points x and x' in the image plane of two different frames would be

$$\begin{aligned} x &= KRX, \\ x' &= KR'X, \end{aligned} \quad (3)$$

where R and R' are 3×3 rotation matrices representing the rotational parameters at two frame timestamps, which will be detailed in the next section. Rearrange (3) and substitute for X ; we get a mapping of all points in one frame to another as (4) shows:

$$x' = KR'R^TK^{-1}x. \quad (4)$$

With these parameters above, we formulate calibration as an optimization problem, as shown in the following equation:

$$(Td, f) = \arg \min \frac{\sum_{j=1}^{N-1} \sum_{i=1}^{n_j} \|x'_i - KR'_jR_j^TK^{-1}x_i\|}{\sum_{j=1}^{N-1} n_j}. \quad (5)$$

And we have the following:

- (1) N is the frame amount, j is frame number, and I is the feature number in the current frame.
- (2) n_j is the feature amount of the j th frame pair.
- (3) $\|\cdot\|$ is the Frobenius norm.
- (4) (x_i, x'_i) is the i th SIFT pair [16] in the current frame pair.

3.2. Estimation of Rotation. After calibration, gyroscope data can be used to estimate the rotation of the device accurately. Gyroscope outputs angular velocities of every axis, so the angular rotation of each axis can be calculated by integrating the angular velocities. For any axis of the gyroscope, let ω_i be its angular velocity and let its corresponding sampling time be Δt_i ; then the angular value $\Delta\Theta_i$ from moment m to moment n can be obtained as shown in the following equation:

$$\Delta\Theta_i = \sum_{i=m}^n \omega_i \Delta t_i. \quad (6)$$

Let $k = (x, y, z)$ be the rotation values of the three gyroscope axes, where x, y, z can be calculated by (6). Let $\Theta = \sqrt{x^2 + y^2 + z^2}$ be the total rotation value; then the rotation matrix R can be given by the Rodrigues formula, as shown in the following equation:

$$R = I + [k]_{\times} \sin \Theta + [k]_{\times}^2 (1 - \cos \Theta), \quad (7)$$

where I is the identity matrix and $[k]_{\times}$ is the cross product matrix of k :

$$[k]_{\times} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}. \quad (8)$$

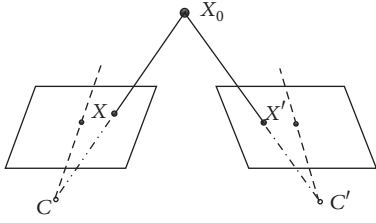


FIGURE 2: Two-view geometry of S^2fM .

Finally, we obtain $R \in SO(3)$, which is a unitary matrix representing the rotation of the device.

3.3. Estimation of Translation. The obtained rotation matrix R is used as known parameter in the estimation of translation based on visual method to generate the final camera pose. Similar to the traditional method, we formulate linear equations using correspondence points and solve them to get camera translation.

3.3.1. Formulation of Linear Equations. As shown in the two-view geometry of the proposed S^2fM algorithm (Figure 2), given a scene point X_0 in Euclidean space, its projections X and X' in camera coordination C and C' have the mapping relation of

$$X = RX' + T, \quad (9)$$

where R is the rotation matrix (3×3) and T is translation 3-vector. Define $[T]_{\times}$ as the cross product matrix of T .

We have

$$T = [T_x \ T_y \ T_z]^T, \quad (10)$$

$$[T]_{\times} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}.$$

Premultiplying both sides of (9) by $X^T [T]_{\times}$ gives

$$X^T [T]_{\times} RX' = 0. \quad (11)$$

Let $E \sim [T]_{\times} R$ be the essential matrix (3×3); then

$$X^T EX' = 0. \quad (12)$$

Equation (12) also holds for image points \tilde{x} and \tilde{x}' , which gives the epipolar constraint [2]:

$$\tilde{x}^T E \tilde{x}' = 0. \quad (13)$$

Writing (13) in the form of pixel coordinate gives

$$\begin{aligned} (K^{-1}\tilde{u})^T EK'^{-1}\tilde{u}' &= 0, \\ \tilde{u}^T (K^{-1T}EK'^{-1})\tilde{u}' &= 0, \end{aligned} \quad (14)$$

$$\tilde{u}^T (K^{-1T} [T]_{\times} RK'^{-1})\tilde{u}' = 0.$$

Rearrange (14):

$$(\tilde{u}^T K^{-1T}) [T]_{\times} (RK'^{-1}\tilde{u}') = 0. \quad (15)$$

Let $A = \tilde{u}^T K^{-1T} = (a_1, a_2, a_3)$ and let $B = RK'^{-1}\tilde{u}' = (b_1, b_2, b_3)^T$, which are now known; then

$$A [T]_{\times} B = 0. \quad (16)$$

Equation (16) can be rearranged as linear equations:

$$\begin{bmatrix} a_3 b_2 - a_2 b_3 \\ a_1 b_3 - a_3 b_1 \\ a_2 b_1 - a_1 b_2 \end{bmatrix}^T \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = 0, \quad (17)$$

where T_x , T_y , and T_z in (17) are unknowns to be estimated.

3.3.2. Solution of Linear Equations. The inputs of (17) are camera intrinsic K , rotation matrix R , and feature correspondences (\tilde{u}, \tilde{v}) and (\tilde{u}', \tilde{v}') . This is typical ternary homogeneous linear equation set and we need only 3 pairs of feature correspondences to solve it (T can also be estimated with 2 pairs of feature correspondences by enforcing certain constraint to reduce the number of freedoms to 2). We extract SIFT correspondence features [16] between frames and introduce RANSAC algorithm [9] to remove mismatches and noise. Then singular value decomposition (SVD) method is employed to solve (17). Since the essential matrix is defined up to scale, the solved translation vector T will also be defined up to scale; that is, $\|T\| = 1$.

Two solutions are still possible due to the arbitrary choices of signs for translation T . The correct one can be easily determined by ensuring that the reconstructed points lie in front of the camera [5], as shown in Figure 3. One pair of feature correspondence is enough to determine the sign of T . First, select a random pair of feature correspondence and random sign of T ; then apply the 3D reconstruction solution by Hartley and Sturm [17] to reconstruct its 3D space coordinate X_0 . If X_0 lies in front of the cameras, as Figure 3(a) shows, the selected sign of T is already correct; otherwise, inverse the sign of T , as shown in Figure 3(b).

4. Experimental Results

Our experimental device is Lenovo Vibe Z, with built-in three-axis gyroscopes ST L3GD20. The gyroscopes run at a frequency of 100 Hz.

We show the experimental results of S^2fM algorithm for CPE. To show the results explicitly, we draw the camera 3D motion computed by the proposed solution to check its validity. Then we compare our solution with traditional solutions (including 8 pt, 6 pt, and 5 pt algorithms. 7 pt algorithm performs similar to 8 pt algorithm so we omit it) and the inertial-based solution proposed by Gaida et al. [7]. But, first of all, we show the calibrating result for the device and gyroscopes used in our experiments.

4.1. Calibrating Result. We record a video of about 10 seconds with a rotation motion around y -axis of the camera and

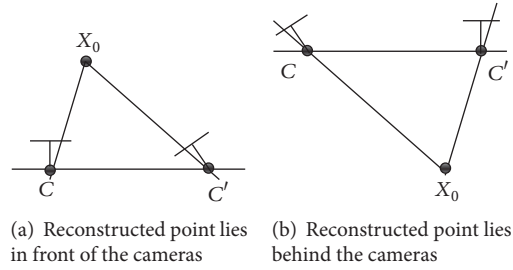
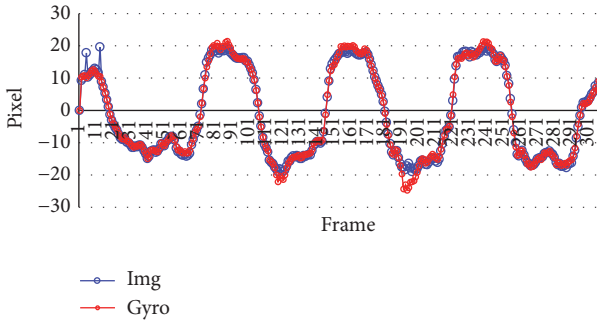
FIGURE 3: Two possible signs of T .

FIGURE 4: Frame horizontal translation between successive frames.

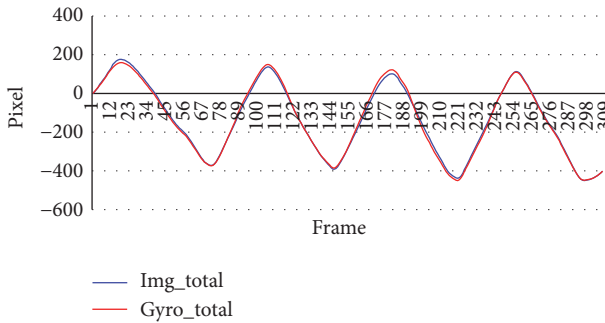


FIGURE 5: Video's total horizontal translation.

record the gyroscope data of all the three axes and then run the calibrating algorithm. Under a pure rotation model, the frame translation motion can be estimated both by SIFT features and by gyroscope data. After calibration, the frame motions estimated by the gyroscopes would align with that estimated by SIFT features. As the results show in Figures 4 and 5, the best parameters align the two motion results well.

In order to show the results more explicitly, the calibration error for every frame is computed as shown in Figure 6. The average calibration error is 0.0024 pixels and the absolute average calibration error is 0.75 pixels per frame. There are some error peaks in the calibration because the motion is not pure rotation (translation exists), but this will not affect the calibrating results.

4.2. CPE Results. We write letters L, E, N, O, V, and O and the word Lenovo in the air with our experimental device and run S^2fM algorithm to estimate the camera pose. Figure 7 shows

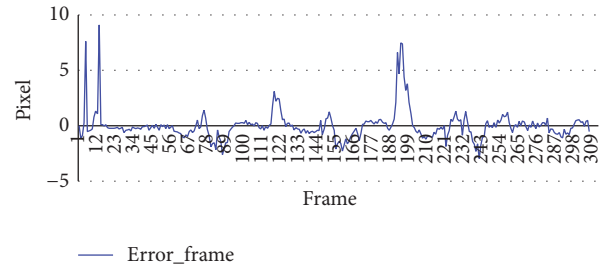


FIGURE 6: Calibration errors.

its results, where every point represents one estimated camera pose and the red one is the start position.

4.3. Evaluation. We compare our solution with 8 pt algorithm [1], 6 pt algorithm [3], 5 pt algorithm, [6] and the fusion method by Gaida et al. [7]. In order to make the experiments as complete as possible, we design six basic camera motion scenes:

- (1) Left-right translation
- (2) Up-down translation
- (3) Forward-backward translation
- (4) Rotation around x -axis
- (5) Rotation around y -axis
- (6) Rotation around z -axis

All possible camera motions can be a combination of the six basic motions above. To suit for different application precisions, different error thresholds (2.0, 1.0, and 0.5 pixels) are set in the linear square solution.

4.3.1. Accuracy. The symmetric squared geometric error E_r is introduced to measure the reprojection error [10], as shown in the following equation:

$$E_r = \frac{(p^T F p')^2}{[F p']_1^2 + [F p']_2^2} + \frac{(p^T F p')^2}{[F^T p]_1^2 + [F^T p]_2^2}, \quad (18)$$

where p and p' are feature correspondences and F is the fundamental matrix. The reprojection errors of all the six motion scenes with different error thresholds are shown in Figure 8.

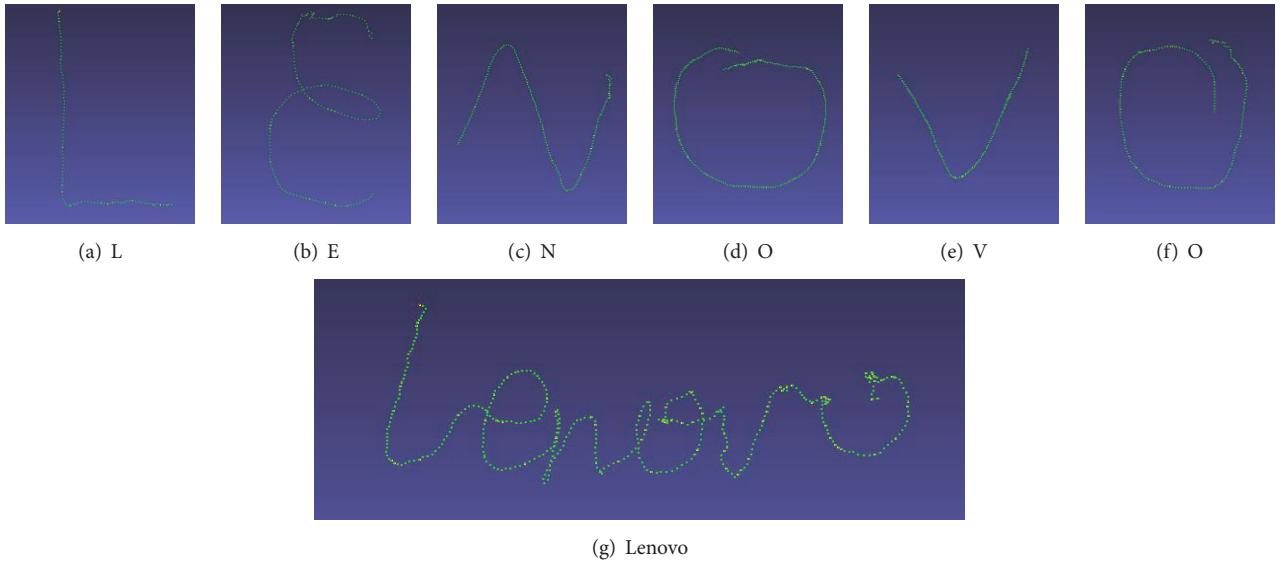


FIGURE 7: Camera pose estimation results.

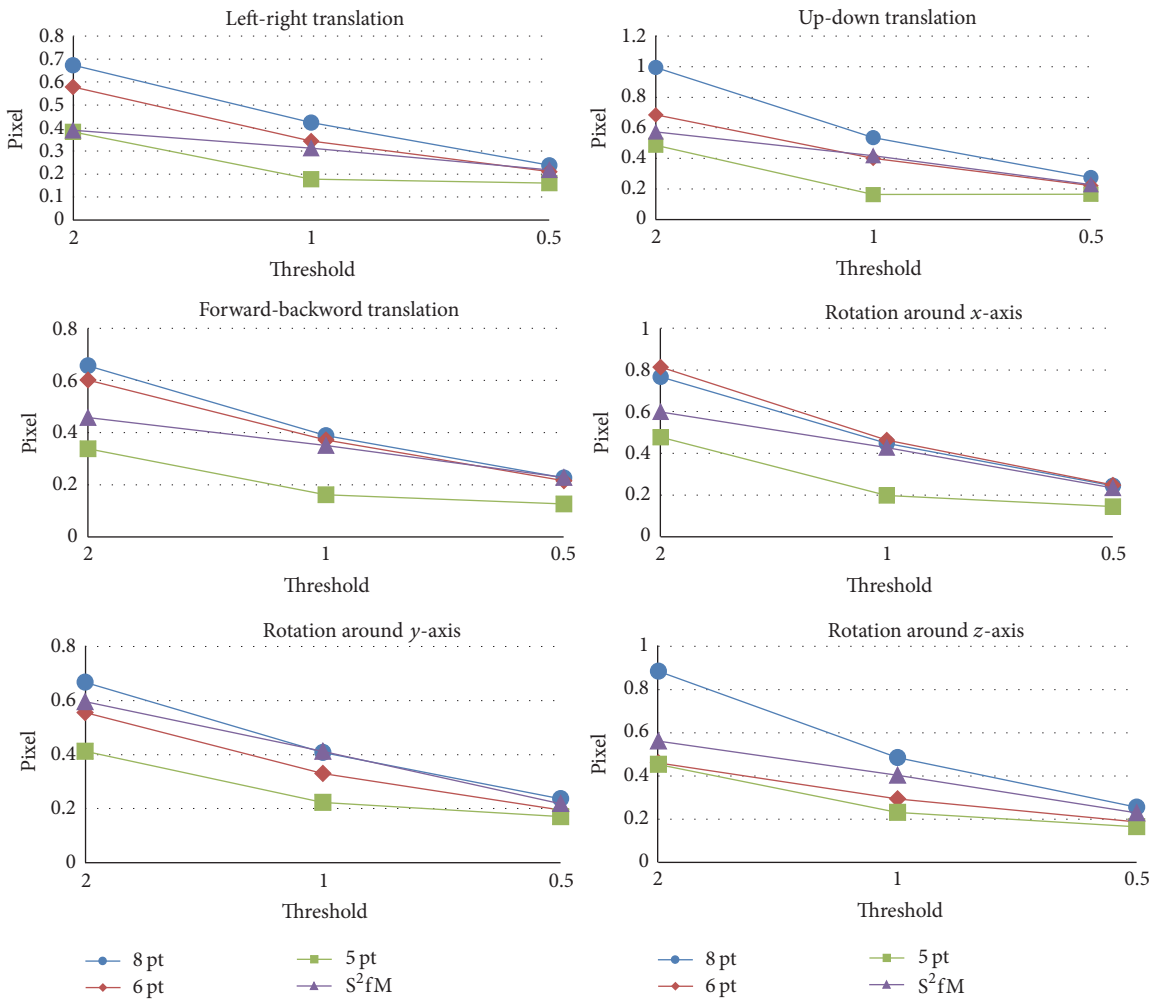


FIGURE 8: CME reprojection errors in 6 basic scenes.

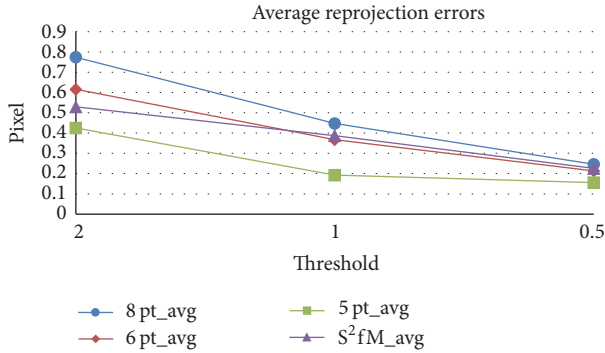


FIGURE 9: Average reprojection errors of 6 basic scenes.

TABLE 1: Average reprojection errors of 6 basic scenes (unit: pixel).

	2	1	0.5
8 pt	0.774027	0.448177	0.246181
6 pt	0.61605	0.36723	0.213302
5 pt	0.425233	0.192617	0.155683
S ² fM	0.529583	0.38755	0.226083
Gaida et al.	Average of 8.69 ¹		

¹The result is given by the author in paper [7].

We compute the average reprojection errors of the 6 basic scenes under different threshold in RANSAC algorithms, as shown in Figure 9 and Table 1. The results show that, in our experiments, Nistér’s 5 pt algorithm generates minimal reprojection error and our solution performs similar to 6 pt algorithm.

4.3.2. Efficiency. We analyze the efficiency of S²fM both in theory and in practice.

(1) *Theoretical Analysis.* Since the camera rotation can be read out from gyroscopes in real time, S²fM costs most in translation estimation by the visual method. The time complexity of traditional algorithms can be analyzed by the number of feature pairs needed in solving the linear equations. In our experiments, we adopt SVD algorithm to solve the linear equations, whose time complex is $O(n^3)$. As a result, the time complex ratio of solving one linear equation set for CPE would be as shown in Table 2 (with S²fM set to be 1).

(2) *Practical Analysis.* In practice, all of the algorithms (S²fM, 8 pt, 6 pt, and 5 pt) need to adopt RANSAC algorithm to estimate the optimal solution of the linear equations and the number of iterations will not be the same. However, for each algorithm, we set its upper limit of RANSAC iteration times to be 128 in our experiments and then compute the average frame-computing time of all the 11715 experiments above with the same SIFT features and the same running environment. With a good number of experiments, the average value should be enough to compare the performances of all the algorithms, as shown in Table 3.

As seen in Table 3, the practical efficiency result generally tallies with the theoretical analysis.

TABLE 2: Time complex ratio of solving one linear equation set for CPE in theory.

S ² fM	1
8 pt	$(8/3)^3$
6 pt	8
5 pt	$(5/3)^3$

TABLE 3: Average frame-computing time for CPE in practice.

S ² fM	16 ms
8 pt	288 ms
6 pt	107 ms
5 pt	59 ms

4.4. Limitations. One main limitation of the proposed solution lies in the drift of gyroscopes. After running a long period of time, the gyroscope data may drift and result in the decrease of estimation accuracy, which can be compensated by timely calibrations, for example, assisted by other inertial sensors to remain drift-free.

Another limitation is that our solution is ideally supposed to perform similar to 5 pt algorithm because they are the most similar in estimating translation, but the experimental results show that it performs worse than 5 pt algorithm in accuracy. The reason that S²fM does not achieve ideal results lies in the focal length calibrating error and the drift of gyroscopes. More accurate calibrating algorithm will be of great help.

What is more, in our experiments, we choose to use 3 pairs of feature correspondences to estimate camera translation but T can also be estimated with 2 pairs of feature correspondences by enforcing certain constraint to reduce the number of freedoms to 2. However, the purpose of this paper is to provide a fusion solution for CPE, not the mathematical technique for solving equation set.

5. Conclusion

Traditionally, CPE has been formulated as the problem of estimating the optimal camera pose given a set of point correspondences. This is the vision-based method. The first 8-point solver was proposed in the 1990s and later 7-point, 6-point, and 5-point solvers were proposed in the 2000s, all of which are based on the feature correspondence problem as reviewed in the literature. These solvers all focus on the mathematical technique of solving the formulated optimal estimation problem. In recent years, when MEMS sensors become accurate enough and become popular in many hand-held devices, MEMS-based methods turn up in CPE-related problems. Generally, sensor data will be coupled with image processing data with data filter, for example, Kalman filter, for robust camera pose estimation. The drawback of this method is also the computing complexity due to the additional data filtering process. This paper proposes a camera pose estimation algorithm with gyroscope sensor, which estimates camera rotation with the built-in gyroscopes of the device and then fuses it with the image data to estimate camera translation. This proposed fusion solution is quite different

from existing fusion solutions in the fusion manner of inertial and visual data. We compare our solution with both the traditional solution and existing fusion solution and the experimental results validate the efficiency of our solution. As for the accuracy, our solution performs similar to 6 pt algorithm, which can be further improved with better focal length calibration and drift compensation techniques.

Under the circumstance that no proper MEMS sensor can estimate accurately enough translation, S^2fM provides a way to fuse the inertial data with visual data to solve the problem.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

References

- [1] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580–593, 1997.
- [2] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003.
- [3] H. Stewénius, D. Nistér, F. Kahl, and F. Schaffalitzky, "A minimal solution for relative pose with unknown focal length," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 789–794, IEEE, San Diego, Calif, USA, June 2005.
- [4] Z. Kukulova, M. Bujnak, and T. Pajdla, "Polynomial eigenvalue solutions to minimal problems in computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1381–1393, 2012.
- [5] D. Nistér, "An efficient solution to the five-point relative pose problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 756–770, 2004.
- [6] H. Li and R. Hartley, "Five-point motion estimation made easy," in *Proceedings of the 18th International Conference on Pattern Recognition (ICPR '06)*, pp. 630–633, Hong Kong, August 2006.
- [7] D. Gaida, A. Stuhlsatz, and H. G. Meier, "Fusion of visual and inertial measurements for pose estimation," in *Proceedings of the Research Workshop on Emotion-, Speech-, and Face Recognition with Advanced Classifiers*, pp. 39–47, Düsseldorf, Germany, May 2008.
- [8] R. Miyano, T. Inoue, T. Minagawa, Y. Uematsu, and H. Saito, "Camera pose estimation of a smartphone at a field without interest points," in *Computer Vision - ACCV 2012 Workshops: ACCV 2012 International Workshops, Daejeon, Korea, November 5-6, 2012, Revised Selected Papers, Part II*, vol. 7729 of *Lecture Notes in Computer Science*, pp. 545–555, Springer, Berlin, Germany, 2013.
- [9] 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 Association for Computing Machinery*, vol. 24, no. 6, pp. 381–395, 1981.
- [10] M. Brückner, F. Bajramovic, and J. Denzler, "Experimental evaluation of relative pose estimation algorithms," in *Proceedings of the 3rd International Conference on Computer Vision Theory and Applications (VISAPP '08)*, pp. 431–438, Funchal, Portugal, January 2008.
- [11] A. Karpenko, D. Jacobs, J. Baek, and M. Levoy, "Digital video stabilization and rolling shutter correction using gyroscopes," Stanford Tech Report CTSR, 2011.
- [12] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski, "Image deblurring using inertial measurement sensors," *ACM Transactions on Graphics*, vol. 29, no. 4, pp. 1989–1997, 2010.
- [13] D. H. Titterton and J. L. Weston, "Strapdown inertial navigation technology," *IEEE Aerospace & Electronic Systems Magazine*, vol. 20, no. 7, pp. 33–34, 2004.
- [14] P. Corke, J. Lobo, and J. Dias, "An introduction to inertial and visual sensing," *The International Journal of Robotics Research*, vol. 26, no. 6, pp. 519–535, 2007.
- [15] A. Goldstein and R. Fattal, "Video stabilization using epipolar geometry," *ACM Transactions on Graphics*, vol. 31, no. 5, article 126, 2012.
- [16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [17] R. I. Hartley and P. Sturm, "Triangulation," *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146–157, 2001.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

