Industrial Robot: An International Journal

Visual-inertial SLAM method based on optical flow in a GPS-denied environment

Chang Chen, Hua Zhu,

Visual-inertial SLAM method based on optical flow in a GPS-denied environment

Chang Chen and Hua Zhu
University School of Mechatronic Engineering, China University of Mining and Technology, Xuzhou, China, and
Jiangsu Collaborative Innovation Center of Intelligent Mining Equipment, Xuzhou, China

Abstract

Purpose – This study aims to present a visual-inertial simultaneous localization and mapping (SLAM) method for accurate positioning and navigation of mobile robots in the event of global positioning system (GPS) signal failure in buildings, trees and other obstacles.

Design/methodology/approach – In this framework, a feature extraction method distributes features on the image under texture-less scenes. The assumption of constant luminosity is improved, and the features are tracked by the optical flow to enhance the stability of the system. The camera data and inertial measurement unit data are tightly coupled to estimate the pose by nonlinear optimization.

Findings – The method is successfully performed on the mobile robot and steadily extracts the features on low texture environments and tracks features. The end-to-end error is 1.375 m with respect to the total length of 762 m. The authors achieve better relative pose error, scale and CPU load than ORB-SLAM2 on EuRoC data sets.

Originality/value – The main contribution of this study is the theoretical derivation and experimental application of a new visual-inertial SLAM method that has excellent accuracy and stability on weak texture scenes.

Keywords Mobile robots, Nonlinear optimization, Optical flow method, Tightly-coupled, Visual-inertial SLAM

Paper type Research paper

1. Introduction

Simultaneous localization and mapping (SLAM) is first applied to robotics with the goal of creating a map of the surroundings in real-time based on sensor data in an unknown environment (Yousif *et al.*, 2015; Gui *et al.*, 2015). It has been successfully applied to mobile robotics, self-driving cars, unmanned aerial vehicles, autonomous underwater vehicles and virtual and augmented reality fields.

Under traditional approaches, using GPS for localization and navigation is the common method. However, satellite signals cannot always be received because of the buildings, trees and walls, as the robot moves autonomously in the city or indoors. In this case, sensor fusions are commonly used to instead of GPS. Among different sensor modalities, visual-inertial setups provide a cheap solution with great potential. This integration not only can replace GPS for location but also can establish a three-dimensional environment map. Nevertheless, the camera and inertial measurement unit (IMU) have disadvantages, as the camera image is indistinct and contains much noise under fast motion. The accelerometer and gyroscope of the IMU have a bias, and the data reliability is truncated at initialization. It is challenging to accurately locate the target in large-scale environments and successfully apply to the mobile robot.

In this study, we propose a new visual-inertial SLAM method, which uses the camera and IMU for tight coupling. The front-end uses the optical flow to track the motion and the back-end adopts nonlinear optimization. A feature extraction method is proposed to distribute features on the image under the textureless scenes. It improves the assumption of constant luminosity to improve the robustness. The visual-inertial SLAM system is successfully performed on the mobile robot and achieves excellent accuracy. Compared with ORB-SLAM2[1] (Mur-Artal and Tardós, 2016) on EuRoC data sets, our method has better relative pose error, scale and CPU load.

2. Related works

At present, the camera and IMU integration SLAM is generally divided into loosely-coupled approaches and tightly-coupled approaches (Martinelli, 2014). Historically, the visual-inertial pose estimation problem has been addressed with filtering. Weiss *et al.* (2011) offered a micro-aircraft position and pose estimation based on monocular camera and IMU in 2011. Relative position and pose were calculated completely in a black box that matches image features and was not dependent on IMU. However, pose estimation is fused with the IMU as a state vector input filter framework only when it was obtained by visual odometry. Calculations were small, but those could not obtain higher localization accuracy. The investigation of Indelman *et al.* (2012) showed a navigation system achieved

The current issue and full text archive of this journal is available on Emerald Insight at: **www.emeraldinsight.com/0143-991X.htm**

Industrial Robot: An International Journal
© Emerald Publishing Limited [ISSN 0143-991X]
[DOI 10.1108/IR-01-2018-0002]

The authors would like to gratefully acknowledge the funding support from the National 863 Program of China (no. 2012AA041504) and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

Received 4 January 2018
Revised 12 February 2018
4 March 2018
Accepted 12 March 2018

valuable results by fusing various sensors, stereo cameras, IMU and GPS. This method was a classic visual-inertial SLAM. One of the limits of this study was that it could not accurately meet the current robot positioning needs. Falquez *et al.* (2016) argued for a dense visual odometry integrating RGBD camera and IMU in 2016. The pose-shift caused by the visual odometry was added to the IMU optimization framework to achieve a good localization. However, the RGBD camera could not be used for accurate localization in a wide range of scenes and outdoors, which was susceptible to greater interference.

In recent years, more studies on the tightly-coupled approaches have been conducted. Mourikis and Roumeliotis (2007) put forward a fusion SLAM based on the EKF filter between monocular camera and IMU, which performed linear calculation of features and precise pose estimation in large-scale scenes. However, this work did not consider localization in strong rotation and weak texture scenarios. Leutenegger *et al.* (2015) showed a keyframe based tightly-coupled visual-inertial odometry, which became the representative of keyframe-based visual-inertial odometry methods. The front-end uses a binary robust invariant scalable key point (BRISK) descriptor for feature matching; the back-end uses the sliding window for nonlinear optimization. However, calculating features and descriptors are more computationally intensive, and the real-time performance of the system was weaker. Lin *et al.* (2017) presented a visual-inertial odometry suitable for various environments. The optical flow served to track features and the loop closure thread was used to strengthen the system performance. However, the same positioning effect could not be obtained in the mobile robot. Concha *et al.* (2016) published a direct and close-coupled visual-inertial odometry that directly matches the image pixels to optimize the photometric error. Even though the information utilization is high, processing speed is slower.

The remainder of this paper is organized as follows: we describe the notation and definitions of the study in Section 3. In Section 4, we describe the construction of the visual-inertial SLAM, then present experiments and results in Section 5. The conclusion is drawn in Section 6.

3. Notation and definitions

We use the following notation throughout this work. $(\cdot)^w$ is the world frame, and the gravity direction is consistent with the z-axis direction. $(\cdot)^b$ is the body frame aligned with the IMU. $(\cdot)^c$ is the camera frame. The classic pinhole camera model (Hartley and Zisserman, 2003) is used to transform the 3D space point $\mathrm{X}^c \in \mathrm{R}^3$ under the camera frame to the 2D point $\mathrm{x}^c \in \Omega \subset R^2$ under the image plane, to construct the projection function π: $\mathrm{R}^3 \in \Omega$:

$$\pi(\mathrm{X}^c) = \begin{bmatrix} f_u \dfrac{X^c}{Z^c} + c_u \\ f_v \dfrac{Y^c}{Z^c} + c_v \end{bmatrix} \quad (1)$$

where $\mathrm{X}^c = [X^c\ Y^c\ Z^c]$, $[f_u\ f_v]^T$ is the focal length of the camera and $[c_u\ c_v]^T$ is the camera's projection center.

The IMU contains a three-axis accelerometer and a three-axis gyroscope, with $a(t)$ representing IMU acceleration, $w(t)$ representing IMU angular velocity, $b_a(t)$ an accelerometer bias and $b_g(t)$ a gyroscope bias. In visual-inertial SLAM system, the world frame, the inertial frame and the camera frame have a rotational transformation relationship. The coordinate transformation between the camera and the IMU can be expressed as $\mathrm{T}_{cb} = [R_{cb} \mid p_{cb}]$, which can be obtained in advance by a multisensory calibration system (Yang and Shen, 2017; Furgale *et al.*, 2013). $_{mea}(\cdot)$ represents the measured value of the data. For ease of notation, exponents and logarithms are expressed as follows:

$$\mathrm{Exp}:\ \mathrm{R}^3 \rightarrow \mathrm{SO}(3)\ ;\ \mathrm{Log}:\ \mathrm{SO}(3) \rightarrow \mathrm{R}^3 \quad (2)$$

4. System construction

This section constructs a visual-inertial SLAM system based on optical flow and contributes to feature extraction, keyframe selection, the assumption of constant luminosity, loop closure, IMU pre-integration and keyframe number in sliding window.

4.1 Model designs

In the purely visual SLAM method, the pose of the camera is estimated according to the multiple view geometry principle in an unfamiliar environment. The target pose contains the rotation and translation of the camera. The six-dimensional state is expressed as:

$$\mu_i = [R_i, p_i] \in \mathrm{R}^6 \quad (3)$$

where R and p are the rotation and translation of the camera, respectively, and pose (R, p) belongs to $SE(3)$. The presented method uses camera pose, velocity and IMU deviations to construct the target state in the tightly-coupled SLAM. The 15-dimensional state variables of the system are expressed as:

$$\mu_i = [R_i, p_i, v_i, b_{ai}, b_{gi}] \in \mathrm{R}^{15} \quad (4)$$

In visual-inertial odometry, the pose, velocity and IMU deviation of each frame can be derived. The pose of the camera's current frame can be estimated using the IMU motion model, so a landmark in local map can be projected onto the current frame and matched with the key points of the current frame. The pose of the current frame is optimized by minimizing the features projection error and the IMU error. We transform the pose estimation problem into a joint optimization problem to estimate the state:

$$\mu = \arg\min_{\mu}\left(\sum r_u^I(\xi) + \mathrm{E}_{\mathrm{IMU}}(i,j)\right) \quad (5)$$

where $\Sigma r_u^{\mathrm{I}}(\xi)$ and $\mathrm{E}_{\mathrm{IMU}}(i, j)$ are the error equations of camera and IMU, respectively. We use the Gauss-Newton algorithm to optimize the entire problem.

The proposed study uses a multi-thread manner to improve computing speed and performance. The entire visual-inertial SLAM system consists of three threads: tracking, mapping and loop closure. The work in the tracking thread is mainly introduced.

4.2 Building visual odometry based on optical flow

4.2.1 Feature extraction

A stereo camera is used in the system, but only monocular data are used after the triangulation of the surrounding environment. Images are prepossessed using histogram equalization before feature extraction. The image is split into fixed size blocks (e.g. 15 × 15 pixels) and the FAST corners (Rosten *et al.*, 2010) on the blocks are initialized with a higher weight depth filter. Unless existing 2D-3D corresponding points are present, abnormal points will be filtered out.

A multi-layer pyramid is used to extract the image features. FAST corners are extracted from each layer of the image pyramids to obtain the best effect without scale constraint. In the image block where no corner is discovered, the pixel with the highest gradient is deleted and an edge feature is initialized. Features are extracted in each grid with the procedure of amending the threshold, which can dynamically change according to the brightness. When the number of points is trivial, the threshold is decreased and the feature is extracted. More uniform distribution of features in the image makes tracking easier and improves the optimization of numerical effects. When gathered locally, the value is easier to stabilize. The feature extraction image and local map are shown in Figure 1. The features are more evenly distributed in the image and tracking will not be lost.

4.2.2 Keyframe selection

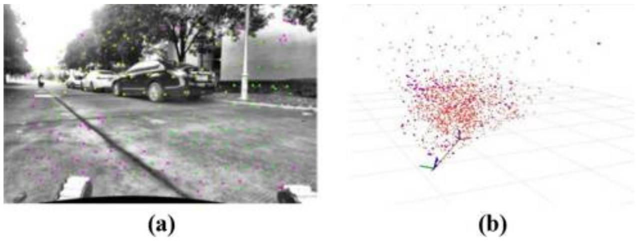
In this work, when the state of the image is estimated, whether the current frame is a keyframe or not depends on the following approaches. The interval between the current frame and the last keyframe is within a fixed threshold (e.g. 0.5 s) to guarantee the accuracy of the system. As the IMU can only provide effective precision in a short period of time, the IMU constraint between the two keyframes will be inaccurate in the case where the time interval from the previous keyframe is long; the rotation of the current frame from the previous keyframe is within a fixed threshold (e.g. 10°).

IMU is used to address pose when there are few inliers provided by camera and temporary landmarks are added to enhance the stability. New landmarks are created for those points that match and have not yet been associated with the points of the local map. Providing this feature is linked to a landmark, the landmark's information is updated. This paper selects the keyframe whose features are successfully fixed in the sliding window and uses the five-point method to resume the rotation and translation of the two keyframes. Based on the triangulation features, the PnP algorithm is used to estimate the pose of all the frames of the sliding window.

4.2.3 Optical flow tracking

This system uses a stereo camera and an IMU for tight coupling, which is similar to Huai *et al.* (2015). Conventional methods are time-consuming, which use descriptors to track features. The Lucas–Kanade optical flow algorithm (Baker and Matthews, 2004) is employed to track features to reduce the calculation. Similarly, Lin *et al.* (2017) used optical flow method for tracking, but they used the Harris corner, while our work used the FAST corner.

Figure 1 (a) Feature extraction image; (b) local map



The optical flow method has a strong assumption, which is the measured luminosity of the same point in each perspective remains the same. This assumption fails when the camera adapts to the brightness of the scene for automatic exposure. We adapt the work of Engel *et al.* (2015) and employ the parameters α and β in the system to rectify the luminosity change between the two images. The photometric residual between building image I_{k-1} and I_k is:

$$r_u^I(\xi) = \alpha I_{k-1}(u) + \beta - I_k(u') \tag{6}$$

$$\alpha = \frac{\sum_{u\in\Omega_L} I_1^l(u) I_2^l(u')}{\sum_{u\in\Omega_L} I_2^l(u') I_2^l(u')} \tag{7}$$

$$\beta = \frac{1}{|\Omega_L|}\sum_i \left(I_1^l(u') - aI_2^l(u)\right) \tag{8}$$

where Ω_L is the set of interior points. ξ represents the transition from I_k to $I_{k\text{-}1}$. $\ddot{u} = \pi(\mathrm{T}_\xi \pi^{-1}(u, D_1(u)))$. The optimization of the target equation uses the Gauss-Newton method.

In terms of reducing drift and lighting effects, this study adapts the work of Hwangbo *et al.* (2009) and Jin *et al.* (2001) to improve the system. Optical flow tracking is difficult to lose when the mobile robot maintains uniform motion (e.g. 1m/s).

4.2.4 Loop closure

The loop closure thread lessens the accumulation error when the mobile robot returns to the place that has already been built to map. Unlike pure visual odometry such as SVO (Forster *et al.*, 2017), our system has a loop closure thread that recognizes where the map was built. When a new keyframe is generated, the loop closure thread uses the bag-of-words model to find candidate keyframes that may revisit the map. DBow3 (Galvez-López and Tardos, 2012) is used on the loop closure thread and the system requires a binary format dictionary to enhance the loading speed.

In the real environment, because of the relatively simple architectural style (e.g. concrete, wall), the image is more coincidental. To avoid the false positive result, the accuracy of the system be enhanced by setting the threshold value properly. For example, the similarity between the current frame and a keyframe is three times more than the latest keyframe. At the same time, we also create a loop closure cache mechanism. It has been determined that the loop closure is correct for a period, to eliminate the mismatch caused by a single picture.

4.3 Inertial measurement unit pre-integration

The IMU data typically have white noise and bias, while the image data does not drift. Therefore, this study uses the camera to determine deviations and uses the IMU to discover the rotation and rapid motion. Accelerometer and gyroscope measurements can be computed as:

$$_{mea}w^b(t) = w^b(t) + b_g(t) + \eta_g(t) \tag{9}$$

$$_{mea}a(t) = R^{\mathrm{T}}(t)(a(t) - g) + b_a(t) + \eta_a(t) \tag{10}$$

where $_{mea}w^b(t)$ and $_{mea}a^b(t)$ account for the measured values of accelerometer and gyroscope respectively. $w^b(t)$ and $a^b(t)$ are the IMU accelerometer and gyroscope real value, respectively.

IMU measurements are issued by bias and noise, but those can be obtained during IMU calibration:

$$\dot{b}_g(t) = \eta_{bg}(t) \tag{11}$$

$$\dot{b}_a(t) = \eta_{ag}(t) \tag{12}$$

The above $\eta_g(t)$, $\eta_a(t)$, $\eta_{bg}(t)$ and $\eta_{ag}(t)$ obey the zero-mean Gaussian distribution.

Pre-integration can avoid repeating constraints caused by the re-parameterization of comparative motion integration and can effectively cut down the amount of computation. Lupton and Sukkarieh (2012) first described the method and showed the process of IMU pre-integration between two frames as a constraint to contract the optimization variables. Forster *et al.* (2015) applied the IMU bias and integral to the SLAM overall framework, and pre-integration theory was further enhanced. We use pre-integration to integrate rotation, velocity and the pose matrix. The IMU data pre-integration result is:

$$R(t+\Delta t) = \mathrm{Exp}\left(\left({}_{mea}w(t) - b_g(t) - \eta_{gd}(t)\right)\Delta t\right) \tag{13-1}$$

$$v(t+\Delta t) = v(t) + R(t)\left({}_{mea}a(t) - b_a(t) - \eta_{ad}(t)\right)\Delta t \tag{13-2}$$

$$p(t+\Delta t) = p(t) + v(t)\Delta t + \frac{1}{2}g\Delta t^2 + \frac{1}{2}R(t)\left({}_{mea}a(t) - b_a(t) - \eta_{ad}(t)\right)\Delta t^2 \tag{13-3}$$

where $R(t) \in \mathrm{SO}(3)$, Δt is the time interval. According to the result in equation (13), the pre-integration between two keyframes can be expressed as:

$$R_j = R_i \prod_{k=i}^{j-1} \mathrm{Exp}\left(\left({}_{mea}w_k - b_{g_k} - \eta_{gd_k}\right)\Delta t\right) \tag{14-1}$$

$$v_j = v_i + \mathrm{g}\Delta t_{ij} + \sum_{k=i}^{j-1} R_k\left({}_{mea}a_k - b_{g_k} - \eta_{gd_k}\right)\Delta t) \tag{14-2}$$

$$p_j = p_j + \sum_{k=i}^{j-1}\left[v_k\Delta t + \frac{1}{2}\mathrm{g}\Delta t^2 + \frac{1}{2}R_k\left({}_{mea}a_k - b_{a_k} - \eta_{ad_k}\right)\Delta t^2\right] \tag{14-3}$$

where $\Delta t_{ij} = \sum_{k=i}^{j-1}\Delta t$, $(\cdot)_i = (\cdot)(t_i)$.

This work calculates the rotation, translation, velocity of IMU and the error term of IMU deviation on account of the robot kinematics. The error of IMU is defined as:

$$\mathrm{E}_{\mathrm{IMU}}(i,j) = \rho\left(\left[\mathrm{e}_R^T \mathrm{e}_v^T \mathrm{e}_p^T\right]\sum\nolimits_I \left[\mathrm{e}_R^T \mathrm{e}_v^T \mathrm{e}_p^T\right]^T\right) + \rho\left(\mathrm{e}_b^T \sum\nolimits_R \mathrm{e}_b\right) \tag{15}$$

$$\mathrm{e}_R = \mathrm{Log}\left(\left(\Delta R_{ij}\mathrm{Exp}(\mathrm{J}_{\Delta R}^{g} b_{gj})\right)^T R_i^{bw} R_j^{bw}\right) \tag{16}$$

$$\mathrm{e}_v = R_i^{bw}\left({}_W v_j^b - {}_W v_i^b - \mathrm{g}^w\Delta t_{ij}\right) - \left(\Delta v_{ij} + \mathrm{J}_g^{\Delta v} b_g^j + \mathrm{J}_a^{\Delta v} b_a^j\right) \tag{17}$$

$$\mathrm{e}_p = R_{\mathrm{BW}}^i\left({}_{\mathrm{w}}p_B^j - {}_{\mathrm{w}}p_B^i - {}_{\mathrm{w}}v_B^i\Delta t_{ij} - \frac{1}{2}\mathrm{g}_{\mathrm{w}}\Delta t_{ij}^2\right) - \left(\Delta p_{ij} + \mathrm{J}_{\Delta p}^{\mathrm{g}} b_g^j + \mathrm{J}_{\Delta p}^{a} b_a^j\right) \tag{18}$$

$$\mathrm{e}_b = b^j - b^i \tag{19}$$

where Σ_I is a pre-integration form of the information matrix obtained by the technique of Mur-Artal and Tardós (2017). Σ_R is the random walk deviation, and $\mathrm{J}_a^{(\cdot)}$ and $\mathrm{J}_g^{(\cdot)}$ are the first-order approximations of the effects of bias variation when the pre-integration is not explicitly calculated. ρ is Huber's weight function.

In practice, the IMU frequency is much higher than the camera. When the IMU frequency is 500 Hz, the camera frequency is 25 Hz. To meet real-time performance, the output of the estimator is directly matched with the experimental measurements of the IMU as the high-frequency feedback of the control loop.

4.4 Nonlinear optimization

In this section, a further nonlinear optimization scheme is presented. The sliding window is used to construct the system variable model, and the optimal scheme is provided to obtain the optimal keyframe number in the sliding window through experiments.

As sliding windows utilize multi-constraint estimation to provide accuracy, sliding windows are used to estimate the state of the camera data and IMU data. We build a fixed window with camera and IMU data in the variable model:

$$\chi = \left[\mu_0, \mu_1, \cdots \mu_n, \mu_c^b, \lambda_0, \lambda_1, \cdots \lambda_m\right] \tag{20}$$

$$\mu_i = \left[R_{b_i}^w, p_{b_i}^w, v_i, b_{a_i}^b, b_{g_i}^b\right], i \in [0, n] \tag{21}$$

where n denotes the number of keyframes in the sliding window; m denotes the number of features in the sliding window; and λ_l is the inverse depth at which the lth feature is first observed in the camera.

We select the number of keyframes through our data set as shown in Table I. Surprisingly, the error and scale do not change linearly with the number of keyframes. As expected, the computation increases with additional keyframes. Finally, the number of keyframes in sliding window is set to 9 to balance accuracy and calculation.

5. Experiments and results

This section experiments the visual-inertial SLAM system on the mobile robot and offline platform.

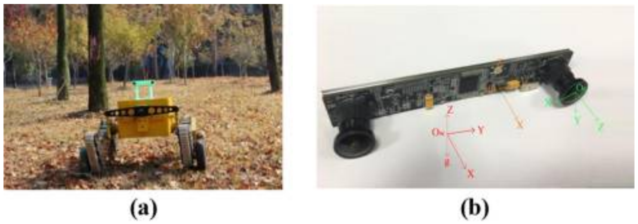
5.1 Application on the mobile robot

The constructed visual-inertial SLAM system is applied to the rocker wheel-track robot, which is illustrated in Figure 2(a). The experiment utilizes a visual-inertial sensor integrated with a consumer-grade IMU and two global shutter cameras. The

Table I Experiments on the number of keyframes

No.	n = 5	n = 7	n = 9	n = 11	n = 13
RMSE (m)	0.244	0.241	0.231	0.240	0.238
Scale	1.043	1.039	1.006	1.003	0.996

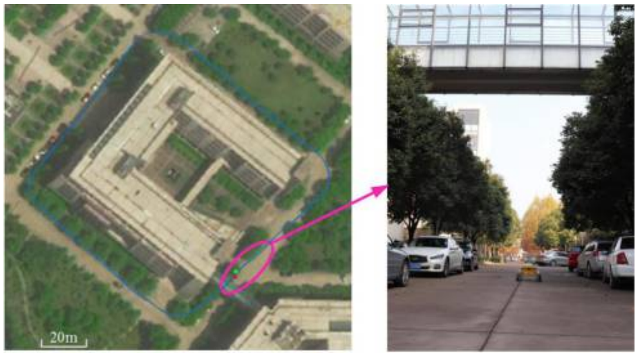
Figure 2 (a) Rocker wheel-track robot; (b) frame pose relationship



world frame, camera frame and IMU frame are illustrated in Figure 2(b). The experiments were performed on an Intel Core i7-3555LE × 4 computers with 8 Gb RAM.

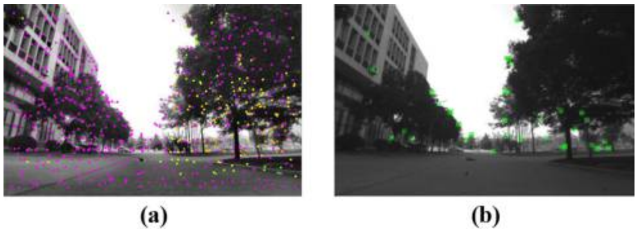
We set the camera frequency to 25 Hz. The system also must be compatible with other tasks while moving the robot. When the robot is stationary, no keyframes are added to maintain the entire tracking stability. To cope with low-texture scenes during the experiment, we set a lower FAST corner threshold to create the tracking success. For example, we set the threshold to 5 instead of 20. The maximal number of features in the state is set to 200 and the algorithm is run with a four-level pyramid. As shown in Figure 3, the robot moved two laps in the 10,000 m^2 campus environment and the scene is not rich around the mobile robot, which are concrete, walls and other low-texture scenes.

Figure 3 Campus environment and robot trajectory



In this large GPS-denied environment, it is hard to provide the ground truth. The end-to-end error is 1.375 m with respect to the total length of 762 m; it is only 0.18 per cent of the total trajectory length. We also used ORB-SLAM2 in the record data sets but tracking always failed. Not only could it not extract enough features on low texture environments but also the features were less evenly distributed. Feature extraction images of our SLAM and ORB-SLAM2 are shown in Figure 4.

Figure 4 (a) Our SLAM feature extraction image; (b) ORB-SLAM2 feature extraction image



5.2 Evaluation on data sets

This investigation uses the EuRoC data sets (Burri *et al.*, 2016) to compare our SLAM algorithm with the ORB-SLAM2 algorithm, which is the best robust algorithm in the current SLAM algorithm. The EuRoC data sets is currently the best choice for comparing different approaches, although acquired by a micro-aerial vehicle. The experiments were performed on the computer provided in Section 5.1. Absolute trajectory error (ATE), relative pose error (RPE), scale and CPU load were calculated. Machine hall data sets and Vicon room one data set were chosen for comparative experiments, as shown in Table II.

In Table II, when all four logical cores are in use, the CPU load is 100per cent, and the average scale error is filled in the form. We achieve better RPE, scale and CPU load than ORB-SLAM2, which means our system drift is much lower than that of ORB-SLAM2 and have good stability. The ATE reaches centimeter level and the RPE reaches the micron level.

6. Conclusion

In this research, we provide a novel optical flow-based, tightly coupled, visual-inertial SLAM system, which can realize robot positioning and navigation under the condition of GPS failure. The system can extract the features on low texture environments and track features steadily. We improve the assumption of constant luminosity and use the optical flow to

Table II The comparison between our method and ORB-SLAM2

	Our SLAM				ORB-SLAM2 (Stereo)			
Sequence	ATE(m)	RPE(m)	Scale	CPU load (%)	ATE(m)	RPE(m)	Scale	CPU load (%)
MH_01_easy	0.056	0.002	0.997	50.51	0.034	0.022	1.004	63.55
MH_02_easy	0.285	0.003	0.997	49.83	0.047	0.024	0.992	61.27
MH_03_medium	0.233	0.005	0.995	48.53	0.040	0.056	0.997	64.32
MH_04_difficult	0.446	0.006	1.014	55.07	0.111	0.064	1.014	60.51
MH_05_difficult	0.269	0.005	1.011	54.18	0.567	0.056	0.990	61.09
Average	0.258	0.004	0.07	51.62	0.159	0.044	0.008	62.15
V1_01_easy	0.124	0.003	1.011	46.07	0.089	0.033	1.005	54.19
V1_02_media	0.204	0.003	1.002	52.77	0.063	0.050	1.008	53.34
V1_03_difficult	0.292	0.016	0.996	52.72	0.156	0.048	0.997	45.01
Average	0.21	0.007	0.057	50.52	0.103	0.044	0.005	50.85

track features and enhance the system robustness. This system can be applied on mobile robots and achieve better accuracy. Its RPE can reach the micron level. The system performs better than ORB-SLAM2 in terms of RPE, scale and CPU load.

Note

1 https://github.com/raulmur/ORB_SLAM2

References

Baker, S. and Matthews, I. (2004), “Lucas-Kanade 20 years on: a unifying framework”, *International Journal of Computer Vision*, Vol. 56 No. 3, pp. 221-255.

Burri, M., Nikolic, J., Gohl, P., Schneider, T., Rehder, J., Omari, S., Achtelik, M.W. and Siegwart, R. (2016), “The EuRoC micro aerial vehicle datasets”, *International Journal of Robotics Research*, Vol. 35 No. 10, pp. 1157-1163.

Concha, A., Loianno, G., Kumar, V. and Civera, J. (2016), “Visual-inertial direct Slam”, *IEEE International Conference on Robotics and Automation*, pp. 1331-1338.

Engel, J., Stückler, J. and Cremers, D. (2015), “Large-scale direct Slam with stereo cameras”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1935-1942.

Falquez, J.M., Kasper, M. and Sibley, G. (2016), “Inertial aided dense & semi-dense methods for robust direct visual odometry”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3601-3607.

Forster, C., Carlone, L., Dellaert, F. and Scaramuzza, D. (2015), *IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation*, Georgia Institute of Technology, Georgia.

Forster, C., Zhang, Z., Gassner, M., Werlberger, M. and Scaramuzza, D. (2017), “Svo: semidirect visual odometry for monocular and multicamera systems”, *IEEE Transactions on Robotics*, Vol. 33 No. 2, pp. 249-265.

Furgale, P., Rehder, J. and Siegwart, R. (2013), “Unified temporal and spatial calibration for multi-sensor systems”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1280-1286.

Galvez-López, D. and Tardos, J.D. (2012), “Bags of binary words for fast place recognition in image sequences”, *IEEE Transactions on Robotics*, Vol. 28 No. 5, pp. 1188-1197.

Gui, J., Gu, D., Wang, S. and Hu, H. (2015), “A review of visual inertial odometry from filtering and optimisation perspectives”, *Advanced Robotics*, Vol. 29 No. 20, pp. 1289-1301.

Hartley, R. and Zisserman, A. (2003), “Multiple view geometry in computer vision”, *Kybernetes*, Vol. 30 Nos 9/10, pp. 1865 -1872.

Huai, J., Toth, C.K. and Grejner-Brzezinska, D.A. (2015), “Stereo-inertial odometry using nonlinear optimization”, *International Technical Meeting of the Satellite Division of the Institute of Navigation*.

Hwangbo, M., Kim, J.S. and Kanade, T. (2009), “Inertial-aided KLT feature tracking for a moving camera”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1909-1916.

Indelman, V., Williams, S., Kaess, M. and Dellaert, F. (2012), “Factor graph based incremental smoothing in inertial navigation systems” *International Conference on Information Fusion*, pp. 2154-2161.

Jin, H., Favaro, P. and Soatto, S. (2001), “Real-time feature tracking and outlier rejection with changes in illumination”, ICCV, pp. 684-689.

Leutenegger, S., Lynen, S., Bosse, M., Siegwart, R. and Furgale, P. (2015), “Keyframe-based visual-inertial odometry using nonlinear optimization”, *International Journal of Robotics Research*, Vol. 34 No. 3, pp. 314-334.

Lin, Y., Gao, F., Qin, T., Gao, W., Liu, T., Wu, W., Yang, Z. and Shen, S. (2017), “Autonomous aerial navigation using monocular visual-inertial fusion”, *Journal of Field Robotics*, Vol. 4.

Lupton, T. and Sukkarieh, S. (2012), “Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions”, *IEEE Transactions on Robotics*, Vol. 28 No. 1, pp. 61-76.

Martinelli, A. (2014), *Closed-Form Solution of Visual-Inertial Structure from Motion*, Kluwer Academic Publishers, Dordrecht.

Mourikis, A.I. and Roumeliotis, S.I. (2007), “A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation”, *IEEE International Conference on Robotics and Automation*, pp. 3565-3572.

Mur-Artal, R. and Tardós, J.D. (2016), “ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras”, *IEEE Transactions on Robotics*, Vol. 33 No. 5, pp. 1255-1262.

Mur-Artal, R. and Tardós, J.D. (2017), “Visual-inertial monocular SLAM with Map reuse”, *IEEE Robotics and Automation Letters*, Vol. 2 No. 2, pp. 796-803.

Rosten, E., Porter, R. and Drummond, T. (2010), “Faster and better: a machine learning approach to corner detection”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32 No. 1, pp. 105-119.

Weiss, S., Scaramuzza, D. and Siegwart, R. (2011), “Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments”, *Journal of Field Robotics*, Vol. 28 No. 6, pp. 854-874.

Yang, Z. and Shen, S. (2017), “Monocular visual–inertial state estimation with online initialization and camera–IMU extrinsic calibration”, *IEEE Transactions on Automation Science and Engineering*, Vol. 14 No. 1, pp. 39-51.

Yousif, K., Bab-Hadiashar, A. and Hoseinnezhad, R. (2015), “An overview to visual odometry and visual SLAM: applications to mobile robotics”, *Intelligent Industrial Systems*, Vol. 1 No. 4, pp. 289-311.

Further reading

Weiss, S. and Siegwart, R. (2011), “Real-time metric state estimation for modular vision-inertial systems”, *Robotics and Automation (ICRA), 2011 IEEE International Conference on, IEEE*, pp. 4531-4537.

Corresponding author

Hua Zhu can be contacted at: zhuhua83591917@163.com