A Tightly Coupled Feature-Based Visual-Inertial Odometry With Stereo Cameras

Lei Yu, Jiahu Qin, *Senior Member, IEEE*, Shuai Wang, Yaonan Wang, and Shi Wang

***Abstract*—Early works have shown that inertial measurement unit (IMU) can help visual odometry to achieve more accurate pose estimation. However, existing methods mainly focus on fusing visual and inertial information in the back end, while ignoring it in the front end. In this article, we present a novel feature-based visual-inertial odometry for stereo cameras, namely FSVIO, which makes full use of visual and inertial information in both the front and the back end. Specifically, we first introduce an IMU-aided feature-based method in the visual processing part of the front end, in which IMU information is used to build robust descriptors for image perspective deformation caused by the camera motion. This differs from the traditional feature-based methods that only use local image information in the descriptor construction. Then, in order to improve the efficiency of feature matching, we apply a fast-tracking method by predicting the position of feature points in the current frame with the help of combining stereo camera and IMU measurements, which also reduces outliers caused by dynamic environment or nonconvexity of the image. Furthermore, the 2D–2D epipolar geometry constraint and the improved Huber norm are introduced into the tightly coupled optimization of the back end, which reduces the influence of incorrect depth estimation from stereo cameras. Finally, our odometry is evaluated on both EuRoC datasets and real-world experiments. The experimental results verified the effectiveness and superiority of FSVIO.**

***Index Terms*—Inertial measurement unit (IMU)-aided visual front end, stereo visual-inertial odometry (VIO), simultaneous localization and mapping (SLAM).**

Manuscript received 14 August 2021; revised 7 December 2021 and 19 February 2022; accepted 27 April 2022. Date of publication 25 May 2022; date of current version 12 December 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61922076 and Grant 61873252 and in part by the Science and Technology Major Project of Anhui Province under Grant 202203a06020011. *(Corresponding author: Jiahu Qin.)*

Jiahu Qin is with the Department of Automation, University of Science and Technology of China, Hefei 230027, China, and also with the Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China (e-mail: jhqin@ustc.edu.cn).

Lei Yu and Shuai Wang are with the Department of Automation, University of Science and Technology of China, Hefei 230027, China (e-mail: yl010093@mail.ustc.edu.cn; wsustcid@mail.ustc.edu.cn).

Yaonan Wang and Shi Wang are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China, and also with the National Engineering Laboratory for Robot Visual Perception and Control Technology, Hunan University, Changsha 410082, China (e-mail: yaonan@hnu.edu.cn; shi_wang@hnu.edu.cn).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TIE.2022.3176304.

Digital Object Identifier 10.1109/TIE.2022.3176304

I. Introduction

MOTION estimation plays a crucial role in various applications, such as service robots, autonomous driving, and augmented reality (AR). Visual odometry (VO) [1]–[4] is a popular approach to estimate motion because of the low-cost and small size of the camera.

For VO, vision processing algorithm in the front end [5] is critical to ensure high-precision pose estimation. It can be roughly summarized into two categories: direct and feature-based methods. Direct methods are first proposed in the work of Irani *et al.* [6] for motion and shape estimation. Since then, direct methods have been widely used for odometry [2], simultaneous localization and mapping (SLAM) [7], and 3-D reconstruction [8]. Recently, a popular monocular odometry based on the direct method is DSO [9], which estimates camera poses by minimizing photometric errors. Based on DSO, there are a series of improved algorithms such as VI-DSO [10], DVIO [11], etc. In these methods, the intensity values of image pixels are directly used as measurements to establish the photometric residual model, which is sensitive to geometric noise and the changes in scene lighting. Feature-based methods take up extra computational complexity for feature extraction and matching to establish the geometric error model. Feature-based methods are proposed in the field of computer vision very early, such as MonoSLAM [12] and PTAM [13]. Recently, ORB-SLAM is presented by Mur-Artal *et al.* [3], [14]. It is a complete SLAM system for monocular, stereo, and RGB-D camera, which uses ORB features as visual measurements. Our work takes up the idea of multiscale FAST corners for scale invariant from ORB-SLAM. For the feature-based methods, there are two main feature matching categories: descriptor matching [15] and KLT tracking [16]. KLT tracking is similar to direct methods and it is also sensitive to illumination change. Unlike direct methods and KLT tracking, descriptor matching is robust to errors in the projection geometry and changes in illumination. More importantly, if the structure of the descriptor is concise, such as BRIEF [17] and Shi-Tomasi [18], the extra computational complexity can be ignored. Therefore, the feature-based methods using descriptor matching are more frequently applied in engineering deployment. However, there are two problems that hinder the application of the feature-based methods using descriptor matching in VO.

1) The first one is that the existing real-time empirical descriptors (e.g., ORB [15]) cannot accurately describe in-plane camera rotation [see Fig. 2(f)] and are sensitive

to out-of-plane camera rotation [see Fig. 2(d) and (e)], thereby resulting in fewer feature points that can be tracked when the camera rotates fastly.

2) The other issue is that the feature-based methods only abstract local information into the feature model for matching, which cannot overcome the influence of image nonconvexity or dynamic environment.

It is noteworthy that IMU information can provide pose prediction to reduce the impacts of these two problems. The method of using IMU measurement to assist VO has been proposed by many researchers, which is called visual-inertial odometry (VIO) [11], [19], [20]. Loosely coupled approaches [21] are simple but rough methods to implement visual inertial fusion. Tightly coupled methods (e.g., MSCKF [22], OKVIS [23], VI-DSO [10], VI-ORB [24], and VINS-FUSION [25], [26]) are more popular in VIO for achieving higher accuracy of pose estimation. Filter-based approaches [22] are early tightly coupled methods, where current camera states are predicted by IMU measurements and are recursively corrected by images. Another more accurate tightly coupled methods are optimization-based methods [23]–[25], which jointly optimize multiframe poses by minimizing visual and inertial residuals in the combined cost function. However, these existing VIO mainly focus on fusing visual and inertial information in the back end [5]. Their visual processing front ends only use IMU information to predict the search range of feature points to be matched in the next frame, which can only deal with the problem (2). In fact, the problem (1) can also be solved by fully fusing visual inertia information in the whole process from feature extraction to feature matching.

Motivated by the aforementioned discussion, we present a novel feature-based stereo VIO, namely FSVIO, which fully utilizes inertial information to assist visual processing in both the front and the back ends. For the visual part of front end, the traditional feature extraction and matching methods are improved. By establishing the mathematical models of image perspective deformation caused by camera motion, the proposed feature extraction module reconstructs a robust IMU-aided descriptor that is invariant to in-plane and out-of-plane camera rotations. Combining the corner depth obtained by stereo and the camera pose predicted by the IMU, the position of feature points in the next frame is predicted for achieving faster and more accurate feature matching. This causes in fewer outliers caused by dynamic environment or nonconvexity of image. In order to further increase the pose estimation accuracy and robustness of VIO, we also improve the back end fusion algorithm. On the basis of the traditional optimization-based method, which only uses 3-D–2-D reprojection and IMU measurement errors, the 2-D–2-D epipolar geometry constraint and the improved Huber norm are added to reduce the influence of incorrect stereo depth estimation.

To summarize, our main contribution is threefold.

1) Two image perspective deformation models for stereo VIO are given to describe the effect of camera motion on feature tracking.
2) IMU information is fully utilized to assist feature extraction and matching. First, a novel IMU-aided descriptor is constructed by the proposed image perspective deformation models, which is insensitive to in-plane and out-of-plane camera rotations. Moreover, a fast IMU-aided feature tracking method is applied, which makes descriptor matching fast and accurate.
3) A complete stereo VIO is proposed, which achieves high-precision pose estimation without loop detection.

The rest of this article is organized as follows. Section II provides an overview of our system pipeline. The IMU-aided front end processing is discussed in Section III. We give a discussion about state management in the sliding window and propose a tightly coupled cost function in which a 2-D–2-D epipolar geometry constraint is introduced into the optimization in Section IV. The experimental evaluation results are shown in Section V. Finally, Section VI concludes this article.

II. System Overview

This section provides the main structure of the proposed stereo VIO, as illustrated in Fig. 1.

The odometry first abstracts and correlates features from visual and inertial sensor inputs. Feature corners are extracted, matched and their depth is estimated from the corrected left and right images, while inertial measurements are integrated to predict the current frame pose. Then, the estimated pose and depth are combined to achieve robust descriptor construction and fast matching for feature association in consecutive image frames of the left camera.

For each new data frame, the associated features are added to the sliding window and local map, which are used to establish visual residual constraints by a series of structure from motion algorithms and inertial residual constraints by the preintegration theory [27]. Finally, based on all the residual models, a tightly coupled optimization is presented to jointly estimate all the model parameters within the sliding window, including the velocity and pose of all frames in the window, the inverse depth of feature points and IMU intrinsic parameters.

III. IMU-Aided Front-End Processing

This section discusses the use of IMU information for visual measurement processing. Camera pose predicted by the IMU is substituted into the image perspective deformation model to construct rotation invariant descriptors and is combined with corner depth obtained by static stereo to achieve real-time accurate feature matching.

A. IMU Short-Term Pose Estimation

Due to short-term high-precision measurements, the IMU can be used to predict the camera pose in the next frame. Consider the true acceleration $\boldsymbol{a}_k$ and angular rate $\boldsymbol{w}_k$ in the body frame, which are measured as noisy sensor readings $\hat{\boldsymbol{a}}_k$ and $\hat{\boldsymbol{w}}_k$ with accelerometer bias $\boldsymbol{b}_{a_k}$ and gyroscope bias $\boldsymbol{b}_{g_k}$ from an IMU, the IMU kinematics discrete integral model is given as follows:

$$\boldsymbol{q}_{b_j}^{w} = \boldsymbol{q}_{b_i}^{w} \otimes \prod_{k=i}^{j-1} \begin{bmatrix} 1 \\ \frac{1}{2}(\hat{\boldsymbol{w}}_k - \boldsymbol{b}_{g_k} - \boldsymbol{n}_{w_k})\delta t \end{bmatrix}$$

$$\boldsymbol{v}_{b_j}^{w} = \boldsymbol{v}_{b_i}^{w} + \sum_{k=i}^{j-1} \left[\boldsymbol{R}_{b_k}^{w} (\hat{\boldsymbol{a}}_k - \boldsymbol{b}_{a_k} - \boldsymbol{n}_{a_k}) - \boldsymbol{g}^{w} \right] \delta t$$

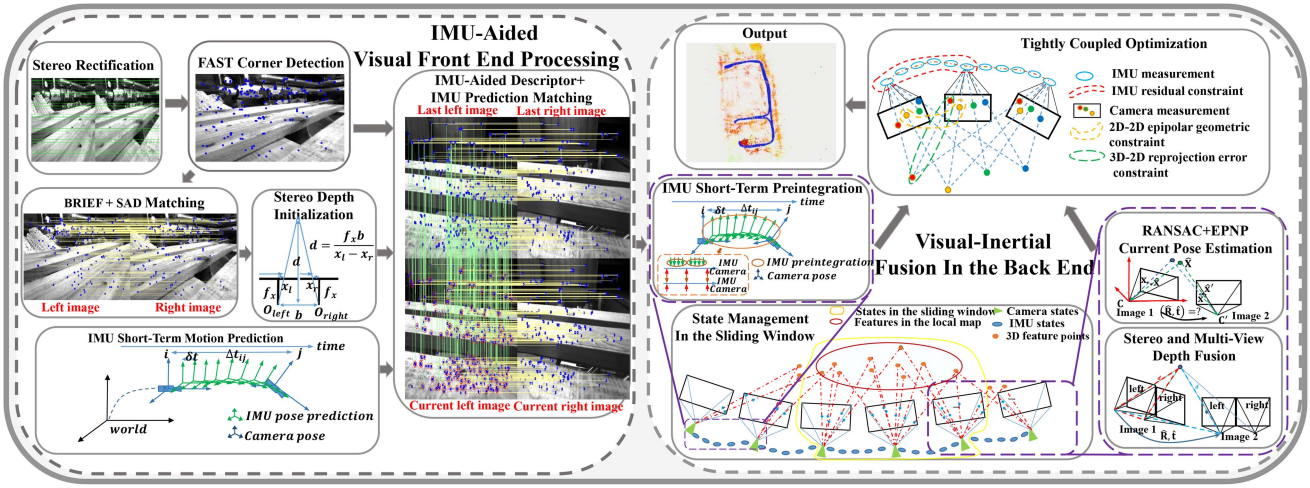

Fig. 1. Core flowchart of the proposed algorithm, which shows that visual-inertial information is fully exploited in the front and back ends. The camera pose predicted by the IMU is used to assist feature descriptor construction and matching in the front end. In the back end, visual-inertia information is fused by tight coupling optimization cost function in which our system jointly estimates six degrees-of-freedom camera poses within the active window by minimizing 3-D–2-D reprojection and IMU measurement errors under the constraints of 2-D–2-D epipolar geometry in the combined cost function.

$$\boldsymbol{t}_{b_j}^{w}=\boldsymbol{t}_{b_i}^{w}+\sum_{k=i}^{j-1}\left\{\boldsymbol{v}_{b_k}^{w}\delta t+\frac{1}{2}\left[\boldsymbol{R}_{b_k}^{w}(\hat{\boldsymbol{a}}_k-\boldsymbol{b}_{a_k}-\boldsymbol{n}_{a_k})-\boldsymbol{g}^{w}\right]\delta t^{2}\right\} \tag{1}$$

where $\boldsymbol{R}_{b_k}^{w}$ is the rotation matrix form of quaternion $\boldsymbol{q}_{b_k}^{w}$. δt is the time interval between two consecutive IMU measurements.

Furthermore, the IMU pose in the body frame is transformed into the camera pose by the following formula:

$$\begin{aligned}\boldsymbol{q}_{c_j}^{w}&=\boldsymbol{q}_{b_j}^{w}\otimes\boldsymbol{q}_{c}^{b}\\ \boldsymbol{t}_{c_j}^{w}&=\boldsymbol{R}_{b_j}^{w}\boldsymbol{t}_{c}^{b}+\boldsymbol{t}_{b_j}^{w}.\end{aligned} \tag{2}$$

B. IMU-Aided Descriptor Construction

Due to large camera motion, perspective deformation occurs between consecutive left image frames taken from the same scene. For camera translation along any direction on the camera plane, image perspective deformation occurs only in the scene that is not parallel to the image plane [see Fig. 2(b)] rather than the scene parallel to the image plane [see Fig. 2(a)]. For the translation along the normal vector direction of the camera plane, image perspective deformation occurs in both planar and nonplanar scenes, which is usually known as the scale change of image and is shown in Fig. 2(c). For camera rotation, both weak in-plane and out-of-plane ones cause severe image perspective deformation, as shown in Fig. 2(d)–(f). These deformations mean the descriptor using only local image information sensitive to the transformation of camera pose. To address this issue, it is necessary to build the mathematical model of image perspective deformation caused by camera motion. Consider the rotation and translation from the camera frame c_j to the camera frame c_i are $\boldsymbol{R}_{c_i}^{c_j}=[\boldsymbol{r}_1^{\top}\ \boldsymbol{r}_2^{\top}\ \boldsymbol{r}_3^{\top}]^{\top}$ and $\boldsymbol{t}_{c_i}^{c_j}=[t_1\ t_2\ t_3]^{\top}$, respectively. Assume the mth 3-D point in the local map is observed as $\boldsymbol{z}_m^{c_i}=[u_m^{c_i}\ v_m^{c_i}]^{\top}$ on the ith image and $\boldsymbol{z}_m^{c_j}=[u_m^{c_j}\ v_m^{c_j}]^{\top}$ on the jth image and its depth in the coordinate system of the camera

a表示相机和图像平面平行；b表示相机和图像平面不平行

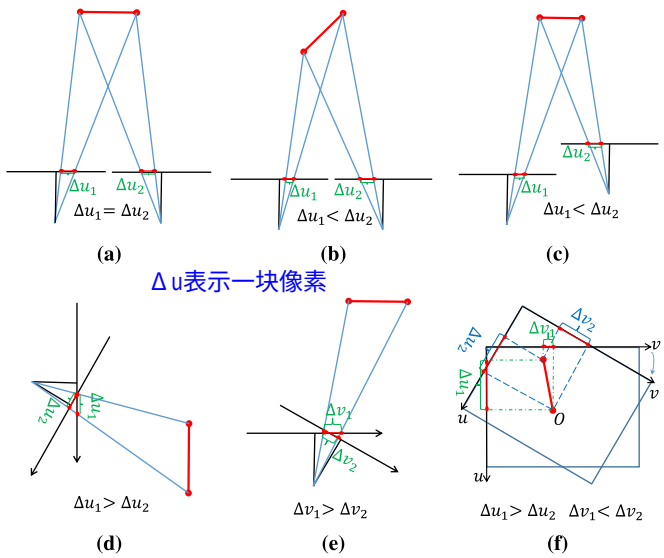

Fig. 2. Illustration of the image perspective deformation caused by camera motion. (a)–(c) Image deformation caused by camera translation. (d)–(f) Image deformation caused by out-of-plane and in-plane camera rotation.

在不同旋转方式下，特征匹配的半径是否固定不变
他研究的是随透视畸变特征深度出现了差异，导致对深度点估计错误出现了误差

c_i is $d_m^{c_i}$, we have

归一化后的相机坐标投影到像素坐标

$$\boldsymbol{z}_m^{c_j}=\Pi\left(\begin{bmatrix}\dfrac{d_m^{c_i}\boldsymbol{r}_1^{\top}\boldsymbol{x}_m^{c_i}+t_1}{d_m^{c_i}\boldsymbol{r}_3^{\top}\boldsymbol{x}_m^{c_i}+t_3} & \dfrac{d_m^{c_i}\boldsymbol{r}_2^{\top}\boldsymbol{x}_m^{c_i}+t_2}{d_m^{c_i}\boldsymbol{r}_3^{\top}\boldsymbol{x}_m^{c_i}+t_3} & 1\end{bmatrix}^{\top}\right) \tag{3}$$

为什么要乘d？

where $\boldsymbol{x}_m^{c_i}=\Pi^{-1}(\boldsymbol{z}_m^{c_i})$ and Π is the projection function based on the well-known pinhole camera model [24], which transforms the normalized coordinate of a 3-D point in the camera reference into a 2-D point on the image plane. Π^{-1} is the inverse projection function, which transforms a 2-D point on the image plane to the normalized coordinate of a 3-D point.

由于透视畸变存在，所以无法得到准确的深度值

From (3), the parameters $\boldsymbol{R}_{c_i}^{c_j}$, $\boldsymbol{t}_{c_i}^{c_j}$, and $d_m^{c_i}$ should be known *a priori* to eliminate the deformation. $\boldsymbol{R}_{c_i}^{c_j}$ and $\boldsymbol{t}_{c_i}^{c_j}$ can be obtained by the IMU. However, for the featured-based algorithm, the depth $d_m^{c_i}$ of image patch pixels associated with each feature point is generally unknown, which makes it impossible to correct the image perspective deformation by (3). In order to solve this problem, two solutions are given in this article.

消除畸变方法1

1) Model 1: Since the depth difference between a corner and its surrounding pixels is small relative to its own depth in most nonplanar scenes, perspective deformation shown in Fig. 2(b) is b considered to be similar to that in Fig. 2(a) and can be ignored. Since camera translations in the sliding window are very small relative to the average depth of all the image corners, image scale change can be well approached by the pyramid method and c image perspective deformation is assumed to be caused only by camera rotation. Based on the aforementioned analysis, we can simplify (3) and give a mathematical model of image perspective deformation caused by camera rotation

相机旋转带来的透视畸变，3d坐标投影到像素坐标

$$\boldsymbol{z}_m^{c_j} = \Pi\left(\begin{bmatrix}\frac{\boldsymbol{r}_1{}^\top \boldsymbol{x}_m^{c_i}}{\boldsymbol{r}_3{}^\top \boldsymbol{x}_m^{c_i}} & \frac{\boldsymbol{r}_2{}^\top \boldsymbol{x}_m^{c_i}}{\boldsymbol{r}_3{}^\top \boldsymbol{x}_m^{c_i}} & 1\end{bmatrix}^\top\right). \tag{4}$$

From the aforementioned equation, the projection position after rotation does not depend on the pixel depth value. This model is applicable to monocular, stereo, and RGB-D cameras.

方法2

2) Model 2: For most scenes, the depth of adjacent pixels on the image plane is similar. The depth value of the feature points can be obtained from our stereo camera. Thus, we can use the depth of the feature points to approximate the depth of image patch pixels around the feature points. Based on the aforementioned assumption, VIO with stereo or RGB-D camera can use (3) to correct the image perspective deformation caused by camera motion.

Significantly, both (3) and (4) can be used by our odometry to achieve perspective deformation correction of the image patch that is used to construct the descriptor. Here, to simplify the notation, let $\boldsymbol{T}_{c_i}^{c_j} = \begin{bmatrix}\boldsymbol{R}_{c_i}^{c_j} & \boldsymbol{t}_{c_i}^{c_j} \\ \boldsymbol{0}^\top & 1\end{bmatrix}$, then (3) and (4) are rewritten as

$$\boldsymbol{z}_m^{c_j} = \text{Pers}\left(\boldsymbol{z}_m^{c_i}, \boldsymbol{T}_{c_i}^{c_j}, d_m^{c_i}\right). \tag{5}$$

According to the proposed image perspective deformation models, we design a novel IMU-aided descriptor based on BRIEF [17]. Unlike the ORB descriptor [15], a popular and improved BRIEF descriptor, which uses image information to construct a local approximate rotation matrix for each corner to roughly correct the image perspective deformation caused only by in-plane camera rotation, our IMU-aided descriptor is insensitive not only to in-plane camera rotation but also to out-of-plane camera rotation due to the mathematical model (5) and the accurate short-term camera motion predicted by the IMU. The pseudocode of the IMU-aided descriptor construction algorithm is given in Algorithm 1. Consider a pixel point set $\mathcal{M}$ from the smoothed image patch $\boldsymbol{p}$, we define a binary test τ as

$$\tau(\boldsymbol{p}; \boldsymbol{a}, \boldsymbol{b}) := \begin{cases} 1 : \boldsymbol{p}(\boldsymbol{a}) < \boldsymbol{p}(\boldsymbol{b}) \\ 0 : \boldsymbol{p}(\boldsymbol{a}) \geq \boldsymbol{p}(\boldsymbol{b}) \end{cases} \tag{6}$$

Algorithm 1: IMU-Aided Descriptor Construction Algorithm.

Input : The pose transformation matrix $\boldsymbol{T}_{c_i}^{c_f}$, the FAST corner $\boldsymbol{z}_m^{c_i}$ and the pixel point set $\mathcal{M} = \begin{pmatrix}\boldsymbol{a}_1 & \cdots & \boldsymbol{a}_n \\ \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_n\end{pmatrix}$.

Output: The n-dimensional descriptor vector $g_n(\boldsymbol{p})$.

$\boldsymbol{T}_{c_i}^{c_f} = (\boldsymbol{T}_{c_i}^{c_f})^{-1}$

for $k = i$ *to* n **do**

  $\boldsymbol{a}_k = Pers(Pers(\boldsymbol{z}_m^{c_i}, \boldsymbol{T}_{c_i}^{c_f}) + \boldsymbol{a}_k, \boldsymbol{T}_{c_f}^{c_i}) - \boldsymbol{z}_m^{c_i}$ 消除畸变的像素坐标a

  $\boldsymbol{b}_k = Pers(Pers(\boldsymbol{z}_m^{c_i}, \boldsymbol{T}_{c_i}^{c_f}) + \boldsymbol{b}_k, \boldsymbol{T}_{c_f}^{c_i}) - \boldsymbol{z}_m^{c_i}$ 消除畸变的像素坐标b

  if $\boldsymbol{p}(\boldsymbol{a}_k) < \boldsymbol{p}(\boldsymbol{b}_k)$ **then**

    $\tau(\boldsymbol{p}; \boldsymbol{a}_k, \boldsymbol{b}_k) := 1$

  else

    $\tau(\boldsymbol{p}; \boldsymbol{a}_k, \boldsymbol{b}_k) := 0$

  end

end

$g_n(\boldsymbol{p}) := \sum_{k=1}^{n} 2^{k-1} \tau(\boldsymbol{p}; \boldsymbol{a}_k, \boldsymbol{b}_k)$

where $\boldsymbol{p}(\boldsymbol{a})$ and $\boldsymbol{p}(\boldsymbol{b})$ are the intensity of pixel points $\boldsymbol{a}$ and $\boldsymbol{b}$, respectively. The descriptor is defined as an n-dimensional vector

$$f_n(\boldsymbol{p}) := \sum_{k=1}^{n} 2^{k-1} \tau\left(\boldsymbol{p}; \boldsymbol{a}_k, \boldsymbol{b}_k\right) \tag{7}$$

with

$$\mathcal{M} = \begin{pmatrix}\boldsymbol{a}_1 & \cdots & \boldsymbol{a}_n \\ \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_n\end{pmatrix}.$$

As with the ORB descriptor, the vector length n is 256 and the binary tests use a Gaussian distribution around the center of the patch. Considering that pose transformation matrix of the camera frame c_i relative to the reference frame c_f (the first image frame after successful system initialization) is $\boldsymbol{T}_{c_i}^{c_f} = \boldsymbol{T}_{c_f}^{w\,-1} \hat{\boldsymbol{T}}_{c_i}^{w}$, where $\boldsymbol{T}_{c_f}^{w}$ has been estimated by our odometry and $\hat{\boldsymbol{T}}_{c_i}^{w}$ can be accurately predicted by the IMU according to the literature [25], we use (5) to construct a corrected version of $\mathcal{M}$ related to a FAST corner $\boldsymbol{z}_m^{c_i}$ in the image frame c_i as follows:

$$\mathcal{M}_{\text{corr}} = \begin{pmatrix}\text{Corr}(\boldsymbol{a}_1) & \cdots & \text{Corr}(\boldsymbol{a}_n) \\ \text{Corr}(\boldsymbol{b}_1) & \cdots & \text{Corr}(\boldsymbol{b}_n)\end{pmatrix} \tag{8}$$

where

delta

$$\text{Corr}(\boldsymbol{a}_k) = \text{Pers}\left(\text{Pers}\left(\boldsymbol{z}_m^{c_i}, \boldsymbol{T}_{c_i}^{c_f}\right) + \boldsymbol{a}_k, \boldsymbol{T}_{c_f}^{c_i}\right) - \boldsymbol{z}_m^{c_i}$$

$$\text{Corr}(\boldsymbol{b}_k) = \text{Pers}\left(\text{Pers}\left(\boldsymbol{z}_m^{c_i}, \boldsymbol{T}_{c_i}^{c_f}\right) + \boldsymbol{b}_k, \boldsymbol{T}_{c_f}^{c_i}\right) - \boldsymbol{z}_m^{c_i}.$$

Then, our IMU-aided descriptor operator becomes

$$g_n(\boldsymbol{p}) := f_n(\boldsymbol{p}) | (\boldsymbol{a}_k, \boldsymbol{b}_k) \in \mathcal{M}_{\text{corr}}. \tag{9}$$

描述子 像素对

It is important that we correct the relative position between each corner and all pixels used for the binary tests associated with the corner instead of their absolute position on the image. In the following, our IMU-aided descriptors based on model 1 and model 2 are called IMU-AIDED1 and IMU-AIDED2, respectively.

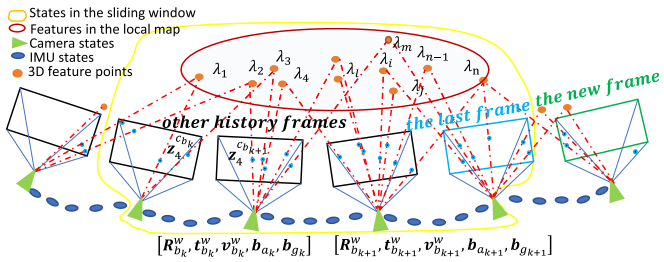

Fig. 3. Illustration of state management in the sliding window. We keep several camera and IMU states and maintain a local map consisting of all the 3-D feature points successfully recovered for tracking and optimization.

C. Feature Tracking Using IMU-Aided Matching

For each new image frame, we only track the corners that were observed by the left and right cameras from stereo cameras at the same time in the previous image frames. In order to improve the efficiency and accuracy of feature matching, we propose a fast matching strategy that combines the information of stereo cameras and IMU to predict the projection pixel position of each corners. Assume that the camera frame c_i is a frame in the sliding window and the camera frame c_j is a new image frame to be matched. With short-term high-precision IMU measurements, we can accurately estimate the translation $\boldsymbol{t}_{c_i}^{c_j}$ and rotation $\boldsymbol{R}_{c_i}^{c_j}$ from the camera frame c_j to the camera frame c_i. Benefiting from the fixed baseline of stereo cameras, we can directly recover the 3-D point $\boldsymbol{X}_m^{c_i}$ related to the mth corner of the ith image frame according to the literature [14]. Then, the projected position $\boldsymbol{z}_m^{c_j}$ of the tracked 3-D point on the jth image plane is predicted as follows:

$$\hat{\boldsymbol{z}}_m^{c_j} = \Pi\left(\frac{\boldsymbol{R}_{c_i}^{c_j}\boldsymbol{X}_m^{c_i} + \boldsymbol{t}_{c_i}^{c_j}}{d_m^{c_j}}\right) \tag{10}$$

where $\hat{\boldsymbol{z}}_m^{c_j}$ denotes the predicted value of $\boldsymbol{z}_m^{c_j}$. Then, from all the corners whose Euclidean distance from the predicted position $\hat{\boldsymbol{z}}_m^{c_j}$ is less than r (we use $r = 10$ based on the image resolution and IMU accuracy in our equipment) on the jth image plane, the corner with the smallest hamming distance between its descriptor and the descriptor of the tracked 3-D point is found as the best matching corner and its observed value on the image plane is considered to be the closest to $\boldsymbol{z}_m^{c_j}$.

IV. Visual-Inertial Fusion in the Back End

A. State Management in the Sliding Window

In order to reduce computational complexity and efficiently utilize front end measurements, the sliding window is used by the proposed odometry, as shown in Fig. 3. In the sliding window, we keep several camera and IMU states and maintain a local map consisting of all the 3-D feature points successfully recovered for tracking and optimization. As the camera moves continuously, we need to constantly update the states in the sliding window. For each new frame, the current camera pose is predicted by the IMU and optimized by EPnP [28] with RANSAC [29]. Based on the estimated pose, the depth of the matching corners between the last and the new image frames is estimated by the linear triangulation method [30]. Consider the pixel coordinates $\boldsymbol{z}_i^{b_k} = [u_i^{b_k}\ v_i^{b_k}\ 1]^\top$ of the feature point i in the camera frame b_k, whose depth is recovered as $\tilde{\lambda}_i^{b_k}$ by the stereo model [14] and is estimated as $\lambda_i^{b_k}$ by the linear triangulation method [30], the point is considered as a new 3-D tracked point to add into the local map if it satisfies

$$\text{th}_1 \le \frac{\lambda_i^{b_k}}{\tilde{\lambda}_i^{b_k}} \le \text{th}_2 \tag{11}$$

where th_1 and th_2 are minimum and maximum thresholds of the ratio (we use $\text{th}_1 = 0.5$ and $\text{th}_2 = 2$ based on trial-and-error). Its 3-D coordinate $\boldsymbol{X}_i^{b_k}$ is recovered as $[\frac{\lambda_i^{b_k}+\tilde{\lambda}_i^{b_k}}{2}u_i^{b_k}\ \frac{\lambda_i^{b_k}+\tilde{\lambda}_i^{b_k}}{2}v_i^{b_k}\ \frac{\lambda_i^{b_k}+\tilde{\lambda}_i^{b_k}}{2}]^\top$. If the tracked feature point number between the new frame and the last frame is less than the threshold or the interval between the two frames is greater than 0.5 s, the new frame state is added to the sliding window while the oldest one is removed.

B. Tightly Coupled Optimization

A tightly coupled optimization framework is used to estimate camera motion by minimizing visual and inertial residuals in the combined cost function. However, unlike the traditional optimization cost function that only includes the IMU residual and the 3-D–2-D visual reprojection error, our cost function also adds the new 2-D–2-D epipolar geometric residual, which can reduce the influence of wrong stereo depth estimation of corners on optimization results. Within the sliding window, the set of all parameters $\mathcal{X}$ to be estimated are defined as

$$\mathcal{X} := \{\mathcal{X}_{\text{common}}, \mathcal{X}_{\text{imu}}, \mathcal{X}_{\text{cam}}\} \tag{12}$$

where $\mathcal{X}_{\text{common}}$ is the common state set of the IMU and camera, including the rotation $\boldsymbol{R}_{b_k}^w$ and translation $\boldsymbol{t}_{b_k}^w$ of the body frame b_k in the sliding window relative to the world coordinate system. $\mathcal{X}_{\text{imu}}$ is the set of IMU-specific states, including the speed, acceleration, and gyroscope bias of the IMU corresponding to the body frame b_k in the sliding window. $\mathcal{X}_{\text{cam}}$ is the set of camera specific states, including the inverse depth value of each tracked 3-D point in the first observed frame.

We jointly optimize all the parameters to be estimated in the sliding window to minimize visual and inertial residual cost functions as follows:

$$\begin{aligned}\arg\min_{\mathcal{X}} \Bigg\{ &\sum_{k\in\mathcal{B}} \|\boldsymbol{E}_{\text{imu}}(\boldsymbol{z}_{b_{k+1}}^{b_k}, \mathcal{X})\|_{\Sigma_i}^2 \\ &+ \sum_{i,j\in\text{obs}(m),m\in\mathcal{L}} \rho\|\boldsymbol{E}_{v3d2d}(\boldsymbol{z}_{b_m}^{c_j}, \mathcal{X})\|_{\Sigma_p}^2 \\ &+ w_f\|\boldsymbol{E}_{v2d2d}\left(\boldsymbol{z}_{b_m}^{c_i}, \boldsymbol{z}_{b_m}^{c_j}, \mathcal{X}\right)\|_{\Sigma_f}^2\Bigg)\Bigg\}\end{aligned} \tag{13}$$

with

$$\rho(s) = \begin{cases} s : s < 1 \\ 2\sqrt{s} - 1 : s \le \frac{0.5(f_x+f_y)}{1+10^{\text{len}(fx,fy)-1}} \\ 0 : \text{otherwise} \end{cases}$$

where $\rho(.)$ is an improved Huber norm, which filters out false visual residuals. len(f_x, f_y) is the maximum number of digits of both f_x and f_y. w_f is the weight of E_{v2d2d} in visual residuals. $\mathcal{B}$ is the set of all measurement frames in the sliding window. $\mathfrak{obs}(m)$ is the set of all image frames where the mth 3-D point in the local map $\mathcal{L}$ can be observed. $\boldsymbol{E}_{\text{imu}}$ is the IMU residual term, $\boldsymbol{E}_{v3d2d}$ is the 3-D–2-D visual reprojection error term, and $\boldsymbol{E}_{v2d2d}$ is the 2-D–2-D epipolar geometric constraint term. Σ_i, Σ_p, and Σ_f are the information matrices corresponding to the residuals. The tightly coupled optimization is essentially a nonlinear least-square problem, which is solved by the iterative algorithm provided by ceres [31]. Here, the Gauss–Newton method is used to explain the application of the iterative algorithms. The combined cost function (13) is simply expressed as $\boldsymbol{E}$. The Jacobian of $\boldsymbol{E}$ with respect to $\mathcal{X}$ is denoted by $\boldsymbol{J} = \frac{\partial \boldsymbol{E}}{\partial \mathcal{X}}$. Then, (13) is solved by nonlinear iterations to make $\boldsymbol{E}$ toward zero as

$$\mathcal{X} \leftarrow \mathcal{X} - (\boldsymbol{J}^\top \boldsymbol{J})^{-1} \boldsymbol{J}^\top \boldsymbol{E}. \tag{14}$$

Detailed definitions of all residual terms are described as follow.

1) IMU Residual: The IMU measurement residual between two body frames b_k and b_{k+1} corresponding to two consecutive image frames is defined as follows:

$$\boldsymbol{E}_{\text{imu}}(\boldsymbol{z}^{b_k}_{b_{k+1}}, \mathcal{X}) = \begin{bmatrix} \boldsymbol{R}^{b_k}_w \left(\boldsymbol{t}^w_{b_{k+1}} - \boldsymbol{t}^w_{b_k} - \boldsymbol{v}^w_{b_k} \Delta t_k + \frac{1}{2} \boldsymbol{g}^w \Delta t_k^2 \right) - \widetilde{\boldsymbol{\alpha}}^{b_k}_{b_{k+1}} \\ \boldsymbol{R}^{b_k}_w \left(\boldsymbol{v}^w_{b_{k+1}} - \boldsymbol{v}^w_{b_k} + \boldsymbol{g}^w \Delta t_k \right) - \widetilde{\boldsymbol{\beta}}^{b_k}_{b_{k+1}} \\ 2 \left[\boldsymbol{q}^{b_k}_w \otimes \boldsymbol{q}^w_{b_{k+1}} \otimes \left(\widetilde{\boldsymbol{\gamma}}^{b_k}_{b_{k+1}} \right)^{-1} \right]_{xyz} \\ \boldsymbol{b}_{a_{k+1}} - \boldsymbol{b}_{a_k} \\ \boldsymbol{b}_{g_{k+1}} - \boldsymbol{b}_{g_k} \end{bmatrix} \tag{15}$$

where Δt_k is the time interval between two body frames b_k and b_{k+1}. $\widetilde{\alpha}^{b_k}_{b_{k+1}}$, $\widetilde{\beta}^{b_k}_{b_{k+1}}$, and $\widetilde{\gamma}^{b_k}_{b_{k+1}}$ are the preintegration [27] terms of IMU measurements with noise, which has been discussed in detail in Appendix A of VINS-MONO [25].

2) 3-D–2-D Visual Residual: Consider the mth 3-D point in the local map, which is first observed as $\boldsymbol{z}^{c_i}_m$ by the camera frame c_i and is observed as $\boldsymbol{z}^{c_j}_m$ by the camera frame c_j, the 3-D–2-D reprojection error based on the pinhole camera model is defined as

$$\boldsymbol{E}_{v3d2d}(\boldsymbol{z}^{c_j}_m, \mathcal{X}) = \boldsymbol{z}^{c_j}_m - \Pi(\boldsymbol{x}^{c_j}_m) \tag{16}$$

where

$$\boldsymbol{x}^{c_j}_m = \frac{\boldsymbol{X}^{c_j}_m}{[\boldsymbol{X}^{c_j}_m]_z} \boldsymbol{X}^{c_j}_m$$

$$= \boldsymbol{R}^c_b \left\{ \boldsymbol{R}^{b_j}_w \left[\boldsymbol{R}^w_{b_i} \left(\boldsymbol{R}^b_c \frac{1}{\lambda_m} \Pi^{-1}(\boldsymbol{z}^{c_i}_m) + \boldsymbol{t}^b_c \right) + \boldsymbol{t}^w_{b_i} - \boldsymbol{t}^w_{b_j} \right] - \boldsymbol{t}^b_c \right\}$$

where $[\boldsymbol{X}^{c_j}_m]_z$ is the third value of the vector $\boldsymbol{X}^{c_j}_m$. $\boldsymbol{R}^b_c$ and $\boldsymbol{t}^b_c$ have been calibrated offline and are fixed variables that are not optimized.

TABLE I
STATISTIC OF THE RUNTIME OF DIFFERENT MODULES

Modules	Thread	Time (ms)
Stereo feature extraction	2	15.8
Stereo matching and depth estimation	1	8.5
Descriptor construction with model 1 (2)	1	10.3 (12.2)
Feature matching	1	0.4
Tightly optimization	1	40.0

3) 2-D–2-D Visual Residual: The 2-D–2-D epipolar geometric residuals are introduced as follows:

$$\boldsymbol{E}_{v2d2d}(\boldsymbol{z}^{c_i}_m, \boldsymbol{z}^{c_j}_m, \mathcal{X}) = \boldsymbol{x}^{c_j\top}_m \boldsymbol{R}^c_b \boldsymbol{R}^{b_j}_w \boldsymbol{R}^w_{b_i} \boldsymbol{R}^b_c [\boldsymbol{x}^{c_i}_m]_\times \cdot \left\{ \boldsymbol{R}^c_b \left[\boldsymbol{R}^{b_i}_w \left(\boldsymbol{R}^w_{b_j} \boldsymbol{t}^b_c + \boldsymbol{t}^w_{b_j} - \boldsymbol{t}^w_{b_i} \right) - \boldsymbol{t}^b_c \right] \right\} ?$$
$$\boldsymbol{x}^{c_i}_m = \Pi^{-1}(\boldsymbol{z}^{c_i}_m), \quad \boldsymbol{x}^{c_j}_m = \Pi^{-1}(\boldsymbol{z}^{c_j}_m). \tag{17}$$

Moreover, it is worth noting that E_{v2d2d} does not play a leading role but serves as an additional penalty constraint in the combined cost function, because it is a nonconvex function, which has a local minimum.

V. EXPERIMENTAL RESULTS

In this section, three experiments are first given to evaluate our algorithm on the EuRoC [32] datasets by using the open source python package EVO [33]. Then, real-world experiments are designed to test the practicality of our work.

A. Evaluation of Tracking

For the sake of evaluating the efficiency of feature extraction and matching, a statistic about the running time of each module of our system is given in Table I. Here, our algorithm is tested on a desktop PC, whose computation resource is an Intel i3-7100 CPU with 3.9 GHz x4. Under the condition that the number of FAST corners in each frame is 1000 and the sliding window size is 10, we take the running time average of the algorithm in sequence MH_01_easy as the statistical result. Descriptor construction with IMU-AIDED1 or IMU-AIDED2 takes 10.3 or 12.2 ms, which shows that our descriptor is a real-time descriptor suitable for SLAM or odometry. In particular, feature point tracking only takes 0.4 ms due to our efficient feature matching strategy.

In order to further verify that our algorithm can effectively solve the image perspective deformation caused by camera motion, the proposed descriptor is compared with other popular and real-time descriptors such as ORB and BRIEF on the EuRoC datasets with many fast motion scenes. We compare the pose estimation accuracy of our odometry with different descriptors to indirectly verify the robustness of each descriptor for the deformation. The root-mean-square error (RMSE) of the absolute trajectory error (ATE) in each sequence is shown Table II. In most sequences, the trajectories obtained by our algorithm using the IMU-aided descriptors are closer to ground truth. It proves that the proposed IMU-aided descriptors are more accurate and robust. We also compare the construction time with different descriptor construction algorithms for 1000 feature

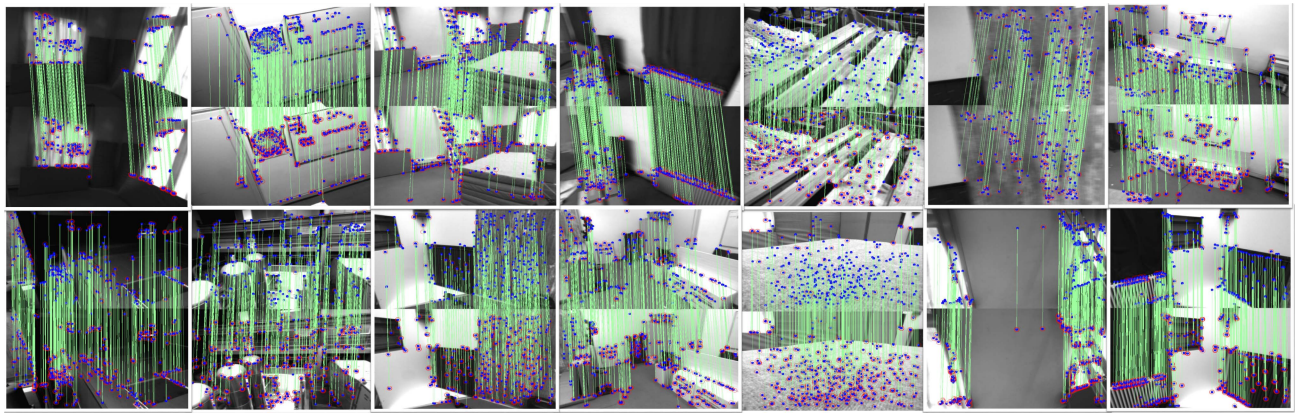

Fig. 4. Visualization of the matching results of our algorithm in different scenarios.

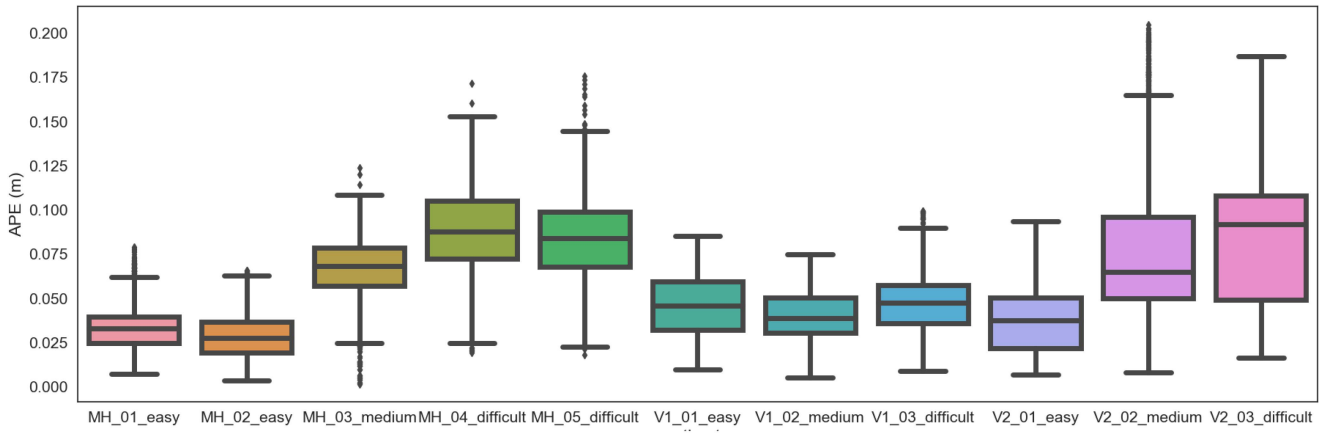

Fig. 5. Statistics of the APE of the trajectories estimated by FSVIO in all sequences from EuRoC datasets.

TABLE II
RMSE [34] OF THE ESTIMATED TRAJECTORY ATE OBTAINED BY FSVIO WITH DIFFERENT DESCRIPTORS

Sequence	ORB	BRIEF	IMU-AIDED1	IMU-AIDED2
MH_01_easy	0.054	0.055	**0.036**	0.043
MH_02_easy	0.032	0.038	**0.026**	0.030
MH_03_medium	0.091	0.078	**0.058**	0.076
MH_04_difficult	0.112	0.105	0.091	**0.088**
MH_05_difficult	0.086	0.088	**0.081**	0.083
V1_01_easy	0.068	0.064	**0.049**	0.052
V1_02_medium	**0.046**	**0.046**	**0.032**	0.046
V1_03_difficult	0.068	0.063	**0.050**	0.060
V2_01_easy	0.059	0.060	**0.045**	0.048
V2_02_medium	**0.091**	0.093	0.093	0.110
V2_03_difficult	0.191	0.175	**0.090**	0.115

TABLE III
COMPARISON OF RUNNING TIME OF DIFFERENT DESCRIPTOR CONSTRUCTON ALGORITHMS

Algorithm	BRIEF	ORB	IMU-AIDED1	IMU-AIDED2
Time	**1.4ms**	3.3ms	10.3ms	12.2ms

points, which is shown in Table III. The proposed descriptor increases the computational complexity due to the accurate correction operation. However, comparing with the improvement of system performance in VIO, the additional operation is worth it. Furthermore, the performance of IMU-AIDED1 is better than that of IMU-AIDED2, which may be due to a significant difference in the depth of pixels around the feature points in our stereo VIO. We suspect that IMU-AIDED2 performs better in VIO with RGB-D cameras due to accurate depth measurement. The following evaluations directly use IMU-AIDED1 to achieve the remaining experiments because of its better performance in our odometry. Remarkably, ORB is usually more robust than BRIEF in a scene where the camera rotates rapidly and the pose estimation accuracy of using BRIEF is lower, but in our algorithm, the performance of using BRIEF and ORB is almost the same because our feature matching strategy combining visual and inertial information excludes most of the outliers.

For evaluating the robustness and accuracy of our matching algorithm, the results of feature matching in multiple scenes are sampled and visualized, which is shown in Fig. 4. The predicted range of each feature point, obtained by using IMU information, is marked with a red circle. The blue point is the feature point

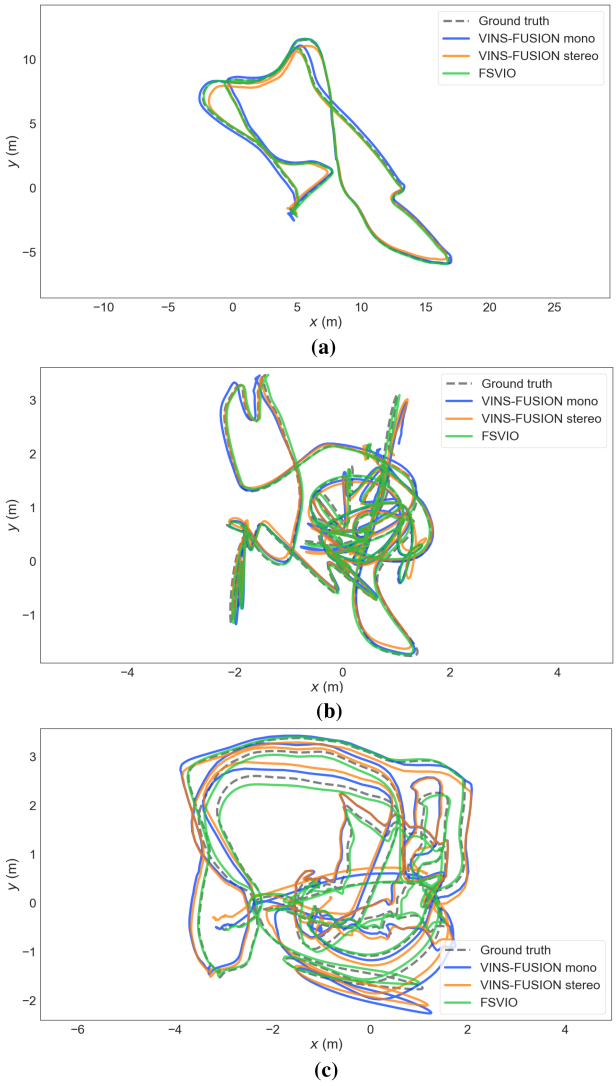

Fig. 6. Comparison of the estimated trajectories obtained by FSVIO and VINS-FUSION in sequences MH_05_difficult, V1_03_difficult, and V2_03_difficult, respectively.

successfully matched between the last or another previous frame of the sliding window and the current frame. The green line connects a feature point and its associated feature point in two consecutive frames. Those lonely blue feature points correspond to the feature points of other historical frames. As can be seen from Fig. 4, the matching results of the proposed algorithm are very accurate in different scenarios.

B. Algorithm Comparison

In order to comprehensively evaluate the performance of FSVIO, we compare our odometry trajectories with the ground truths in all sequences of EuRoC datasets. Fig. 5 is the absolute pose error (APE) statistics of the trajectories estimated by FSVIO in all sequences from EuRoC datasets, which shows that our algorithm has an RMSE less than 0.1 m on most sequences and especially less than 0.05 m on sequences MH_01_easy, MH_02_easy, V1_01_easy, V1_02_medium, V1_03_difficult, and V2_01_easy.

We also compare the performance of our algorithm to other state-of-the-art VIO including OKVIS [23], VI-DSO [10], VINS-FUSION [25], and Basalt [35] on the EuRoC datasets. The evaluation results in terms of RMSE are summarized in Table IV. Our approach not only runs successfully on all sequences, but also shows the best performance in most sequences. While our system is a VIO without closed loop [36], the long-term drift is very small. In particular, while our system directly use the IMU measurement processing algorithm from VINS-FUSION, the accuracy of our VIO outperforms VINS-FUSION in all sequences of EuRoC datasets. In order to more intuitively show the performance improvement brought by our algorithm, we compared the estimated trajectories of FSVIO and VINS-FUSION in sequences MH_05_difficult, V1_03_difficult, and V2_03_difficult, as shown in Fig. 6.

C. Real-World Experiments

In this section, the proposed odometry is tested by several real-world experiments, in which the closed-loop motion trajectory is designed so that the pose error between the start and the end points can be used to evaluate the estimation accuracy of the odometry. The datasets in both indoor and outdoor environments are captured by stereo cameras (Intel Realsense D455), which records 640 × 480 stereo images at 15 Hz and IMU data with 200 Hz.

1) Room Experiment: We walk around the room with the device in hand, and then, go back to the place of departure, as illustrated in Fig. 7(a). In particular, there are some scenes where our camera rotates rapidly in the experiment. The visualization of pose estimation in our system is shown in Fig. 7(f) and (k), which shows that the shape of our estimated trajectory is very close to the real one. The length of the estimated trajectory is about 43 m and the difference between the start and the end positions is 0.04 m, which demonstrates that our system can perform high-precision pose estimation in an indoor texture-rich environment and work stably under fast rotation.

2) Corridor Experiment: As shown in Fig. 7(b), the corridor is selected to verify the robustness of the system. The camera moves forward approximately 25 m, and then, returns to the origin. There are some similar and weak texture scenes in the experiment. For some images with similar texture, we use IMU to predict motion pose to assist feature tracking, which can effectively reduce false matching. For weak texture regions, we use IMU pose prediction to avoid pose estimation failure. The estimated results are shown in Fig. 7(g) and (l). The estimated path length is 53 m and the position error of the start and the end points is 0.13 m, which shows that our system is robust.

3) Outdoor Experiment: The proposed algorithm is also tested in a long outdoor travel. The first outdoor experiment occurs in the campus garden, as shown in Fig. 7(c). The person with the visual-inertial equipment walks 188 m around the garden. Comparing Fig. 7(c) with Fig. 7(h), the estimated trajectory is consistent with the contour of the garden. The start to end error of the estimated trajectory in Fig. 7(m) is 0.7 m. The second

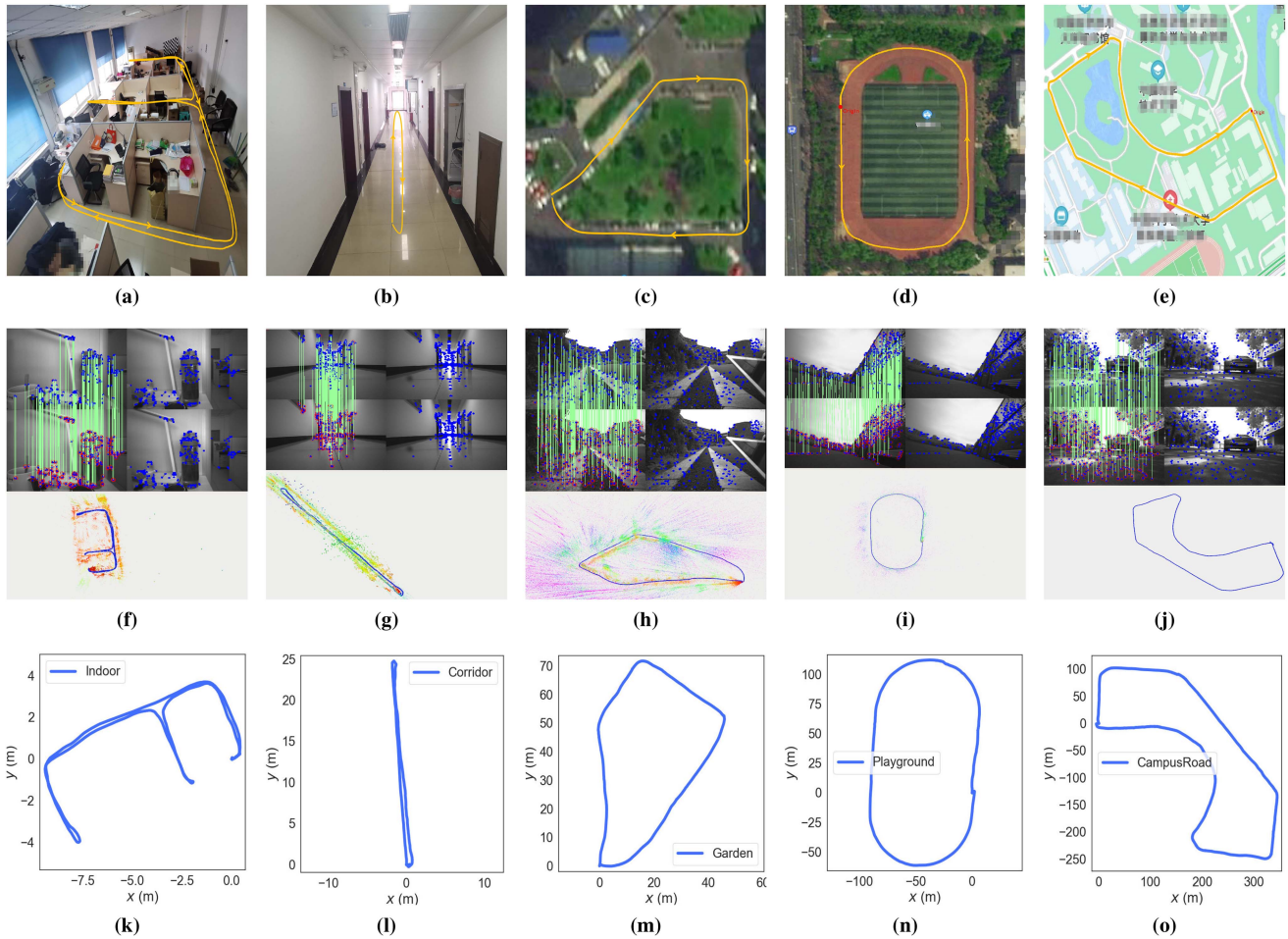

Fig. 7. (a)–(e) Real motion path of the camera in room, corridor, garden, playground, and campus road, respectively. (f)–(j) Visualization of real-time motion estimation results of FSVIO in different scenes. (k)–(o) Estimated trajectories by FSVIO in different scenes.

TABLE IV
RMSE OF THE ESTIMATED TRAJECTORY ATE OBTAINED BY DIFFERENT METHODS

Sequence	OKVIS [23] mono	OKVIS [23] stereo	VI-DSO [10] mono	VINS-FUSION [26] mono	VINS-FUSION [26] stereo	Basalt [35] stereo	**FSVIO stereo**
MH_01_easy	0.34	0.23	0.06	0.18	0.24	0.07	**0.04**
MH_02_easy	0.36	0.15	0.04	0.09	0.28	0.05	**0.03**
MH_03_medium	0.30	0.23	0.12	0.17	0.23	**0.06**	**0.06**
MH_04_difficult	0.48	0.32	0.13	0.21	0.39	0.12	**0.09**
MH_05_difficult	0.47	0.36	0.12	0.25	0.19	0.12	**0.08**
V1_01_easy	0.12	**0.04**	0.06	0.06	0.10	0.05	0.05
V1_02_medium	0.16	0.08	0.07	0.09	0.10	0.05	**0.03**
V1_03_difficult	0.24	0.13	0.10	0.18	0.11	0.10	**0.05**
V2_01_easy	0.12	0.10	0.04	0.06	0.12	**0.04**	0.05
V2_02_medium	0.22	0.17	0.06	0.11	0.10	**0.05**	0.09
V2_03_difficult	X	X	0.17	0.26	0.27	X	**0.09**

outdoor experiment scene is chosen as the playground, as shown in Fig. 7(d). The bicycle carries the camera and moves quickly around the playground for about 460 m. The distance between the start and end points in Fig. 7(n) is 3.04 m. The distance estimation error per meter is only 6.5 mm. The third outdoor experiment scene is located in the road around the campus. We take the camera for a quick ride around the campus road, and then, go back to the start point, as shown in Fig. 7(e). The whole ride is about 1.3 km and the distance between the start and the end points in Fig. 7(o) is 10.79 m. The error of distance estimation is about 9 mm per meter.

Finally, the distance between the start point and the end point from the estimated trajectories in all real scenes is summarized as shown in Table V. Since FSVIO mainly improves from VINS-FUSION, we also compare the performance of FSVIO and VINS-FUSION based on stereo cameras in the table. The

TABLE V
COMPARISON OF THE DISTANCE BETWEEN THE START AND THE END POSITIONS FROM THE TRAJECTORIES OF POSE ESTIMATION BASED ON DIFFERENT ALGORITHMS IN FIVE DIFFERENT REAL SCENES

sequence	Room 43m	Corridor 50m	Garden 188m	Playground 460m	Campus 1300m
VINS-FUSION stereo	0.24m	0.95m	1.10m	8.42m	15.80m
FSVIO	**0.04m**	**0.13m**	**0.70m**	**3.04m**	**10.79m**

experimental results show that our algorithm can achieve smaller drift and better performance in most outdoor scenes.

VI. CONCLUSION

In this article, we presented a novel VIO with stereo cameras, which combined visual and inertial measurements in both the front and the back ends. Two image perspective deformation models were established to fuse visual and inertial measurements for the novel IMU-aided descriptor construction. The proposed descriptor was robust to camera motion, which can be invariant not only to in-plane camera rotation but also to out-of-plane camera rotation. We combined the corner depth obtained by stereo and the camera pose predicted by the IMU to predict the position of corners in the next frame for reducing more outliers caused by dynamic environment or nonconvexity of image. A 2-D–2-D epipolar geometric constraint and an improved Huber norm were introduced into the optimization, which reduced the influence of the incorrect stereo depth estimation. In the evaluation experiment, our IMU-aided descriptor can run in real time and performed better than other popular descriptors that were often used in the field of VIO or SLAM. Our VIO method achieved SOTA performance on the EuRoC datasets and estimates ego-motion robustly and accurately in real-world experiments.

In the future work, we plan to apply the proposed IMU-AIDED2 to VIO with RGB-D camera, and also improve the IMU-aided descriptor for loop detection.

REFERENCES

[1] R. Li, S. Wang, and D. Gu, "DeepSLAM: A robust monocular slam system with unsupervised deep learning," *IEEE Trans. Ind. Electron.*, vol. 68, no. 4, pp. 3577–3587, Apr. 2021.

[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, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular slam system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[4] V. Hoang, M.-H. Le, and K.-H. Jo, "Motion estimation based on two corresponding points and angular deviation optimization," *IEEE Trans. Ind. Electron.*, vol. 64, no. 11, pp. 8598–8606, Nov. 2017.

[5] C. Cadena *et al.*, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.

[6] M. Irani and P. Anandan, "About direct methods," in *Vision Algorithms: Theory and Practice*. Berlin, Germany: Springer, 2000, pp. 267–277.

[7] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 3748–3754.

[8] R. A. Newcombe *et al.*, "KinectFusion: Real-time dense surface mapping and tracking," in *Proc. IEEE Int. Symp. Mixed Augmented Reality*, 2011, pp. 127–136.

[9] D. Caruso, J. Engel, and D. Cremers, "Large-scale direct SLAM for omnidirectional cameras," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 141–148.

[10] L. Von Stumberg, V. Usenko, and D. Cremers, "Direct sparse visual-inertial odometry using dynamic marginalization," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 2510–2517.

[11] J. Jiang, J. Yuan, X. Zhang, and X. Zhang, "DVIO: An optimization-based tightly coupled direct visual-inertial odometry," *IEEE Trans. Ind. Electron.*, vol. 68, no. 11, pp. 11212–11222, Nov. 2020.

[12] A. J. Davison *et al.*, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[13] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proc. IEEE ACM Int. Symp. Mixed Augmented Reality*, 2007, pp. 225–234.

[14] 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.

[15] E. Rublee *et al.*, "ORB: An efficient alternative to sift or surf," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.

[16] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. Artif. Intell.*, 1981, pp. 674–679.

[17] M. Calonder *et al.*, "Brief: Binary robust independent elementary features," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.

[18] J. Shi and Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1994, pp. 593–600.

[19] X. Zhang, B. Xian, B. Zhao, and Y. Zhang, "Autonomous flight control of a nano quadrotor helicopter in a GPS-denied environment using on-board vision," *IEEE Trans. Ind. Electron.*, vol. 62, no. 10, pp. 6392–6403, Oct. 2015.

[20] Y. Zhou *et al.*, "Toward autonomy of micro aerial vehicles in unknown and global positioning system denied environments," *IEEE Trans. Ind. Electron.*, vol. 68, no. 8, pp. 7642–7651, Aug. 2021.

[21] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart, "Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 957–964.

[22] 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.*, 2007, pp. 3565–3572.

[23] S. Leutenegger *et al.*, "Keyframe-based visual-inertial odometry using nonlinear optimization," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 314–334, 2014.

[24] 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.

[25] 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.

[26] T. Qin *et al.*, "A general optimization-based framework for local odometry estimation with multiple sensors," 2019, *arXiv:1901.03638*.

[27] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual-inertial odometry," *IEEE Trans. Robot.*, vol. 33, no. 1, pp. 1–21, Feb. 2017.

[28] V. Lepetit *et al.*, "EPNP: An accurate o(n) solution to the PNP problem," *Int. J. Comput. Vis.*, vol. 81, pp. 155–166, 2009.

[29] M. A. Fischler, "Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[30] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[31] S. Agarwal *et al.*, "Ceres solver." [Online]. Available: http://ceres-solver.org

[32] M. Burri *et al.*, "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.

[33] M. Grupp, "EVO: Python package for the evaluation of odometry and SLAM," 2017. [Online]. Available: https://github.com/MichaelGrupp/evo

[34] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 573–580.

[35] V. Usenko, N. Demmel, D. Schubert, J. Stückler, and D. Cremers, "Visual-inertial mapping with non-linear factor recovery," *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 422–429, Apr. 2020.
[36] P. Newman and K. Ho, "SLAM-loop closing with visually salient features," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2005, pp. 635–642.

Lei Yu received the B.E. degree in automation from North China Electric Power University, Baoding, China, in 2018. He is currently working toward the M.S. degree in automation with the University of Science and Technology of China, Hefei, China.

His current research interests include simultaneous localization and mapping and autonomous navigation of mobile robots.

Jiahu Qin (Senior Member, IEEE) received the first Ph.D. degree in control science and engineering from the Harbin Institute of Technology, Harbin, China, in 2012, and the second Ph.D. degree in systems and control from the Australian National University, Canberra, Australia, in 2014.

He is currently a Professor with the Department of Automation, University of Science and Technology of China, Hefei, China. His current research interests include multiagent systems, cyber-physical systems, and complex dynamical networks.

Shuai Wang received the B.E. degree in automation from Northeast Forestry University, Harbin, China, in 2016. He is currently working toward the Ph.D. degree in control science and engineering with the University of Science and Technology of China, Hefei, China.

His current research interests include perception and control in robotics.

Yaonan Wang received the Ph.D. degree in electrical engineering from Hunan University, Changsha, China, in 1994.

He was a Postdoctoral Research Fellow with the National University of Defense Technology, Changsha, from 1994 to 1995. From 1998 to 2000, he was a Senior Humboldt Fellow in Germany. Since 1995, he has been a Professor with the College of Electrical and Information Engineering, Hunan University, Changsha, China. His current research interests include robotics and image processing.

Shi Wang received the Ph.D. degree in engineering and computer science from the Australian National University, Canberra, Australia, in 2014.

During 2013–2014, he was a Postdoctoral Fellow with the National Institute of Informatics, Tokyo, Japan. He is currently an Associate Professor with the College of Electrical and Information Engineering, Hunan University, Changsha, China. His research interests include quantum coherent feedback control, quantum network analysis and synthesis, and multiagent systems.