2021 14th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI) | 978-1-6654-0004-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/CISP-BMEI53629.2021.9624443

A New Semantic SLAM Mapping Algorithm Based on Improved YOLOv5

Weixiang Shen, Yongxing Jia*,Mingcan Li, Junchao Zhu
College of Communications Engineering
Army Engineering University of PLA
Nanjing, China

Abstract—**Visual SLAM (V-SLAM) uses cameras for information input. In mapping, the spatial geometric information of the point cloud is used, which lacks the semantic information of the objects in the environment. This paper proposes a new semantic mapping algorithm based on improved YOLOv5. Firstly, A Pyramid Scene Parsing Network (PSPNet) segmentation head is added to YOLOv5 for performing semantic extraction of the environment. Subsequently, the robot pose is estimated with the ORB-SLAM2 framework. Finally, the semantic images, the depth images and the pose transformation matrix are sent to a mapping module to fuse a dense point cloud semantic map. Experiments show that the algorithm in this paper builds an accurate semantic map on KITTI dataset. Combined with the depth map that eliminates interference factors, it has good accuracy and robustness for semantic mapping in large-scale scenarios.**

Keywords- SLAM, YOLOv5, Semantics, Mapping

I. Introduction

Simultaneous Localization and Mapping (SLAM) is the process of autonomous localization and mapping of the robot during robot motion in unknown environments through its sensors. The robot builds an incremental map base on the positioning. SLAM is also known as the key technology of modern robot intelligence. There are many types of SLAM systems. V-SLAM is a SLAM system that mainly takes the image as the source of information perception. V-SLAM is widely used because of its low cost, simple sensor, small size, cost-effectiveness, and large amount of available information. In the early stage, V-SLAM used an extended Kalman filter to optimize the camera pose estimation and the accuracy of map construction. With computing ability and algorithm improvement, BA optimization, pose optimization, and other methods gradually become the mainstream.

Although the environmental map which is built by the traditional SLAM system meets the positioning needs of the robot to some extent, it is not enough to support the robot to complete the navigation and obstacle avoidance tasks independently, and its interactive ability is insufficient. Semantic SLAM plays a pivotal role in modern SLAM. It not only obtains the geometric structure information in the environment when it is mapping, but also identifies independent individuals in the environment to obtain semantic information such as its position, posture, and functional properties in order to handle complex scenes and accomplish more intelligent tasks. SemanticFusion [1] implements the semantic map building of the indoor environment. It becomes a classical algorithm for building pixel-level semantic maps and constructs a complete semantic SLAM system that combines the traditional SLAM framework with semantic segmentation. It optimizes the segmentation effect of Convolutional Neural Networks (CNN) for single pictures using the feature point matching relationship output in the SLAM system, which significantly improves the accuracy of semantic maps. But this algorithm requires significant computational resources and lacks real-time performance. DA-RNN [2] completes the semantic information extraction of the video frames and integrates the semantic information with the 3D map built by KinectFusion [3]. However, it still needs significant computational resources. At the same time, although the algorithm improves the accuracy of semantic segmentation to a certain extent, the improvement effect is limited. For this problem, the researchers propose some lightweight semantic map building methods. SEMANTIC-RTAB-MAP (SRM) for semantic map building is proposed based on RTABMAP [4] and You only look once (YOLO) [5]. Instead of directly using the deep learning method, it uses the YOLO V2 algorithm to estimate the object position roughly. Then it detects the edge of the object on the deep image using the Canny operator and processes the boundaries based on the regional growth algorithm to complete accurate object segmentation. This algorithm improves the real-time performance of semantic map building. The building of scene-oriented semantic map algorithms mainly uses deep learning methods to map 2D semantic information to 3D point clouds. At present, the relevant research of algorithms mainly focuses on semantic segmentation methods and semantic fusion methods.

On the one hand, from the perspective of map application, the scene-oriented semantic map can assist the robot with a better understanding of the environment. On the other hand, considering the effect of the algorithm, such algorithms need pixel-level semantic segmentation of all objects in the scene. The semantic segmentation performance of algorithms is good, and they can build high-precision semantic maps. So this paper proposes a new semantic SLAM mapping algorithm based on improved YOLOv5, which uses RGB-D camera data to build an outdoor semantic dense 3D point cloud map. The rest of this paper is organized as follows. The related work of each module is presented in Section II, the main modules of the proposed system are introduced in Section III, the algorithm and some modules are evaluated in Section IV, this paper is summarized in Section V.

978-1-6654-0004-6/21/$31.00 ©2021 IEEE

II. Related Work

A. Semantic Extractor

Accurate detection and classification of objects facilitate the building of precision maps. The challenge of obtaining semantic information lies in accurately getting object pixel-level classification. With the rapid development of deep learning, traditional target detection algorithms based on manually extracted features gradually turn to those found on deep neural networks. In conventional machine vision, feature points and descriptors are usually used for part and object recognition, such as SIFT, SURF, FAST, etc. R-CNN [6] is turned out to be the first algorithm to apply deep learning to object detection. It draws on the idea of the sliding window method, recognizes regions by region, and uses CNN to extract a fixed length for each area. R-CNN uses Support Vector Machine (SVM) for object classification, which causes it to take up a considerable amount of memory and use many calculations. Fast R-CNN [7] abandons multiple SVM classifiers based on R-CNN and combines classification with regression. As a single network, Fast R-CNN realizes end-to-end training and improves the speed of the original RCNN. Based on Fast-RCNN, Faster-RCNN [8] uses RPN instead of the initial selective search for feature extraction. It integrates feature extraction, proposed extraction, boundary box regression and classification into a network, and its detection speed can improv significantly. With the continuous update of object detection algorithms, mainstream methods include Mask R-CNN [9], YOLO, SDD [10], etc. Mask R-CNN is a small and flexible general object instance segmentation framework, which uses the idea of Faster-RCNN. Its feature extraction uses the ResNet-FPN architecture and adds a mask prediction branch. The algorithm used in this paper is to add a PSPNet [11] segmentation head based on YOLOv5. Experiments shows that it can achieve real-time semantic segmentation and object detection.

B. Advanced SLAM Framework

At present, most pose estimation methods rely on traditional methods, such as geometric constraints and nonlinear optimization. MonoSLAM [12] is the first real-time monocular vision SLAM system. Mono-SLAM uses a monocular camera to recover the 3D trajectory moved by the robot in an unknown environment. The application scenario of Mono-SLAM is narrow. Its limited number of road signs and its sparse feature points are easily lost. PTAM [13] proposes the front-end and back-end for the first time, which implements parallelization of tracking during mapping. PTAM uses nonlinear optimization instead of the traditional filter as the back-end, and it introduces a keyframe mechanism that significantly reduces the data volume processing time. Due to the emergence of PTAM, the backend optimization method of V-SLAM gradually shifts to nonlinear optimization. PTAM can only be used for a small scene, and its tracking thread is easy to lose. Nowadays, the development of V-SLAM has makes significant progress, and many excellent algorithms are emerged, such as Semi-Direct Monocular Visual Odometry (SVO) [14], Large -Scale Direct monocular SLAM (LSD-SLAM) [15], Direct Sparse Odometry (DSO) [16], etc. ORB-SLAM2 [17] inherits PTAM excellently and improves the computation speed of the system by approximately ten times by extracting ORB features as an observation symbol. This measure meets the requirements of real-time operation. Currently, ORB-SLAM2 is one of the most popular SLAM frameworks.

C. Mapping Module

Outputting an accurate environmental model is one of the goals of the SLAM system. Under the constraints of the localization data, the mapping module integrates the original observed image or its feature points into the virtual space to complete exact matching and de-duplication. Many classic SLAM systems only use the spatial geometric information of the point cloud to build sparse and dense maps. The light point cloud map abstracts the map and works only on the robot's localization.

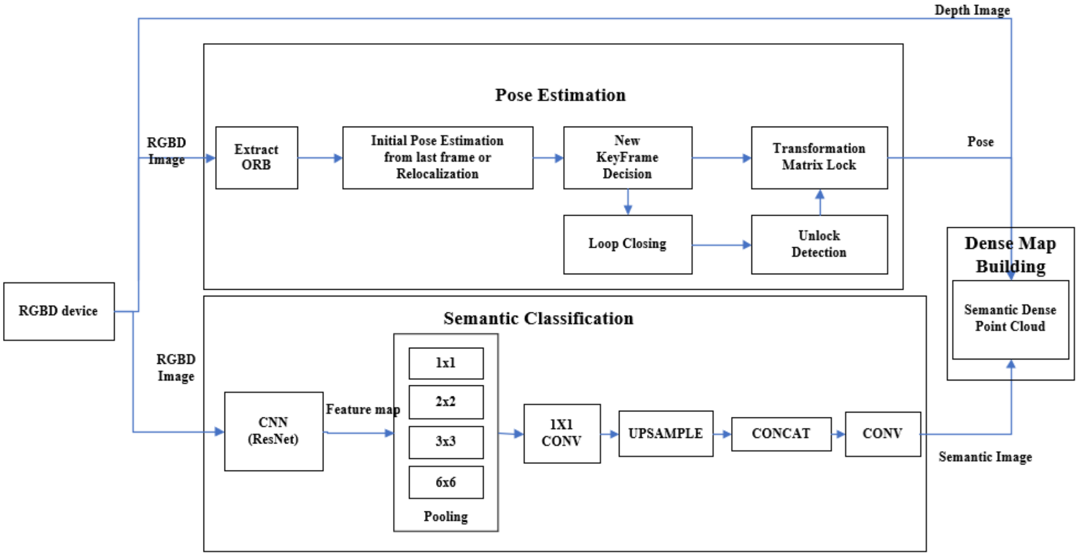

Figure 1. The System Outline

Normal dense point cloud map lacks semantic information about objects in the environment. The system in this paper uses a mapping module based on the improved YOLOv5, which can build accurate semantic maps in a large-scale environment.

III. Porposed Methed

In many robot applications, it is necessary to obtain the semantic information and geometry of the surrounding environment, especially autonomous driving and automatic obstacle avoidance. As the carrier of environmental semantic information, the function and application of map have been extensively studied. The algorithm used in this paper builds the dense point cloud semantic maps based on the improved YOLOv5 to realize the pixel-level description. As shown in Fig.1, the system is mainly divided into three modules. The three modules are described in detail as follow.

A. Semantic Classification Module

In this module, RGB images need to be classified at the pixel level. It is necessary to choose a state-of-the-art (SOTA) deep learning method as the core of the module. YOLOv5 is selected as the semantic extractor backbone after comparing the excellent current algorithms. YOLOv5 is the v5 version of YOLO. It is an object recognition and positioning algorithm based on deep neural networks which can be used in real-time systems. YOLOv5 mainly has four versions. The basic version is YOLOv5s, and other versions have widened and deepened the network of this version to varying degrees. Take YOLOv5s as an example. Its network structure mainly includes four aspects: Input, Backbone, Neck, and Prediction. It uses the Mosaic data augmentation operation on the input side, and it uses random scaling, random cropping, and random arrangement methods for splicing to improve the training speed of the model and the accuracy of the network. YOLOv5 proposes an adaptive anchor box calculation and an adaptive image scaling method. In the previous version of YOLO, the anchor box uses different programs to calculate other datasets. It combines anchor box and object detection to adapt to the optimal anchor box for object detection automatically. The improved adaptive image scaling algorithm uses minimal black borders during scaling and filling.

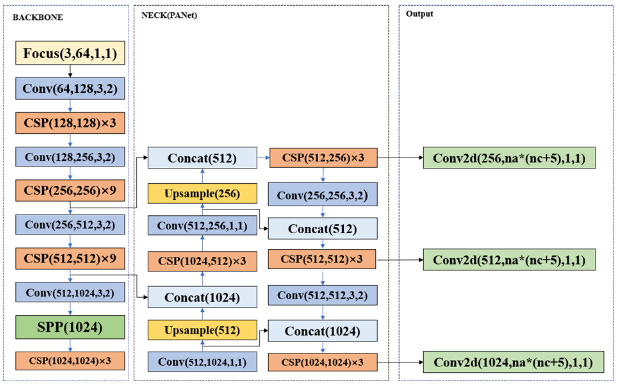

Figure 2. The network structure of YOLOv5s

YOLOv5 adds a Focus structure in Backbone more than YOLOv4. In YOLOv5s, the input picture (608*608*3) is the first image sliced (304*304*32), and then 32 convolution kernels are used for convolution to turn it into a feature map with a specification of 304*304*32. It uses the Cross Stage Partial Network (CSPNet) structure. It reduces the amount of calculation and optimizes the redundancy of the network gradient information. Furthermore, to improve the learning power of the convolutional network, Spatial Pyramid Pooling (SPP) performs multi-scale fusion with 1 × 1,5 × 5,9 × 9,13 ×13 maximum pooling methods. It uses the FAN+PAN structure in Neck and adopts the CSP2 structure designed for reference from CSPNet to strengthen the ability of network feature integration. Fig. 2 shows the network structure of YOLOv5s.

GIoU is used as the loss function of the Bounding box in YOLOv5. Combining (1) and (2), GIoU defines that A and B are two convex shapes. C is the smallest convex shape containing A and B. For the matrix box, C is the smallest matrix box containing A and B. GIoU is calculated the ratio of C not covering A and B to the total area of C, and IoU of A and B is used to subtract this ratio.

$$A, B \subseteq s \in R^n, C \subseteq s \in R^n \tag{1}$$

$$GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|} \tag{2}$$

During the post-processing of object detection, YOLOv5 uses a Gaussian weighting non-maximum suppression (NMS) method to filter multiple object frames. It can be seen in (3). In the original NMS, the scores for windows with IoU above the threshold are set to zero.

$$s_i = \begin{cases} s_i, & \text{IoU}\left(\mathcal{M}, b_i\right) < N_t \\ 0, & \text{IoU}\left(\mathcal{M}, b_i\right) \geq N_t \end{cases} \tag{3}$$

In the modified Gaussian-weighted NMS, is the anchor boxes with the highest current score and is the anchor boxes to be processed in (3). The more the score of IoU of and, the more the decrease in the score of. This algorithm in this paper sorts the different anchor boxes by their confidence levels instead of directly deleting them. The correct anchor box can be better selected in this mechanism, and errors and lost choices are largely avoided.

$$s_i = s_i e^{-\frac{IoU\left(\mathcal{M}, b_i\right)^2}{\sigma}}, \forall b_i \notin \mathcal{D} \tag{4}$$

YOLOv5 cannot complete the semantic segmentation task, this paper integrates the PSPNet segmentation layer into YOLOv5, and the object is segmented more accurately at the pixel level.

B. Pose Estimation Module

To make the obtained camera pose more accurate, ORB-SLAM2 that the advanced SLAM algorithm is used in this paper. It adds support for stereo cameras and RGB-D cameras based on ORB-SLAM. ORB-SLAM2 is a front-end visual odometer based on the feature point method, a three-thread parallel complete SLAM system. This system includes ORB feature extraction and feature matching tracking threads. After the ORB features are extracted, the Iterative Closest Point (ICP) is used to match the feature points between the image frames, and the centroid positions of the two matched 3D points are calculated in (5). Combining (6) and (7)，the calculation of the rotation matrix R^* and the translation vector t^* is completed through the de-centroid coordinates of each issue.

$$q_i = p_i - p, q_i^{'} = p_i^{'} - p' \tag{5}$$

$$R^* = \arg\min_R \frac{1}{2}\sum \left\| q_i - Rq_i^{'} \right\|^2 \tag{6}$$

$$t^* = p - Rp' \tag{7}$$

The least-square method (Bundle Adjustment) is used to minimize the projection error, reduce the posture jitter, and optimize the camera posture. After that, the original mapping module is discarded, and the mapping process is carried out in the dense semantic mapping module.

C. Dense Map Building Module

The RGB-D camera can collect color images and depth images at the same time. The formation of dense point cloud images requires a combination of camera pose, color map, and depth map. The translation vector and the quaternion constitute the camera external parameter pose. The coordinate points in the three-dimensional world are used to map to the two-dimensional pixel plane in order to obtain the coordinate, the distance of each pixel in the real world, and the camera internal parameter. The point cloud space coordinates are obtained through (8) and (9). The dense semantic map is formed by splicing the point clouds.

$$u = f_x \frac{X}{Z} + c_x \tag{8}$$

$$v = f_y \frac{Y}{Z} + c_y \tag{9}$$

IV. Experiment Result and Analysis

The experimental environment is Ubuntu 18.04 system, Inter Core i9 3.3GHz CPU, Nvidia RTX5000 GPU, 128GB memory. The algorithm is implemented in C++ and Python in the ROS dynamic environment. Before assembling a complete system, this paper tests the functional modules and compares them with other SOTA algorithms to complete the overall real-time and accuracy evaluation. Three semantic extractors are trained on the COCO dataset, and a segmentation model for outdoor scenes is obtained by comparison. These semantic extractors are tested on the KITTI dataset.

A. Comparison of Semantic Extractor

Considering the real-time nature of the SLAM system and the accuracy of mapping, TABLE I tests the object detection accuracy and running speed of the YOLO series algorithms to select the appropriate backbone of the algorithm.

YOLOv5s has a simple structure and can be used in real-time conditions. It is only 28.99MB, which is 11.7% of YOLOv4 [18], and the object detection accuracy is 15.9% higher than YOLOv3-Tiny [19]. YOLOv5s is selected as the framework of the algorithm.

TABLE I. Yolo Algorithm Comparison

Method	MAP@0.5 (%)	FPS	Size (MB)
YOLOv3-Tiny	34.8	200	33.97
YOLOv4	56.5	65	246.19
YOLOv5s	55.4	140	28.99

Mean Intersection over Union (mIoU) is an essential index of semantic segmentation performance. TABLE II provides the mIoU data of different algorithms based on deep learning, and this paper trains them under the COCO dataset and verifies them on the KITTI dataset. It can be seen in TABLE II that the network structure with the segmentation head of PSPNet has a better mIoU. In general, the performance gap between the standard semantic segmentation algorithm and the algorithm provided in this paper is the largest, by 13.8%. However, under the YOLOv5 basic framework, the performance difference of the three algorithms with different segmentation heads is slight, and the difference between the highest and the lowest is only 4.8%. Experiments show this proposed algorithm expands the receptive image field by increasing a small amount of explicit memory and computation while deepening the basic YOLOv5 detection layer and adding the PSPNet segmentation layer. It has an excellent performance in the experiment. Adding a semantic segmentation head to the object detection algorithm helps to improve the semantic segmentation ability and more accurately segment the object at the pixel level.

TABLE II. Algorithmic Performance Comparison

Algorithm	Segmentation Head	Backbone	MIoU(%)
DeeplabV3+	-	Resnet101	59.7
PSPNet	-	Resnet101	68.6
BiSeNetv2	-	Resnet101	68.4
YOLOv5s	DeeplabV3+	Resnet101	68.3
YOLOv5s	BiSeNetv2	Resnet101	69.4
YOLOv5s	PSPNet	Resnet101	73.5

The first row of Fig. 3 is selected KITTI original dataset, the second and third rows are the object detection and semantic segmentation results based on the YOLOv5s segmentation layer of DeeplabV3+ [20], the fourth and fifth rows are the object detection and semantic segmentation results based on the YOLOv5s segmentation layer of BiSeNetv2 [21], and the sixth and seventh rows are the object detection and semantic segmentation results based on the YOLOv5s segmentation layer of PSPNet. From the images in the experimental results, it can be seen with the naked eye that the YOLOv5s with PSPNet as the segmentation head performs best in the dataset.

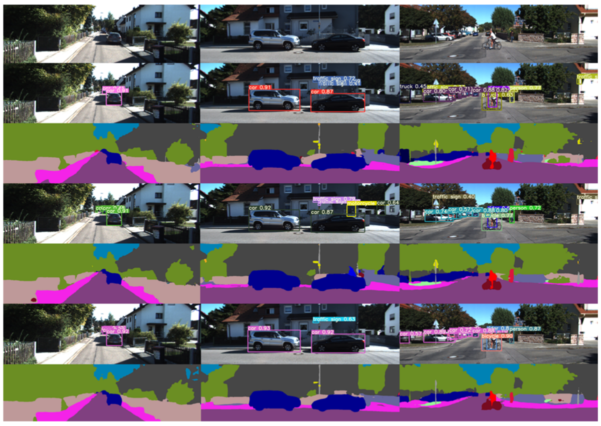

Figure 3. Performance presentation of the different algorithms

B. *Comparison of Mapping*

Fig. 4 is a semantic map obtained by excluding the sky part of the depth map. During the mapping process, due to the sky and other influencing factors (picture on the right), there will be shadows in the roads on the map, which will block the observation of streets and obstacles. When the influencing factors are removed (left picture), the road is clearly visited.

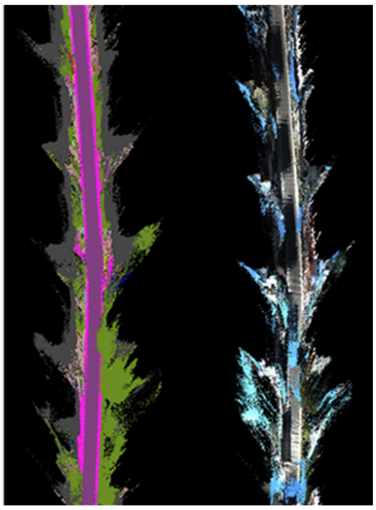

Figure 4. Semantic images with no distractions

Fig. 5 is a comparison between the dense point cloud map and the dense point cloud semantic map. The comparison shows that the dense point cloud map can accurately identify objects. A clear car can be seen in the matrix box of the picture. The dense point cloud semantic map has clear roads, trees and specific object semantic information.

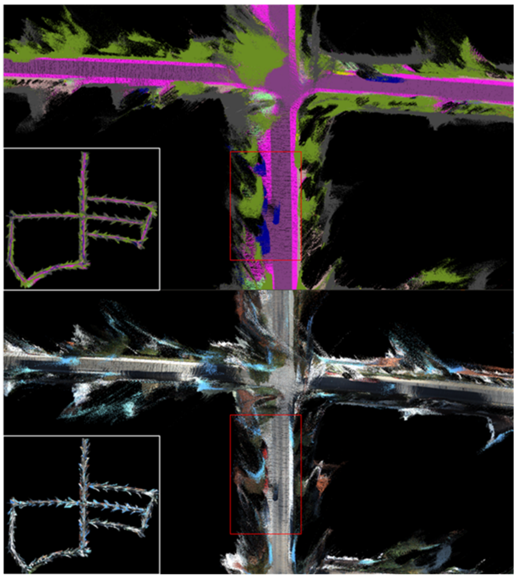

Figure 5. Precise semantic image on KITTI dataset

V. Conclusion

This paper proposes a new semantic mapping algorithm based on improved YOLOv5. First, an excellent semantic extractor is trained based on the COCO dataset to obtain the observed image and perform pixel-level object classification. At the same time, the RGB-D data is input into the ORB-SLAM2 framework to get real-time pose estimation. Subsequently, three-dimensional point cloud dense maps are built using pose estimation and depth maps of excluding interference factors. This paper selects an appropriate algorithm based on the evaluation and visual effects to achieve accurate semantic segmentation, which plays a crucial role in building an accurate map. On the other hand, the map building module realizes accurate large-scale outdoor semantic 3D mapping with good application prospects in robot navigation. The system uses a loosely coupled method to connect each module. Therefore, with this structure, better object detection and semantic segmentation algorithms can replace the content in the module at any time. Experiments show that the algorithm in this paper has good robustness and practicability in the building of dense semantic maps in large environments.

References

[1] McCormac J, Handa A, Davison A, et al. SemanticFusion: dense 3D semantic mapping with convolutional neural networks // 2017 IEEE International Conference on Robotics and automation (ICRA). Singapore, 2017: 4628

[2] Xiang Y, Fox D. DA-RNN: semantic mapping with data associated recurrent neural networks. arXiv preprint (2017-05-30) [2020-11-09]. https://arxiv.org/abs/1703.03098v2

[3] Izadi S, Kim D, Hilliges O, et al. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera // Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology. Santa Barbara, 2011: 559

[4] Labbe M, Michaud F. Appearance-based loop closure detection for online large-scale and long-term operation. IEEE Trans Rob, 2013, 29（3）: 734

[5] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779-788.

[6] Girshick R , Donahue J , Darrell T , et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation// IEEE Computer Society. IEEE Computer Society, 2013.

[7] R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 1440-1448.

[8] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017.

[9] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2980-2988.

[10] Liu W , Anguelov D , Erhan D , et al. SSD: Single Shot MultiBox Detector. European Conference on Computer Vision, 2016

[11] H. Zhao, J. Shi, X. Qi, X. Wang and J. Jia, "Pyramid Scene Parsing Network," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 6230-6239.

[12] Davison A J , Reid I D , Molton N D , et al. MonoSLAM: real-time single camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(6):1052-1067.

[13] Klein G , Murray D . Parallel Tracking and Mapping for Small AR Workspaces// IEEE & Acm International Symposium on Mixed & Augmented Reality. ACM, 2008.

[14] Forster C , Pizzoli M , D Scaramuzza*. SVO: Fast semi-direct monocular visual odometry// IEEE International Conference on Robotics & Automation. IEEE, 2014.

[15] Engel J , Schps T , Cremers D . LSD-SLAM: Large-scale direct monocular SLAM// European Conference on Computer Vision. Springer, Cham, 2014

[16] Direct Sparse Odometry. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2016:1-1.

[17] RMur-Artal, TArDos J D . ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras. 2016.

[18] Bochkovskiy A , Wang C Y , Liao H . YOLOv4: Optimal Speed and Accuracy of Object Detection. 2020.

[19] Redmon J , Farhadi A . YOLOv3: An Incremental Improvement. arXiv e-prints, 2018.

[20] Chen L C , Papandreou G , Kokkinos I , et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4):834-848.

[21] Yu C , Wang J , Peng C , et al. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation// European Conference on Computer Vision. Springer, Cham, 2018.