ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

# Overview of Visual SLAM for Mobile Robots

**Chen Wei, Aihua Li**

*Rocket Force University of Engineering, Xi'an Shaanxi 710038, China*

***Abstract:*** *Simultaneous Localization and Mapping (Simultaneous Localization and Mapping) technology refers to the technology of self-localization and construction of environmental maps based on visual sensors. It plays an important role in the field of autonomous mobile robots and autonomous vehicle navigation. This article introduces the classic framework and basic theory of visual SLAM, as well as the common methods and research progress of each part, enumerates the landmark achievements in the visual SLAM research process, and introduces the latest ORB-SLAM3. Finally, the current problems and future research directions of visual SLAM are proposed.*

***Keywords:*** *Mobile robots, Visual SLAM, Automatic navigation, ORB-SLAM3*

## Introduction

With the continuous improvement of human life science and technology, mobile robots capable of autonomous navigation appear more and more frequently in our lives. The key to autonomous navigation technology of mobile robots lies in sensing the environment, positioning, building maps and path planning. SLAM technology is the technology of simultaneous positioning and map construction. The main research is to estimate one's own position and construct a map of the surrounding environment while moving in an unknown environment. It plays a very important role in the autonomous navigation of mobile robots. Because visual sensors can provide richer information and are inexpensive, the research on visual SLAM has become a research hotspot in the field of robotics and computer vision in the past two decades. Visual SLAM technology also shows great application value in market applications.

This article mainly introduces the classic framework and research content of visual SLAM technology, discusses recent research progress at home and abroad, and analyzes the problems to be solved in visual SLAM and future research trends.

## 1. Visual SLAM Technology

### *1.1. Classic Framework*

After twenty years of research, the visual SLAM framework has been basically mature, including five steps of sensor data reading, visual odometry, back-end nonlinear optimization, loop detection and mapping [1], such as shown in Figure 1

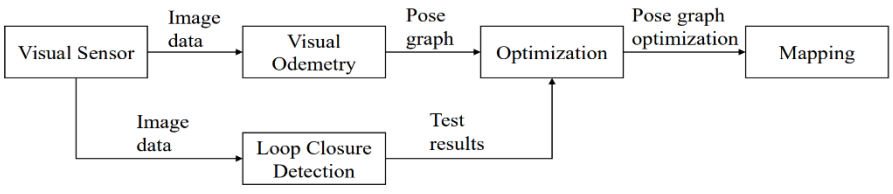


*Figure 1: Visual SLAM technology framework diagram*

Among them, the sensor data reading mainly uses the camera to collect image information and preprocess it. Visual Odemetry is also known as the front-end, the task is to use neighboring images to estimate camera movement and local maps. Optimization, also known as the back-end, is used to receive the camera pose and loop detection information estimated by the front-end, and optimize it to obtain a global motion trajectory and map. Loop Closure Detection is also called closed loop detection. The task is to determine whether the robot has returned to the origin and solve the problem of time drift. Mapping is to use the previously estimated motion trajectory to construct a suitable map according to

ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

the application requirements.

The basic principle of the SLAM problem

Assuming that the robot observes a certain mark point y at the actual position x, the observation data is z, and the motion input is u, then the mathematical equation to express the SLAM problem is:

$$x_i = f\left(x_{i-1}, u_i, v_i\right)$$

$$z_{i,j} = g\left(y_j, x_i, w_{i,j}\right)$$

Among them, v and w are observation noise.

### *1.2. The Main Research Content of Visual SLAM*

#### *1.2.1 Visual Odometry*

The visual odometer estimates the pose of the camera based on the information in the adjacent images. The algorithms currently used can be divided into two categories: feature point method and direct method.

1) Feature point

The step of the feature point method is to first extract the appropriate feature points from the image. These feature points can be matched in adjacent images. Then, according to the geometric relationship of the matched point pairs, estimate what the camera does when shooting these adjacent images. sports. Feature points are representative points in the image. Commonly used feature points include Harris corner points [2], FAST corner points [3], GFTT corner points (Shi-Tomasi corner points)[4] and so on. However, these corner points do not have scale invariance[5] and cannot meet the matching in the case of camera movement, so the researchers designed the famous SIFT[6], SURF[7], ORB[8] feature points.

Among them, the SIFT feature considers the changes in image illumination, scale, and rotation caused by camera motion, and has the advantages of distinguishability, rotation invariance, and scale invariance [9]. However, because the calculation dimension is too high, it occupies a lot of computing resources. The SURF feature is an acceleration of the SIFT feature (3 to 7 times). ORB (Oriented FAST and Roated BRIEF) feature is the best feature point applied at this stage. According to the author's test in the literature[8], to extract 1000 feature points in the same image, the time taken for ORB is 15.3ms, the time taken by SURF is 217.3ms, and the time taken by SIFT is 5228.7ms. It can be seen that the ORB feature not only guarantees rotation and scale invariance, but also guarantees good performance.

However, point features also have their limitations. For example, it is difficult to extract enough feature points in a weak texture environment, so Lu [9] et al. proposed to introduce line features into visual odometry, and Pumarola et al. proposed PL-SLAM [10]. The accuracy of ORB-SLAM is improved by adding line features. Concha et al. introduced super-pixel features into MonoSLAM [11], which solved the problem of insufficient point features in weak texture environments.

After obtaining the matched point pairs, the camera pose estimation can be carried out. According to the different cameras used, there are 3 solutions for pose estimation: When using a monocular camera, since there are only 2D coordinates, use the method of epipolar geometry; when using When the binocular camera or RGB-D camera has 3D coordinates, use the Iterative Closest Point (ICP) method to solve; when the obtained coordinates are a set of 3D and a set of 2D, use the PnP method to solve[1].

Disadvantages of feature point method:

The extraction of feature points is very time-consuming, ignoring most of the image information except for the feature points, and the number of feature points is small in a weak texture environment.

2) Direct method

In order to overcome the shortcomings of the feature point method, researchers have proposed a direct method to estimate camera pose and motion based on pixel brightness information. New visual SLAMs such as SVO [12], LSD-SLAM [13], and DSO [14] have appeared. algorithm.

ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

The direct method evolved from the optical flow method, based on the gray-level invariance assumption [1] (that is, the pixel gray value of the same spatial point is unchanged in images of different viewing angles), and the camera pose is the optimization variable, Solve the optimal camera pose by minimizing the photometric error. According to the number of pixels used, it can be divided into sparse direct method, semi-dense direct method and dense direct method.

SVO[11] proposed by Forster et al. is a semi-direct visual odometer, which combines the feature point method and the direct method, and uses the feature point block to match the camera pose transformation, and obtains a faster processing speed. In 2017, the author expanded the functions of multi-robot collaboration and IMU inertial devices on its basis [15].

#### *1.2.2. Optimization*

Through the processing of the visual odometer, a local error map and motion track can be obtained. In order to reduce the accumulated error and obtain a global map, a back-end optimization is required. The essence of back-end optimization is to find the optimal solution of Equation 1. According to whether Markov property is considered, the back-end optimization methods can be divided into two categories. One is the filtering method. It is believed that the state at this moment is only related to the state at the previous moment, including the extended Kalman filter (EKF), Particle Filter (PF), etc., the other is a non-linear optimization method, considering that the state at this moment is related to the state at all previous moments, including BA (Bundle Adjustment), pose map method, graph Optimization, etc., are currently the mainstream solutions to SLAM problems.

1) Filtering method

The early SLAM problem mainly used the filtering method. The classical Kalman filter is not suitable for the actual navigation scene of nonlinear and non-Gaussian [16], so Smith et al. first proposed the application of extended Kalman filter in the literature [17] The theory of SLAM, Moutarlier et al. put it into practice [18]. Since EKF requires the system to be approximately linear and continuous, it is easy to cause cumulative errors for nonlinear systems. Then Murphy proposed SLAM algorithm based on particle filter [19]. PF-SLAM uses weighted random samples to approximate the system state [16], which has a great advantage when dealing with nonlinear systems. However, the particle filter takes up a lot of space to store a large number of particles, which limits its application in real-time large-scale mapping [20].

2) Non-linear optimization method

The more commonly used nonlinear optimization methods are the Gauss Newton method and the Levenberg-Marquardt method, and the graph optimization method is the product of combining nonlinear optimization with graph theory [1]. At present, the open source nonlinear optimization libraries include Ceres and g2o [21], which are widely used in nonlinear optimization problems. Lu et al. first proposed the SLAM method based on graph optimization, using nonlinear least squares to solve [22]. Duckett et al. proposed an optimization method based on relaxation and proved that the method must be able to find the optimal solution [23]. Olson et al. used the stochastic gradient descent method [24], and Grisetti et al. used a tree structure to improve the update efficiency of the pose [25]. Aiming at the problem of robot pose in non-Euclidean space, Grisetti et al. proposed to optimize in the manifold to improve the accuracy of the algorithm [26].

#### *1.2.3. Loop Closure Detection*

Since the visual odometer estimates the robot's movement and map based on neighboring images, its errors will inevitably accumulate over time. The task of loop detection is to detect that the robot has passed through the same place through sensor data, provide data for the back end, eliminate accumulated errors, and build a globally consistent trajectory and map [1], which is conducive to the correct operation and reconfiguration of the system over a long period of time. Positioning work.

Sivic et al. applied the bag-of-words model to closed-loop detection technology [27]. The basic idea is to cluster the feature points extracted from the image with the k-means algorithm into a word containing several "words", and then calculate the similarity with the target image based on the words contained in the tested image. Set a similarity threshold. If the similarity between the current image and a certain key frame is higher than 3 times the similarity between the current image and the previous key frame [1], it is considered that there may be a loop.

It can be seen that finding suitable image features is the basis of loop detection. In order to cope with loop detection in complex environments, researchers have conducted more extensive explorations.

ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

Oliva proposed the Gist descriptor [28], which uses Gabor filters to extract information from different directions and frequencies. Chen et al. first proposed the use of convolutional neural network (CNN) for location recognition [29], which opened the door for deep learning to be applied to SLAM closed-loop detection. Later, Arandjelovic [30], Lopez-Antequera [31], Naseer[32] and others fine-tuned the design of CNN to make it more suitable for closed-loop detection.

#### *1.2.4. Mapping*

There are many forms of maps. According to whether they accurately describe the location of objects in the map, they can be divided into two types: metric map and topological map. Because the topological map and the multi-reaction are the relationship between objects, which reduces the requirement for precise positional relationship, and is not suitable for the application research of SLAM [1], the mapping module mainly constructs the metric map. Among them, a map with only landmark points is called a sparse map, which can meet the needs of positioning; the corresponding one expresses all the objects seen by the camera, and can support the system to realize the functions of navigation, obstacle avoidance, and map reconstruction. In augmented reality, which includes the interaction function between people and the map, it is necessary to construct a further semantic map.

In the construction of dense maps, Forster et al. proposed the method of epiline search and block matching [33] to find the position of the pixel in different images, and use the depth filter [34] to determine the position of the point. However, in monocular vision or binocular vision, block matching is highly dependent on the texture of the object, and mismatching is prone to occur, which affects the mapping effect. Civera et al. proposed the concept of inverse depth [35] to be applied to SLAM, which achieved good numerical stability and was widely used in the existing SLAM framework [36].

After the appearance of the Kinect camera, SLAM research based on RGB-D has become a hot topic. Henry et al. first proposed the RGB-D SLAM framework [37], and used RGB-D images for three-dimensional reconstruction. The point cloud map can be easily generated from the RGB-D image, and the map can be displayed quickly. Then Poisson reconstruction [38] and surfel reconstruction [39] appeared, which made the map display better.

Point cloud maps have the disadvantages of large scale, waste of resources, and inability to handle moving objects. The octree map [40] can solve these problems. Divide the three-dimensional squares into eight evenly, and divide them layer by layer to form an octree. The octree structure is shown in the figure. In addition to being easy to compress and update, Burri et al. proposed a navigation method based on the octree [41] by using the feature of the octree to query the occupied points, which improved the navigation efficiency.

## 2. Several Schemes of Current SLAM research

MonoSLAM proposed by Davison et al. has a milestone significance in visual SLAM research. This is a monocular vision system, the back end adopts EKF method, can construct sparse map online. But the disadvantage is that it is easy to cause accumulated errors, the amount of calculation is large, and the application scenarios are small. PTAM was proposed by Klein and others, which created a dual-threaded structure for tracking and mapping, which had a very important influence on the later SLAM framework. And PTAM uses nonlinear optimization methods for the first time in the back-end, making researchers realize the huge potential of nonlinear optimization in the back-end of SLAM. PTAM also has the shortcomings of small application scenarios and easy tracking loss. ORB-SLAM [36] was first proposed by Mur-Artal et al. in 2015. It inherited and expanded the two-threaded structure of PTAM to a three-threaded structure. The effect is much better than that of PTAM. ORB-SLAM started as a monocular vision SLAM based on ORB features, and later ORB-SLAM2[42] expanded to binocular cameras and RGB-D cameras, which greatly broadened the scope of application. Due to the rotation invariance and scale invariance of ORB features, the system can still perform loop detection in a large range of motion. In 2020, ORB-SLAM3[43] was released, which expanded the IMU fast initialization algorithm based on maximum posterior estimation, which greatly improved the accuracy of the algorithm; the SLAM positioning algorithm based on multiple sub-maps was used for relocation and map fusion; support pinholes Camera model and fisheye camera model. It is currently the most accurate and comprehensive SLAM system.

ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

## 3. Problems and Development Trends

### *3.1. Problems*

1) Feature dependence

Visual SLAM has a very serious dependence on environmental features. Feature extraction and matching in weak texture environments, loop detection, etc. will all be affected by it. To solve this kind of problem, you can introduce more advanced features such as line features, surface features, super pixel features; another method is to add other types of sensors such as laser or IMU, which can combine the features of several different sensors to achieve higher Precision and robustness.

2) Real-time

SLAM is called real-time positioning and mapping technology, and real-time is always an issue that needs to be considered. This is related to the speed at which the robot completes tasks in an unfamiliar environment. To solve the real-time problem, it is necessary to consider the complexity of the algorithm, as well as the problem of reducing the computational complexity of large-scale scenes by means of sub-maps.

### *3.2. Future Development Trends*

1) Deep learning

As an equally popular research topic in recent years, deep learning has played an increasingly important role in the field of image processing, and each step of visual SLAM involves image processing. It can be seen that deep learning is also promising in visual SLAM. DeepVO [44] proposed by Wang et al. in 2017 is widely used in visual odometry. Lin Zhaohao et al. proposed a loop detection algorithm based on Mask_R_CNN [45], and merged with the traditional BoW algorithm, and achieved good results. In the construction of semantic map, Zhang Ting proposed to bring deep learning semantic segmentation into ORB-SLAM [46] to improve the positioning accuracy in a dynamic environment.

2) Multi-sensor fusion

Nowadays, there are various types of sensors that can be used in SLAM, including monocular, binocular, pinhole, fisheye, depth camera and laser, IMU, etc. How to better integrate these sensors to improve the accuracy and robustness of SLAM systems has become a recent trend Hot spot. The general method is by setting a sliding window. Methods were proposed to use deep learning to extract sensor features. In ORB-SLAM3, the IMU fast initialization based on maximum posterior estimation is used, so that the accuracy of the monocular and binocular inertial system is greatly improved compared with other methods.

3) Multi-robot collaboration

Multi-robot collaborative visual SLAM can overcome the problem of single viewing angle in indoor environment. The multi-drone cooperative SLAM system proposed by Schmuck[47] et al. uses a server to connect multiple drones, so that each drone can see the perspective of other drones, effectively improving a single drone. The human-machine trajectory has increased the scale of map construction.

## References

*[1] Gao X, Zhang T. Fourteen Lectures on Visual SLAM: From Theory to Practice [M]. BeiJing: Publishing House of Electronics Industry, 2019.*
*[2] Harris C, Stephens M. A combined corner and edge detector[J]. Alvey Vision Conference, 1988, 1988(3):147-151.*
*[3] Rosten E, Drummond T. Machine learning for high-speed corner detection[C]//Computer Vision – ECCV 2006. Berlin, Heidelberg：Springer, Berlin, Heidelberg, 2006: 430-443.*
*[4] Shi J B, Tomasi C. Good features to track[C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Seattle, WA, USA: IEEE, 1994: 593-600.*
*[5] Hu K, Wu J, Zheng F，et al．A Survey of visual odometry[J]Journal of Nanjing University of Information Science &Technology(Natural Science Edition), 2021, 13( 3): 269-280.*

ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

*[6] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110*
*[7] Bay H, Tuytelaars T, van Gool L. SURF: speeded up robust features[C]//Computer Vision – ECCV 2006. Berlin, Heidelberg: Springer, Berlin, Heidelberg, 2006: 404-417.*
*[8] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF[C]// International Conference on Computer Vision. Barcelona, Spain: IEEE, 2011:2564-2571.*
*[9] Lu Y, Song D Z. Robust RGB-D odometry using point and line features[C]// IEEE International Conference on Computer Vision (ICCV), 2015: 3934-3942*
*[10] Pumarola A, Vakhitov A, Agudo A, et al. PL-SLAM: Real-time monocular visual SLAM with points and lines[C]// IEEE International Conference on Robotics and Automation (ICRA), 2017: 4503-4508.*
*[11] Concha A, Civera J. Using superpixels in monocular SLAM[C]. 2014 IEEE International Conference on Robotics and Automation. Hong Kong: IEEE, 2014: 365-372*
*[12] Forster C. Pizzoli M., Scaramuzza D. SVO: Fast semi-direct monocular visual odometry[C]// 2014 IEEE International Conference on Robotics and Automation. Hong Kong: IEEE, 2014: 15-22.*
*[13] Engel J, Schöps T, Cremers D. LSD-SLAM: Large-Scale Direct Monocular SLAM[C]//Computer Vision – ECCV 2014. Cham: Springer International Publishing, 2014: 834-849.*
*[14] Engel J, Koltun V, Cremers D. Direct sparse odometry[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(3): 611-625*
*[15] Forster C, Zhang Z C, Michael G, et al. SVO: semidirect visual odometry for monocular and multicamera systems [J]. IEEE Trans on Robotics, 2017, 33(2): 249-265*
*[16] Sun H, Tong Z, Tang S, et al. SLAM Research Based on Kalman and Particle Filter [J]. Software Guide, 2018, 17(12): 1–4.*
*[17] Smith R C, Cheeseman P. On the Representation and Estimation of Spatial Uncertainty [J]. International Journal of Robotics Research, 1986, 5(4): 56-68*
*[18] DEMIM F, NEMRA A, LOUADJ K. Robust SVSF-SLAM for Unmanned Vehicle in Unknown Environment [J]. IFAC-PapersOnLine, 2016, 49(21): 386-94.*
*[19] Murphy P. Bayesian Map Learning in Dynamic Environments[C]//Neural Information Processing Systems. Denver, USA: NIPS, 1999: 1015-1021*
*[20] Liang M, Min H, Luo R. Graph-based SLAM: A Survey [J]. Robot, 2013, 35(4): 500-512.*
*[21] KüMMERLE R, GRISETTI G, STRASDAT H, et al. G2o: A general framework for graph optimization[C] //2011 IEEE International Conference on Robotics and Automation. Shanghai, China: IEEE, 2011: 3607-3613.*
*[22] Lu F, Milios E. Globally consistent range scan alignment for environment mapping[J]. Autonomous robots, 1997, 4(4): 333-349.*
*[23] Duckett T, Marsland S, Shapiro J. Learning globally consistent maps by relaxation[C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2000: 3841-3846.*
*[24] Olson E, Leonard J, Teller S. Fast iterative alignment of pose graphs with poor initial estimates[C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2006: 2262-2269.*
*[25] Grisetti G, Stachniss C, Grzonka S, et al. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent[M]//Robotics: Science and Systems III. Cambridge, USA: MIT Press, 2008: 65-72*
*[26] Grisetti G, Kummerle R, Stachniss C, et al. Hierarchical optimization on manifolds for online 2D and 3D mapping[C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2010: 273-278.*
*[27] Sivic J, Zisserman A. Video Google: A text retrieval approach to object matching in videos[C]//IEEE International Conference on Computer Vision. Piscataway, USA: IEEE Computer Society, 2003: 1470-1477.*
*[28] Oliva A, Torralba A. Building the gist of a scene: The role of global image features in recognition[J]. Progress in Brain Research, 2006, 155(2): 23-36.*
*[29] Chen Z., Lam O., Jacobson A., et al, Convolutional neural network-based place recognition[C]. 2014 Australasian Conference on Robotics and Automation, Melbourne, Australia: ACRA, 2014: 1–8*
*[30] Arandjelovic R, Gronat P, Torii A, et al. NetVLAD: CNN architecture for weakly supervised place recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, USA: IEEE, 2016: 5297-5307.*
*[31] Lopez-Antequera M, Gomez-Ojeda R, Petkov N, et al. Appearance-invariant place recognition by discriminatively training a convolutional neural network[J]. Pattern Recognition Letters, 2017, 92(1): 89-95.*

ISSN 2706-655X Vol.3, Issue 7: 1-7, DOI: 10.25236/IJFET.2021.030701

*[32] Naseer T, Oliveira G L, Brox T, et al. Semantics-aware visual localization under challenging perceptual conditions[C]. IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2017: 2614-262.*

*[33] Pizzoli M, Forster C, Scaramuzza D. REMODE: Probabilistic, monocular dense reconstruction in real time[C]. 2014 IEEE International Conference on Robotics and Automation. Hong Kong: IEEE，2014: 2609–2616.*

*[34] Vogiatzis G., Hernández C. Video-based, Real-Time Multi View Stereo [J]. Image and Vision Computing, 2011, 29(7):434-441.*

*[35] Civera, J.; Davison, A.J.; Montiel, J. Inverse Depth Parametrization for Monocular SLAM [J]. IEEE Transactions on robotics, 2008, 24(5): 932–945.*

*[36] Mur-artal R, Montiel J M M, Tardós J D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System [J]. IEEE Transactions on Robotics, 2015, 31(5): 1147-63.*

*[37] Henry P，Krainin M，Herbst E，et al. RGB-D mapping：Using Kinect-style depth cameras for dense 3D modeling of indoor environments[J]. International Journal of Robotics Research, 2012, 31(5): 647-663.*

*[38] Kazhdan M, Bolitho M, Hoppe H. Poisson surface reconstruction [C]. The fourth Eurographics symposium on Geometry processing. Cagliari, Sardinia, Italy; Eurographics Association. 2006: 61–70.*

*[39] Stuckler J, Behnke S. Multi-resolution surfel maps for efficient dense 3D modeling and tracking [J]. Journal of Visual Communication and Image Representation, 2014, 25(1): 137-47.*

*[40] Hornung A, Wurm K M, Bennewitz M, et al. OctoMap: an efficient probabilistic 3D mapping framework based on octrees [J]. Autonomous Robots, 2013, 34(3): 189-206.*

*[41] Burri M, Oleynikova H, Achtelik M W, et al. Real-time visual-inertial mapping, re-localization and planning onboard MAVs in unknown environments[C]// 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems. Hamburg, Germany: IEEE, 2015: 1872-1878.*

*[42] MUR-ARTAL R, TARDóS J D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras [J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-62.*

*[43] CAMPOS C, ELVIRA R, RODRíGUEZ J J G, et al. ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM [J]. IEEE Transactions on Robotics, 2021, 1-17.*

*[44] Wang S, Clark R, Wen H, et al. DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks[C]. IEEE International Conference on Robotics and Automation. Singapore, 2017: 2043-2050.*

*[45] Lin Z, Xu Y. Loop detection algorithm based on Mask R-CNN[J]. Electronics and Software Engineering, 2021, 05): 71-3.*

*[46] Zhang T, Cai Y, Chen L. Construction of Visual Semantic Map Based on Deep Learning [J]. industrial control computer;process computer, 2020, 33(11): 94-96.*

*[47] Schmuck P, Chli M. Multi-UAV collaborative monocular SLAM[C]. IEEE International Conference on Robotics and Automation. Singapore, 2017:3863-3870.*