

A Double Stage EKF-based Stereo Visual Inertial Odometry

Xiaogang Xiong¹, Wenqing Chen¹, Zhichao Liu² and Qiang Shen³

Abstract—This paper presents an algorithm for the real-time and robust stereo VIO (visual inertial odometry) based on two-stage EKF. In the first stage, we perform data fusion from accelerometer and gyroscope of IMU. During the second stage, we perform the data fusion the stereo camera and the result of the first stage. Due to the two stage fusion, the proposed algorithm can achieve a very high precision of VIO estimations. At the same time, the proposed algorithm maintains the advantage of computational efficiency filter-based approaches. We call our approach DS-VIO (double stage visual inertial odometry) and evaluate it by comparing with the state-of-art approaches based on optimization and filtering approaches, including OKVIS, ROVIO, VINS-MONO and S-MSCKF. The comparison results show that DS-VIO can achieve comparable or even better performances in terms of the RMS error on the EuRoC dataset.

I. INTRODUCTION

In GPS-denied environments, for instance, indoors and in urban canyons, it is essential for mobile robot platforms such as micro aerial vehicles (MAVs) to know their own pose for operations. In recent years, this pose estimation problem is popularly solved by the combination of visual information from cameras and measurements from an Inertial Measurement Unit (IMU), which is usually referred to as Visual Inertial Odometry (VIO)[1], [2]. Compared to lidar based approaches, VIO requires only a lightweight and small-size sensor package, making it the preferred technique for mobile robot platforms.

Realtime VIO solutions are critical for robots to response quickly to corresponding environments. At the same time, computation efficiency of VIO solutions is also important for robots with limited payloads and computation power such as MAVs. It is well-known that filter-based VIO solutions are generally more computationally efficient than that of optimization based methods [1] while stereo cameras are more robust to hostile environments compared with monocular camera. In order to solve the above problem of pose estimation in a realtime and robust manner, we propose a new VIO solution based on the extended Kalman filter (EKF) and information from stereo cameras in this paper.

To achieve a precise and robust pose estimation, the algorithm we present here is a dual stage EKF-based VIO solution, which is referred as to DS-VIO. It is consisted of two EKF filters and each stage has one EKF filter. We demonstrate that our algorithm perform comparable or even

higher precision than other state-of-art VIO algorithms. The primary contribution of our work is a new filter-based VIO framework that contains a dual stage EKF filter. The first stage perform the sensor fusion of the measurements of accelerometer and gyroscope. The second stage performs the fusion of the measurements of IMU and stereo camera. The dual stage of EKF allows the VIO solution achieve higher precision and robustness. To show the effects, we provide detailed comparisons between our DS-VIOS with other state-of-art open-source VIO approaches including OKVIS, ROVIO, VINS-MONO and S-MSCKF on the EuRoC dataset through experiments. The results demonstrate that our DS-VIO is able to achieve similar or even better performance than these state-of-art VIO approaches in term of precision and robustness.

The rest of the paper is orgnized as follows: Section II introduces the related work. Section III and Section IV provide the mathematical details of the first and second stage of EKF filter. Section V describes the enforcement mechanism of dual stage EKF filter. The experiments comparing our DS-VIO with state of the art open-source VIO approaches is conducted in Section VI. Finally, Section VII draws some conclusions of this paper.

II. RELATED WORK

There are a large number of work which fuses measurements from cameras and IMUs to perform pose estimations, including VIO and VI-SLAM (Simultaneous Localization and Mapping). The related work is discussed from three different aspects: (i) loosely or tightly coupled solutions. (ii) filter-based or optimization-based solutions. (iii) direct or indirect solutions.

Loosely coupled solutions process the measurements from IMUs and cameras separately. Some methods process the images for computing relative-motion estimates between consecutive poses firstly [3], [4], [5] and then fuse the result with IMU measurements. In contrast, in [6], [7], IMU measurements are used to compute rotation estimates and fuse the result with an visual estimation algorithm. Loosely coupled solutions lead to a reduction in computational cost and information loss [8]. However, the tightly coupled solutions can achieve higher accuracy [9], [10], [11], [12], [13]. In this work, we are interested in tightly coupled solutions and the proposed DS-EKF is also a tightly coupled solution.

Existing tightly coupled solutions can be divided into filter-based or optimization-based solutions. The latter generally attains higher accuracy because re-linearization at each iteration can better deal with their nonlinear measurement models, like [14], [15], [12], [16]. However, the procedure for

¹ X. Xiong and W. Chen are with Faculty of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen, P.R. China xiongxg@hit.edu.cn

² UBTECH Robotics, Shenzhen, China

³ Arizona State University, School for Engineering of Matter, Transport and Energy, USA qiang.shen@asu.edu

multiple iterations lead to heavy calculation burden. As for filter-based solutions, to date, the majority of the proposed algorithms are EKF-based methods like [17], [18], [19], [20], [11], [21], [13], also, Uncented Kalman Filter in [22], and Particle Filter in [23]. In order to improve the accuracy of filter-based solutions, some researchers focus on the consistency of estimator. The works in [24], [25] present the First Estimate Jacobian EKF (FEJ-EKF). Observability Constraint EKF (OC-EKF) is presented in [26], [27], [13]. The key idea of FEJ-EKF is to choose the same or first-ever available estimations for all the state variables as the linearization points. However, OC-EKF guarantees observability of the linearized system by ensuring the rank of nullspace of the nonlinearized system not changed after linearized. The OC-EKF is applied in our DS-VIO algorithm.

The vast majority of pose estimation methods have relied on the usage of point features, which are detected and tracked in the images like [11], [12]. The methods in [28], [29] employ edge information while the method in [2] employs both point and line features. All the above methods are classified as indirect solutions. The methods in [30], [31] present a direct method which employ the image-intensity measurements directly. The direct methods exploit more information from the images than that of indirect methods.

III. FIRST STAGE EKF FILTER DESCRIPTION

The mathematical description in this paper follows the formulations in [11]. The gyroscope state is defined as:

$$\mathbf{X}_g = [\mathbf{q}_g^T \quad \mathbf{b}_g^T]^T$$

where \mathbf{q}_g is the unit quaternion describing the rotation from inertial frame (G) to the body frame (I), $\mathbf{b}_g \in \mathbb{R}^{3 \times 1}$ describes the bias of gyroscope measurement. Here, the body frame (I) is assumed to be fixed to IMU frame.

A. Process Model

The state equation in the time-continuous form is given as follows:

$$\dot{\mathbf{q}}_g = \frac{1}{2} \mathbf{\Omega}(\boldsymbol{\omega}(t)) \mathbf{q}_g(t) \quad (1a)$$

$$\dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t) \quad (1b)$$

where $\boldsymbol{\omega} = [\omega_x \ \omega_y \ \omega_z]^T$ is the rotational velocity in the IMU frame,

$$\mathbf{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -(\boldsymbol{\omega} \times) & \boldsymbol{\omega} \\ \boldsymbol{\omega}^T & 0 \end{bmatrix}, (\boldsymbol{\omega} \times) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ \omega_y & \omega_x & 0 \end{bmatrix}$$

and the gyroscope bias \mathbf{b}_g is modeled as a random walk process with random walk rate \mathbf{n}_{wg} . The gyroscope measurements $\boldsymbol{\omega}_m$ is:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b}_g + \mathbf{n}_g \quad (2)$$

By applying the expectation operator in (1), we can get the estimates of the evolving gyroscope state:

$$\dot{\hat{\mathbf{q}}}_g = \frac{1}{2} \mathbf{\Omega}(\hat{\boldsymbol{\omega}}) \hat{\mathbf{q}}_g \quad (3a)$$

$$\dot{\hat{\mathbf{b}}}_g = \mathbf{0}_{3 \times 1} \quad (3b)$$

where $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_g$.

To propagate the uncertainty of the state, the discrete time state transition matrix should be computed, as for (3a),

$$\Phi_{k_q} = \exp\left(\int_{t_k}^{t_{k+1}} \frac{1}{2} \mathbf{\Omega}(\hat{\boldsymbol{\omega}}(\tau)) d\tau\right)_{4 \times 4} \approx \left(\mathbf{I} + \frac{t_k}{2} \mathbf{\Omega}\right)$$

where $t_k \triangleq t(k)$ and $h := t_{k+1} - t_k$ is the time interval between two IMU measurements. As for (3b), in a similar manner, one has

$$\Phi_{k_b} = \mathbf{0}_{3 \times 3}.$$

Therefore, we can get

$$\mathbf{X}_g(k+1) = \begin{bmatrix} \mathbf{q}_g(k+1) \\ \mathbf{b}_g(k+1) \end{bmatrix} = \begin{bmatrix} \Phi_{k_q} \mathbf{q}_g(k) \\ \Phi_{k_b} \mathbf{b}_g(k) \end{bmatrix} + \mathbf{w}(k).$$

Therefore, one can obtain the following expression:

$$\mathbf{X}_g(k+1) \approx \mathbf{H}_p \mathbf{X}_g + \mathbf{w}(k) \quad (4)$$

where \mathbf{H}_p is the Jacobian of process model with respect to the gyroscope state, which is shown in Appendix, $\mathbf{w}(k)$ is a 7×1 vector describe the process noise, of which the covariance matrix is as follows:

$$\mathbf{Q}_{1,k} = \begin{bmatrix} \sigma_q^2 \mathbf{I}_4 & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{4 \times 4} & \sigma_b^2 \mathbf{I}_3 \end{bmatrix}.$$

Finally, the propagated covariance of the gyroscope state is described as:

$$\mathbf{P}_{G|G_{k+1}|k} = \mathbf{H}_p \mathbf{P}_{G|G_k|k} \mathbf{H}_p^T + \mathbf{Q}_{1,k}$$

B. Measurement Model

The first correction stage uses data from the accelerometers to correct the gyroscope state, the measurement is,

$$\mathbf{z}(k) = \mathbf{a}(k) = [a_x(k) \ a_y(k) \ a_z(k)]^T \quad (5)$$

According to [32], $\hat{\mathbf{z}}(k)$ can be presented by

$$\hat{\mathbf{z}}(k) = \mathbf{R}_n^b \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = -g \begin{bmatrix} 2q_1q_3 - 2q_0q_2 \\ 2q_0q_1 + 2q_2q_3 \\ q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}$$

where g is the constant g-force acceleration, and

$$\mathbf{R}_n^b = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) \\ & 2(q_1q_3 - q_0q_2) \\ & 2(q_2q_3 + q_0q_1) \\ & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}.$$

The residual of the measurement can be approximated as

$$\mathbf{r}(k) = \mathbf{z}(k) - \hat{\mathbf{z}}(k) = \mathbf{H}_g \mathbf{X}_g + \mathbf{v}(k) \quad (6)$$

where \mathbf{H}_g is the Jacobian of measurement with respect to the gyroscope state, which is shown in Appendix, $\mathbf{v}(k) \in \mathbb{R}^{3 \times 1}$ describes the measurement noise, of which the covariance matrix is $\sigma_a^2 \mathbf{I}_3$,

IV. SECOND STAGE EKF FILTER DESCRIPTION

The evolving IMU state is defined as follows:

$$\mathbf{X}_I = [\begin{smallmatrix} I_G \bar{\mathbf{q}}^T & \mathbf{b}_g^T & {}^G \mathbf{v}_I^T & \mathbf{b}_a^T & {}^G \mathbf{p}_I^T \end{smallmatrix}]^T$$

where ${}^G \mathbf{v}_I$ and ${}^G \mathbf{p}_I \in \mathbb{R}^{3 \times 1}$ are vectors describing the IMU position and velocity in frame G , and $\mathbf{b}_a \in \mathbb{R}^{3 \times 1}$ describes the bias of accelerometer measurement. The bias is modeled as random walk processes and the random walk rate is \mathbf{n}_{wa} . The IMU error-state is defined as:

$$\tilde{\mathbf{X}}_I = [\delta \theta_I^T \quad \tilde{\mathbf{b}}_g^T \quad {}^G \tilde{\mathbf{v}}_I^T \quad \tilde{\mathbf{b}}_a^T \quad {}^G \tilde{\mathbf{p}}_I^T]^T$$

where the standard additive error defined as $\tilde{x} \triangleq x - \hat{x}$ with $x \in \{\mathbf{b}_g^T, {}^G \mathbf{v}_I^T, \mathbf{b}_a^T, {}^G \mathbf{p}_I^T\}$ and $\hat{x} \in \{\hat{\mathbf{b}}_g^T, {}^G \hat{\mathbf{v}}_I^T, \hat{\mathbf{b}}_a^T, {}^G \hat{\mathbf{p}}_I^T\}$ is used for position, velocity and bias, respectively, and $\delta \theta_I^T$ represents the quaternion error [11], [33]. Ultimately, we add the N camera poses in the IMU state, we can get the EKF state vector, at time-step, it can be defined as:

$$\mathbf{X}_E = [\mathbf{X}_{I_k}^T \quad {}^{C_1} \bar{\mathbf{q}}^T \quad {}^G \mathbf{p}_{C_1}^T \quad \cdots \quad {}^{C_N} \bar{\mathbf{q}}^T \quad {}^G \mathbf{p}_{C_N}^T]^T$$

where the ${}^{C_i} \bar{\mathbf{q}}^T, {}^G \mathbf{p}_{C_i}^T, i = 1, 2, \dots, N$ is the camera attitude and position, respectively. The EKF error state vector is defined as:

$$\tilde{\mathbf{X}}_E = [\tilde{\mathbf{X}}_{I_k}^T \quad \delta \theta_{C_1}^T \quad {}^G \tilde{\mathbf{p}}_{C_1}^T \quad \cdots \quad \delta \theta_{C_N}^T \quad {}^G \tilde{\mathbf{p}}_{C_N}^T]^T$$

A. Process Model

The continuous-time system model is described as :

$$I_G \dot{\bar{\mathbf{q}}} = \frac{1}{2} \Omega(\omega(t)) I_G \bar{\mathbf{q}}(t) \quad (7a)$$

$$\dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t) \quad (7b)$$

$${}^G \dot{\mathbf{v}}_I(t) = {}^G \mathbf{a}(t) \quad (7c)$$

$$\dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa}(t) \quad (7d)$$

$${}^G \dot{\mathbf{p}}_I(t) = {}^G \mathbf{v}_I(t) \quad (7e)$$

where ${}^G \mathbf{a}$ is the body acceleration in the global frame (G). The accelerometer measurement \mathbf{a}_m is:

$$\mathbf{a}_m = C({}^I_G \bar{\mathbf{q}})({}^G \mathbf{a} - {}^G \mathbf{g}) + \mathbf{b}_a + \mathbf{n}_a$$

where $C(\cdot)$ denotes a rotational matrix, \mathbf{n}_g and \mathbf{n}_a are measurement noise, ${}^G \mathbf{g}$ is gravitational acceleration expressed in the local frame. From (1) to (4), we can get the equations for propagating the estimates of the evolving IMU state by applying expectation operator:

$$I_G \dot{\bar{\mathbf{q}}} = \frac{1}{2} \Omega(\hat{\omega}) I_G \hat{\bar{\mathbf{q}}} \quad (8a)$$

$$\dot{\hat{\mathbf{b}}}_g = \mathbf{0}_{3 \times 1} \quad (8b)$$

$${}^G \dot{\hat{\mathbf{v}}}_I = C({}^I_G \hat{\bar{\mathbf{q}}})^T \hat{\mathbf{a}} + {}^G \mathbf{g} \quad (8c)$$

$$\dot{\hat{\mathbf{b}}}_a = \mathbf{0}_{3 \times 1} \quad (8d)$$

$${}^G \dot{\hat{\mathbf{p}}}_I = {}^G \hat{\mathbf{v}}_I \quad (8e)$$

where $\hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a$. Therefore, the linearized continuous-time model for the IMU error-state is:

$$\dot{\tilde{\mathbf{X}}}_I = \mathbf{F} \tilde{\mathbf{X}}_I + \mathbf{G} \mathbf{n}_I \quad (9)$$

where $\mathbf{n}_I = [\mathbf{n}_g^T \quad \mathbf{n}_{wg}^T \quad \mathbf{n}_a^T \quad \mathbf{n}_{wa}^T]^T$, \mathbf{F} and \mathbf{G} are shown in appendix. The state equation model can be computed as follows:

$$\tilde{\mathbf{X}}_E(k+1) = \Phi_k \tilde{\mathbf{X}}_E(k) + \mathbf{Q}_{2,k} \quad (10)$$

where the computation of transform matrix Φ_k and the discrete time noise covariance \mathbf{Q}_k can be found in [13]. So the propagated covariance of the IMU state is described as:

$$\mathbf{P}_{I|I_{k+1}|k} = \Phi_k \mathbf{P}_{I|I_k|k} \Phi_k^T + \mathbf{Q}_{2,k}$$

where the covariance matrix $\mathbf{P}_{I|I_k|k}$ and $\mathbf{P}_{I|I_{k+1}|k}$ are described as:

$$\mathbf{P}_{E|E_k|k} = \begin{bmatrix} \mathbf{P}_{I|I_k|k} & \mathbf{P}_{I|C_k|k} \\ \mathbf{P}_{C|I_k|k} & \mathbf{P}_{C|C_k|k} \end{bmatrix}$$

$$\mathbf{P}_{E|E_{k+1}|k} = \begin{bmatrix} \mathbf{P}_{I|I_{k+1}|k} & \Phi_k \mathbf{P}_{I|C_k|k} \\ \mathbf{P}_{C|I_{k+1}|k} \Phi_k^T & \mathbf{P}_{C|C_k|k} \end{bmatrix}.$$

When recording a new camera, we should add the camera pose into the state by using the following expressions:

$${}^G \hat{\bar{\mathbf{q}}} = {}^I_G \bar{\mathbf{q}} \otimes {}^I_G \bar{\mathbf{q}}$$

$${}^G \hat{\mathbf{p}}_C = {}^G \hat{\mathbf{p}}_I + C({}^I_G \hat{\bar{\mathbf{q}}})^T I \mathbf{p}_C$$

The augmented covariance is given by:

$$\mathbf{P}_{k|k} \leftarrow \begin{bmatrix} \mathbf{I}_{6N+15} \\ \mathbf{J} \end{bmatrix} \mathbf{P}_{k|k} \begin{bmatrix} \mathbf{I}_{6N+15} \\ \mathbf{J} \end{bmatrix}^T \quad (11)$$

where \mathbf{J} is shown in Appendix.

B. Measurement Model

The second correction stage employs data from the images to correct the IMU state, of which the measurement model we adopted here follows [11], [13]. It is based the fact that static features in the world can be observed from multiple camera poses showing constraints among all these camera poses. Consider the case of a single feature f_j observed by the stereo cameras at time step i . Assume this feature are observed by both left and right cameras, and the left and right cameras poses are represented as $({}^{C_{i,1}} \bar{\mathbf{q}}, {}^G \mathbf{p}_{C_{i,1}})$ and $({}^{C_{i,2}} \bar{\mathbf{q}}, {}^G \mathbf{p}_{C_{i,2}})$. For the feature f_j at time step i , the stereo measurement is,

$$\mathbf{z}_i^j = \begin{bmatrix} u_{i,1}^j \\ v_{i,1}^j \\ u_{i,2}^j \\ v_{i,2}^j \end{bmatrix} = \begin{bmatrix} {}^{C_{i,1}} Z_j & \mathbf{0}_{2 \times 2} \\ \mathbf{0}_{2 \times 2} & {}^{C_{i,2}} Z_j \end{bmatrix} \begin{bmatrix} {}^{C_{i,1}} X_j \\ {}^{C_{i,1}} Y_j \\ {}^{C_{i,2}} X_j \\ {}^{C_{i,2}} Y_j \end{bmatrix}$$

where $[{}^{C_{i,k}} X_j \quad {}^{C_{i,k}} Y_j \quad {}^{C_{i,k}} Z_j]^T, k = 1, 2$ represent the positions of feature f_j in the left and right camera frame are given by,

$${}^{C_i} \mathbf{p}_{f_j} = \begin{bmatrix} {}^{C_{i,k}} X_j \\ {}^{C_{i,k}} Y_j \\ {}^{C_{i,k}} Z_j \end{bmatrix} = C({}^{C_i} \bar{\mathbf{q}})({}^G \mathbf{p}_{f_j} - {}^G \mathbf{p}_{C_{i,k}})$$

where ${}^G \mathbf{p}_{f_j}$ is the position of feature f_j in the global frame, it can be computed the least square method shown in [11].

Once ${}^G\mathbf{p}_{f_j}$ is obtained, the measurement residual can be computed by:

$$\mathbf{r}_i^j = \mathbf{z}_i^j - \hat{\mathbf{z}}_i^j.$$

By linearizing about the estimates for the camera poses and feature position, the residual can be approximated as:

$$\mathbf{r}_i^j \approx \mathbf{H}_{X_i}^j \tilde{\mathbf{X}} + \mathbf{H}_{f_i}^j {}^G\tilde{\mathbf{p}}_{f_j} + \mathbf{n}_i^j$$

where \mathbf{n}_i^j is the noise of the measurement, $\mathbf{H}_{X_i}^j$ and $\mathbf{H}_{f_i}^j$ are the Jacobian matrixes of the measurement \mathbf{z}_i^j with respect to the state and the feature position, respectively, of which the value of these two Jacobian matrixes can be found in [13], ${}^G\tilde{\mathbf{p}}_{f_j}$ is the error in the position estimate of f_j . By stacking all the observations of feature f_j , one can obtain:

$$\mathbf{r}^j \approx \mathbf{H}_X^j \tilde{\mathbf{X}} + \mathbf{H}_f^j {}^G\tilde{\mathbf{p}}_{f_j} + \mathbf{n}^j.$$

Because the position of feature ${}^G\mathbf{p}_{f_j}$ is computed by using the state estimate \mathbf{X} , the error ${}^G\tilde{\mathbf{p}}_{f_j}$ is correlated with the error $\tilde{\mathbf{X}}$. The form of residual cannot be directly used for update in the EKF. By projecting \mathbf{r}^j to the nullspace of \mathbf{H}_f^j , one has

$$\mathbf{r}_o^j = \mathbf{A}^T \mathbf{r}^j \approx \mathbf{A}^T \mathbf{H}_X^j \tilde{\mathbf{X}} + \mathbf{A}^T \mathbf{n}^j = \mathbf{H}_{X,o}^j \tilde{\mathbf{X}} + \mathbf{n}_o^j.$$

where \mathbf{A} denotes the unitary matrix whose columns form the basis of the left nullspace of \mathbf{H}_f^j [11].

V. DUAL STAGE EKF MECHANISM

The structure of the proposed dual stage of EKF filter is shown in Fig.1. The first stage of EKF is designed to combine the measurements from accelerometer and gyroscope. By implementing the first stage EKF filter, data from accelerometer is used as a corrective measure by taking into account the gravitational force to curb the error of the orientation estimate. The role of the second stage EKF is fusing measurements from IMU and stereo cameras. The images from stereo cameras can present a natural complement to IMU to curb the errors and drift.

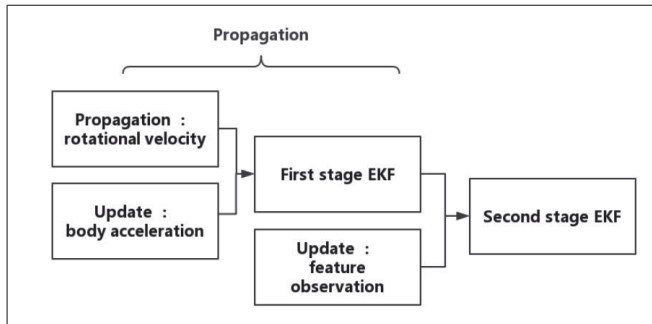


Fig. 1. The block diagram of illustrating the structure of proposed dual stage EKF filter

As shown in Fig.1, the first stage of EKF is executed immediately as the IMU measurements are acquired. For each IMU measurement received, one has:

- **Propagation:** The rotational velocity ω_m is used to propagate the gyroscope state \mathbf{X}_g , and covariance matrix $\mathbf{P}_{G|G}$.

- **update:** The body acceleration \mathbf{a}_m is used to perform an EKF update.

The second stage of EKF contains the first stage, and the first stage is regarded as the propagate part in the second stage EKF:

- **Propagation:** Whenever a new IMU measurement is received, it propagates the IMU state \mathbf{X}_I and covariance matrix $\mathbf{P}_{E|E}$.
- **Image registration:** Whenever new stereo images are acquired, it (i) augments the IMU state and the corresponding covariance matrix; (ii) operates image front-end processes, including feature extraction and feature matching.
- **update:** EKF update is performed when (i) a feature has been tracked in a number of images (the number is 3 in our algorithm) is no longer detected; (ii) camera poses will be discarded when the largest number of camera poses in the IMU state has been reached.

We present a new strategy to discard old camera poses in the above **update**. The oldest non-keyframe is discarded according to two criteria whenever new stereo images are received. The two criteria from [12] is employed here for keyframe selection. The first one is the average parallax apart from the previous keyframe and the second one is tracking quality.

VI. EXPERIMENTS

In this section, we perform two experiments to evaluate the proposed DS-VIO algorithm. In the first experiment, we compare the proposed algorithm with other state-of-the-art algorithms on public dataset by analyzing the Root Mean Square Error (RMSE) metrics. In the second experiment, we compare the proposed DS-VIO with S-MSCKF in the indoor environment. S-MSCKF is selected because it is also one of stereo and EKF filter based approaches.

A. Dataset Comparison

The proposed DS-VIO algorithm (stereo-filter) is compared with the state of the art approaches including OKVIS [14] (stereo optimization), ROVIO [33] (monocular filter), VINS-MONO [12] (monocular optimization) and S-MSCKF [13] (stereo filter) on the EuRoC dataset [34]. These methods are different combinations of monocular, stereo, filter-based, and optimization-based methods. Among these methods, S-MSCKF is a tightly-coupled filtering-based stereo VIO algorithm which is closely related to our work. During the experiment, only the images from the left camera are used for monocular camera based algorithms like ROVIO and VINS-MONO. In order to perform a convictive experiment, only the performance of VIO is conducted and the functionality of loop closure is disabled for VINS-MONO. For each algorithm, its performance is evaluated for repeating experiments ten times and the mean value is treated as the result.

The EuRoC MAV Dataset is a collection of visual-inertial datasets collected on-board a Micro Aerial Vehicle (MAV). It contains synchronized 20Hz stereo images and 200Hz IMU messages, accurate motion and structure ground-truth.

Some parts of the dataset exhibit very dynamic motions and different image brightness, which renders stereo matching and feature tracking more challenging.

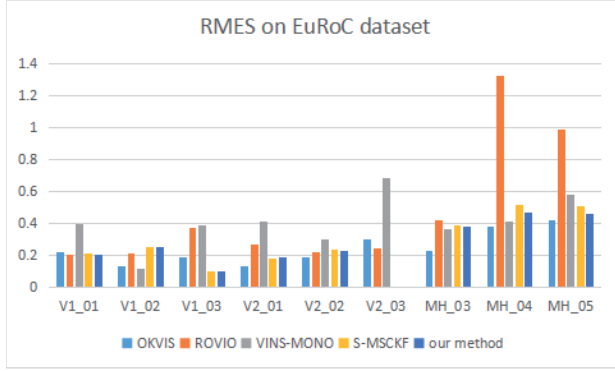


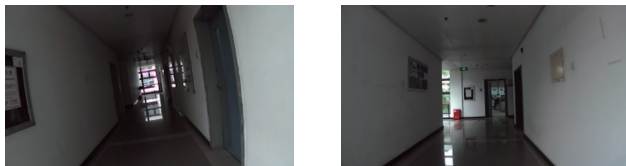
Fig. 2. Root Mean Square Error (RMSE) results

Figure 2 shows the RMSE results of our proposed DS-VIO algorithm and other state of the art algorithms including OKVIS, ROVIO, VINS-MONO and S-MSCKF on the EuRoC datasets. Because both our proposed method and S-MSCKF employ the KLT optical flow algorithm for feature tracking and stereo matching, they do not work properly on “V2.03” dataset. As pointed out in [13], the rapid change of brightness in “V2.03” dataset causes failures in the stereo feature matching. On the rest datasets, our method achieves comparable performance with other methods. For the most similar method S-MSCKF, we can see that our method has better performance than that of S-MSCKF on all the datasets.

B. Indoor Experiment

In the indoor experiment, for the sake of fairness, we only compare S-MSCKF with our DS-VIO because both algorithms are stereo-filter based approaches. We choose rectangular corridor in our laboratory building as the experiment area. We encounter low light and texture-less condition in the corridor environment, as shown in Fig.3. The sensor suite we use is ZED-mini, which contains stereo cameras (30hz) and an IMU (800hz). With ZED-mini in our hands, we walk along the rectangular corridor for a circle around 1m/s.

Fig.4 shows the comparison results between our DS-VIO and S-MSCKF. Fig.4(a) shows the trajectories in the xy plane and Fig.4(b) shows the trajectories in three dimensions. The blue line represents trajectory of DS-VIO while the dotted line represents the trajectory of S-MSCKF. One can



(a) Low light

(b) Texture-less

Fig. 3. Images for the features of corridor environment

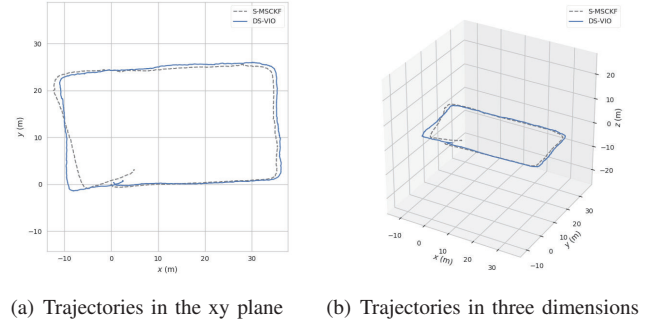


Fig. 4. Results of S-MSCKF and DS-VIO

that the blue trajectory is a nonstandard rectangle but the dotted trajectory is far away from a rectangle. This comparison shows that the proposed DS-VIO achieves a smaller cumulative error than that S-MSCKF. One can attribute the archived senior performance to the first stage of EKF filter of the proposed DS-VIO, which employs the complementary characteristics between accelerometer and gyroscope.

VII. CONCLUSIONS

In this paper, we present a robust and efficient filter-based stereo VIO. It employs a dual stage of EKF to perform the state estimation. This dual stage EKF filter employ the complementary characters of IMU and stereo cameras as well as accelerometer and gyroscope. The accuracy and robustness of the proposed VIO is demonstrated by experiments of the EuRoC MAV Dataset and indoor environment by comparing with the state of the art VIO algorithms. The further work should explore how to achieve better accuracy and efficiency by selecting feature points which have some certain characters.

APPENDIX

The \mathbf{H}_p in (4) is,

$$\mathbf{H}_p = \begin{bmatrix} \mathbf{H}_{p1} & \mathbf{H}_{p2} \\ \mathbf{0}_{3 \times 4} & \mathbf{I}_3 \end{bmatrix}$$

where

$$\mathbf{H}_{p1} = \begin{bmatrix} 1 & -0.5\hat{\omega}_x t & -0.5\hat{\omega}_y t & -0.5\hat{\omega}_z t \\ 0.5\hat{\omega}_x t & 1 & 0.5\hat{\omega}_y t & -0.5\hat{\omega}_z t \\ 0.5\hat{\omega}_y t & -0.5\hat{\omega}_z t & 1 & 0.5\hat{\omega}_x t \\ 0.5\hat{\omega}_z t & 0.5\hat{\omega}_y t & -0.5\hat{\omega}_x t & 1 \end{bmatrix}$$

and

$$\mathbf{H}_{p2} = \begin{bmatrix} 0.5q_1 t & 0.5q_2 t & 0.5q_3 t \\ -0.5q_0 t & 0.5q_3 t & -0.5q_2 t \\ -0.5q_3 t & -0.5q_2 t & 0.5q_1 t \\ 0.5q_2 t & -0.5q_1 t & -0.5q_0 t \end{bmatrix},$$

$\hat{\omega} = [\hat{\omega}_x \ \hat{\omega}_y \ \hat{\omega}_z]^T$ is the estimation of rotational velocity.

The \mathbf{H}_g in (6) is,

$$\mathbf{H}_g = \begin{bmatrix} 2gq_2 & -2gq_3 & 2gq_0 & -2gq_1 & 0 & 0 & 0 \\ -2gq_1 & -2gq_0 & -2gq_3 & -2gq_2 & 0 & 0 & 0 \\ -2gq_0 & 2gq_1 & 2gq_2 & 2gq_3 & 0 & 0 & 0 \end{bmatrix}$$

where g is the g-force acceleration. The \mathbf{F} and \mathbf{G} in (9) are as follows:

$$\mathbf{F} = \begin{bmatrix} -(\hat{\omega} \times) & -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ -C(\hat{\mathbf{q}}^T) \cdot (\hat{\mathbf{a}} \times) & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -C(\hat{\mathbf{q}}^T) & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}$$

and

$$\mathbf{G} = \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -C(\hat{\mathbf{q}}^T) & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_3 \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}.$$

The \mathbf{J} in (11) is follows:

$$\mathbf{J} = \begin{bmatrix} C(\hat{\mathbf{q}}^T) & \mathbf{0}_{3 \times 9} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{6 \times N} \\ (C(\hat{\mathbf{q}}^T) \mathbf{p}_I) \times & \mathbf{0}_{3 \times 9} & \mathbf{I}_3 & \mathbf{0}_{6 \times N} \end{bmatrix}.$$

REFERENCES

- [1] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, "Robust stereo visual inertial odometry for fast autonomous flight," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, April 2018.
- [2] F. Zheng, G. Tsai, Z. Zhang, S. Liu, C.-C. Chu, and H. Hu, "Pi-vio: Robust and efficient stereo visual inertial odometry using points and lines," *arXiv preprint arXiv:1803.02403*, 2018.
- [3] J. Ma, S. Susca, M. Bajracharya, L. Matthies, M. Malchano, and D. Wooden, "Robust multi-sensor, day/night 6-dof pose estimation for a dynamic legged vehicle in gps-denied environments," in *IEEE International Conference on Robotics and Automation*, 2012, pp. 619–626.
- [4] J. P. Tardif, M. George, M. Laverne, and A. Kelly, "A new approach to vision-aided inertial navigation," in *Ieee/rsj International Conference on Intelligent Robots and Systems*, 2010, pp. 4161–4168.
- [5] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 4531–4537.
- [6] L. Matthies, "Fully self-contained vision-aided navigation and landing of a micro air vehicle independent from external sensor inputs," *Proc Spie*, vol. 8387, p. 25, 2012.
- [7] T. Oskiper, Z. Zhu, S. Samarasekera, and R. Kumar, "Visual odometry system using multiple stereo cameras and inertial measurement unit," in *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, 2007, pp. 1–8.
- [8] M. Li, "Visual-inertial odometry on resource-constrained systems," *Dissertations Theses - Gradworks*, 2014.
- [9] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008.
- [10] K. Konolige, M. Agrawal, and J. Solà, "Large-scale visual odometry for rough terrain," in *Robotics Research - the International Symposium, Isrr 2007, November 26-29, 2007 in Hiroshima, Japan*, 2010, pp. 201–212.
- [11] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Robotics and automation, 2007 IEEE international conference on*. IEEE, 2007, pp. 3565–3572.
- [12] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [13] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C. J. Taylor, and V. Kumar, "Robust stereo visual inertial odometry for fast autonomous flight," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 965–972, 2018.
- [14] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2014.
- [15] T. Lupton and S. Sukkarieh, *Visual-Inertial-Aided Navigation for High-Dynamic Motion in Built Environments Without Initial Conditions*. IEEE Press, 2012.
- [16] V. Usenko, J. Engel, J. Stückler, and D. Cremers, "Direct visual-inertial odometry with stereo cameras," in *IEEE International Conference on Robotics and Automation*, 2016, pp. 1885–1892.
- [17] E. S. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *International Journal of Robotics Research*, vol. 30, no. 4, pp. 407–430, 2011.
- [18] J. Kelly, S. Saripalli, and G. S. Sukhatme, *Combined Visual and Inertial Navigation for an Unmanned Aerial Vehicle*. Springer Berlin Heidelberg, 2008.
- [19] J. Kelly and G. S. Sukhatme, "Visual-inertial simultaneous localization, mapping and sensor-to-sensor self-calibration," in *IEEE International Conference on Computational Intelligence in Robotics and Automation*, 2009, pp. 360–368.
- [20] M. Kleinert and S. Schleith, "Inertial aided monocular slam for gps-denied navigation," in *Multisensor Fusion and Integration for Intelligent Systems*, 2010, pp. 20–25.
- [21] P. Pinies, T. Lupton, S. Sukkarieh, and J. D. Tardos, "Inertial aiding of inverse depth slam using a monocular camera," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 2797–2802.
- [22] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, 2011.
- [23] T. Yap, M. Li, A. I. Mourikis, and C. R. Shelton, "A particle filter for monocular vision-aided odometry," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5663–5669.
- [24] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, *A First-Estimates Jacobian EKF for Improving SLAM Consistency*. DBLP, 2009.
- [25] —, "Observability-based rules for designing consistent ekf slam estimators," *International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, 2010.
- [26] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 158–176, 2017.
- [27] S. I. Roumeliotis, D. G. Kottas, C. Guo, and J. Hesch, "Observability-constrained vision-aided inertial navigation," 2016.
- [28] J. J. Tarrio and S. Pedre, "Realtime edge-based visual odometry for a monocular camera," in *IEEE International Conference on Computer Vision*, 2016, pp. 702–710.
- [29] H. Yu and A. I. Mourikis, "Edge-based visual-inertial odometry," in *Ieee/rsj International Conference on Intelligent Robots and Systems*, 2017, pp. 6670–6677.
- [30] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "Imu preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation (supplementary material)," *Georgia Institute of Technology*, 2015.
- [31] X. Zheng, Z. Moratto, M. Li, A. I. Mourikis, X. Zheng, Z. Moratto, M. Li, and A. I. Mourikis, "Photometric patch-based visual-inertial odometry," in *IEEE International Conference on Robotics and Automation*, 2017, pp. 3264–3271.
- [32] S. Sabatelli, M. Galgani, L. Fanucci, and A. Rocchi, "A double-stage kalman filter for orientation tracking with an integrated processor in 9-d imu," *IEEE Transactions on Instrumentation Measurement*, vol. 62, no. 3, pp. 590–598, 2013.
- [33] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct ekf-based approach," in *Ieee/rsj International Conference on Intelligent Robots and Systems*, 2015, pp. 298–304.
- [34] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *International Journal of Robotics Research*, vol. 35, no. 10, pp. 1157–1163, 2016.