Research on an Unmanned Platform Odometer Based on Vision and Inertial Navigation Fusion Technology

Yue Dong[1], Qiang Luo[2(✉)], Ruochen Yang[2], Zhe Du[3], and Luhua Zhang[3]

[1] Unmanned System Research Institute, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, China
[2] Xi'an Modern Control Technology Research Institute, Xi'an 710065, Shaanxi, China
`296556515@qq.com`
[3] Shanghai Electro-Mechanical Engineering Institute, Shanghai 201109, China

Abstract. Visual inertial odometer is a device that takes camera and IMU as input and takes the position and attitude of unmanned platform as output. The traditional odometer is an important auxiliary navigation mode in the unmanned navigation system. But the traditional odometer cannot adapt to many new developments. Therefore, this paper aims to make a study on a new odemeter: visual inertial odometer (VIO). This paper mainly studies the robustness and applicable environment of the visual inertial odometer. Firstly, this paper explains what is visual inertial odometer, and puts forward the significance of this research. After that, simulation experiments are carried out to verify the robustness and accuracy of the visual inertial odometer. Then the visual inertial odometer is tested based on hardware. Finally, through the analysis of the experimental results and errors, the applicable environment and conditions of the visual inertial odometer are obtained.

Keywords: Visual navigation · Inertial navigation · Pose estimation · Monocular vision inertial odometer

1 Introduction

Visual inertial odometer (VIO) is a new odometer which integrated the traditional visual odometer and the inertial odometer [1]. The VIO's purpose is to estimate unmanned-system's motion and trajectory, so as to achieve the purpose of navigation and positioning.

With the development of unmanned navigation technology, the working environment of navigation system has become more diverse and complex. For example, in modern warfare, unmanned systems may be in a denied environment [2]. In the denied environment, radio navigation and GPS navigation are likely to fail, so a new navigation is needed to replace the failed navigation. The VIO is not affected by the denied environment.

In the mean time, the traditional inertial navigation is easy to accumulate errors [3]. The pure visual odometer cannot work well when the unmanned system moving too fast.

Different from both of the above, VIO are not affected by the environment and motion speed. Therefore, the research of the VIO is very significant. This paper mainly studies the robustness of the VIO and its application conditions.

2 Model and Algorithm

In this paper, the visual-inertial odometer uses the Vins-Mono open source code [4]. This open source code combined with own hardware to form a VIO for experiments. The frame of visual inertial odometer is shown in the following figure (Fig .1):
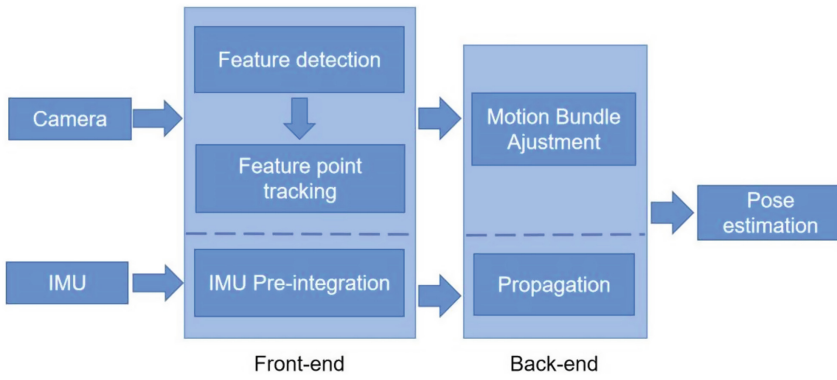

Fig. 1. The frame of the VIO

As the front-end of the VIO, camera and IMU provide the information for VIO. The back-end of the VIO receives the data from the front-end for pose estimation. According to the fusion mode of front-end's information, it can be divided into tight coupling and loose coupling. There are two pose estimation methods of back-end: optimization and filtering. The methods of close coupling combined filtering mainly be used by ROVIO (robust visual inertial odometry) [5] and MSCKF (multi-state constraint Kalman filter) [6, 7].

At present, the commonly used method is the combination of tight coupling and optimization, such as Vins-Mono. In Vins-Mono, the front-end adopts Harris corner detection algorithm [8] and KLT [9] optical flow tracking method, and the back-end adopt BA (Bundle Ajustment) algorithm [10]. Some key algorithms are introduced below.

Harris Corner Detection Algorithm: Assume that $E(u, v)$ is the pixel change in the small window, I is the gray value of the image.

$$E(u, v) = [u, v]M \begin{bmatrix} u \\ v \end{bmatrix} \tag{1}$$

$$M = \sum_{x,y} w_{x,y} \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{2}$$

$$R(x, y) = \det(M) - k \, (tr(M))^2 \tag{3}$$

Defined a judgement function as R(x,y). If R(x,y) bigger than a certain value, we can say that this point is corner point.

Bundle Adjustment Algorithm: The conversion relationship between pixel coordinates and space point coordinates is as follows.

$$s_i \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = K \exp(\hat{\xi}) \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \tag{4}$$

The (4) has errors, we add up the errors to construct the least square problem.

$$\xi^* = \arg\min \frac{1}{2} \sum_{i=1}^{n} \left\| u_i - \frac{1}{s_i} K \exp(\hat{\xi}) \, P_i \right\|_2^2 \tag{5}$$

The coordinates of three-dimensional points can be obtained by solving the least square problem (5).

KLT Algorithm: KLT algorithm is an algorithm for image matching by solving the offset d. Suppose I and J are two consecutive pictures, the pixel value of point (x,y) is I(x,y) and J(x,y) respectively.

$$\varepsilon(d) = \iint \left[J(x + \frac{d}{2}) - I(x - \frac{d}{2}) \right]^2 \omega(x) dx \tag{6}$$

Taylor expand $\frac{\partial \varepsilon}{\partial d}$.

$$\frac{\partial \varepsilon}{\partial d} = 2 \iint \left(J(x + \frac{d}{2}) - I(x - \frac{d}{2}) \right) \left(\frac{\partial J(x + \frac{d}{2})}{\partial d} - \frac{\partial I(x - \frac{d}{2})}{\partial d} \right) \omega(x) dx \tag{7}$$

Let (7) equal zero to solved.

IMU Pre-integration: IMU Pre-integration solves the problem that the amount of calculation of PVQ update is too large. Assume that the bias of IMU is known and the value of Pre-integration is only related to the measurement of IMU. By integrating the IMU data directly, the value of Pre-integration can be obtained:

$$\begin{cases} \alpha_{bibj} = \iint_{t \in [i,j]} (q_{bibt} a^{bt}) \delta t^2 \\ \beta_{bibj} = \int_{t \in [i,j]} (q_{bibt} a^{bt}) \delta t \\ q_{bibj} = \int_{t \in [i,j]} q_{bibt} \otimes \begin{bmatrix} 0 \\ \frac{1}{2} \omega^{bt} \end{bmatrix} \delta t \end{cases} \tag{8}$$

The new IMU Pre-integration formula of PVQ is obtained:

$$
\begin{bmatrix} p_{\omega bj} \\ v_j^{\omega} \\ q_{\omega bj} \\ b_j^a \\ b_j^g \end{bmatrix} = \begin{bmatrix} p_{\omega bi} + v_{\omega}^i \Delta t - \frac{1}{2} g^{\omega} \Delta t^2 + q_{\omega bi} \alpha_{bibj} \\ v_i^{\omega} - g^{\omega} \Delta t + q_{\omega bi} \beta_{bibj} \\ q_{\omega bi} q_{bibj} \\ b_i^a \\ b_i^g \end{bmatrix}
\tag{9}
$$

The (9) is the final formula of IMU Pre-integration, it can simplify IMU integral operation.

3   Mainly Results

The visual inertial odometer can work normally in different environments and tracks, and has good stability and robustness. However, its positioning accuracy is affected by the following aspects:

1) Lighting conditions: In poor or complex lighting conditions, the APE errors of visual inertial odometry will be bigger.
2) Speed: The movement speed should not be too fast, not more than 17m/s; The speed should also not be too slow.
3) IMU: Before using the visual-inertial odometry, the IMU should be fully activated.
4) Length of trajectory: With the increasement of the length of trajectory, the errors will become bigger.

As the difficulty of the experiment increases, the error of the experiment will also increase. And from the experiments, If the trajectory vibrates or the robot suddenly stops moving, the VIO is prone to drift. The reasons are as follows:

1) If the robot runs too slowly, the image update speed will be too slow, and the feature points cannot be updated quickly, so the camera pose transformation cannot be calculated by the new feature points and the old feature points.
2) Due to the slow movement of the robot, the inertial components are not fully activated.
3) Vibration in a small range will reduce the accuracy of IMU, and it will make the camera unable to process rapidly changing images.

4   Experimental Results and Error Analysis

4.1   Euroc Dataset Simulation Results and Error Analysis

Euroc dataset [11] is commonly used in the simulation test of visual inertia algorithm.

The upper left corner of the Fig. 2 is the real-time image of the camera. The lower left corner of the picture shows the ring detection results, and the right part of the Fig. 3 shows the output trajectory of the unmanned platform.
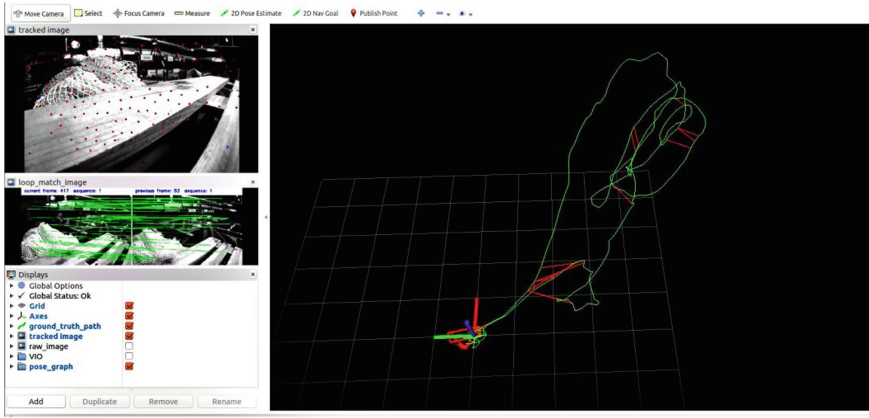

Fig. 3. Output results of dataset experiment

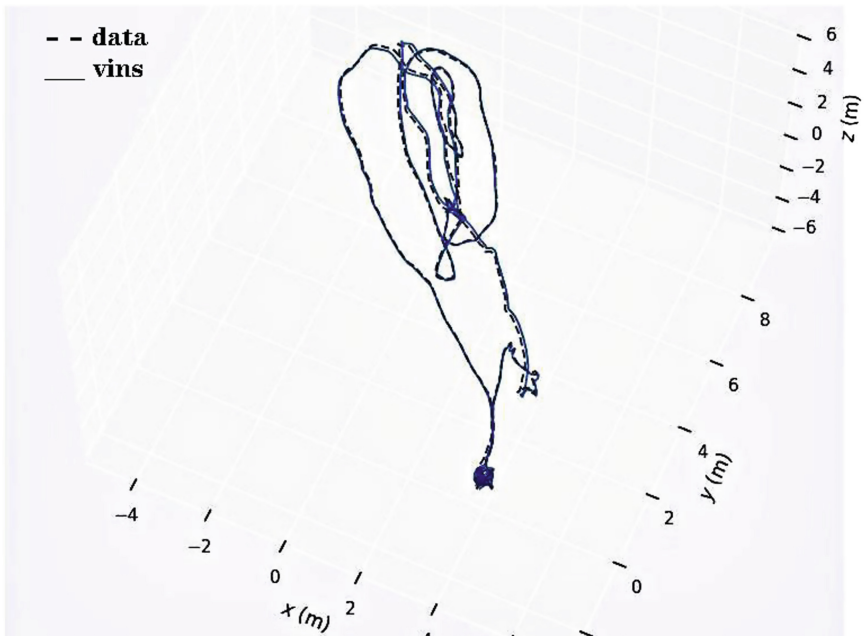The following figure shows the comparison between the output trajectory and the real trajectory:

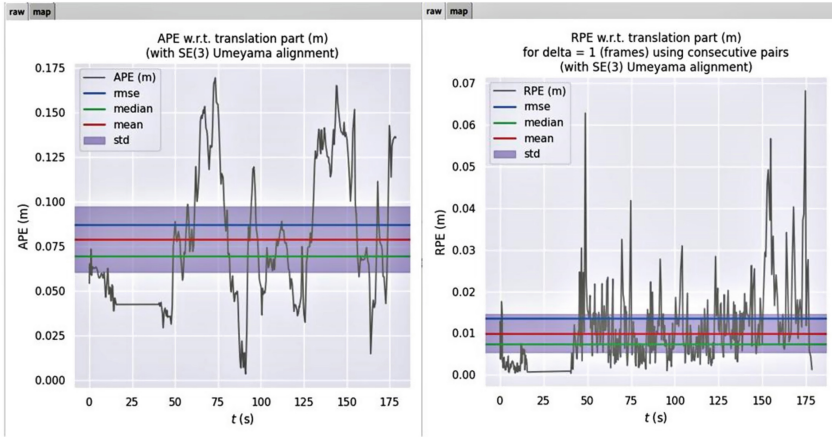**Fig. 4.** Output trajectories vs true trajectories

Fig. 5. PE&RPE error of simulation experiment

The solid line in the Fig. 4 is the output trajectory of visual inertial odometer, while the dotted line represents the real trajectory. There is little difference between the real trajectory and the output trajectory.

Table 1. Error table of simulation experiment.

	APE-mean	APE-min	APE-max	RPE-mean	RPE-min	RPE-max
Error	0.078913	0.003406	0.169524	0.009952	0.000378	0.068184

The figure of APE error and RPE error is as follows:

It can be seen from the Table 1 and Fig. 5 that both the APE error and the RPE error are maintained within a relatively small range.

4.2 Experimental Results and Error Analysis Based on Hardware

The experimental hardware is as follows (Fig. 6):

Fig. 6. Experimental hardware

The basic idea of the experiment based on the hardware is to let the robot carry a VIO and walk twice in the same path. The two paths can be as close as possible by setting

road markings. But limited by the experimental condition, the two paths may not be exactly the same and the experimental error may become larger. However, as long as the error is within the acceptable range, it will not affect the analysis of VIO's performance. After the above description, then we can compare the errors of the two paths, analyze the errors, and get the characteristics of the visual inertial odometer (Fig. 7).
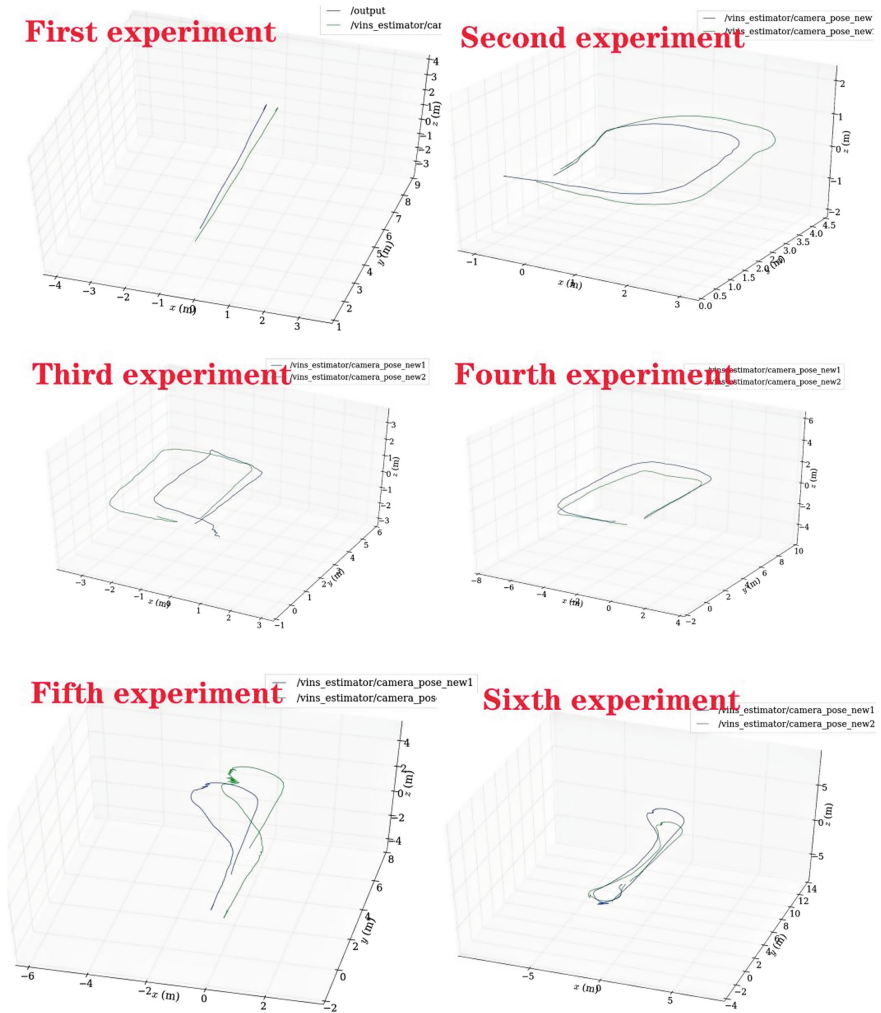

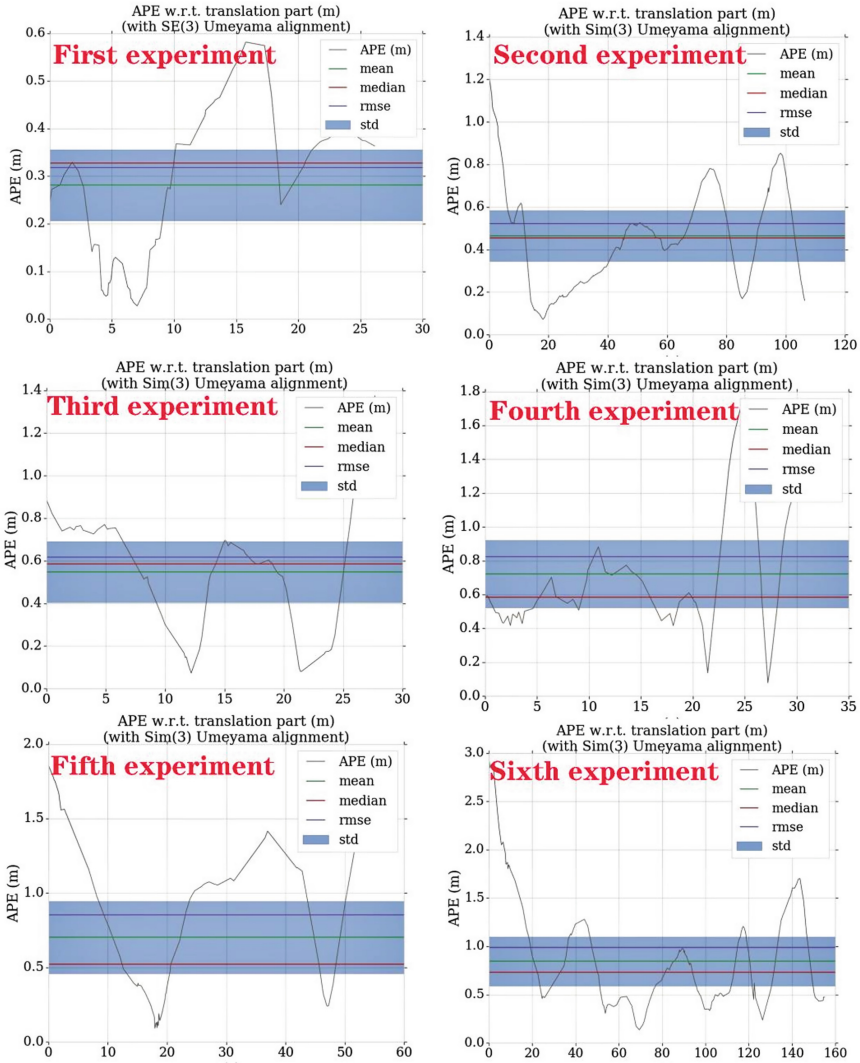**Fig. 7.** Output trajectory of six experiments

Fig. 8. APE error of six experiments

It can be seen from the Table 2 and Fig. 8 that the average value of APE error and RPE error is within a relatively small range. The visual inertial odometer can work normally in different environments and tracks, and has good stability and robustness. However, in the fifth and sixth experiments, the error becomes bigger. After carefully observing the fifth and sixth experiments, it can be seen that the output trajectory vibrated in these two experiments (Fig. 9).

In the actual experiment, the speed of the robot at the corner will also become very low. So, the low speed and vibration caused the error to become larger.

Table 2. Error table of six experiments.

|	APE-mean	APE-min	APE-max	RPE-mean	RPE-min	RPE-max
TheFir	0.227576	0.012622	0.692916	0.027238	0.000378	0.560324
TheSec	0.464942	0.073091	1.213435	0.030403	0.000618	0.392573
TheThi	0.547895	0.072934	1.375854	0.082922	0.005244	0.534687
TheFou	0.721566	0.078212	1.699692	0.121585	0.007112	0.924192
TheFiv	0.702418	0.091417	1.857750	0.104320	0.001957	0.876218
TheSix	0.846011	0.136444	2.929616	0.069932	0.033430	1.057165
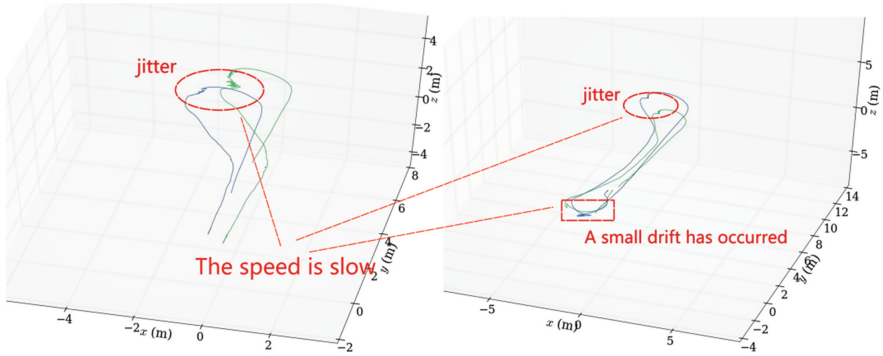
Fig. 9. The fifth and the sixth experiment output trajectory description

The experimental environment and trajectory of the six experiments are as follows (Fig. 10):

Fig. 10. Six experiments' environments

Table 3. Table of six experiments.

	Indoor/outdoor	Lighting conditions	Environmental information	Trajectory shape	Overview
Case1	Indoor	Good	Generally	Straight line	Easy
Case2	Indoor	Good	Generally	Closed loop	Medium
Case3	Indoor	Generally	Less	Closed loop	Medium
Case4	Outdoor	Generally	Good	Closed loop	Difficult
Case5	Indoor	Generally	Less	Rregular	Disaster
Case6	Outdoor	Poor	Good	Irregular	Disaster

The specific information for six experiments has listed in the Table 3. As the difficulty of experiment increases, the error of experiment also increases.

5 Conclusions

This visual inertial odometer can work in different environments and has robustness. The positioning accuracy of this visual inertial odometer is affected by illumination conditions, environmental information, track length and so on. But in most cases, this visual inertial odometer has good positioning accuracy.

The visual inertial odometer industry also still faces some problems:

1) When the robot changes direction violently or vibrates, the positioning effect of VIO is not good, and it is easy to drift. The robustness of the system needs to be further improved.
2) The computational complexity of the system needs to be further improved and optimized.

For the above problems, the following are some suggestions for improvement:

1) Adding trajectory shape optimization algorithm and vibration judgement algorithm: When the vibration judgement algorithm detected that the robot is vibrating, the trajectory shape optimization algorithm will be used to optimize the trajectory, to smooth the trajectory.
2) Adding environment judgment algorithm: When the operating environment is judged to be good, the VIO can reduce the number of feature points, to reduce the amount of calculation. While in difficult environments, the VIO can increase the amount of feature points appropriately.

Today, most unmanned aerial vehicles and robots are equipped with a variety of sensors, among these sensors, cameras and inertial components are very common. For example, the United States' global Eagle UAV is equipped with optical and infrared cameras. Dajiang UAV is equipped with high-definition cameras and inertial devices.

And VIO has low requirements for cameras and inertial devices. The price of a camera is about 200 yuan, and the electronic inertial component MEMS is just only about twenty yuan. Therefore, the VIO is very economical and feasible. And this VIO can obtain better positioning performance under the conditions of good lighting conditions and sufficient information of the surrounding environment (which can be satisfied in most cases), and can provide auxiliary navigation for unmanned aerial vehicles or robots.

References

1. MUR-ARTAL, Raúl; TARDÓS, Juan D. Visual-inertial monocular SLAM with map reuse. *IEEE Robotics and Automation Letters*, 2017, 2.2: 796–803
2. Balamurugan, G., Valarmathi, J., Naidu, V.P.S., Survey on UAV navigation in GPS denied environments. In,: International conference on signal processing, communication, power and embedded system (SCOPES). IEEE **2016**, 198–204 (2016)
3. Ronnback S. Development of a INS/GPS navigation loop for an UAV[D]. Lulea Tekniska University of Technology,2000
4. Qin Tong, Li Peiliang, Shen Shaojie. VINS-Mono:a robust and versatile monocular visual-inertial state estimator[J].IEEE Trans on Robotics, 2018, 34(4):1004–1020
5. Bloesch M,Omari S,Hutter M,et al.Robust visual inertial odometry using a direct EKF-based approach[C]//Proc of IEEE / RSJ International Conference on Intelligent Robots and Systems. Washington DC:IEEE Computer Society,2015: 298–304
6. Mourikis AI,Roumeliotis S I. A multi-state constraint Kalman filter for vision-aided inertial navigation[C]//Proc of IEEE International Conference on Robotics and Automation.Piscataway,NJ:IEEE Press,2007:3565–3572
7. Li Mingyang,Mourikis AI.High-precision,consistent EKF-based visual-inertial odometry[J].International Journal of Robotics Re-search,2013,32(6) : 690–711
8. C.Harris and M.Stephens,A combined corner and edge detector, in Alvey vision conference,vol. 15,pp.10–5244,Citeseer,1988
9. Lucas B D, Kanade T. An iterative image registration technique with an application to stereo vision[C] International Joint Conference on Artificial Intelligence. Morgan Kaufmann Publishers Inc. 1981:674–679
10. Li, et al.: Bundle adjustment method using sparse BFGS solution[J]. Remote Sensing Letters **9**(8), 789–798 (2018)
11. Burri, M., et al.: The EuRoC micro aerial vehicle datasets[J]. The International Journal of Robotics Research **35**(10), 1157–1163 (2016)