International Journal of Control, Automation and Systems 16(4) (2018) 1962-1970
http://dx.doi.org/10.1007/s12555-017-0200-5
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Visual Inertial Odometry with Pentafocal Geometric Constraints

Pyojin Kim, Hyon Lim, and H. Jin Kim*

Abstract: We present the sliding-window monocular visual inertial odometry that is accurate and robust to outliers by employing a new observation model grounded on the pentafocal geometric constraints. The previous approaches are dependent on the unknown 3D coordinates of the features to estimate the ego-motion. However, the inaccurate 3D position of the features can lead to poor performance in motion estimation. To overcome these limitations, we utilize the pentafocal geometry relationship between five images as camera observation model, which makes it unnecessary to estimate the 3D position of the features. Furthermore, we apply the pentafocal constraints in the 1-point random sample consensus (RANSAC) algorithm to find incorrect feature correspondences. We demonstrate the effectiveness of the proposed algorithm in two types of experiments: the KITTI driving scene dataset and the EuRoC micro aerial vehicle (MAV) flying dataset, both qualitatively and quantitatively. It shows more accurate state estimation performance compared to the well-known stereo visual odometry algorithm and current state-of-the-art visual inertial odometry methods.

Keywords: One-point RANSAC, pentafocal geometry, relative pose estimation, visual inertial odometry.

1. INTRODUCTION

Odometry is one of the significant elements to enable autonomous robot navigation, which incrementally integrates the estimated relative motion of an agent without any prior map, 3D scene model, or agent's dynamic model [1]. It is also useful in many other applications such as map-based localization, 3D reconstruction, and augmented reality.

As onboard sensors used in odometry algorithms, inexpensive, lightweight, and passive sensors such as camera and an inertial measurement unit (IMU) have received significant attention. Especially, monocular, stereo, and visual inertial odometry techniques have been actively investigated in robotics and computer vision community. In the methods that estimate the egomotion of a moving vehicle with only a single camera [2–5], there is the limitation that the camera motion can be recovered only up to a scale factor without any prior metric information [6]. In contrast, there is no scale ambiguity in stereo visual odometry approaches, and they show successful egomotion estimation results in [7–9]. However, the operating range depends on the baseline between the two cameras, and a pair of cameras are relatively more expensive than a single camera and IMU sensor. Increased computational complexity and memory consumption are also associated compared to the processing of only a single camera. Vision-only methods are sensitive to motion blur, low texture, etc, leading to a drop in accuracy or even failure. To overcome such limitations of vision-only navigation, a method of using a camera and IMU sensor at the same time, which are complementary to each other, has received much attention in the past few years [10–12]. As an exteroceptive sensor, a moving monocular camera can perceive appearance and geometry of a three-dimensional surrounding environment up to unknown scale. Complementarily, as a proprioceptive sensor, IMU can observe the metric scale and provide

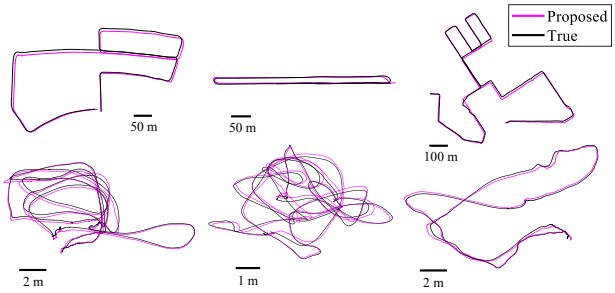

Fig. 1. Trajectory estimation results, showing accurate 6-DoF state estimation of the proposed method in the KITTI driving scene (top) and the EuRoC MAV flying (bottom) dataset.

Manuscript received April 6, 2017; revised November 13, 2017; accepted December 29, 2017. Recommended by Associate Editor Huaping Liu under the direction of Editor Duk-Sun Shim. This work was supported by the Seoul National University Research Grant in 2015, the National Research Foundation of Korea (NRF) grant funded by the Ministry of Science, ICT & Future Planning (2014M1A3A3A02034854), and the Technology Innovation Program (10067206) funded by the Ministry of Trade, industry & Energy (MI, Korea).

Pyojin Kim, Hyon Lim, and H. Jin Kim are with the School of Mechanical and Aerospace Engineering, Seoul National University, 1 Gwanack-ro, Gwanack-gu, Seoul, Korea (e-mails: {rlavywls, hyonlim, hjinkim}@snu.ac.kr).
* Corresponding author.

the accurate attitude of the body frame regardless of the surrounding environment.

Despite some recent success, monocular visual inertial approaches [11–14] are still challenging in terms of accuracy and robustness to outliers. The dependence of the measurement model on the unknown (estimated) 3D coordinate of the features can lower the accuracy if the estimated 3D position of the features is incorrect. The accuracy of motion estimation can also deteriorate rapidly due to wrong data associations so called outliers caused by many factors that cannot be treated easily such as independently moving objects in the scene, occlusion, illumination changes, and image noise.

To address such issues, we propose a sliding-window monocular visual inertial odometry, which is more accurate and robust to outliers, by employing the pentafocal geometric constraints. The pentafocal geometry relationship between five images is used as the camera observation model, which makes the proposed algorithm without estimating the 3D position of the features. To perform robust motion estimation to outliers which are tracked incorrectly or located on the moving objects in the dynamic scene, the pentafocal geometric constraints are also utilized in the 1-point RANSAC [15]. The proposed algorithm is validated with the large-scale KITTI datasets [16] in which the total traveling distance of a recording platform is longer than 1 km. The EuRoC micro aerial vehicle (MAV) flying datasets [17] are also used to evaluate the performance of the proposed algorithm in the indoor long-distance flight environments.

This paper is organized as follows. Related works are discussed in Section 2. In Section 3, the notation used throughout this paper and the proposed sliding-window visual inertial odometry are described in detail. After validation and evaluation results are presented in Section 4, the conclusion is made in Section 5.

2. RELATED WORK

During the last decade, visual odometry (VO) and visual simultaneous localization and mapping (V-SLAM) have been active areas of research in the robotics and computer vision communities. We will briefly review VO and V-SLAM algorithms related to the tightly-coupled visual-inertial solution which will provide better accuracy and robustness than loosely-coupled ones.

One of the most popular methodologies in VO and V-SLAM is a probabilistic filtering approach, which sequentially updates the position of the features and the current location of the camera. Real-time camera tracking with the filtering approach is successfully implemented only using a single camera in monoSLAM [2]. Similar to monoSLAM, [18] jointly estimates a 3D position of feature points and camera motion with inverse depth parametrization (IDP) in the extended Kalman filter (EKF) for good feature initialization. Further, 1-point RANSAC [15] is proposed for robust estimation to reject spurious feature matches. In [19], inertial information from IMU is coupled tightly into the monoSLAM by replacing the constant velocity motion model with the IMU motion model. The main problem of this EKF-based VO and V-SLAM methods is that the dimension of the state vector and covariance matrix grows rapidly as the number of mapped features increases, causing high computational complexity over time.

To solve the above problem, a sliding-window filter for incremental SLAM [20] is proposed to keep the complexity of the filter by removing the oldest camera pose and distant landmarks. To only focus on the camera motion, lightweight sliding-window filtering based VO methods [10, 21] have gained in popularity, which do not include the estimated 3D feature position in the filter state vector, so called multi-state constraint Kalman filter (MSCKF). The past camera poses in the filter state vector are used to estimate the 3D position of the features with least-squares minimization process, and the estimated features are re-projected to the past camera poses for measurement update. In [11], the camera motion and IMU-camera alignment are estimated in the Kalman filtering framework in real time. Local bundle adjustment (LBA) is employed in [22] to address the problem of growing computational cost over time. However, it is still heavier than the sliding-window method, and it is difficult to be integrated into the Kalman filter framework.

The requirement of a least square minimization to estimate the 3D position of the features in MSCKF can lead to a drop in accuracy or even odometry failure if the 3D feature position is inaccurately estimated. To eliminate the needs of the estimation of 3D feature position, we propose a sliding-window visual inertial odometry using the pentafocal geometric constraints (combination of bifocal and trifocal tensors) between five images as the camera observation model. Furthermore, the pentafocal geometric constraints are also chosen as the update model in the 1-point RANSAC algorithm [15] to perform robust motion estimation to outliers. Similar research can be found in [23] and [24]. Compared to [23], we use the combination of multiple bifocal and trifocal tensors, resulting in more accurate state estimation. The outlier rejection in [24], which is performed by computing the motion hypothesis with two feature correspondences and gyroscopic data from IMU, cannot be easily implemented directly in the tightly-coupled visual inertial odometry like MSCKF. Thus, we choose to use the 1-point RANSAC.

3. FILTER SETUP

The goal of the proposed filtering approach is to estimate accurate egomotion of the visual inertial sensor rig, which consists of the camera frame {C} and the IMU

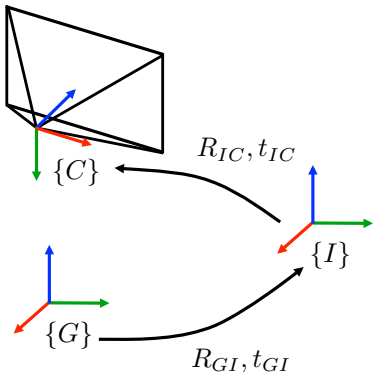

Fig. 2. IMU-camera coordinates system. R_{IC},t_{IC} are rotational and translational part of IMU-camera extrinsic calibration parameters.

frame {I} with respect to the global inertial frame {G} as shown in Fig. 2.

Note that $R_{AB} \in SO(3)$ is the rotation matrix from {B} to {A} and $t_{AB} \in \mathbb{R}^3$ is the translation vector from the origin of frame {A} to the origin of frame {B} expressed in {A}.

3.1. Structure of the UKF state vector

The overall structure of the proposed filtering algorithm is similar with [10]. The filter state vector is composed of the current IMU state vector and the last four poses of the IMU frame only, which means that it does not include a 3D position of the tracked feature points for reducing computational burden. Like [10] and [11], the IMU measurements are used for propagation of IMU state vector, and the visual information, i.e., the detected, tracked keypoints, is served in the filter update step for every captured image. The overview of the proposed sliding-window visual inertial odometry is drawn in Fig. 3.

The IMU state vector $\mathbf{x}_{\text{IMU}}$ is defined as follows:

$$\mathbf{x}_{\text{IMU}} = \begin{bmatrix} {}^{G}\mathbf{p}_I^\top & {}^{G}\bar{q}_I^\top & {}^{G}\mathbf{v}_I^\top & \mathbf{b}_a^\top & \mathbf{b}_g^\top \end{bmatrix}^\top \in \mathbb{R}^{16}, \quad (1)$$

where ${}^{G}\mathbf{p}_I^\top$ and ${}^{G}\bar{q}_I^\top$ are the position vector and the unit quaternion for rotation of the IMU frame expressed in the global frame, respectively. ${}^{G}\mathbf{v}_I^\top$ is the velocity vector of the IMU frame with respect to the global frame, and $\mathbf{b}_a^\top$ and $\mathbf{b}_g^\top$ are the biases affecting the accelerometer and gyroscope measurements.

The IMU error state vector $\tilde{\mathbf{x}}_{\text{IMU}}$ is described as follows:

$$\tilde{\mathbf{x}}_{\text{IMU}} = \begin{bmatrix} {}^{G}\tilde{\mathbf{p}}_I^\top & \boldsymbol{\delta\theta}_I^\top & {}^{G}\tilde{\mathbf{v}}_I^\top & \tilde{\mathbf{b}}_a^\top & \tilde{\mathbf{b}}_g^\top \end{bmatrix}^\top \in \mathbb{R}^{15}. \quad (2)$$

The more detailed explanation of the IMU state vector and each component in the Eq. (1) and (2) can be found in [10] and [11].

The UKF state vector $\mathbf{X}_k$ consists of the current IMU state vector and history of the last four poses of the IMU frame as follows:

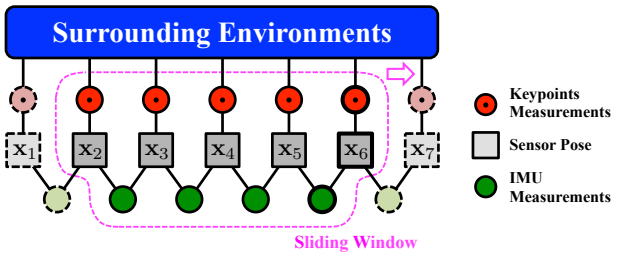

Fig. 3. Overview of the proposed visual inertial odometry.

$$\mathbf{X}_k = \begin{bmatrix} \mathbf{x}_{k,\text{IMU}}^\top & {}^{G}\mathbf{p}_{I_{k-4}}^\top & {}^{G}\bar{q}_{I_{k-4}}^\top & \cdots & {}^{G}\mathbf{p}_{I_{k-1}}^\top & {}^{G}\bar{q}_{I_{k-1}}^\top \end{bmatrix}^\top \in \mathbb{R}^{44}, \quad (3)$$

where the subscript k denotes the time step. Contrary to [2] and [19], the dimension of the filter state vector is fixed to 44; hence the computational complexity associated with the state vector is constant over time.

The UKF error state vector $\tilde{\mathbf{X}}_k$ can be defined in the same way as the IMU error state is defined:

$$\tilde{\mathbf{X}}_k = \begin{bmatrix} \tilde{\mathbf{x}}_{k,\text{IMU}}^\top & {}^{G}\tilde{\mathbf{p}}_{I_{k-4}}^\top & \boldsymbol{\delta\theta}_{I_{k-4}}^\top & \cdots & {}^{G}\tilde{\mathbf{p}}_{I_{k-1}}^\top & \boldsymbol{\delta\theta}_{I_{k-1}}^\top \end{bmatrix}^\top \in \mathbb{R}^{39}. \quad (4)$$

3.2. Propagation model

The continuous-time differential equation for the time evolution of the filter state vector is described as follows:

$$\dot{\mathbf{X}}_k = \begin{bmatrix} \dot{\mathbf{x}}_{k,\text{IMU}} \\ {}^{G}\dot{\mathbf{p}}_{I_{k-4}} \\ {}^{G}\dot{\bar{q}}_{I_{k-4}} \\ \vdots \\ {}^{G}\dot{\mathbf{p}}_{I_{k-1}} \\ {}^{G}\dot{\bar{q}}_{I_{k-1}} \end{bmatrix} = \begin{bmatrix} f(\mathbf{x}_{k,\text{IMU}}, a_m, w_m) \\ 0_{3\times1} \\ 0_{4\times1} \\ \vdots \\ 0_{3\times1} \\ 0_{4\times1} \end{bmatrix}, \quad (5)$$

where $\mathbf{x}_{k,\text{IMU}}$ is the IMU state vector at time step k defined in Eq. (1) and a_m, w_m are linear acceleration and angular velocity measurements from IMU. The zeros in the process model of the last four poses indicate that there is no time evolution in the propagation step.

The time evolution of the IMU state vector in Eq. (5) is written as follows:

$$\begin{bmatrix} {}^{G}\dot{\mathbf{p}}_I \\ {}^{G}\dot{\bar{q}}_I \\ {}^{G}\dot{\mathbf{v}}_I \\ \dot{\mathbf{b}}_a \\ \dot{\mathbf{b}}_g \end{bmatrix} = \begin{bmatrix} {}^{G}\mathbf{v}_I \\ \frac{1}{2}\Omega(w_m - b_g)\,{}^{G}\bar{q}_I \\ R_{GI}\,(a_m - b_a) - {}^{G}\mathrm{g} \\ \mathbf{n}_{ba} \\ \mathbf{n}_{bg} \end{bmatrix}, \quad (6)$$

where $\Omega(\cdot)$ denotes quaternion multiplication and ${}^{G}\mathrm{g}$ is the gravitational acceleration expressed in the global frame. The white Gaussian noise processes, $\mathbf{n}_{ba}$ and $\mathbf{n}_{bg}$, propagate the accelerometer and gyro biases.

For every IMU state vector $\mathbf{x}_{k,\text{IMU}}$, we perform the fourth-order Runge Kutta integration between the time

step $k-1$ and k. Because we adopt the similar IMU propagation model and numerical methods to discrete-time implementation used in [10], the more detailed explanations are referred to the paper.

3.3. Observation model

The observation model for filter update is the pentafocal geometric constraint from the five consecutive camera poses, which has a balance between computational cost and accuracy. The camera observations from the five camera poses are bundled not for camera pose, but for tracked feature points. We utilize the observations of the same physical points from the five consecutive camera poses to impose pentafocal geometric constraints to the related five camera poses, i.e., the maximum number of the bi- and trilinear geometric constraints are applied to update the camera poses. It is noteworthy that when we impose geometric constraints to the five camera poses for updating the filter, it is *not* necessary to include or estimate the position of the 3D feature points in the filter state vector. It can lead to more accurate state estimation, and the computational complexity related to the dimension of the filter state vector does not change over time, contrary to [19] and [15].

To promote understanding in this paper, we only represent the single point f_j observed from the five consecutive camera poses. The description applies to multiple feature points $j = 1, \cdots, n$ where n is the number of tracked feature points in the five consecutive camera poses from $\{C_{k-4}\}$ to $\{C_k\}$.

Assuming that the current time step k is five, we consider a point correspondence across five views: $m_1, \cdots, m_5$. The pentalinear relationships describing the projection of the feature point $\mathbf{P}_{f_j}$ into the five images can be written as follows:

$$\begin{bmatrix} T_{11} & \tilde{m}_1 & 0 & \cdots & 0 \\ T_{21} & 0 & \tilde{m}_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \cdots \\ T_{51} & 0 & 0 & \cdots & \tilde{m}_5 \end{bmatrix} \begin{bmatrix} \mathbf{P}_{f_j} \\ -\lambda_1 \\ \vdots \\ -\lambda_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{7}$$

where $\tilde{m}_k = [m_k{}^\top, 1]^\top$ is the homogeneous form of the feature point location m_k and $T_{AB} = [R_{AB}, t_{AB}] \in \mathbb{R}^{3\times4}$ is the rigid body transformation matrix from {B} to {A}. λ_k is the scale factor corresponding to the feature points. Instead of adopting the above pentalinear relations directly, the equivalent geometric relationships, i.e., multiple bifocal and trifocal geometric constraints, are used to update the filter state vector as shown in Fig. 4.

The bifocal geometric constraints, i.e. epipolar geometry constraints, between the two views can be written as follows:

$$\tilde{m}_2^\top F_{12} \tilde{m}_1 = 0, \tag{8}$$

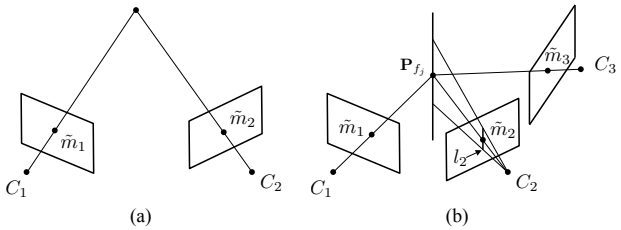

Fig. 4. Geometries used in the pentafocal constraints. (a) Bifocal and (b) trifocal geometry.

where F_{12} is the fundamental matrix written as $F_{12} = K^{-\top} R_{12}^\top [t_{12}\times] K^{-1}$. K presents the camera intrinsic parameters which can be obtained from camera calibration.

The point transfer equation based on the trifocal tensor between the three different viewpoints can be written as follows:

$$\tilde{m}_3 = K \left(\sum_{i=1}^{3} \bar{m}_1^i \mathrm{T}_i^\top \right) l_2, \tag{9}$$

where $\bar{m}_k = K^{-1}\tilde{m}_k$ is the feature point in the normalized image plane, and $\bar{m}_k^i$ is the i-th element in the feature point location vector $\bar{m}_k$. $l_2 \in \mathbb{R}^3$ is the corresponding line passing through the feature point m_2 in the second image. $\mathrm{T} \in \mathbb{R}^{3\times3\times3}$ is the trifocal tensor, which involves the relative rigid body transformation and can be calculated with the given formula as follows:

$$\mathrm{T} = \{\mathrm{T}_1, \mathrm{T}_2, \mathrm{T}_3\}, \tag{10}$$

where the i-th rigid body transformation matrix is $\mathrm{T}_i = a_i b_4^\top - a_4 b_i^\top \in \mathbb{R}^{3\times3}$. They can be calculated with the column vectors of the followings: $T_{21} = [R_{21}, t_{21}] = [a_1|a_2|a_3|a_4] \in \mathbb{R}^{3\times4}$ and $T_{31} = [R_{31}, t_{31}] = [b_1|b_2|b_3|b_4] \in \mathbb{R}^{3\times4}$. The more detailed descriptions of each component and equation are referred to [25].

The five images can maximally impose ten independent bifocal and ten independent trifocal geometric constraints to the filter state vector. The observation residual term per tracked feature point f_j can be formulated as follows:

$$\mathbf{r}_j = \mathbf{z}_j - \hat{\mathbf{z}}_j \in \mathbb{R}^{30}, \tag{11}$$

where

$$\hat{\mathbf{z}}_j = h(\mathbf{X}_k, \{m_1, \cdots, m_5\}) = \begin{bmatrix} \tilde{m}_2^\top F_{12} \tilde{m}_1 \\ \vdots \\ \tilde{m}_5^\top F_{45} \tilde{m}_4 \\ K(\sum_i \tilde{m}_{1i} T_i^{jk}) l_2 \\ \vdots \\ K(\sum_i \tilde{m}_{3i} T_i^{jk}) l_4 \end{bmatrix} \in \mathbb{R}^{30},$$

where $\mathbf{X}_k$ is the filter state vector written in (3), and F_{12}, T_i^{jk} are the fundamental matrix and trifocal tensor calculated by the relative pose of each camera frame.

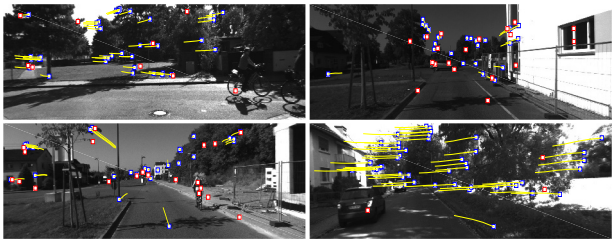

Fig. 5. The outlier rejection results with the 1-point RANSAC.

After the filter state vector $\mathbf{X}_k$ and the covariance matrix are updated with (11), the oldest pose of IMU is discarded and replaced with the next state to keep only five IMU poses in the filter state vector.

$$\mathbf{X}_{k+1,initial} = T_c \mathbf{X}_{k,final}, \tag{12}$$

where

$$T_c = \begin{bmatrix} I_{7\times7} & 0_{7\times9} & 0_{7\times7} & 0_{7\times7} & 0_{7\times7} & 0_{7\times7} \\ 0_{9\times7} & I_{9\times9} & 0_{9\times7} & 0_{9\times7} & 0_{9\times7} & 0_{9\times7} \\ 0_{7\times7} & 0_{7\times9} & 0_{7\times7} & I_{7\times7} & 0_{7\times7} & 0_{7\times7} \\ 0_{7\times7} & 0_{7\times9} & 0_{7\times7} & 0_{7\times7} & I_{7\times7} & 0_{7\times7} \\ 0_{7\times7} & 0_{7\times9} & 0_{7\times7} & 0_{7\times7} & 0_{7\times7} & I_{7\times7} \\ I_{7\times7} & 0_{7\times9} & 0_{7\times7} & 0_{7\times7} & 0_{7\times7} & 0_{7\times7} \end{bmatrix}.$$

The proposed sliding-window odometry method is straightforward and easy to implement. The dimension of the filter state vector and covariance matrix are all fixed contrary to MSCKF [10].

3.4. 1-Point RANSAC

We employ a random sample consensus (RANSAC) algorithm [26] similar to the 1-point RANSAC used in [15] to reject the problematic feature points, which are tracked incorrectly or located on the moving objects in the dynamic scene. The pentafocal geometric constraints are chosen as update model in the 1-point RANSAC since they do not require a heavy computational load such as nonlinear optimization for estimating the 3D feature position. Since only well-tracked static feature points over five consecutive frames can satisfy the pentalinear relations, we can detect the problematic feature points which are located on moving objects or tracked incorrectly as shown in Fig. 5.

In Fig. 5, blue squares denote the inlier feature points in the update step. Red squares, however, indicate the problematic feature points tracked incorrectly or located on the moving objects. It can be seen that the feature points marked with the red square are located on independently moving objects such as driving cars, human. Therefore, the pentafocal geometric constraints used in the RANSAC process make the proposed algorithm robust with respect to the dynamic environment and mismatched feature points.

4. EVALUATION

We test the proposed sliding-window visual inertial odometry on the two types of experiments with the KITTI driving scene dataset [16] and the EuRoC MAV flying dataset [17], both qualitatively and quantitatively, and analyze in terms of estimation accuracy in detail. In the KITTI driving scene dataset, we use the rectified image sequences from PointGray Flea2 grayscale single camera and inertial information from OXTS RT3003 GPS/IMU sensor for estimating egomotion of the recording platform at 10 Hz. In the EuRoC MAV flying dataset, we utilize the left images in the stereo camera rig and IMU information at 20 Hz and 200 Hz respectively, which are captured by a visual-inertial (VI) sensor unit [27] attached to an AscTec Firefly MAV. We utilize accurate spatio-temporal alignment of the IMU and camera sensor data from [16, 17].

We detect the feature points for updating the filter state vector, and track them with SURF [28] within only five consecutive image sequences. The maximum number of the tracked feature points is 50, and the bucketing mechanism [29] is used to reduce drift rate of the proposed algorithm by making the distribution of the feature points uniform. The proposed visual odometry algorithm is written and tested in MATLAB R2015a, and all of the calculations are performed on a desktop with Intel i5 3.2 GHz and 8 GB memory.

For quantitative evaluation and comparison of the proposed algorithm, we select three types of error metrics: root mean square error (RMSE) of the relative pose error (RPE), absolute trajectory error (ATE) defined in [30], and the final drift error (FDE) which is the end point position error divided by the total traveling distance of a recording platform. We compare the proposed method with the monocular and stereo visual odometry [8], which estimate the 3D position of the features for egomotion estimation. Current state-of-the-art visual inertial odometry algorithms [23], [13] are also compared to the proposed algorithm in terms of accuracy.

4.1. Qualitative results

We choose several sequences in the KITTI and MAV dataset, which include sudden light variations, frequent on-the-spot rotations, and some jerky motion of the camera to evaluate the proposed method qualitatively. Fig. 6 shows the estimated trajectories of each VO method on the Residential #3 in the KITTI dataset, which is about 3.6 km traveling distance. We plot the estimated trajectories of the proposed method (magenta), monocular visual odometry (red), stereo visual odometry (green) [8], and the visual inertial odometry (blue) [23] with the ground truth path (black). Two selected close-ups show that the estimated path with the proposed method is more consistent than other resulting trajectories. The point cloud of the 3D feature points (gray dots), which are used for updating

Table 1. Evaluation results with the KITTI dataset.

Environment	Absolute Trajectory Error [m]				Final Drift Error [%]				Length [m]	# of frame
	Proposed	Mono VO	Stereo VO	TVIO	Proposed	Mono VO	Stereo VO	TVIO		
Residential #1	**7.851**	83.391	23.583	35.919	**0.267**	9.392	1.719	2.871	1830	2300
Residential #2	10.299	52.274	**9.356**	13.118	**0.937**	5.014	1.283	2.401	1227	1104
Residential #3	**12.082**	173.248	71.081	20.604	**0.262**	6.850	2.085	0.515	3667	4500
Residential #4	**10.025**	181.928	14.297	40.441	**0.752**	13.432	2.747	6.425	920	1224
Residential #5	13.862	80.095	36.125	**12.320**	**1.110**	5.891	2.674	1.313	1797	2400
Residential #6	**35.480**	142.534	93.320	60.267	**1.222**	3.529	3.481	2.030	5064	4663

Table 2. Evaluation results with the EuRoC MAV dataset.

Environment	Absolute Trajectory Error [m]				Final Drift Error [%]				Length [m]	# of frame
	Proposed	ROVIO	Stereo VO	TVIO	Proposed	ROVIO	Stereo VO	TVIO		
Vicon Room #1	0.391	0.439	**0.320**	0.500	**0.685**	1.153	0.764	0.827	78.74	1954
Vicon Room #2	**0.227**	0.413	1.239	Fail	**1.110**	2.050	4.005	Fail	36.27	2104
Vicon Room #3	**0.196**	0.324	2.145	0.505	**0.234**	0.479	4.362	2.914	82.77	2203
Machine Hall #1	0.401	0.430	**0.337**	Fail	1.768	**0.869**	1.163	Fail	72.83	2757
Machine Hall #2	**0.233**	0.643	0.918	0.445	**0.123**	0.420	1.679	0.417	127.06	2251
Machine Hall #3	**0.244**	0.929	1.461	1.043	**0.337**	1.456	1.372	1.932	88.33	1578
Machine Hall #4	**0.389**	0.885	1.021	0.559	**0.299**	1.314	0.897	0.551	94.24	1832

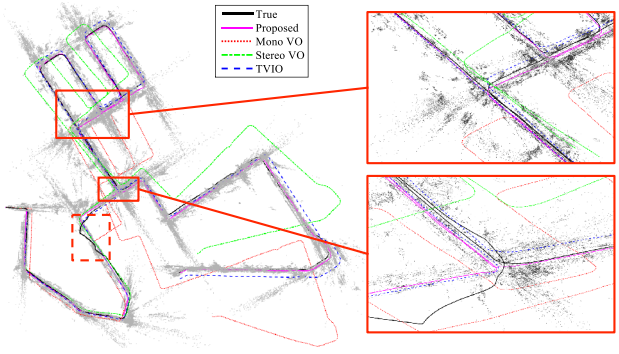

Fig. 6. Top view of the estimated trajectories on the Residential #3 in the KITTI dataset.

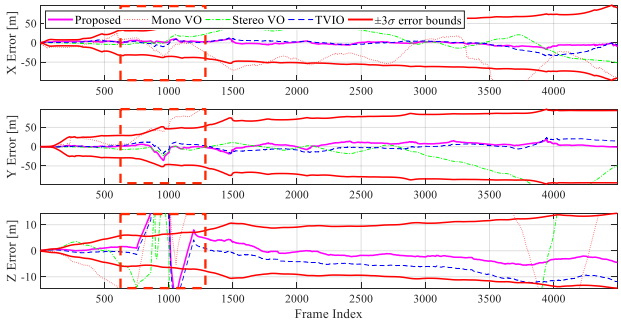

Fig. 7. Position error with $\pm 3\sigma$ error bounds from the proposed method.

the filter state of the proposed method, is reconstructed consistently throughout the entire trajectory. The above results suggest that the estimated trajectory with the proposed method is highly *consistent* and accurate compared to the other trajectory estimation results.

The estimation errors for the position are also compared against the corresponding $\pm 3\sigma$ bounds computed using the estimated covariance (red lines) with the proposed algorithm in Fig. 7. We can see that the computed covariance corresponds to the position error from the proposed method. The winding line in the ground truth trajectory shown in the dashed rectangles of Figs. 6 and 7 is not the actual traveling path of the recording platform, but the erroneous GPS data from the KITTI dataset.

The trajectory estimation results of Machine Hall #2 in the EuRoC MAV dataset, which consists of the images taken during some jerky motion of the MAV, are plotted in Fig. 8. Estimated trajectories of the proposed (magenta), ROVIO (red) [13], stereo VO (green) [8], and TVIO (blue) [23]. Two close-up views of the estimated trajectories confirm that the magenta curve is more similar than other curves to the black curve. We plot the absolute trajectory error (ATE) with respect to the frame number. The proposed method shows the lowest growth rate of the absolute trajectory error.

Please refer to the video clips submitted with this paper showing more details about the evaluations.[1]

4.2. Quantitative results

We report the root mean square error (RMSE) of the error metric for the resulting camera trajectories of the KITTI and the EuRoC MAV dataset in Tables 1 and 2.

[1] Video available at `https://youtu.be/24nnjRNjI1k`

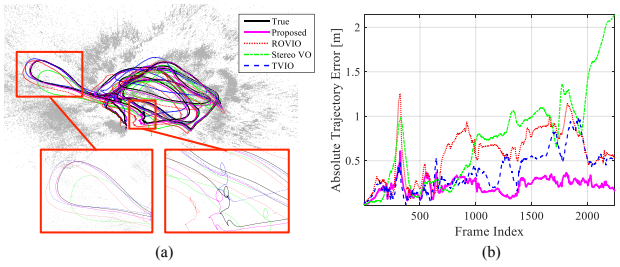

Fig. 8. Trajectory estimation results for each VO method with Machine Hall #2 in the EuRoC MAV dataset.

The smallest error for each sequence is highlighted. It shows that the proposed method is superior to the existing mono, stereo, and visual inertial odometry methods in most cases, for both KITTI and EuRoC MAV datasets. The overall final drift error of the proposed algorithm in the KITTI dataset is about 0.8 % while driving the total distance of 14.5 km. We can also observe that the absolute trajectory error (ATE) of the proposed algorithm for each KITTI and MAV dataset is very small regardless of the total traveling distance, which means that the performance of the proposed algorithm is maintained not only in the short distance, but also in the long traveling distance over 5 km.

In particular, we select Vicon Room #1 in the EuRoC MAV dataset from Table 2 to analyze the estimation results in detail. We perform the statistical analysis of the relative error metric to figure out the trend of the odometric drift error over trajectory segments of different length: $[10, 20, 30, 40, 50, 60]$ m. Fig. 9 shows translational and rotational error distribution of each VO method depending on the different length of the interval used in the calculation of the relative drift error defined in [30] and [31]. The proposed algorithm shows the smallest increase in the drift error as the traveled distance increases from 10 to 60. The average, median, and variance of the translational relative error values of the proposed method are very small in every distance length interval compared to other methods as illustrated in Fig. 9. Although ROVIO (red) has the small median of the rotational relative error values, the error range of the ROVIO boxplot is wider than the error range of the proposed method. We also observe that the heading angle accuracy of the proposed algorithm is high enough while the stereo visual odometry cannot estimate yaw direction of the vehicle accurately.

We plot comparison of average reprojection error values for each VO method in Fig. 10 for analyzing the accuracy of the proposed method from the different point of view. The reprojection error of the tracked feature points is one of the significant performance indexes in the VO methods [8], [23] because they estimate the optimal camera motion which minimizes the reprojection errors. We cannot plot the reprojection error values of ROVIO because it estimates the motion of the camera by minimizing the photometric error of the warped patches rather than the reprojection error. In both Residential #3 in the KITTI dataset (top) and Machine Hall #2 in the EuRoC MAV dataset (bottom) graphs in Fig. 10, the proposed method shows outstanding results compared to the other VO methods. The reprojection errors for the proposed method are the lowest, which means that the estimated camera motions with the proposed method are more accurate and closer to the actual camera movements than the estimation results of the other VO methods. The overall average values of the reprojection error for each VO method in top graph Fig. 10 are 0.16, 0.78, and 0.45 for the proposed algorithm (magenta), stereo VO (green), and TVIO (blue) respectively. Fig. 10 also shows that the pentafocal geometric constraints employed in the proposed method generate smaller reprojection error compared to the trifocal tensor-based reprojection error model from TVIO [23].

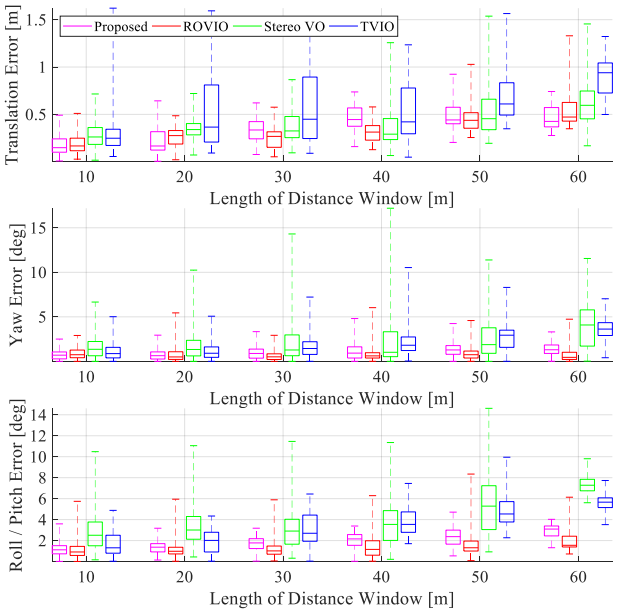

Fig. 9. Comparison of the tendency of the relative translational and rotational error with respect to the different length interval with Vicon Room #1.

We analyze the computation time between two different observation models, which are the pentafocal geometry model in the proposed method and the reprojection error model including nonlinear optimization for estimation of the 3D feature position from MSCKF in Fig 11. The pentafocal geometry constraints in the proposed method, which do not require the estimation of the 3D position of the features, take about 60 ms averagely. The reprojection error model in the MSCKF, which includes the nonlinear optimization process for estimating 3D feature position, takes about 80 ms averagely. It shows that the pentafocal observation model runs about 20 ms faster than the typical reprojection error model in average. The proposed

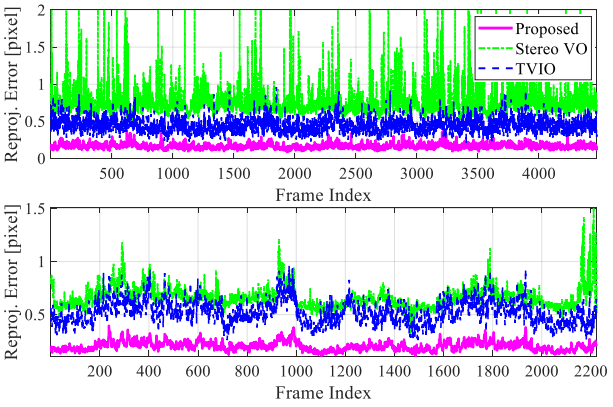

Fig. 10. Average reprojection error of the feature points for each VO method. Residential #3 in the KITTI dataset (top) and Machine Hall #2 in the EuRoC MAV dataset (bottom).

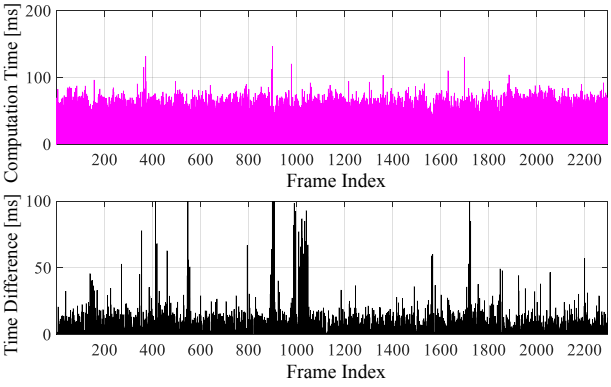

Fig. 11. Computation time for the proposed method (top) and the difference of computation time between the pentafocal geometry model and reprojection error model from MSCKF (bottom).

method written in unoptimized MATLAB codes runs in almost real time at 17 Hz, suggesting potential when implemented in C/C++ in the future.

5. CONCLUSION

In this paper, we propose pentafocal geometry for camera observation model in a sliding-window monocular visual inertial odometry. To avoid motion estimation error due to the inaccurate estimation of the 3D feature position, the pentafocal geometric constraints are employed as camera observation model, which makes the proposed visual inertial odometry without estimating 3D feature positions at all. Furthermore, the pentafocal geometry between five images is also utilized in 1-point RANSAC to select the static and reliable feature points, resulting in robust motion estimation results. From the extensive evaluation on the real experimental datasets, our method shows more accurate and precise motion estimation results with KITTI and EuRoC MAV dataset compared to other monocular, stereo, and visual inertial odometry methods. Additionally, since the 3D position of the features is not computed explicitly in the pentafocal observation model, the computation time of the proposed method is reduced compared to the reprojection error observation model. Future work will focus on the computational formal complexity analysis of the pentafocal geometric constraints compared to other popular observation models used in the visual inertial odometry.

REFERENCES

[1] D. Scaramuzza and F. Fraundorfer, "Visual odometry [tutorial]," *IEEE Robotics & Automation Magazine*, vol. 18, no. 4, pp. 80-92, 2011.

[2] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052-1067, 2007.

[3] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," *Proc. of European Conference on Computer Vision (ECCV)*, pp. 834-849, 2014.

[4] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15-22, 2014.

[5] S. Choi, J. Park, and W. Yu, "Simplified epipolar geometry for real-time monocular visual odometry on roads," *International Journal of Control, Automation and Systems*, vol. 13, no. 6, pp. 1454-1464, 2015.

[6] P. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, Springer, 2011.

[7] D. Nistér, O. Naroditsky, and J. Bergen, "Visual odometry," *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. I-I, 2004.

[8] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: dense 3d reconstruction in real-time," *Proc. of IEEE Intelligent Vehicles Symposium (IV)*, pp. 963-968, 2011.

[9] J. Engel, J. Stückler, and D. Cremers, "Large-scale direct SLAM with stereo cameras," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1935-1942, 2015.

[10] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3565-3572, 2007.

[11] E. S. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *The International Journal of Robotics Research*, vol. 30, no. 4, pp. 407-430, 2011.

[12] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314-334, 2015.

[13] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct EKF-based approach," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 298-304, 2015.

[14] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," *Robotics: Science and Systems (RSS)*, 2015.

[15] J. Civera, O. G. Grasa, A. J. Davison, and J. Montiel, "1-point RANSAC for EKF-based structure from motion," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3498-3504, 2009.

[16] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231-1237, 2013.

[17] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *The International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157-1163, 2016.

[18] J. Civera, A. J. Davison, and J. M. Montiel, "Inverse depth parametrization for monocular SLAM," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932-945, 2008.

[19] P. Piniés, T. Lupton, S. Sukkarieh, and J. D. Tardós, "Inertial aiding of inverse depth SLAM using a monocular camera," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2797-2802, 2007.

[20] G. Sibley, L. Matthies, and G. Sukhatme, "A sliding window filter for incremental SLAM," *Unifying Perspectives in Computational and Robot Vision*, pp. 103-112, 2008.

[21] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual-inertial odometry," *The International Journal of Robotics Research*, vol. 32, no. 6, pp. 690-711, 2013.

[22] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Generic and real-time structure from motion using local bundle adjustment," *Image and Vision Computing*, vol. 27, no. 8, pp. 1178-1193, 2009.

[23] J.-S. Hu and M.-Y. Chen, "A sliding-window visual-IMU odometer based on tri-focal tensor geometry," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3963-3968, 2014.

[24] C. Troiani, A. Martinelli, C. Laugier, and D. Scaramuzza, "2-point-based outlier rejection for camera-imu systems with applications to micro aerial vehicles," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5530-5536, 2014.

[25] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge university press, 2003.

[26] 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. 381-395, 1981.

[27] J. Nikolic, J. Rehder, M. Burri, P. Gohl, S. Leutenegger, P. T. Furgale, and R. Siegwart, "A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM," *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pp. 431-437, 2014.

[28] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," *Proc. of European Conference on Computer Vision (ECCV)*, pp. 404-417, 2006.

[29] B. Kitt, A. Geiger, and H. Lategahn, "Visual odometry based on stereo image sequences with RANSAC-based outlier rejection scheme," *Proc. of IEEE Intelligent Vehicles Symposium (IV)*, pp. 486-492, 2010.

[30] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 573-580, 2012.

[31] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3354-3361, 2012.

Pyojin Kim received the B.S. degree in Mechanical Engineering from Yonsei University in 2013. He is currently pursuing the M.S. and Ph.D. degrees in the Department of Mechanical and Aerospace Engineering at Seoul National University. His research interests include 3D computer vision, visual odometry, and visual SLAM.

Hyon Lim received the B.S. and M.S. degrees in Electronic and Electrical Engineering from Inha University in 2008 and 2010, and the Ph.D. degree in the Department of Mechanical and Aerospace Engineering from Seoul National University in 2015. His research interests include computer vision, real-time visual SLAM, and applications of unmanned aerial vehicles.

H. Jin Kim received the B.S. degree from Korea Advanced Institute of Technology (KAIST) in 1995, and the M.S. and Ph.D. degrees in Mechanical Engineering from University of California, Berkeley (UC Berkeley), in 1999 and 2001, respectively. From 2002 to 2004, she was a Postdoctoral Researcher in Electrical Engineering and Computer Science (EECS), UC Berkeley. In September 2004 she joined the Department of Mechanical and Aerospace Engineering at Seoul National University, Seoul, Korea, as an Assistant Professor where she is currently a Professor. Her research interests include intelligent control of robotic systems and motion planning.