Available online at www.sciencedirect.com

ScienceDirect

Journal of the Franklin Institute 357 (2020) 3893–3906

www.elsevier.com/locate/jfranklin

Temporal delay estimation of sparse direct visual inertial odometry for mobile robots

Ruping Cen[a], Xinyue Zhang[a], Yulin Tao[a], Fangzheng Xue[a,*], Yuxin Zhang[b]

[a] *College of Automation, Chongqing University, Chongqing 400044, China*
[b] *College of Material Science and Engineering, Chongqing University, Chongqing 400044, China*

Received 9 June 2019; received in revised form 18 November 2019; accepted 25 November 2019
Available online 15 January 2020

Abstract

Most visual inertial odometry(VIO) system can get an ideal result on datasets, which typically use an external device to synchronize measurements between the camera and the inertial measurement unit(IMU). But for an inexpensive homemade Cameras-IMU system, the accuracy of these algorithms usually degrades or even fails because of the temporal misalignment between the camera and the IMU. In this article, we focus on the time synchronization problem in direct VIO system, and we propose a method to calibrate the temporal delay between the camera and the IMU. To achieve this, the temporal delay parameter is added into the state variable of the extended Kalman filter(EKF), and the brightness error of the landmark is used as the error function to update the EKF filter. Finally, we construct a sparse direct VIO system with online temporal delay estimation, and make a comparison with the VINS-mono and the ROVIO. The experiments show that the brightness error of the landmark can be used to accurately calibrate the temporal delay between the image and the IMU measurements in direct VIO system, and the method considering the temporal delay will improve the performance of the direct VIO system.

* Corresponding author.
E-mail address: xuefangzheng@cqu.edu.cn (F. Xue).

https://doi.org/10.1016/j.jfranklin.2019.11.075

1. Introduction

Localization, navigation and control are three basic prerequisites for autonomous robots to fulfill missions automatically in an unknown environment. Over the past decades, the robot control has been a hot research topic and has achieved abundant achievements[1–4] in the field of motion control and path tracking [5]. Focused on synchronization positioning and map construction, the SLAM (Simultaneous Localization and Mapping) allows robots to start from unknown locations in an unknown environment, locate their position and pose by repeatedly observing the environment during movement, and build an incremental map of the environmental, so as to achieve simultaneous localization and map building.

Since this concept was proposed by Smith [6] in 1986, the sensor of the SLAM system has developed from the sonar to laser [7], monocular [8–10], stereo [11,12], RGBD [13] and the combined system of the camera and the IMU [14,15]. There are inherent scale problems in monocular vision SLAM systems, and Mur-Artal et al. [14] show that the algorithms with only visual measurements fail when the camera moves fast or the texture information of the environment is insufficient. Therefore, such algorithms usually lack of robustness. Leutenegger et al. [16] employ an IMU as an additional sensor to improve both the robustness and the accuracy of the system. The VIO fuses data from vision sensors with those from an IMU to estimate the position and orientation of a camera. This sensor pair is ideal, since it has inherent complementary advantages [17]. Specifically, the IMU sensors are sensitive to vibration, so the IMU can accurately respond to the rapid movement of the robot, but the measurements of the IMU will drift over a long period of time [18]. Besides, although the camera cannot cope with fast motion, the image does not drift. Therefore, the camera can be used to correct the drift of the IMU measurements, and the IMU can be used to improve the ability of fast movement of the camera.

Sensor unsynchronization is a common problem in multi-sensor fusion systems. In the VIO system, although the external triggering strategy can be used to synchronize the image and IMU measurements on the hardware, the time offset between the sensors still exists due to data sampling, transmission and operating system response delay. And this parameter of time offset is not a constant but a variable which changes over time. The existing works based on the IMU have addressed a variety of issues in visual inertial odometry systems, such as improving the robustness of the system with sliding window frame work [19],calibrating the external parameters between the camera and the IMU online [15,20] or the internal parameters of cameras [21], solving the inconsistency of the EKF estimator [22] and increasing the particle depletion of UKF-SLAM [23]. However, there are few works focusing on modelling the delay time between image data and IMU measurements. The classic VIO or VIO-SLAM framework (such as VI-ORB [14], ROVIO [24]) processes the camera and the IMU measurements when the time offset is known in advance or the time synchronization is not considered. So these algorithms can only work in a dataset or a visual-inertial system using an external synchronization trigger device which is usually expensive.

For the estimation of sensor delay in the VIO system. The first researchers who study the problem of time-offset calibration in VIO systems are Kelly and Sukhatme [20,25]. They estimate the rotation parameter from each individual sensor at first, and then estimate the delay via Time Delay Iterative Closest Point (TD-ICP), which only uses orientation information for cost functions. But this works as a principled offline method.

Choi et al. [26] employs the Probability Density Function (PDF) to model the uncertainty of delay, and uses the probability distribution of delay to solve the uncertainty of time delay

parameters. Merwe et al. [27] solve the problem with almost the same formulation but the sigma-point Kalman filter is used instead of EKF. Tungadi [28] estimate the delay between the laser and the odometric measurements by minimizing the hysteresis in positioning data over closed-loop trajectories. However, the noise is not considered in laser or odometric measurements. These works estimate the delay between sensors offline and they believe that the delay parameter is constant. However, the actually delay parameter varies with the computing environment.

Moreover, Li [29] proposes an online approach to estimate the time delay between the measurements of the camera and the IMU. The delay parameters are treated as an additional state of the EKF, and the reprojection errors of landmark is used to update of the time offset between sensors in real-time. The experiment shows that the time delay will be stable after a period of time, and the result proves that the delay parameters are observable. In addition, Qin [30] also proposes an online approach to calibrate temporal offset between the camera and the IMU measurements, which assumes that the time delay parameter is unknown and constant. Specifically, the author shifts forward or backwards the features according to the time offset $\boldsymbol{td}$ and models the feature reprojection error as a vision factor in the cost function.

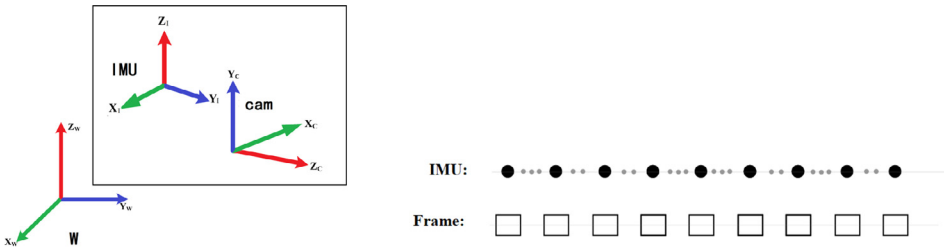
Our algorithm is different from MSCKF [29,31] and compared with which there are solid improvements in this paper. The existing works [29,30] are based on sparse features, however we use the sparse direct method for the temporal calibration. The direct method is based on the assumption of image gray value inconvenience, which directly uses pixel gay values instead of feature descriptors to establish the connection between the two views. The advantage of the direct methods is that they do not require extracting and matching descriptors, which are computationally expensive. The main contributions of this paper are now listed as follows:

(1) We provide the evidence that the unsynchronization between the camera and the IMU measurements can cause performance degradation of the sparse direct VIO system, but the temporal delay parameter can be calibrated by the brightness error of the landmark.
(2) We propose a temporal delay estimate model based on the brightness error of landmark for sparse direct VIO system, and a detailed derivation of the model is given.
(3) We construct a sparse direct VIO system with online temporal delay estimation model and compare it with the VINS-mono [32] and the ROVIO. The results show that the temporal delay value can be accurately estimated by the brightness error of the landmark, and the approach considering the temporal delay between the camera and IMU measurements will bring the improvement of the performance for the sparse direct VIO system.

Our paper is organized as follows. Section 2 demonstrates the approach in mathematical form. Section 3 show the results of our system on EUROC dataset.

2. Methodology

In this section, the theory of our algorithm is addressed. The algorithm can be mainly divided into three parts. Firstly, the measurement model of the IMU is described as a combination of constant values and the noise, which is modelled as a time-varying parameter with random walk characteristics. Secondly, the coarse pose of the IMU is estimated based on the kinematic model. Thirdly, the error function is constructed from the intensity error of the landmark point between two views, which is used to update the state variable of the EKF.



(a) IMU and camera coordinate (b) The rates for IMU and Frame

Fig. 1. IMU and Camera rigid coordinate relationship, the ***Tcb*** denotes the rotation and translation between IMU coordinate and camera coordinate.

2.1. IMU model and the system states

The IMU sensor contains a 3D gyroscope and a 3D linear accelerometer. Usually, the accelerometer and gyro suffer from white noise and a slowly varying bias [33]. With these, the IMU measurements modelled as follows:

$$\begin{aligned} \tilde{w_b}(t) &= w_b(t) + b^w(t) + n(w) \\ \tilde{a_b}(t) &= R_{wb}^T(a_w(t) - g) + b^a(t) + n(a) \end{aligned} \tag{1}$$

where $\tilde{w_b}(t) \in \mathbb{R}^3$ and $\tilde{a_b}(t) \in \mathbb{R}^3$ are the measurements of rotational velocity and linear accelerations expressed in IMU body coordinate, $w_b(t) \in \mathbb{R}^3$ is the true value of the angular velocity expressed in the body coordinate frame, $a_w(t) \in \mathbb{R}^3$ is the acceleration of the sensor in the earth coordinate frame, the rotation matrix $R_{wb} \in \mathbb{R}^{3\times3}$ maps points from the body coordinate $\boldsymbol{B}$ to earth coordinate $\boldsymbol{W}$, and g is the gravity vector. The $b^w(t) \in \mathbb{R}^3$ and $a^w(t) \in \mathbb{R}^3$ are slowly varying bias, which can be described as follows:

$$\begin{aligned} b^w(t+1) &= b^w(t) + \eta^w(t) \\ b^a(t+1) &= b^a(t) + \eta^a(t) \end{aligned} \tag{2}$$

where $\eta^a(t)$, $\eta^w(t)$, $n(a)$ and $n(w) \in \mathbb{R}^3$ are white noises. We define the IMU state $\mathbf{X}_{imu}$ as a 17×1 vector, which includes the 3D IMU pose, linear velocity of IMU v_w in the global coordinate $\boldsymbol{W}$, and the time-varying IMU bias (b^a and b^w).

$$\mathbf{X}_{imu} = \begin{bmatrix} \bar{q}_{wb} & p_w & v_w & b^a & b^w & t_d \end{bmatrix}^T_{17\times1} \tag{3}$$

The $\bar{q}_{wb}$ is a unit quaternion that describes the rotation in the IMU coordinate $\boldsymbol{B}$ with respect to the global coordinate $\boldsymbol{W}$, p_w and v_w are the IMU position and velocity information in the global coordinate $\boldsymbol{W}$, b^a and b^w are the biases of IMU, and the t_d is the time delay between the IMU and the camera.

By the way, the camera used in this system is rigidly attached with IMU, and the rotation and translation matrix between the camera and the IMU is defined as $T_{bc} = [R_{bc}|t_{bc}]$. The coordinate of camera and IMU can be seen in Fig. 1a, and the overview of the camera and the IMU synchro diagram can be found in Fig. 1b.

2.2. Propagation model

Assuming the IMU measurements remain constant in the interval time $\triangle t$. According to the kinematic model, the position and velocity of the IMU in the earth coordinate frame $\boldsymbol{W}$ from time k to $k+1$ are inferred as:

$$\begin{aligned}
\dot{q_{bw}}(k+1) &= \frac{1}{2}\Omega(\breve{w}_k)\bar{q_{bw}}(k) \\
\dot{P_w}(k+1) &= v_w(k) \\
\dot{v_w}(k+1) &= R_{wb}(\bar{q})\bar{a}(k) \\
\dot{b^a}(k+1) &= n(a) \\
\dot{b^w}(k+1) &= n(w) \\
\dot{t^d}(k+1) &= n(t_d)
\end{aligned} \tag{4}$$

where the $\Omega(.)$ is an operator which changes the gyro measurements vector $\breve{w}_k$ into a matrix:

$$\Omega(w) = \begin{bmatrix} -w^{\times} & w \\ -w^T & 0 \end{bmatrix}_{4\times 4}, \ where \ w^{\times} = \begin{bmatrix} 0 & -w_z & w_y \\ -w_x & 0 & w_z \\ w_y & -w_x & 0 \end{bmatrix} \tag{5}$$

while $\breve{w}_k = \tilde{w_b}(k) - b^w(k)$ and $\breve{a}_k = \tilde{a_b}(k) - b^a(k)$ denote the kth values of rotation speed and acceleration of IMU sensor. Similarly, we follow the approach [34] of quaternion error to update quaternion error.

$$\bar{\delta q} = \bar{q} \otimes \hat{\bar{q}}^{-1} \simeq \begin{bmatrix} \frac{1}{2}\delta\theta^T & 1 \end{bmatrix}^T \tag{6}$$

From the above expressions, the denotes the small rotation. This approach allows us to describe the pose disturbance by a dimensional vector. The system states error can be rewritten as:

$$\tilde{X}_{imu} = \begin{bmatrix} \theta_{wb} & p_w & v_w & b^a & b^w & t_d \end{bmatrix}^T_{16\times 1} \tag{7}$$

So the propagation model of IMU states vector is described by:

$$\dot{\tilde{X}}_{imu}(k+1) = F * \tilde{X}_{imu}(k) + G * n_{imu} \tag{8}$$

where n_{imu} is the process noise about the IMU biases.

$$n_{imu} = \begin{bmatrix} n_g^T & n_{wg}^T & n_a^T & n_{wa}^T \end{bmatrix}^T \tag{9}$$

The $\boldsymbol{F}$ is an error-state transform matrix corresponding to the IMU state, and the $\boldsymbol{G}$ is the input noise matrix derived in [31].

$$F = \begin{bmatrix} -w^{\times} & 0_3 & 0_3 & 0_3 & -I_3 & 0 \\ 0_3 & 0_3 & 0_3 & -I_3 & 0_3 & 0 \\ -R_{wb}(\bar{q_k}) * a^{\times} & 0_3 & 0_3 & R_{wb}(\bar{q_k}) & 0_3 & 0 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 & 0 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{10}$$

$$G = \begin{bmatrix} -I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & -R_{wb}(\bar{q}_k) & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \\ 0_3 & I_3 & 0_3 & 0_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{11}$$

Based on the previously derived equation, when the processer receives the IMU measurements, the IMU states are calculated by 4-th order Runge–Kutta numerical integration over Eq. (4). Finally, the state transform matrix $\Phi_{imu}(k)$ and the covariance matrix $P_{II_{kk}}$ of IMU are given as:

$$\begin{aligned} \Phi_{imu}(k) &= exp \int_{t_k}^{t_{k+1}} F\,dt \\ Q_{imu}(k) &= exp \int_{t_k}^{t_{k+1}} Gn_{imu}G^T\,dt \end{aligned} \tag{12}$$

The covariance matrix of the VIO system at step $\boldsymbol{k}$ can be described as:

$$P_{II_{k+1|k}} = \begin{bmatrix} \Phi_{imu}P_{II_{k|k}}\Phi_{imu}^T + Q_{imu} & \Phi_{imu}P_{IC_{k|k}} \\ P_{IC_{k|k}}^T\Phi_{imu}^T & P_{CC_{k|k}} \end{bmatrix} \tag{13}$$

Every time a new image is received, the camera pose can be obtained from the IMU Propagation.

$$\begin{aligned} R_{cw}(k) &= R_{bc}^T * R_{bw}\left(\hat{\bar{q}}_{bw}\right) \\ t_{cw}(k) &= p_w(k) + R_{bw}^T\left(\hat{\bar{q}}_{bw}\right) * t_{bc} \end{aligned} \tag{14}$$

where, $R(q)$ is the rotation matrix corresponding to the quaternion q, R_{bc} and t_{bc} are the external parameters between the camera and the IMU, which can be calibrated by Kalibr toolbox [35].

2.3. EKF update mode

Once we have obtained the estimated pose of camera, we project the feature points in the map into the current image. And the residual of the pixel value between the current frame and reference frame is used to update the EKF filter. This method is called direct image align algorithm, which is based on the assumption that the pixel value is invariant under a small views transform. Compared with the feature point method, this method does not need to calculate and match the descriptor of image features, and the computational cost of this method is low. Therefore, its suitable for robotic systems that require real-time computing.

To simplify the derivation, A signal 3D map point $P_j = [X_j, Y_j, Z_j] \in \mathbb{R}^3$ is considering, which shown in Fig. 2. $R_{C_2C_1}$ is the rotation matrix and $t_{C_2C_1}$ is a translation vector between two views ***C1*** and ***C2***. The project points p_j^1 and p_j^2 can be obtained according to the projection

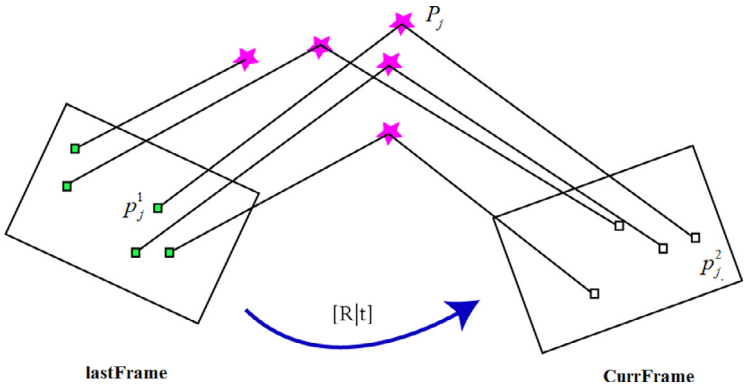


Fig. 2. The 3D map point is projected into the current and the previous view.

relationship:

$$p_j^1 = \begin{bmatrix} u_j^1 \\ v_j^1 \end{bmatrix} = C\frac{1}{Z_j}KP_j = \pi(P_j), where\ C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$
$$p_j^2 = \begin{bmatrix} u_j^2 \\ v_j^2 \end{bmatrix} = C\frac{1}{Z_j}K(R_{C_2C_1}P_j + t_{C_2C_1}) = \pi(R_{C_2C_1}P_j + t_{C_2C_1}) \tag{15}$$

where $\boldsymbol{K}$ is the intrinsic matrix of the camera, which projects 3D map points into the pixel coordinate system, and $\pi(.)$ denotes the projection function.

$$\pi(P_j) = C\frac{1}{Z_j}KP_j \tag{16}$$

Because of the noise, the projection relation cannot be perfectly satisfied, so the photometric error of 3D point P_j in the current frame is given by:

$$r_j = I_1(p_j^1) - I_2(\pi(R_{C_2C_1}P_j + t_{C_2C_1})) \tag{17}$$

where, the I(.) denotes the intensity function of the pixel.

For accuracy, we consider the fact that the image is sampled at the time $\boldsymbol{t}$, but the IMU measurements are sampled at time $\boldsymbol{t+td}$, so the $\boldsymbol{td}$ is a temporal delay between IMU measurements and images. So we need to propagate the IMU pose up to $\boldsymbol{t+td}$ and compute the residual at that point. By using Eqs. (14) and (15), the photometric error of point P_j can be rewritten as:

$$r_j = I_1(p_j^1) - I_2(\pi(R_{bc}^T R_{bw}(\hat{\tilde{q}}_{t+t_d})(P_j - P_w(t+t_d)) + t_{bc})) \tag{18}$$

The system state X as defined as follows, which include the IMU state and N camera pose.

$$X = \begin{bmatrix} X_{imu} | & X_{C1} & X_{C2} & \dots & X_{C1} \end{bmatrix}^T \tag{19}$$

where

$$X_{C1} = \begin{bmatrix} q_{c_iw} & p_{c_iw} \end{bmatrix} \tag{20}$$

The q_{c_iw} and p_{c_iw} are the rotation and translation parameters of the camera C_i. Applying chain derivation rule to Eq. (18), the Jacobian of $r_j(.)$ respects to system states X is given by:

$$H_{\theta,j} = -J_j R_{cb} R_{bw} (\hat{\hat{q}}_{t+t_d}) \lfloor P_j - p_w(t+t_d) \rfloor^{\times} \tag{21}$$

$$H_{p,j} = -J_j R_{cb} R_{bw} (\hat{\hat{q}}_{t+t_d}) \tag{22}$$

$$H_{v,j} = \mathbf{0} \tag{23}$$

$$H_{b^a,j} = \mathbf{0} \tag{24}$$

$$H_{b^w,j} = \mathbf{0} \tag{25}$$

$$H_{t_d^w,j} = -J_j R_{cb} \lfloor \breve{w}(t+t_d) \rfloor^{\times} (P_j - p_w(t+t_d)) - J_j R_{cb} R_{bw} (\hat{\hat{q}}_{t+t_d}) v_w(t+t_d) \tag{26}$$

$$H_{R^{c_iw},j} = J_j \lfloor R(q_{c_iw}) * (\pi^{-1}(P_j^1) - p_{c_iw}) \rfloor^{\times} \tag{27}$$

while the Jacobian matrix J_j is:

$$J_j = \frac{\partial I_2}{\partial p_j} \frac{\partial p_j}{\partial P_j} \tag{28}$$

where $\frac{\partial I_2}{\partial p_j}$ is image gradient and $\frac{\partial p_j}{\partial P_j}$ is described as:

$$\frac{\partial p_j}{\partial P_j} = \begin{bmatrix} \frac{f_x}{Z_j} & 0 & \frac{-f_x X_j}{Z_j^2} \\ 0 & \frac{f_y}{Z_j} & \frac{-f_y Y_j}{Z_j^2} \end{bmatrix} \tag{29}$$

So, the Jacobian matrix H is given by:

$$H = \begin{bmatrix} H_{imu} & \dots & H_{cam_i} \dots \end{bmatrix} \tag{30}$$

where

$$H_{imu} = \begin{bmatrix} H_{\theta,j} & H_{p,j} & \mathbf{0} & \mathbf{0} & \mathbf{0} & H_{t_d,j} \end{bmatrix}_{1\times 16} \tag{31}$$

$$H_{cam_i} = \begin{bmatrix} H_{R_{c_iw},j} & H_{t_{c_iw},j} \end{bmatrix}_{1\times 6} \tag{32}$$

When a new image is recorded, the state of the IMU is updated and the pose of the current frame is updated by Eq. (14). And then, the oldest state of the camera is rejected, and adding the current camera state into the system state X. Finally, the system states and covariance matrix are updated at the time ***t+td***, with the standard measurement update procedure.

$$X_{k|k} = X_{k|k-1} + \sum_j K_j r_j \tag{33}$$

$$P_{k|k} = P_{k|k-1} - \sum_j K_j H_j P_{k|k-1} \tag{34}$$

$$K_j = P_{k|k-1} H_j^T (H_j P_{k|k-1} H_j^T + R_j)^{-1} \tag{35}$$

Where, the ***j*** is the index of the features which are observed in the current views. By the way, we assume that the depth of the features in the reference frame is known in the above derivation. However, the RGB camera cannot measure the depth of the pixel directly. So the new landmarks are still detected in every image, and the initial distance can be computed by triangulation. When this feature is observed by the camera in the future, the LM algorithm is employed to further optimize the depth of the landmark.

3. Experiments

In this part, we separately evaluated ROVIO and our method on the RUROC benchmark dataset [36], which contained hardware-synchronized stereo images and IMU measurements. The RUROC dataset was recorded in a machine hall at ETH Zurich, and the ground truth of the dataset was provided by the VICON motion capture device and Leica-Multistation system. The dataset included 11 sequences, and the entire dataset contained different scenes, such as camera rapid rotation, fast move, low light scenes, low textures, and structure scene. These scenes were usually used to evaluate the performance of SLAM or VIO systems. By the way, the system processor is a laptop with an Intel CPU i7-4710MQ (four cores @ 2.50 GHz) and 8 GB RAM, and all experimental results were obtained on this platform. In addition, the experimental code is modified from ROVIO [37].

3.1. The influence of temporal parameters

Firstly, we generated the '**Offset-Data**' on MH_04_difficult sequence in EUROC datasets and evaluated the ROVIO system on new datasets for exploring the influence of the temporal delay parameter ***td***. The ROVIO is a robust VIO system, which directly used intensity errors of image patches to algin image. But this VIO system without modelling the time offset parameters between images and IMU measurements. In addition, the '**Offset-Data**' was generated by manually shift the timestamp of the IMU measurements on MH_04_difficult from ***t*** to ***t+td***. Where the $td > 0$ denoted the IMU measurements are ahead of the images, and the $td < 0$ denoted the IMU measurements lag behind images. The results are shown in Fig. 3.

Fig. 3a shows the trajectory of the ROVIO system on '**Offset-Data**' with different ***td*** value. The black curve is the ground truth obtained from VICON system, and the red, green, blue, pink and sky blue curves denote 0 ms, −5 ms, −10 ms, −15 ms, −20 ms temporal delay respectively. Moreover, It can be clearly observed that the trajectory of the camera was farther away from the ground truth with increased the temporal delay parameter, although the robot started from the same point. More interestingly, the temporal delay parameter also affects the initialization of the monocular VIO. With the increase of ***td***, the position in the initialization phase also shifts the truth trajectory.

Secondly, to further describe the relationship between RMSE and temporal delay of VIO systems, Fig. 3b show the RMSE of ROVIO, VINS and our method on the '**Offset-Data**' based for scenes when only ***td*** changes. It can be easily noticed that the RMSE of the ROVIO

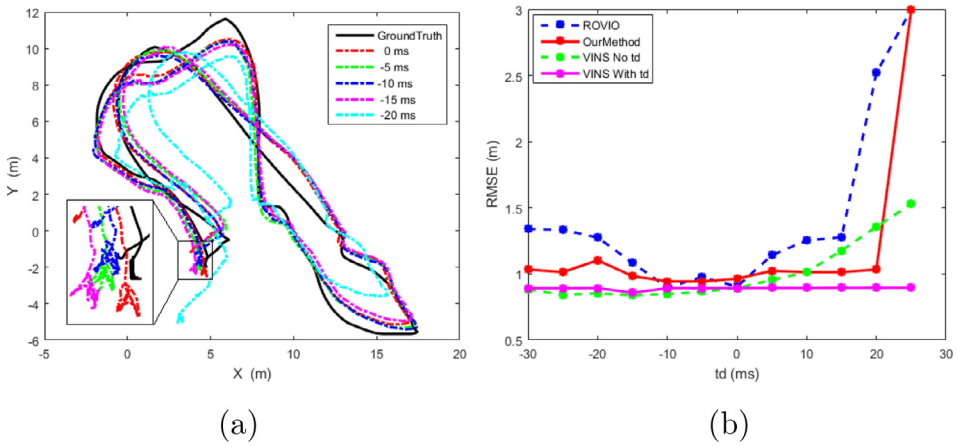


Fig. 3. The impacts of temporal delay parameters. (a) The Trajectory of ROVIO System with a time offset td on MH_04_difficult sequence. (b) The RMSE of ROVIO, VINS and our method with a different time offset td on MH_04_difficult sequence.

system increased with the temporal delay parameters, and similar trends can be found in VINS-No-td. This is mainly because integrating the IMU measurements gives the VIO system an overshooting camera pose. In the VIO system, the IMU measurements are typically integrated as an initial estimate of camera motion. If the IMU measurements are not synchronized with the camera in time, the IMU measurements will reflect an overshoot or negative overshoot motion, which causes a bias in the image matching. So temporal delay has a greater impact on direct image alignment method than feature method. This assumption can be confirmed by the RMSE curves of VINS no-td and ROVIO in Fig. 3b. Unexpectedly, Three systems had a better robust tolerance for IMU measurements lag ($\boldsymbol{td} < 0$) in Fig. 3b, and the system failed to estimate the camera pose when the delay parameter $\boldsymbol{td} > 25$ ms.

Finally, these results provide substantial evidence for the original assumptions that the unsynchronization between the images and IMU measurements can cause performance degradation of the VIO system, and the time delay estimation model can significantly improve the performance of the VIO system.

3.2. System performance in EUROC dataset

To further evaluate the performance of our method, we compared with the VINS and ROVIO system on the new dataset. The new dataset was obtained by manually shift the timestamp of the IMU measurements on all sequence of EUROC dataset with $\boldsymbol{td}$ equal 5 ms. It is due to that the frequency of the camera and IMU are usually 20 Hz and 200 Hz, respectively, so the maximum of the temporal delay between the IMU and the camera should be less than 5 ms. And the results were shown in Fig. 4. Fig. 4a showed the trajectories of ROVIO, VINS and our method on MH_03_medium sequence. The blue, green and red curves are the ROVIO, VINS and our method, respectively. In addition, the black curve showed the ground truth value of camera provided by VICON device. For clarity, Fig. 4b plotted the error curve of the position estimation relative to the true value during the tracking process. It could be clearly seen that the position tracking error of our algorithm is smaller than the ROVIO in

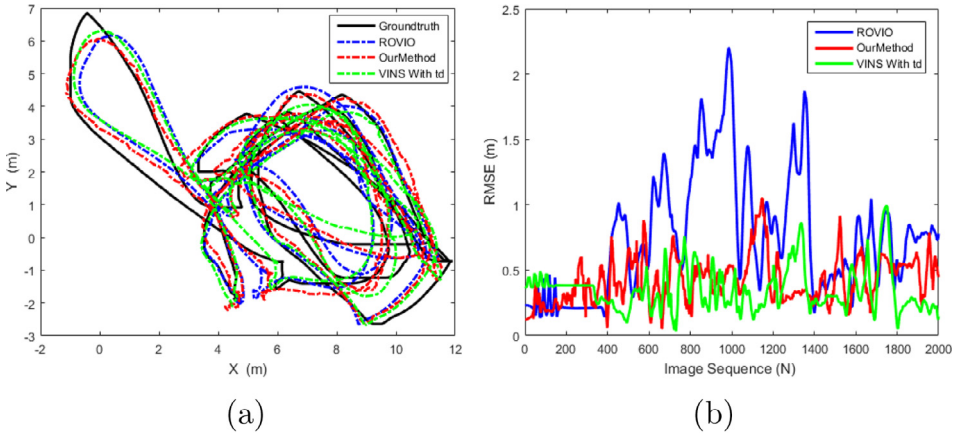


Fig. 4. (a) Trajectory in MH_03_medium sequence, compared with ROVIO. (b) The trajectory error curve of ROVIO, VINS and our method on the MH_03_medium sequence (***td*** = 5 ms).

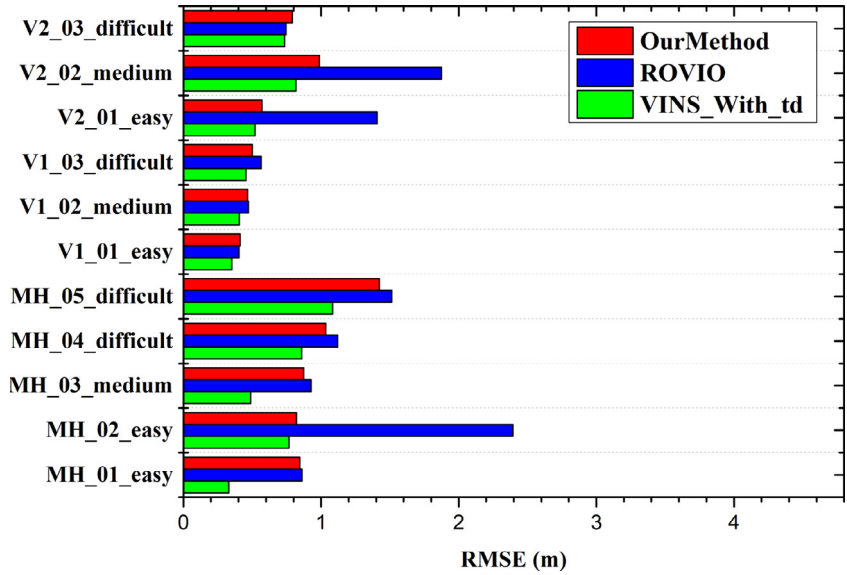


Fig. 5. The RMSE of ROVIO, VINS and our method and on the EUROC dataset with ***td*** = 5 ms.

the whole process on MH_03_medium sequence with ***td*** equal to 5 ms. Thus, the trajectory estimated by our method nicely fitted the ground truth than ROVIO, which indicated that our algorithm is more robust and smooth than ROVIO. Because once the time delay parameter ***td*** is considered, a more precise camera pose can be calculated by integrating IMU measurements during state augmentation. So, the quality of image matching will be improved with precise initial camera pose, which in turns a better tracking.

For generalization performance, the trajectory RMSE for each sequence on new EUROC dataset with ***td*** equal to 5 ms shown in Fig. 5. The method proposed in this paper outperformed the ROVIO system in most cases which can be seen in Fig. 5. However, the VINS-with-td

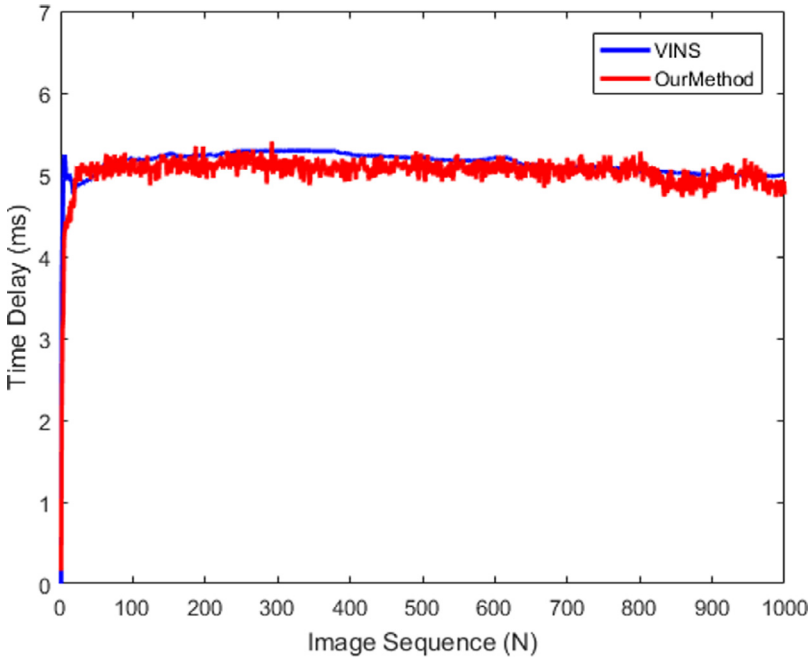


Fig. 6. The temporal delay estimated value on MH_04_difficult.

maintains excellent performance on new datasets, which confirms that VIO systems based on feature points are more robust than direct-method VIO systems when there is illumination variation in the environment. So the photometric error calibration model and the multi-state observation constraints may be a good way to further improve our system robustness.

Fig. 6 showed the estimated value of temporal delay on MH_04 sequence with ***td*** equal 5 ms. In Fig. 6, both VINS and our methods can accurately estimate the time offset between sensors, this indicates that the brightness error of the landmark can be used to calibrate the temporal delay between the image and the IMU measurements in direct VIO system. However, the temporal delay curve estimated by our algorithm has a larger variance than VINS-Mono, which may be due to the error introduced by the direct method in the image matching step. As well know, there are some regions with illumination variations on the MH_04 sequence, which will break the hypothesis of intensity invariant in the direct method. So the photometric camera calibration model can be used to further improve the performance of the algorithm. Besides, when the system runs for a period of time, the estimated value of ***td*** begins to deviate, which may be due to the cumulative error in the image tracking process.

To summarize, it is clear that the impact of the temporal delay parameter ***td*** on VIO system is obvious, so the temporal delay parameter cannot be ignored directly. In addition, the temporal delay between different sensors is inherent and inevitable, and the temporal delay is time-varying and varies on different hardware platforms due to the operating system's response and data transfer operations. However, the temporal delay parameter can be calibrated by the brightness error of the landmark, and using the temporal delay model significantly improve the performance of the VIO system, especially for the VIO systems based on direct methods.

So this approach can be applied to the visual and the inertial sensor fusion system without hardware synchronization device.

4. Conclusion

In this article, we focus on the problem of the time synchronization between the camera and the IMU. We propose a method that calibrates the temporal delay between the camera and the IMU measurements by using intensity error of landmark directly, which can significantly improve the performance of direct VIO systems. Moreover, we conduct a comparative experiment with Vins-mono and ROVIO on the new dataset by manually shift the time stamp of the IMU measurements on EUROC dataset, which simulates the problem of time asynchronism in self-made cameras. The experiments show that the temporal delay between the images and IMU measurements can cause a decrease in the performance of the VIO system. So modelling temporal delay are necessary for multi-sensor fusion systems, and the approach considering the temporal delay between the camera and IMU measurements will bring about the improvement of the performance for the VIO system. To some extents, this work can be seen as an improvement of the direct VIO system, and this approach can be applied to the visual and the inertial sensor fusion system without hardware synchronization device. It is important to develop the visual-inertial fusion system for robot localization and navigation.

Finally, we are interested in time consistency in multi-sensor fusion system. As a next step, we plan to further improve the performance of the VIO systems based on direct methods by using a photometric error model. And we plan to derive a mathematical model for online estimation of time-delay in a multi-sensor data fusion system and to mathematically prove the observability of the delay parameters.

References

[1] M.V. Basin, P.C.R. Ramírez, F. Guerra-Avellaneda, Continuous fixed-time controller design for mechatronic systems with incomplete measurements, IEEE/ASME Trans. Mechatron. 23 (1) (2018) 57–67.
[2] N. Karaboga, A new design method based on artificial bee colony algorithm for digital IIR filters, J. Frankl. Inst. 346 (4) (2009) 328–348.
[3] M. Basin, J.J. Maldonado, Optimal mean-square state and parameter estimation for stochastic linear systems with poisson noises, Inf. Sci. 197 (2012) 177–186.
[4] Y. Bian, J. Peng, C. Han, Finite-time control for nonholonomic mobile robot by brain emotional learning-based intelligent controller, Int. J. Innov. Comput. Inf. Control 14 (2018) 683–695.
[5] Y. Wang, U. Yang, S. Wang, Path tracking control of an indoor transportation robot utilizing future information of the desired trajectory, Int. J. Innov. Comput. Inf. Control 14 (2) (2018) 561–572.
[6] R.C. Smith, P. Cheeseman, On the representation and estimation of spatial uncertainty, Int. J. Robot. Res. 5 (4) (1986) 56–68.
[7] X. Xiong, A. Adan, B. Akinci, D. Huber, Automatic creation of semantically rich 3d building models from laser scanner data, Autom. constr. 31 (2013) 325–337.
[8] S.S. Mehta, C. Ton, Z. Kan, J.W. Curtis, Vision-based navigation and guidance of a sensorless missile, J. Frankl. Inst. 352 (12) (2015) 5569–5598.
[9] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, D. Scaramuzza, SVO: semidirect visual odometry for monocular and multicamera systems, IEEE Trans. Robot. 33 (2) (2016) 249–265.
[10] A.J. Davison, I.D. Reid, N.D. Molton, O. Stasse, MonoSLAM: real-time single camera SLAM, in: IEEE Transactions on Pattern Analysis & Machine Intelligence, volume 6, 2007, pp. 1052–1067.
[11] R. Mur-Artal, J.D. Tardós, Orb-slam2: an open-source slam system for monocular, stereo, and rgb-d cameras, IEEE Trans. Robot. 33 (5) (2017) 1255–1262.
[12] K. Sun, K. Mohta, B. Pfrommer, M. Watterson, S. Liu, Y. Mulgaonkar, C.J. Taylor, V. Kumar, Robust stereo visual inertial odometry for fast autonomous flight, IEEE Robot. Autom. Lett. 3 (2) (2018) 965–972.

[13] P. Henry, M. Krainin, E. Herbst, X. Ren, D. Fox, RGB-d mapping: using kinect-style depth cameras for dense 3d modeling of indoor environments, Int. J. Robot. Res. 31 (5) (2012) 647–663.
[14] R. Mur-Artal, J.D. Tardós, Visual-inertial monocular SLAM with map reuse, IEEE Robot. Autom. Lett. 2 (2) (2017) 796–803.
[15] Z. Yang, S. Shen, Monocular visual-inertial state estimation with online initialization and camera-IMU extrinsic calibration, IEEE Trans. Autom. Sci. Eng. 14 (1) (2016) 39–51.
[16] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, P. Furgale, Keyframe-based visual-inertial odometry using nonlinear optimization, Int. J. Robot. Res. 34 (3) (2015) 314–334.
[17] C. Forster, L. Carlone, F. Dellaert, D. Scaramuzza, On-manifold preintegration for real-time visual-inertial odometry, IEEE Trans. Robot. 33 (1) (2016) 1–21.
[18] M.F. Abdel-Hafez, On the development of an inertial navigation error-budget system, J. Frankl. Inst. 348 (1) (2011) 24–44.
[19] J. Engel, V. Koltun, D. Cremers, Direct sparse odometry, IEEE Trans. Pattern Anal. Mach. Intell. 40 (3) (2017) 611–625.
[20] J. Kelly, G.S. Sukhatme, Visual-inertial sensor fusion: localization, mapping and sensor-to-sensor self-calibration, Int. J. Robot. Res. 30 (1) (2011) 56–79.
[21] G. Seetharaman, H. Bao, G. Shivaram, Calibration of camera parameters using vanishing points, J. Frankl. Inst. 331 (5) (1994) 555–585.
[22] J.A. Hesch, D.G. Kottas, S.L. Bowman, S.I. Roumeliotis, Consistency analysis and improvement of vision-aided inertial navigation, IEEE Trans. Robot. 30 (1) (2013) 158–176.
[23] R. Havangi, Unscented h-infinity filtering based simultaneous localization and mapping with evolutionary re-sampling, J. Frankl. Inst. 352 (11) (2015) 4801–4825.
[24] M. Bloesch, M. Burri, S. Omari, M. Hutter, R. Siegwart, Iterated extended kalman filter based visual-inertial odometry using direct photometric feedback, Int. J. Robot. Res. 36 (10) (2017) 1053–1072.
[25] J. Kelly, G.S. Sukhatme, A general framework for temporal calibration of multiple proprioceptive and exteroceptive sensors, in: Experimental Robotics, Springer, 2014, pp. 195–209.
[26] M. Choi, J. Choi, J. Park, W.K. Chung, State estimation with delayed measurements considering uncertainty of time delay, in: 2009 IEEE International Conference on Robotics and Automation, IEEE, 2009, pp. 3987–3992.
[27] R.V.D. Merwe, E. Wan, S. Julier, Sigma-point kalman filters for nonlinear estimation and sensor-fusion: applications to integrated navigation, in: AIAA Guidance, Navigation, and Control Conference and Exhibit, 2004, p. 5120.
[28] F. Tungadi, L. Kleeman, Time synchronisation and calibration of odometry and range sensors for high-speed mobile robot mapping, in: Proceedings of the Australasian Conference on Robotics and Automation, 2008.
[29] M. Li, A.I. Mourikis, 3-d motion estimation and online temporal calibration for camera-IMU systems, in: 2013 IEEE International Conference on Robotics and Automation, IEEE, 2013, pp. 5709–5716.
[30] T. Qin, S. Shen, Online temporal calibration for monocular visual-inertial systems, in: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2018, pp. 3662–3669.
[31] A.I. Mourikis, S.I. Roumeliotis, A multi-state constraint kalman filter for vision-aided inertial navigation, in: Proceedings 2007 IEEE International Conference on Robotics and Automation, IEEE, 2007, pp. 3565–3572.
[32] T. Qin, P. Li, S. Shen, Vins-mono: a robust and versatile monocular visual-inertial state estimator, IEEE Trans. Robot. 34 (4) (2018) 1004–1020.
[33] D. Tedaldi, A. Pretto, E. Menegatti, A robust and easy to implement method for IMU calibration without external equipments, in: 2014 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2014, pp. 3042–3049.
[34] M. Li, A.I. Mourikis, High-precision, consistent EKF-based visual-inertial odometry, Int. J. Robot. Res. 32 (6) (2013) 690–711.
[35] J. Rehder, J. Nikolic, T. Schneider, T. Hinzmann, R. Siegwart, Extending kalibr: calibrating the extrinsics of multiple IMUs and of individual axes, in: 2016 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2016, pp. 4304–4311.
[36] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M.W. Achtelik, R. Siegwart, The euroc micro aerial vehicle datasets, Int. J. Robot. Res. 35 (10) (2016) 1157–1163.
[37] M. Bloesch, S. Omari, M. Hutter, R. Siegwart, Robust visual inertial odometry using a direct EKFbased approach, in: 2015 IEEE/RSJ international Conference on Intelligent Robots and Systems (IROS), IEEE, 2015, pp. 298–304.