

Received September 10, 2019, accepted October 3, 2019, date of publication October 10, 2019, date of current version October 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2946352

A Novel Feedback Mechanism-Based Stereo Visual-Inertial SLAM

JINQIANG BAI^{1,3}, JUNQIANG GAO², YIMIN LIN², ZHAOXIANG LIU², SHIGUO LIAN², (Member, IEEE), AND DIJUN LIU³

¹School of Electronic Information Engineering, Beihang University, Beijing 100083, China

²Department of AI, CloudMinds Technologies Inc., Beijing 100102, China

³Morningcore Technology Company Ltd., Beijing 100083, China

Corresponding author: Junqiang Gao (devin.gao@cloudminds.com)

ABSTRACT Simultaneous Localization and Mapping (SLAM) combining visual and inertial measurements has achieved significant attention in the community of Robotics and Computer Vision. However, it is still a challenge to balance real-time requirements and accuracy. Therefore, this paper proposes a feedback mechanism for stereo Visual-Inertial SLAM (VISLAM) to provide accurate and real-time motion estimation and map reconstruction. The key idea of the feedback mechanism is that the frontend and backend in the VISLAM system can promote each other. The results of the backend optimization are fed back to the Kalman Filter (KF)-based frontend to reduce the motion estimate error caused by the well-known linearization of the KF estimator. Conversely, this more accurate motion estimate of the frontend can accelerate the backend optimization since it provides a more accurate initial state for the backend. In addition, we design a relocalization and continued SLAM framework with the feedback mechanism for the application of autonomous robot navigation or continuing SLAM. We evaluated the performance of the proposed VISLAM system through experiments on public EuRoC dataset and real-world environments. The experimental results demonstrate that our system is a promising VISLAM system compared with other state-of-the-art VISLAM systems in terms of both computing cost and accuracy.


INDEX TERMS Kalman filter, nonlinear optimization, visual and inertial sensor fusion, visual-inertial simultaneous localization and mapping.

I. INTRODUCTION

Visual Odometry (VO) [1], [2] and Visual Simultaneous Localization and Mapping (VSLAM) [3], [4] techniques, as the solutions of localization and mapping in GPS-denied environments, have been extensively studied in many applications to computer vision, Augmented and Virtual Reality (AR&VR), and mobile robotics [5]. However, the pure vision-based VO/VSLAM methods are sensitive to the challenging scenarios, such as textureless surfaces, motion blur, occlusions and illumination changes [6]–[9]. To address these problems, Visual Inertial Odometry (VIO) or Visual Inertial Simultaneous Localization and Mapping (VISLAM) techniques [10] fuse Inertial Measurement Unit (IMU) data to the VO/VSLAM system and achieve more robustness and higher accuracy even in the above challenging scenarios. Most existing VIO/VISLAM approaches focus on monocular

systems (e.g., MSCKF [11], ROVIO [12], OKVIS [13], [14], VIORB [15], VINS-Mono [16], Maplab [17], PL-VIO [18] and SR-ISWF [19]). Although the scale in monocular VIO/VISLAM systems can be observed from the IMU's accelerometer, it is usually imprecise since the dominant gravity vector is unable to be subtracted accurately from the noisy acceleration measurements [20]. However, stereo vision-based VIO/VISLAM methods [20]–[35] can provide additional scale information by employing an epipolar constraint and achieve higher accuracy [36].

The stereo vision-based VIO/VISLAM is generally divided into two methodologies [10]: filtering-based (e.g., PIRVS [30], S-MSCKF [31] and Trifo-VIO [34]) and optimization-based (e.g., ICE-BA [33] and VINS-Fusion [35]). The comparison in [10], [37] found that the latter has more potential than the former in terms of localization accuracy, while the former has advantages in terms of computing cost. To balance the real-time requirements and accuracy, we propose a feedback mechanism that combines

The associate editor coordinating the review of this manuscript and approving it for publication was Heng Wang .

the filtering-based and optimization-based approaches into one VISLAM system. Inspired by the idea of [38], we feed the optimized state produced by the backend back to the frontend to correct the state estimation. In return, these more accurate states obtained from the frontend can accelerate the optimization process of the backend. In a word, the feedback mechanism makes the frontend and backend in the VIO/VISLAM system be able to promote each other, and thus, improves the efficiency and the accuracy of VIO/VISLAM.

On the other hand, according to the sensor fusion type, VIO/VISLAM can also be further categorized into either loosely-coupled or tightly-coupled approaches. Loosely-coupled approaches fuse two separate estimators (one for processing images and the other one for IMU measurements) to obtain the final relative motion, whereas tightly-coupled approaches joint the vision and IMU measurements into one estimator to find the optimal estimates. Generally, tightly-coupled approaches are more accurate and robust than loosely-coupled approaches [18]. Therefore, we use the tightly-coupled method for our VISLAM system.

In addition, in many applications such as environment mapping and autonomous robot navigation, the system should have the ability of relocalization in a previous built map and continuing SLAM in the previous unknown parts of the environment [29]. Thus, we also design a long-term relocalization framework with the proposed feedback mechanism to detect the revisited environments and continue to map the unvisited environments more accurately.

In summary, the main contributions of this work are as follows:

- To the best of our knowledge, this paper is the first tightly-coupled stereo VISLAM system that combines filtering-based frontend and optimization-based backend through a feedback mechanism, resulting in a significant improvement in accuracy and efficiency of the 6 DoF pose estimation.
- With the proposed feedback mechanism, we also design a long-term relocalization framework that is able to perform visual-inertial localization in a previous visited environment as well as carry on map reconstruction in an unvisited area.
- We compare the performance of the proposed system with other state-of-the-art VIO/VISLAM systems on both the EuRoc dataset [39] and real-world scenarios. The experimental results demonstrate that our system has a good performance on both accuracy and efficiency.

The remainder of the paper is organized as follows. The previous relevant literatures are presented in Section II. In Section III, the proposed VISLAM system is described in detail. Subsequently, in Section IV, the relocalization and continued SLAM framework is introduced. Section V shows the experimental results and the comparisons with other state-of-the-art VIO/VISLAM methods. Finally, conclusions and potential future works are given in Section VI.

TABLE 1. State-of-the-art VIO/VISLAM systems.

Year	Paper	Approach	Fusion Type	System Type
2010	[21]	Filtering-based	Loosely-coupled	VIO
2012	[22]	Filtering-based	Tightly-coupled	VISLAM
2014	[23]	Filtering-based	Tightly-coupled	VIO
2015	[24]	Optimization-based	Tightly-coupled	VISLAM
2015	[25]	Filtering-based	Tightly-coupled	VIO
2015	[26]	Optimization-based	Tightly-coupled	VISLAM
2016	[27]	Optimization-based	Tightly-coupled	VISLAM
2016	Duo-VIO [28]	Filtering-based	Loosely-coupled	VIO
2017	[29]	Optimization-based	Tightly-coupled	VISLAM
2018	PIRVS [30]	Filtering-based	Tightly-coupled	VISLAM
2018	S-MSCKF [31]	Filtering-based	Tightly-coupled	VIO
2018	[32]	Optimization-based	Tightly-coupled	VISLAM
2018	ICE-BA [33]	Optimization-based	Tightly-coupled	VISLAM
2018	Trifo-VIO [34]	Filtering-based	Tightly-coupled	VIO
2018	[20]	Filtering-based	Tightly-coupled	VIO
2019	VINS-Fusion [35]	Optimization-based	Tightly-coupled	VISLAM

II. RELATED WORK

Much research has been conducted on VIO/VISLAM problem in the past few decades, we refer to the review paper [10] to discuss the stereo VIO/VISLAM systems developed in the last 10 years.

As mentioned in Section I, VIO/VISLAM approaches can be classified into filtering-based and optimization-based methods, and can be further divided into loosely-coupled and tightly-coupled systems according to sensor fusion type [38]. State-of-the-art VIO/VISLAM systems over the last 10 years are listed in Table 1. Loosely-coupled approaches such as [21] and Duo-VIO [28] estimate the pose by the visual and IMU measurements separately. Consequently, the accumulated drift in VO cannot be eliminated by using the IMU measurements, resulting in a sub-optimal estimate [38]. In contrast, the tightly-coupled methods such as PIRVS [30], S-MSCKF [31], ICE-BA [33], Trifo-VIO [34], VINS-Fusion [35] and [22]–[27], [29], [32], fuse the state of the camera and IMU into one model to optimally exploit the visual and inertial measurements, thus achieve higher precision at the cost of increased computation. Furthermore, for these tightly-coupled approaches, two methodologies have been widespread: filtering-based (e.g., PIRVS [30], S-MSCKF [31], Trifo-VIO [34], and [22], [23], [25]) and optimization-based (e.g., ICE-BA [33], VINS-Fusion [35], and [24], [26], [27], [29], [32]).

In the early stage of the VIO/VISLAM, filtering-based methods dominate the main research focus, which operates on the mean and covariance of the probabilistic distribution based on the Kalman Filter (KF) framework [38]. According to the measurement information processing method, two modern filtering-based solutions to the VIO/VISLAM problem are prevalent [25]: Extended Kalman Filter (EKF)-based [23], [30] and Sliding Window Filter (SWF)-based methods [20], [22], [31], [34]. The state vector of EKF-based algorithms contain both the pose of the body and a set of feature positions, so on condition that these features are continuously observed and included in the state vector, the estimated pose relative to these features will not drift [38]. However, EKF-based methods are inconsistent, i.e., the state uncertainties are underestimated since the Jacobians in the linearized

model of a VIO/VISLAM system have different observability properties than the actual nonlinear system [25]. Besides, the filter state includes only the most recent pose, a given update step can never modify past poses even if later feature measurements could constrain them, resulting in sub-optimal estimations of both motion and feature positions. Comparably, the SWF-based methods maintain and update a sliding window of past camera poses in the state vector, and use the feature measurements to impose probabilistic constraints on these poses, thus keep computational complexity only linear in the number of features by excluding feature positions from the filter state vector [38], instead of cubic like EKF-based approaches [34]. The S-MSCKF [31] can be considered as a hybrid of EKF-based and SWF-based method in the sense that it maintains a variable window of poses and applies batch updates using all observations of each landmark. Thus, the backbone of our frontend estimator utilizes the S-MSCKF framework due to its computational efficiency and accurate position tracking.

With the development of computer technology and the improvement of computer processing speed, optimization-based VIO/VISLAM has been widely developed. Optimization-based methods usually divide the whole SLAM framework into a frontend and backend. The frontend is used for pose estimation and map construction, whereas the backend is responsible for loop closure detection and pose optimization. The comparison in [10], [37] shows that optimization-based methods have better accuracy than filtering-based methods due to its capability to relinearize the state at each iteration, which avoids integrated error from linearization. The VISLAM system in [24] uses a keyframe-based and nonlinear optimization-based VIO estimator [13] to estimate the state of a Micro Aerial Vehicle (MAV). Since the VIO suffers from slow drift over time, a local map based on the output of the VIO is built in the background and then the drift is corrected through relocalization against this local map. Besides, Bundle Adjustment (BA) is used to further reduce errors and drift in the odometry. The VISLAM system in [29] also builds on the keyframe-based VIO method [13], but it uses image retrieval techniques to detect the revisited locations and then performs loop closures to reduce the drift. Besides, it has a simple but effective multi-frame verification method for relocalization. However, the former two systems need to repeatedly compute the IMU integration when the linearization point changes. The proposed framework in [26] uses two windows of constraints (i.e., a spatial and a temporal window) in the frontend to refine the states by a nonlinear optimization, and routinely maintains the map in the backend once a new keyframe is inserted as well as detect loop. However, the circular tracking of point features used in the frontend often results in dense tracks. The VIO system in [27] estimates the camera pose, velocity and IMU biases simultaneously by minimizing a combined photometric and inertial energy function [4]. Different from the point features-based VIO frontend, it is fully direct, i.e., the geometry is estimated in the form of semi-dense depth maps instead of sparse keypoints. Although the

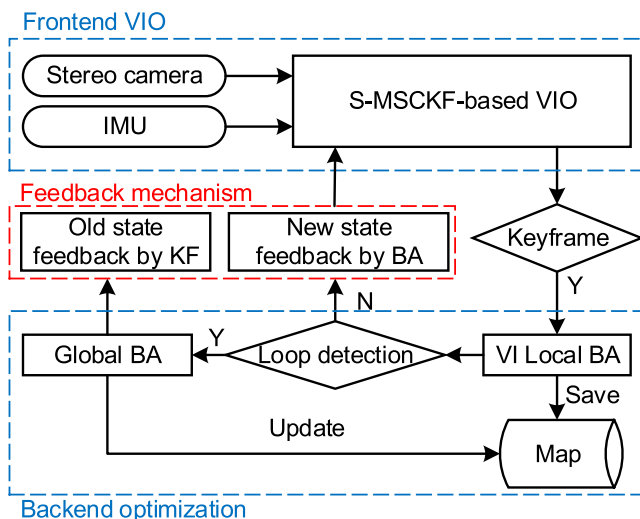


FIGURE 1. The proposed VISLAM system with feedback mechanism.

information utilization is high, the processing speed becomes slower. The VISLAM method in [32] employs the Lucas-Kanade optical flow algorithm to build the VO in the frontend and uses nonlinear optimization in the backend to estimate the state of camera and IMU. Although the accuracy and robustness are excellent even on some textureless scenes, it is unsuitable for tracking the motion in fast-moving scenes. The ICE-BA [33] for VISLAM optimization generalizes the BA to jointly optimize the visual and inertial measurements. It re-uses the intermediate results of previous optimization to avoid redundant new computation, thus increases the solver speed remarkably and can be applied to most sliding window-based VISLAM systems. Recently, a general optimization-based VISLAM framework [35] was proposed that extends the previous work [16] to adapt to multiple sensors, in which, the state of the system and a representation of the environment are estimated by local BA in one thread, and loops are closed in lightweight manner in parallel thread.

Both filtering-based and optimization-based methods have their merits, thus we tightly fuse both methods via a feedback mechanism to achieve a better accuracy, robustness and efficiency.

III. THE PROPOSED VISLAM SYSTEM BASED ON FEEDBACK MECHANISM

The proposed tightly-coupled stereo VISLAM system, as shown in Fig. 1, includes three modules: (1) the filtering-based frontend VIO, (2) the optimization-based backend with loop closure, (3) the state feedback. The frontend VIO is based on the S-MSCKF (Stereo Multiple State Constraint KF) VIO [31], fusing the visual and IMU measurements in a tightly-coupled way to efficiently estimate the 3D pose, velocity, bias. The backend uses the keyframe-based VISLAM [35] to optimize the state in a sliding window, and following the approach in ORB-SLAM2 [3], performs global BA to reduce the drift when the loop closure was detected. To make the frontend estimation have a higher precision,

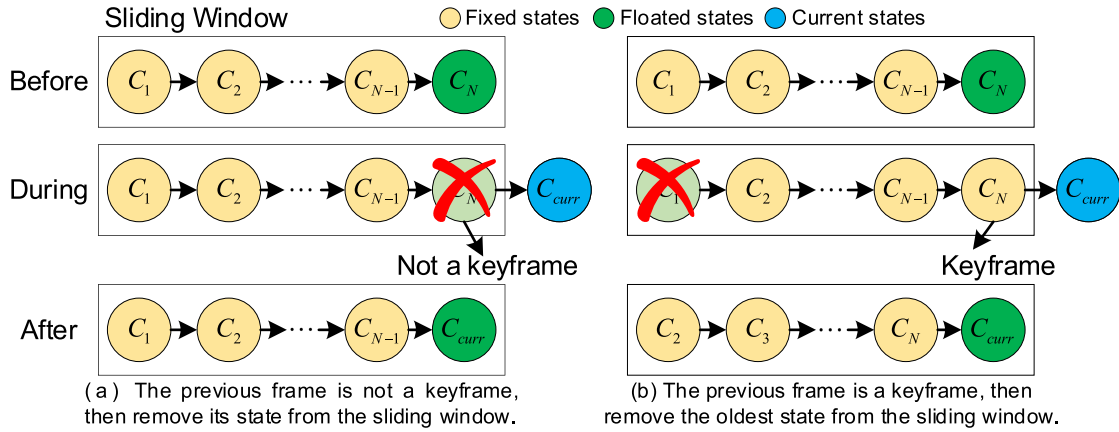


FIGURE 2. The marginalization mechanism.

we feed the state after the local or global BA optimization in the backend back to the frontend. Moreover, the state update by the feedback also provide a more accurate initial state for the backend, resulting in a faster optimization. In the following, we describe these modules in detail.

A. FRONTEND VIO

The backbone of the frontend VIO is S-MSCKF [31] whose key idea is to maintain and update a sliding window of camera poses using feature track observations without including features in the state vector. Instead, 3D feature positions are estimated through multi-view triangulation and subsequently marginalized, which reduces the computational cost considerably and make its complexity linear in the number of features.

Different from the update mechanism of S-MSCKF [31], we use all the map points that are observed by the camera frames (at least two camera frames) in the sliding window to update the EKF state. This is because in textureless scenarios (e.g., corridor), the number of the detected map points is inherently less, the observation can be more accurate by using the constraints of all the map points. Whereas, in the scenarios with rich texture, we limit the number of the map points to 800 for saving computing resources. The map point observed by more frames is more likely to be selected for EKF update since it provides more constraints between frames.

Additionally, to make the sliding window contain more long-term constraints, the keyframe selection strategy in [3] is used to marginalize the camera state (one camera state at each time step). When the number of camera state in the sliding window reaches the preset threshold N and a new image is captured, we will determine the previous frame whether is a keyframe. If the previous frame is a keyframe, then the oldest state is removed; otherwise, the previous state is removed. Therefore, we can ensure the previous $N - 1$ frames in the sliding window are keyframes. Note that the latest frame is always kept in the sliding window since it has the new measurements information. Fig. 2 shows the working scenarios of the above marginalization approach.

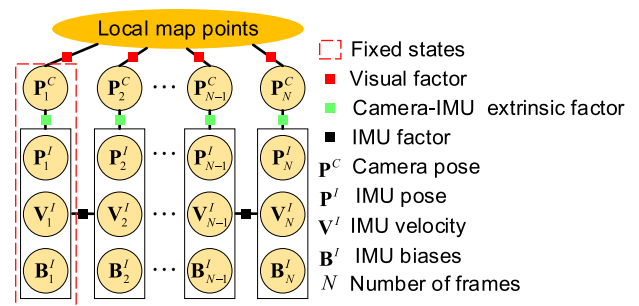


FIGURE 3. VI local BA. The oldest frame is fixed during optimization to serve as a prior information.

B. BACKEND OPTIMIZATION

When a new keyframe is detected in the frontend and the backend is idle, the states in the sliding window (including $N - 1$ keyframes and the current frame which may not be a keyframe) will be transferred to the backend to perform the nonlinear optimization. In this section, an unified formulation of the visual and inertial measurements [16] is used to jointly optimize the full state in the sliding window.

To accelerate the optimization, we remove the prior term from the object function of VINS-Mono [16], and define the visual-inertial BA as:

$$\chi^* = \underset{\chi}{argmin} \left\{ \sum_{(i,j) \in K} \|r_{Iij}\|_{\Sigma_{ij}}^2 + \sum_{i \in K, l \in C_i} \|r_{Cil}\|_{\Sigma_{Cil}}^2 \right\} \quad (1)$$

where r_{Iij} and r_{Cil} are the IMU and visual measurement residual respectively, Σ_{ij} and Σ_{Cil} are the corresponding covariance matrices, K is the set of all keyframes, and C_i is the camera measurements at keyframe i . As the camera can observe multiple landmarks, we let $l \in C_i$ represent a landmark l is seen at keyframe i . Detailed definitions of the residuals are the same with the definitions in [16], [40].

The local BA optimizes the last N frames in the sliding window and all map points seen by those N frames. However, as in Fig. 3, the states of the oldest keyframe serving as the

prior information, are fixed during optimization to obtain a consistent and smooth result with the global map since the prior information was removed in (1).

C. FEEDBACK MECHANISM

The EKF-based frontend VIO can estimate the frame state efficiently. However since the accumulation of the linearization errors and the absence of the loop closure, the error of the state estimate will accumulate as time goes on. If the state provided in the frontend drifts too much, the local BA is prone to fall into the local minimum as the function of visual measurement is non-convex. Thus we introduce the feedback mechanism to constrain the error of the state estimated from the frontend VIO, and conversely accelerate the optimization in the backend.

When a keyframe is detected in the frontend, its corresponding states will be transmitted to the backend to perform the nonlinear optimization. After the optimization, the optimized results will be fed back to the frontend to update the state. However, the current state of the frontend is different from the optimized state since the frontend and backend are performed in different threads. As shown in Fig. 4, the current state of the frontend can be classified into two parts: (1) the old state that has been transferred into the backend, and (2) the new state. We use two methods to update the state: (1) The common KF update rule for the old state, and (2) the nonlinear optimization method for the new state. The update methods are detailed in the following section.

1) OLD STATE UPDATE

As shown in Section III-B, the state in the sliding window for the backend optimization includes N poses in the IMU frame. Thus we first transform the optimized results, i.e. poses $\hat{P}_{I_k} = ({}^I_G \hat{q}_k, {}^I_G \hat{p}_{I_k})$, $k = 1, \dots, N$, from the IMU frame to camera frame, as

$$\begin{aligned} {}^C_G \hat{q}_k &= {}^C_I q \otimes {}^I_G \hat{q}_k \\ {}^C_G \hat{p}_{C_k} &= C({}^I_G \hat{q}_k)^T {}^I p_C + {}^G \hat{p}_{I_k} \end{aligned} \quad (2)$$

where $\hat{P}_{C_k} = ({}^C_G \hat{q}_k, {}^C_G \hat{p}_{C_k})$ is the optimized pose in the camera frame, $({}^C_I q, {}^I p_C)$ is the extrinsic parameters between the IMU frame and the camera frame, \otimes is the multiplication of quaternions, and $C(\cdot)$ is the function of converting quaternion to the corresponding rotation matrix. Then the optimized camera state \hat{P}_{C_k} and its covariance matrix Σ_{C_k} are used to update the old camera state P_{C_k} and its covariance matrix P_Σ estimated from the frontend VIO according to the KF update rule as:

$$\begin{aligned} H_k &= I \\ r_k &= \hat{P}_{C_k} - P_{C_k} \\ K_k &= P_\Sigma H_k^T (H_k P_\Sigma H_k^T + \Sigma_{C_k})^{-1} \\ P_{C_k}^* &= P_{C_k} + K_k r_k \\ P_\Sigma^* &= (I - K_k H_k) P_\Sigma (I - K_k H_k)^T + K_k \Sigma_{C_k} K_k^T \end{aligned} \quad (3)$$

where $P_{C_k}^*$ and P_Σ^* are the updated camera state and its covariance.

2) NEW STATE UPDATE

Generally, when the backend nonlinear optimization finished, the frontend has received new frames. Based on the marginalization mechanism (see Section III-A), the new frames contain one or more keyframes (mostly one keyframe in our real test) and the current frame (maybe or maybe not a keyframe). For these new frames, we use the nonlinear optimization to improve their estimations. As shown in Fig. 4, when the backend optimization is performed, the state estimation of the last keyframe P_N^C is accurate enough. Thus, to save the computation resources, we only use the last keyframe and the new frames for optimization through minimizing the objective function which is the same with (1). Note that the last keyframe as the prior information is fixed during the optimization, and the local map points are firstly updated according to the optimized state obtained from the backend, then they will be fixed to only optimize the new camera states.

IV. RELOCALIZATION AND CONTINUED SLAM FRAMEWORK WITH FEEDBACK MECHANISM

In many practical applications such as autonomous navigation of mobile robot or augmented reality, relocalization in a previously built map or seamlessly continued SLAM in new parts of the previous environment is a desirable property for the SLAM system. Thus with the proposed feedback mechanism, we design a relocalization and continued SLAM framework which is shown in Fig. 5. The proposed framework contains frontend VIO, backend optimization and feedback mechanism, which are similar to the proposed VISLAM system. The processes of relocalization and continued SLAM are as follows.

A. RELOCALIZATION

The frontend VIO estimates and updates the state that includes the current IMU state and N camera poses as the Section III-A does. The backend optimization is similar to the one in Section III-B. The different is that the local map points were obtained from the previously built map instead of the frontend, and this local map points are fixed during the optimization. As is shown in Fig. 6, when current frame is a keyframe, we use the DBoW2 [41] to align current frame with the keyframe in the built map. Then N keyframes before and after the aligned keyframe are selected to update the local map points by retrieving the points corresponding to the keyframes in the built map. Besides, the map points of the previous local map, which are observed by the frames (at least two frames) in the current sliding window, are reserved. However, when current frame is not a keyframe, the local map points will not be updated and directly used to optimize the camera and IMU state. This is because the local map points already include the subsequent map points which are sufficient for optimization. The state vector in the backend is defined as:

$$\chi = \left\{ x_{I_1}, \dots, x_{I_N}, \underbrace{\lambda_1, \dots, \lambda_m}_{fixed} \right\} \quad (4)$$

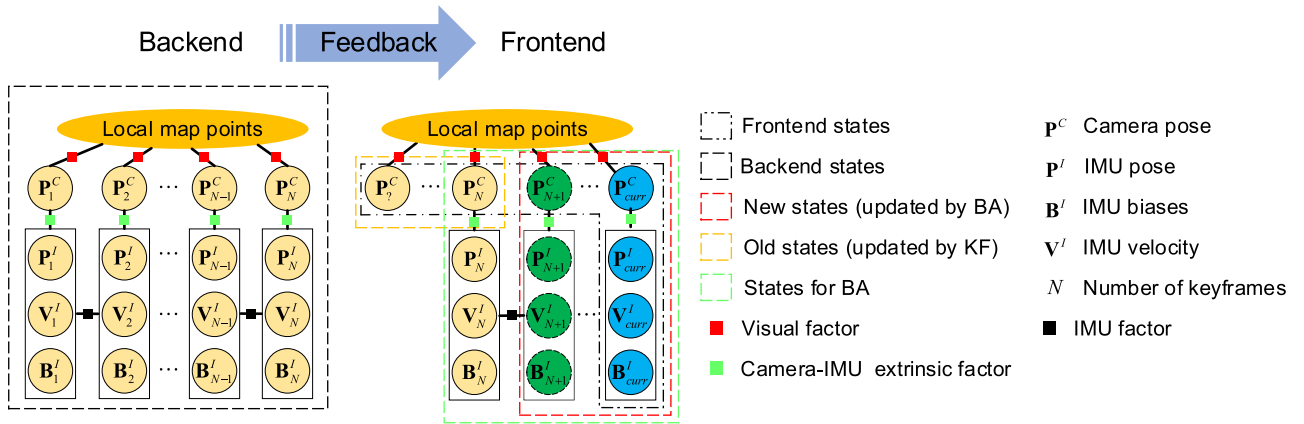


FIGURE 4. State feedback to frontend.

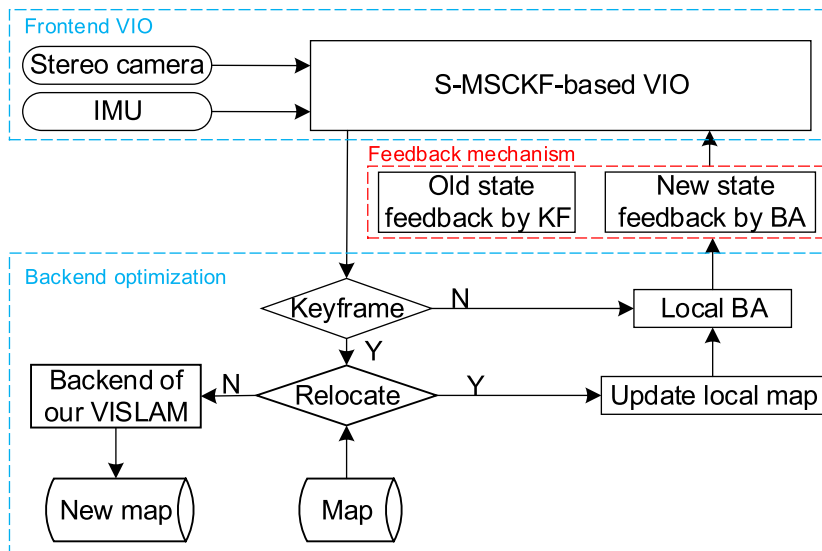


FIGURE 5. Relocalization and continued SLAM framework based on the state feedback mechanism.

where x_{I_k} , including pose ${}^G p_{I_k}^T$, velocity ${}^G v_{I_k}^T$ and biases of acceleration b_a^T and gyroscope b_g^T , is the IMU state at the time that the k^{th} image is captured, N is the total number of keyframes, m is the total number of map points in the sliding window, and λ_k is the inverse depth of the k^{th} map point, and is fixed during the optimization since the previous built map has been optimized. Thus, only the IMU state needs to be optimized.

After the backend optimization, the optimized state will be fed back to frontend VIO to correct the state in the frontend according to the feedback mechanism in Section III-C.

B. CONTINUED SLAM

When it lasts for a long time that the current frame is unable to be aligned with the frames of the previous built map, or when the SLAM is interrupted due to the low battery power and is needed to go on, the continued SLAM module will work.

As shown in Fig. 5, when the relocalization failed for a long time, the backend of our proposed VISLAM system will start and create a new map. If the relocalization succeeded during the new SLAM, the new map can be merged to the previous built map by transforming all key frame poses in the new map to the previous map frame. After the merging, the loop closure technique can create additional constraints between the map parts which further improves the consistency of the trajectories.

V. EXPERIMENTS

We evaluate the proposed VISLAM system both on the EuRoC dataset [39] and real-world scenarios. We first compare the proposed system qualitatively and quantitatively with other state-of-the-art systems on the public EuRoC dataset to show the accuracy and efficiency. Then we evaluate the relocalization module on the EuRoC dataset. Finally, the performance of our algorithm is validated again in both indoor

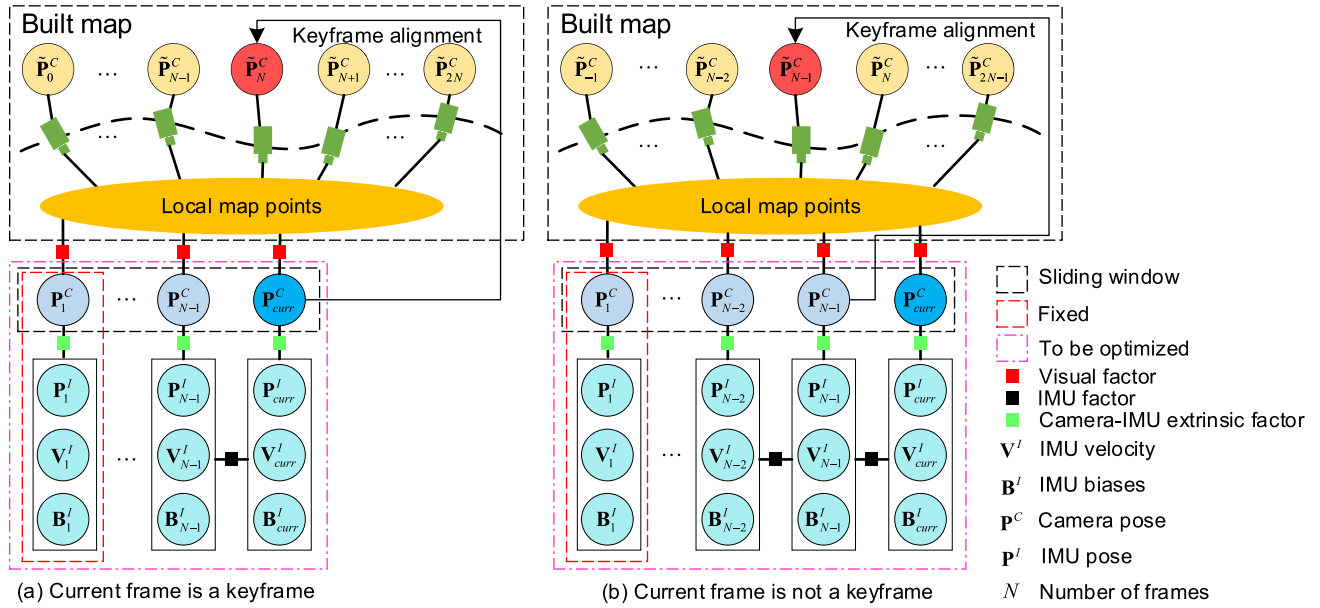


FIGURE 6. Backend optimization of the relocalization and continued SLAM framework.

and outdoor real-world environment using the sensor of Intel RealSense depth camera D435i.

A. EVALUATION ON EUROC DATASET

The EuRoC dataset has 11 sequences, which was recorded by a MAV in three different indoor environments. According to the illumination, texture and motion dynamics, the sequences are classified as easy, medium and difficult levels. The dataset provides stereo images at 20Hz, IMU measurements at 200Hz and ground truth at 200Hz. We successfully perform our algorithm on all sequences of EuRoC dataset in real-time.

In this experiments, we compare the proposed algorithm with S-MSCKF [31], a state-of-the-art filtering-based VIO, and VINS-Fusion [35], a state-of-the-art optimization-based VISLAM that can work with three different combinations of sensors (only the combination of stereo cameras and IMU is used for comparison in this paper). We use the SLAM trajectory evaluation tool [42] to calculate the Absolute Trajectory Errors (ATE). Fig. 7 qualitatively shows the comparison between the ground truth and the trajectories estimated by our and other start-of-the-art algorithms on MH_01_easy, MH_03_medium and MH_05_difficult sequence. For quantitative analysis, the Root Mean Square Errors (RMSE) of ATE for all sequences in EuRoC dataset are shown in Table 2.

From Table 2, we can see that our method outperforms VINS-Fusion and S-MSCKF in most of the sequences. The S-MSCKF method performs worst on average. This is because it has no loop closure and optimization to correct the estimated error caused by linearization. The results obtained from the proposed method without feedback are similar to that achieved by the VINS-Fusion method since they have the similar backend optimization to reduce the estimated

error. However, our method with feedback reduced the error of 50% compared with the VINS-Fusion. This is because the feedback mechanism makes the frontend VIO estimate more accurately, and the accurate result obtained from frontend, in return, makes the backend optimization obtain a more accurate result.

To evaluate the efficiency of the proposed algorithm, the average time of processing one frame, the CPU load and the memory utilization of our algorithm, S-MSCKF and VINS-Fusion are measured with the laptop equipped with Intel Core i7-6600U CPU @ 2.6 GHz \times 4 and an 8GB RAM. As shown in Table 3, the S-MSCKF has a highest efficiency since it is only based on EKF algorithm, but it achieves the worst accuracy as shown in Table 2. Compared with VINS-Fusion, the proposed algorithm performs more efficiently. This is because our method performs the local window optimization only when a keyframe, instead of every frame (VINS-Fusion), is detected, which saves a lot of computation resources. Besides, our feedback mechanism makes the optimization converge more rapidly as the frontend provides a more accurate initial state. In addition, we remove the marginalization term in our objective function (see (1)) to make local optimization faster while ensuring the accuracy. Thus our system maintains higher efficiency with sufficient accuracy.

B. RELOCALIZATION TEST

To evaluate the performance of our relocalization and continued SLAM framework, we still use the EuRoC dataset since some sequences are recorded in the same scene. In this experiments, the proposed method and two state-of-the-art VSLAM systems, i.e., Maplab [17] and ORBSLAM2 [3], creat the global map in the MH_01 and V1_01 sequences,

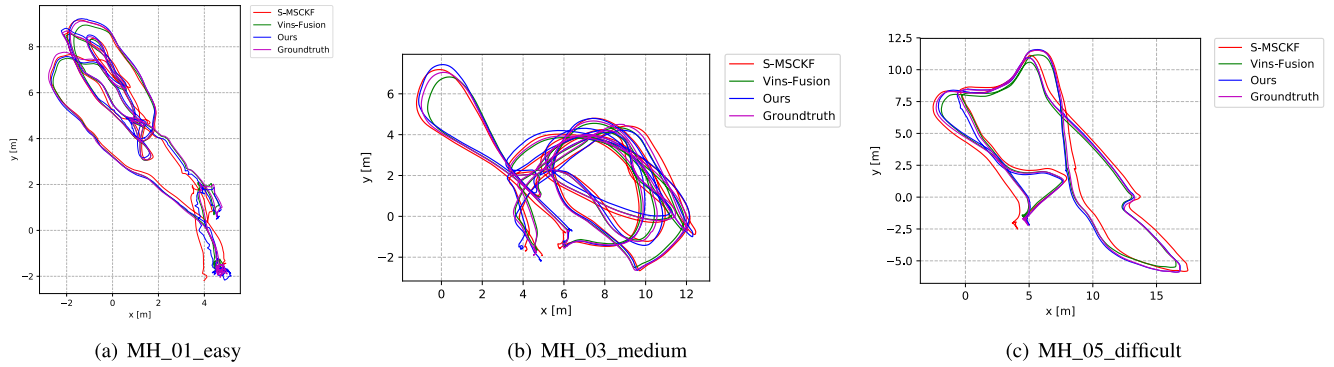


FIGURE 7. Comparison between the ground truth and the trajectories estimated by our algorithm, S-MSCKF and VINS-Fusion method, which is viewed from the gravity direction.

TABLE 2. RMSE of the trajectories estimated by our algorithm, S-MSCKF and VINS-Fusion on EuRoC dataset. The best results are given in bold.

Sequence	Length (m)	S-MSCKF (m)	VINS-Fusion (m)	Ours without feedback (m)	Ours with feedback (m)
MH_01_easy	80.6	0.23	0.24	0.15	0.10
MH_02_easy	73.5	0.23	0.18	0.21	0.11
MH_03_medium	130.9	0.20	0.23	0.18	0.08
MH_04_difficult	91.7	0.35	0.39	0.30	0.14
MH_05_difficult	97.6	0.21	0.19	0.17	0.06
V1_01_easy	58.6	0.06	0.10	0.15	0.10
V1_02_medium	75.9	0.16	0.10	0.14	0.06
V1_03_difficult	79.0	0.28	0.11	0.18	0.12
V2_01_easy	36.5	0.07	0.12	0.13	0.08
V2_02_medium	83.2	0.15	0.10	0.12	0.06
V2_03_difficult	86.1	0.37	0.27	0.16	0.11
Average	81.2	0.21	0.18	0.17	0.09

TABLE 3. Efficiency comparison of our algorithm, S-MSCKF and VINS-Fusion on EuRoC dataset.

Sequence	Frames	Average processing time (ms)			CPU load (%)			Memory utilization (MB)		
		S-MSCKF	VINS-Fusion	Ours	S-MSCKF	VINS-Fusion	Ours	S-MSCKF	VINS-Fusion	Ours
MH_01_easy	3682	25.03	74.43	53.50	19.02	50.42	31.99	65.24	1166.86	525.34
MH_02_easy	3040	24.42	68.95	34.82	19.48	54.20	42.14	66.05	868.65	525.71
MH_03_medium	2700	26.45	75.23	49.49	23.38	52.19	32.72	65.04	967.68	525.97
MH_04_difficult	2033	25.77	65.05	63.02	20.40	50.07	35.91	65.59	755.42	525.02
MH_05_difficult	2273	27.26	66.56	60.89	23.56	48.45	34.79	66.42	819.23	525.33
V1_01_easy	2912	26.32	71.35	51.36	19.20	49.08	31.26	66.51	917.88	525.68
V1_02_medium	1710	24.50	56.40	39.91	18.96	44.76	37.32	66.15	678.60	525.77
V1_03_difficult	2149	31.54	36.28	40.90	20.36	38.57	27.78	65.79	702.81	525.29
V2_01_easy	2280	35.10	67.65	42.90	21.60	47.55	28.38	65.25	731.73	526.33
V2_02_medium	2348	34.30	54.70	36.89	20.98	45.91	25.54	65.01	762.51	524.66
V2_03_difficult	1922	32.60	30.93	33.17	17.72	32.45	24.98	64.15	691.94	513.05
Average	2459	28.48	60.68	46.08	20.42	46.70	32.07	65.56	823.94	524.38



FIGURE 8. The estimated trajectory in indoor environment.

and then perform their corresponding relocalization module on MH_02 and V2_02 sequences respectively. Table 4 shows the translation and orientation RMSE of relocalization. The

experimental results demonstrate that the proposed relocalization framework with feedback mechanism outperforms the other two methods on both translation and orientation accuracy. Besides, compared with the relocalization framework without feedback mechanism, the relocalization framework with feedback reduced the error of translation 27.66%, orientation 40.83% for MH sequence and translation 23.19%, orientation 16.33% for V sequence. This also verified that the feedback mechanism can improve the state estimation accuracy.

C. REAL-WORLD EVALUATION

We perform the experiments in both indoor (an office) and outdoor (a park) environment. In indoor environment, we hold the sensor device by hand and walk in normal pace, start and end at the same location after two circles. As shown

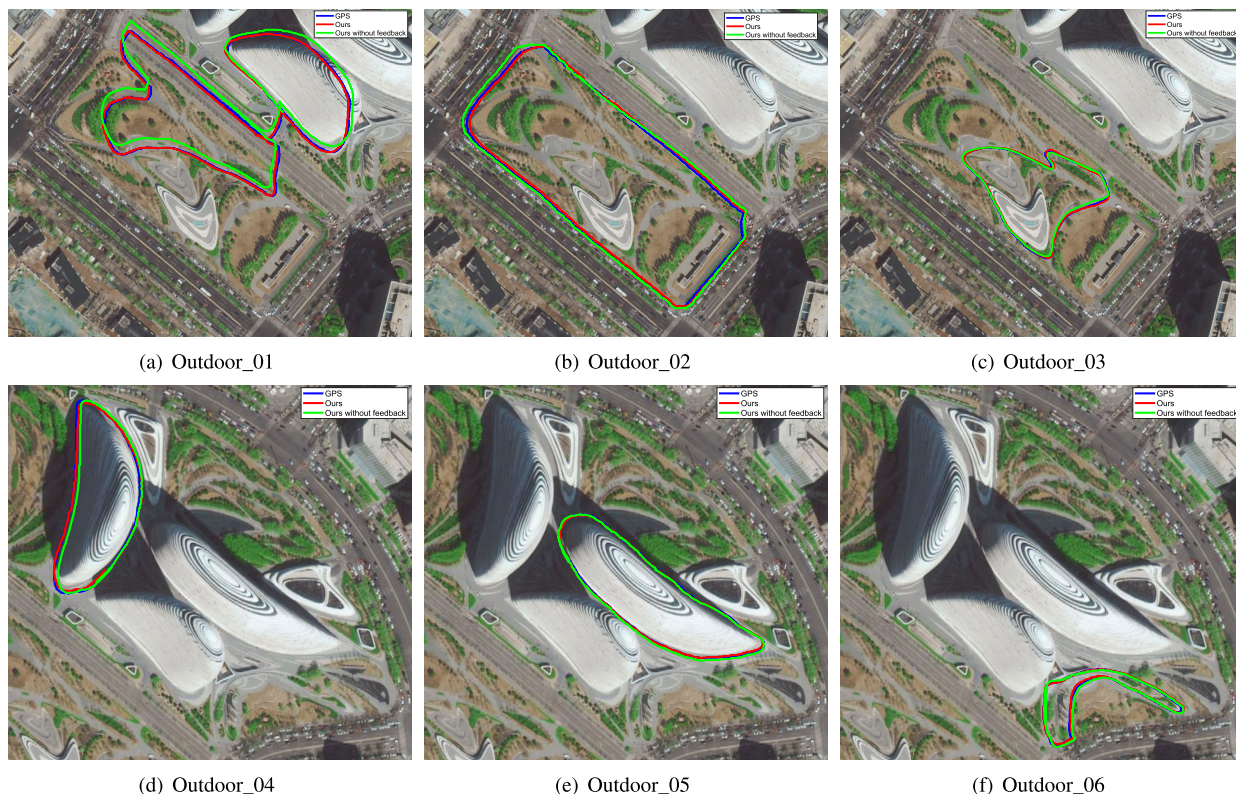


FIGURE 9. The estimated trajectories in outdoor environment.

TABLE 4. Translation and orientation RMSE of relocalization module on EuRoc dataset.

Method	Map: MH_01, ReLoc: MH_02		Map: V1_01, ReLoc: V2_02	
	Translation(m)	Orientation(°)	Translation(m)	Orientation(°)
Maplab	0.082	1.43	0.057	1.57
ORB_SLAM2	0.084	0.78	0.121	1.14
Ours without feedback	0.094	1.2	0.069	1.47
Ours with feedback	0.068	0.71	0.053	1.23

in Fig. 8, the estimated trajectory has no noticeable drifts when we circle in the office. The end-to-end error is 0.26 m with respect to the total length of $121.11\text{ m} \times 2$, it is just the 0.11% of the total trajectory length.

In outdoor environment, we use another GPS device with positioning accuracy $\leq 5\text{ cm}$ to obtain the ground truth. We also hold the sensor device and walk in normal pace around a park and an office building. We perform the state estimation using our algorithm with and without the feedback mechanism and compare the results with GPS. For quantitative analysis, we walked three paths around the park and the office building respectively. The estimated trajectories are shown in Fig. 9 and the translation RMSE are show in Table 5. As same as the dataset experiment, the method with feedback achieves better results than the method without feedback on all paths.

VI. CONCLUSION

In this paper, we presented a feedback mechanism for tightly-coupled stereo VISLAM, which makes the filtering-based frontend and optimization-based backend in the

TABLE 5. Translation RMSE of outdoor test.

Sequence	Length (m)	Ours without feedback(m)	Ours with feedback(m)
Outdoor_01	~1102	3.21	0.84
Outdoor_02	~733	1.83	0.48
Outdoor_03	~405	1.01	0.44
Outdoor_04	~433	1.29	0.42
Outdoor_05	~492	0.88	0.38
Outdoor_06	~369	1.03	0.32

VIO/VISLAM system be able to promote each other, and thus improves the estimation accuracy of the frontend VIO and the convergence speed of backend optimization. Moreover, a relocalization and continued SLAM framework with the feedback mechanism was introduced to make our system can be applied in real robot navigation.

Point feature-based VISLAM is prone to fail in textureless scenes or motion blurred images. In future work, we will consider the line feature or use the deep learning technique to extract more robust feature for better accuracy and robustness. We also aim to extend our framework with GPS to achieve locally accurate and globally aware pose estimation for outdoor application.

ACKNOWLEDGMENT

The authors would like to thank the CloudMinds Technologies Inc. for providing the sensors, and thank Chaopeng Wang and Shangying Liang to help us evaluate our system. They would also like to thank the anonymous reviewers for the insightful comments and valuable suggestions.

REFERENCES

- [1] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Hong Kong, May/June 2014, pp. 15–22.
- [2] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, Mar. 2018.
- [3] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source slam system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.
- [4] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, Sep. 2014, pp. 834–849.
- [5] K. Yousif, A. Bab-Hadiashar, and R. Hoseinnezhad, "An overview to visual odometry and visual SLAM: Applications to mobile robotics," *Intell. Ind. Syst.*, vol. 1, no. 4, pp. 289–311, Dec. 2015.
- [6] L. Han, Y. Lin, G. Du, and S. Lian, "DeepVIO: Self-supervised deep learning of monocular visual inertial odometry using 3d geometric constraints," Jun. 2019, *arXiv:1906.11435*. [Online]. Available: <https://arxiv.org/abs/1906.11435>
- [7] Y. Lin, Z. Liu, J. Huang, C. Wang, G. Du, J. Bai, S. Lian, and B. Huang, "Deep global-relative networks for end-to-end 6-DoF visual localization and odometry," Dec. 2018, *arXiv:1812.07869*. [Online]. Available: <https://arxiv.org/abs/1812.07869>
- [8] M. R. U. Saputra, A. Markham, and N. Trigoni, "Visual SLAM and structure from motion in dynamic environments: A survey," *ACM Comput. Surv.*, vol. 51, no. 2, pp. 1–36, Jun. 2018.
- [9] G. Younes, D. Asmar, E. Shamma, and J. Zelek, "Keyframe-based monocular SLAM: Design, survey, and future directions," *Robot. Auton. Syst.*, vol. 98, pp. 67–88, Dec. 2017.
- [10] C. Chen, H. Zhu, M. Li, and S. You, "A review of visual-inertial simultaneous localization and mapping from filtering-based and optimization-based perspectives," *Robotics*, vol. 7, no. 3, p. 45, Aug. 2018.
- [11] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Roma, Italy, Apr. 2007, pp. 3565–3572.
- [12] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct EKF-based approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Hamburg, Germany, Sep. 2015, pp. 298–304.
- [13] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual-inertial odometry using nonlinear optimization," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 314–334, 2015.
- [14] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial SLAM using nonlinear optimization," in *Robotics: Science and Systems*. Berlin, Germany: MIT Press, 2013.
- [15] R. Mur-Artal and J. D. Tardós, "Visual-inertial monocular SLAM with map reuse," *IEEE Robot. Automat. Lett.*, vol. 2, no. 2, pp. 796–803, Apr. 2017.
- [16] T. Qin, P. Li, and S. Shen, "VINS-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, Aug. 2018.
- [17] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, and R. Siegwart, "Maplab: An open framework for research in visual-inertial mapping and localization," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1418–1425, Jul. 2018.
- [18] Y. He, J. Zhao, Y. Guo, W. He, and K. Yuan, "Pl-vio: Tightly-coupled monocular visual-inertial odometry using point and line features," *Sensors*, vol. 18, no. 4, p. 1159, Apr. 2018.
- [19] K. Wu, A. Ahmed, G. A. Georgiou, and S. I. Roumeliotis, "A square root inverse filter for efficient vision-aided inertial navigation on mobile devices," in *Robotics: Science and Systems*. Rome, Italy: MIT Press, 2015.
- [20] M. K. Paul and S. I. Roumeliotis, "Alternating-stereo VINS: Observability analysis and performance evaluation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 4729–4737.
- [21] J.-P. Tardif, M. George, M. Laverne, A. Kelly, and A. Stentz, "A new approach to vision-aided inertial navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Taipei, Taiwan, Oct. 2010, pp. 4161–4168.
- [22] T. Lupton and S. Sukkariieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 61–76, Feb. 2012.
- [23] E. Asadi and C. L. Bottasso, "Tightly-coupled stereo vision-aided inertial navigation using feature-based motion sensors," *Adv. Robot.*, vol. 28, no. 11, pp. 717–729, Jan. 2014.
- [24] M. Burri, H. Oleynikova, M. W. Achtelik, and R. Siegwart, "Real-time visual-inertial mapping, re-localization and planning onboard MAVs in unknown environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Hamburg, Germany, Sep. 2015, pp. 1872–1878.
- [25] L. E. Clement, V. Peretroukhin, J. Lambert, and J. Kelly, "The battle for filter supremacy: A comparative study of the multi-state constraint Kalman filter and the sliding window filter," in *Proc. 12th Conf. Comput. Robot. Vis.*, Halifax, NS, Canada, Jun. 2015, pp. 23–30.
- [26] J. Huai, C. K. Toth, and D. A. Grejner-Brzezinska, "Stereo-inertial odometry using nonlinear optimization," in *Proc. Int. Technic. Meeting Satell. Division Inst. Navigat.*, Tampa, FL, USA, Sep. 2015, pp. 2087–2097.
- [27] V. Usenko, J. Engel, J. Stückler, and D. Cremers, "Direct visual-inertial odometry with stereo cameras," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Stockholm, Sweden, May 2016, pp. 1885–1892.
- [28] N. de Palézieux, T. Nägeli, and O. Hilliges, "Duo-VIO: Fast, light-weight, stereo inertial odometry," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Daejeon, Korea, Oct. 2016, pp. 2237–2242.
- [29] A. Kasyanov, F. Engelmann, J. Stückler, and B. Leibe, "Keyframe-based visual-inertial online SLAM with relocalization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Vancouver, BC, Canada, Sep. 2017, pp. 6662–6669.
- [30] Z. Zhang, S. Liu, G. Tsai, H. Hu, C.-C. Chu, and F. Zheng, "Pirvs: An advanced visual-inertial SLAM system with flexible sensor fusion and hardware co-design," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 1–7.
- [31] 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 Robot. Autom. Lett.*, vol. 3, no. 2, pp. 965–972, Apr. 2018.
- [32] C. Chen and H. Zhu, "Visual-inertial SLAM method based on optical flow in a GPS-denied environment," *Ind. Robot*, vol. 45, no. 3, pp. 401–406, May 2018.
- [33] H. Liu, M. Chen, G. Zhang, H. Bao, and Y. Bao, "ICE-BA: Incremental, consistent and efficient bundle adjustment for visual-inertial SLAM," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake, UT, USA, Jun. 2018, pp. 1974–1982.
- [34] F. Zheng, G. Tsai, Z. Zhang, S. Liu, C.-C. Chu, and H. Hu, "Trifo-VIO: Robust and efficient stereo visual inertial odometry using points and lines," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, Oct. 2018, pp. 3686–3693.
- [35] T. Qin, J. Pan, S. Cao, and S. Shen, "A general optimization-based framework for local odometry estimation with multiple sensors," Jan. 2019, *arXiv:1901.03638*. [Online]. Available: <https://arxiv.org/abs/1901.03638>
- [36] M. K. Paul, K. Wu, J. A. Hesch, E. D. Nerurkar, and S. I. Roumeliotis, "A comparative analysis of tightly-coupled monocular, binocular, and stereo vins," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Singapore, May/June 2017, pp. 165–172.
- [37] J. Delmerico and D. Scaramuzza, "A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Brisbane, QLD, Australia, May 2018, pp. 2502–2509.
- [38] M. Quan, S. Piao, M. Tan, and S.-S. Huang, "Accurate monocular visual-inertial SLAM using a map-assisted EKF approach," *IEEE Access*, vol. 7, pp. 34289–34300, 2019.
- [39] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.
- [40] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "IMU preintegration on manifold for efficient visual-inertial maximum-a-posteriori estimation," in *Robotics: Science and Systems*. Rome, Italy: MIT Press, 2015.

- [41] D. Galvez-López and J. D. Tardos, “Bags of binary words for fast place recognition in image sequences,” *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012.
- [42] Z. Zhang and D. Scaramuzza, “A tutorial on quantitative trajectory evaluation for visual(-inertial) odometry,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, Oct. 2018, pp. 7244–7251.



JINQIANG BAI received the B.E. and M.S. degrees in electronic information and engineering from the China University of Petroleum (East China), Shandong, China, in 2012 and 2015, respectively. He is currently pursuing the Ph.D. degree with the School of Electronic Information Engineering, Beihang University, Beijing, China.

His current research interests include computer vision, deep learning, and navigation for mobile robotics.



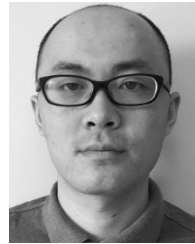
JUNQIANG GAO was born in Pingdingshan, China, in 1991. He received the B.S. degree in biomedical engineering and the M.S. degree in control engineering from the Southwest University of Science and Technology, Mianyang, China, in 2017.

Since 2017, he has been a Machine Vision Engineer with CloudMinds Technologies Inc., Beijing, China. His current research interests include VSLAM and sensor fusion.



YIMIN LIN received the B.S. degree from the Beijing Institute of Technology, in 2008, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, in 2014.

In 2014, he joined Huawei Technologies, Inc., as an Algorithm Engineer. Since 2016, he has been a Senior Engineer with CloudMinds Technologies Inc., Beijing, China. His research interests include the areas of computer vision, deep learning, autonomous navigation, and sensor fusion.



ZHAOXIANG LIU received the B.S. and Ph.D. degrees from the College of Information and Electrical Engineering, China Agricultural University, Beijing, China, in 2006 and 2011, respectively.

He joined VIA Technologies, Inc., Beijing, in 2011. From 2012 to 2016, he was a Senior Researcher with the Central Research Institute, Huawei Technologies, Beijing. He has been a Senior Engineer with CloudMinds Technologies Inc., Beijing, since 2016. His current research interests include computer vision, deep learning, robotics, and human-computer interaction.



SHIGUO LIAN (M'04) received the Ph.D. degree from the Nanjing University of Science and Technology, China.

He was a Research Assistant with the City University of Hong Kong, Hong Kong, in 2004. From 2005 to 2010, he was a Research Scientist with France Telecom Research and Development Beijing, Beijing, China. He was a Senior Research Scientist and the Technical Director of the Huawei Central Research Institute, Beijing, from 2010 to 2016. Since 2016, he has been a Senior Director of CloudMinds Technologies Inc., Beijing. He has authored over 80 refereed international journal articles covering topics of artificial intelligence, multimedia communication, and human-computer interface. He has authored or coedited over ten books. He holds over 50 patents.

Dr. Lian is on the editor board of several refereed international journals.



DIJUN LIU received the Ph.D. degree from the China University of Petroleum (East China), Shandong, China.

He was the Director of the China Institute of Communications. He has been the Chief Scientist with the China Academy of Telecommunication Technology, Beijing, China, and a Professor with Beihang University, Beijing. He has over 20 years of prized academic research, industrial development, and entrepreneurship (as the Chief Scientist, the Vice President, and the CEO) in semiconductor and communication. Since 2018, he has been the Vice President with Morningcore Technology Company Ltd., Beijing.

Dr. Liu was a recipient of the 2016 National Science and Technology Progress Special Award by the China State Council. He was the Chairman of the China Communications Integrated Circuit Committee.

...