IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 7, 2020, accepted November 24, 2020, date of publication November 27, 2020,
date of current version December 10, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3041038

PSPNet-SLAM: A Semantic SLAM Detect Dynamic Object by Pyramid Scene Parsing Network

XUDONG LONG, WEIWEI ZHANG, AND BO ZHAO
School of Mechanical and Automotive Engineering, Shanghai University of Engineering Science, Shanghai 201620, China

Corresponding author: Weiwei Zhang (zwwsues@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 51805312, in part by the Shanghai Sailing Program under Grant 18YF1409400, in part by the Training and Funding Program of Shanghai College young teachers under Grant ZZGCD15102, in part by the Scientific Research Project of Shanghai University of Engineering Science under Grant 2016-19, in part by the Science and Technology Commission of Shanghai Municipality under Grant 19030501100, and in part by the Shanghai University of Engineering Science Innovation Fund for Graduate Students under Grant 18KY0613.

ABSTRACT Simultaneous Localization and Mapping (SLAM) plays an important role in the computer vision and robotic field. The traditional SLAM framework adopts a strong static world assumption for convenience of analysis. It is very essential to know how to deal with the dynamic environment in the entire industry with widespread attention. Faced with these challenges, researchers consider introducing semantic information to collaboratively solve dynamic objects in the scene. So, in this paper, we proposed a PSPNet-SLAM: Pyramid Scene Parsing Network SLAM, which integrated the Semantic thread of pyramid structure and geometric threads of reverse ant colony search strategy into ORB-SLAM2. In the proposed system, a pyramid-structured PSPNet was used for semantic thread to segment dynamic objects in combination with context information. In the geometric thread, we proposed a OCMulti-View Geometry thread. On the one hand, the optimal error compensation homography matrix was designed to improve the accuracy of dynamic point detection. On the other hand, we came up with a reverse ant colony collection strategy to enhance the real-time performance of the system and reduce its time consumption during the detection of dynamic objects. We have evaluated our SLAM in public data sheets and real-time world and compared it with ORB-SLAM2, DynaSLAM. Many improvements have been achieved in this system including location accuracy in high-dynamic scenarios, which also outperformed the other four state-of-the-art SLAM systems coping with the dynamic environments. The real-time performance has been delivered, compared with the geometric thread of the excellent DynaSALM system.

INDEX TERMS PSPNet-SLAM, dynamic, semantic, OCMulti-view geometry.

I. INTRODUCTION

SLAM is a cutting-edge relevant technology in the field of robot movement. When a robot collects data information from the surrounding environment through sensors, it uses relevant effective information to conduct self-positioning and surrounding environment map construction. An interdependent relationship between map construction and positioning can be found here as a continuous iterative process. Accurate positioning depends on a correct map and construct it as required. In the process, the continuous optimization algorithm and the loop detection map accuracy are used to correct the scale drift in the re-access to a certain position. At present, according to the collected information of sensor slam can be divided into laser radar SLAM and visual SLAM. Although the laser radar applied in the slam technology shows the advantages of high precision and reliability, its expensiveness and much increased information demand, gradually lead it to the SLAM technology based on vision sensors for the development direction of industry products which fall to the ground.

With the development of CPU and GPU, more and more powerful capabilities of graphics processing have been shown. As camera is not only cheap, but also lightweight and reliable, it has been used as the data acquisition sensor of visual SLAM, which has been seen the rapid development in the past decade. The camera can be detailly divided into the monocular camera, stereo camera, RGB-D camera, etc.

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Shorif Uddin.

However, it is found that real depth can't be measured by the stereo camera and can only be used to calculate the depth through calibration, correction and matching, which will waste a lot of computing resources. The RGB-D camera can simply and directly calculate the depth through its stereo, structured light and TOF technology.

In recent years, the field of visual SLAM has attracted a large number of researchers with emergence of many excellent SLAM system frameworks such as MonoSLAM [1], ORB-SLAM [2], ORB-SLAM2 [3], LSD-SLAM [4], SVO [5], DynaSLAM [6], which can achieve satisfactory performance while mobile robots are used in a static environment or some dynamic elements moves in space. Although these excellent SLAM systems currently perform well in ideal static environments to precisely locate and map something, them, they are still required to be test in our reality space (indoor and outdoor) where exists numerous moving objects. for example, walkers, animals, or other dynamic objects. The accuracy of LSD-SLAM, ORB-SLAM, ORB-SLAM2 and other systems in the real dynamic space is significantly reduced, or even the test system completely fail toward them.

In this paper, we propose a real-time parallel semantic SLAM system to deal with the problem of dynamic objects faced by the running robot. The system based on the ORB-SLAM2 algorithm framework, adopts semantic segmentation with a multi-view combination method to extract dynamic objects through the establishment of parallel semantic thread. In the semantic thread, we used an efficient PSPNet [34] to segment dynamic objects in which a pyramid structure neural net was designed for connecting contextual information. In the position estimation and dynamic object detection threads of low-cost tracking and multi-view Geometry, we design an OCMulti-View Geometry thread. The dual thread collaboratively works to extract the dynamic objects in the scene, so as to improve the accuracy of the self-positioning of the SALM system with more real-time performance and more robustness of dynamic point detection.

In summary, we highlight our main contribution below:

- We proposed the algorithm framework of PSNet-SLAM, and introduced the PSPNet network of the pyramid structure as a parallel semantic thread on the basis of ORB-SLAM2. The use of the network can effectively utilize the characteristics of context information, so that we can segment dynamic objects in continuous frames more quickly and reliably.
- The optimal compensation homography matrix is proposed in the geometry thread, which compensates for the position offset and lack of feature points in the front and rear frames in the projection transformation, and optimizes the position of the projection point. Improve the robustness of system performance.
- In the process of determining dynamic feature points, a reverse ant colony search strategy is proposed, which uses the characteristics of community distribution of dynamic feature points to search on a pre-set route. When a dynamic feature point is detected, it will shift to the dynamic feature point community, which avoids a dynamic and static judgment on all feature points, saves time consumption and improves the real-time performance of the system.

In the rest of this paper, we discuss the related works first. Then, the proposed system is described in detail. The experimental results are detailly explained in the third part. Finally, the paper is concluded.

II. RELATE WORK

At present, the SLAM framework can be divided into two major categories according to the type of data acquisition sensor: The first type is laser SLAM that uses lidar as the sensor. In this field, mapping [7] is a typical SLAM algorithm based on Rao-Blackwellized Particle Filters. Google's Cartographer [8] is the newest SLAM algorithm based on Lidar input, which provides a good loop closure detection.

Visual SLAM divides surrounding obstacles into two categories according to the movement attributes of static and dynamic objects. In the scene with only static objects, more famous cases can be listed such as MonoSLAM [1], PTAM [9], ORB-SLAM [2], which use ORB feature to detect feature points. Later, Mur-Artal [3] proposed the ORB-SLAM2 algorithm, which increases the accuracy of object detection and map construction. These SLAM frameworks can perform well in a static environment during the experiments.

The position of dynamic objects is detected and judged through geometric information in traditional solutions. For instance, A. kundu, K.M. Krishna *et al.* [10] estimates the distance between the matching feature and the epipolar line in the next frame of the image under the use of fundamental matrix. When the distance reached a predetermined threshold, the object was considered as a dynamic one. CoSLAM [11] uses the triangulation consistency between the two frames to project the feature points from the previous image into the current one, and calculated its error of reprojection. When the value was less than the threshold, it was judged as a static feature point, otherwise dynamic feature points. Piaggio, Fornaro *et al.* [12], Chivil, Mezzaro *et al.* [13], Handa, Sivaswamy, KM Krishna *et al.* [14], propose to utilize the detected difference between person moving and background inflow vectors. W. Tan, H. Liu [15] verifies the changes of objects in the scene by projecting map features into the current frame. Wangsiripitak and Murray [16] also proposes a dynamic object tracking detection scheme.

With the rise of the neural network, the gradual introduction of the SLAM semantic information system not only identifies and classifies, the moving object in a dynamic environment, but make a segmentation and filtration of them. It can be learned from human's common sense and experience that the dynamic objects are usually people, cars, etc., which can move itself. In recent years, the development of deep learning shows that computer tasks such as object detection and semantic segmentation can be solved excellently and its accuracy can even outperform human being. Up to now, there are many excellent neural networks used in SLAM systems. McCormac, Handa *et al.* [17] combine the CNNs with a dense SLAM method to lead maps with semantics that establishes more accurate tracking and map. Kaveti *et al.* [18] proposed a refocusing method based on EM optimization, which uses semantic segmentation to detect dynamic objects in a single time step to initialize the background. Jiyu Cheng *et al.* [19] proposed to use CRF-RNN network to detect dynamic objects in the environment. Lingni and Stuechler *et al.* [20] presents a novel deep neural network to predict semantic segmentation in a self-supervised way, which can enforce multi-view consistency during the training. In [21] proposes an RTFNet architecture in a dark environment at night, uses ResNet to extract features, and combines the encoder to restore the pixel resolution to improve the robustness of the system.

In Detect-SLAM [22], they are introduced SSD [23] network to detect people, animals, cars and other dynamic objects in the environment. During the time, as long as SSDs have identified people, animals, and cars as dynamic objects and regarded them as potential moving ones, it will delete all the features of the object area. However, in the current research, it is found that deep learning is only not good at dealing with the mathematical problems in SLAM, but also has the problem of insufficient calibration data sets, resulting in the inability of the detection accuracy of moving objects for excellent performance. Therefore, it is impossible to completely replace the traditional SLAM target detection module with deep learning and neuromorphic vision sensor [38] at this stage.

The combination of geometric methods and deep learning has become the research direction of the SLAM system to deal with dynamic environmental problems in the next stage. There are, many state-art-of SALM systems [24], [25] contributed. S-SLAM [26] proposed a combination of the SLAM system and SegNet [27] to filter the moving objects through semantic information and motion feature points in dynamic scenes. Liang and Zhang *et al.* [28] proposed another combination of ORB-SLAM2 and YOLO [29], which utilizes the dual-module of moving object detection and moving camera real-time positioning to remove dynamic objects and obtain a semantic map of the scene. In the study of Berta Bescos' DynaSLAM [6], the introduction of Mask-RCNN [30] parallel threads combined with Multi-view Geometry threads uses a fully convolutional neural network to segment objects and transform projections of feature points, jointly removing dynamic objects. The in-painting. Semantic Optical Flow SLAM [31] is proposed based on DynaSLAM. Semantic and geometric threads are tightly coupled to make full use of the elements hidden in semantic and geometric information to eliminate dynamic features. Yuxiang Sun *at al.* [32] proposed a method of using dense optical flow and reprojection error image group pixels to derive the foreground likelihood map to infer moving objects and initialize static scenes. Vincent *et al.* [33] proposed a DoTMask framework that uses a combination of YOLA and EKF tracking modes to segment dynamic objects in the scene. For above schemes, pure geometric or semantic information show better performance for filtering while detecting the direction of moving objects, compared with the traditional ORB-SAM2. However, taking some aspects into consideration, for example, the correlation between objects in semantic information, the accumulation of projection errors between image frames in the geometric thread and the real-time performance of the system, it indicates that there is still room for improvement of the research.

In this article, on the basis of excellent scholars, we propose a SLAM system that combines PSPNet semantic thread and OCMulti-view geometric thread. Under the ORB-SLAM2 system, a PSPNet parallel semantic thread with a pyramid structure is introduced to detect dynamic objects by all related semantic information. e.g. A book originally defined as a static object, which was carried out later, would be detected as a dynamic object during the combination of related information of the entire semantic thread in PSPNet. On the geometric thread, from one point, the optimal error compensation homography matrix is introduced to compensate for the feature point shift phenomenon caused by the projection transformation error of the feature points of the previous frame, and effectively remove the dynamic feature points at the edge of the object. From the other point, we propose a reverse ant colony search strategy in which the characteristics of dynamic point community distribution was used to selectively detect feature points, thereby improving the time-consuming geometric threads and the real-time performance of the system. In total, our system combines PSPNet and OCMulti-View Geometry into ORB-SLAM2, showing its excellent robustness and real-time performance.

III. SYSTEM DESCRIPTION

Figure 1 gives an overview of our system. First of all, the RGB channels passed through a PSPNet that segmented out pixel-wise all the a priori dynamic content., such as people, vehicles or animals. In the ORB-SLAM2 framework, we proposed two parallel threads, which were PSPNet and OCMulti-View Geometry to increase the accuracy and robustness in dynamic environment. First, we refined the segmentation of the dynamic objects previously trained by the PSPNet. Second, we used the Hybrid module to judge whether candidate features were static extraction points.

For that purpose, it is necessary to know the camera pose, for which OCMulti-View Geometry thread has been implemented to localize the camera within the already created scene map. These segmented frames were the ones to obtain the camera trajectory and the map of the scene. It is noticed that Notice that if the moving objects in the scene do use the PSPNet classification, and fusion the OCMulti-View Geometry model stage would promote the accuracy of detecting the dynamic content and self-location accuracy.

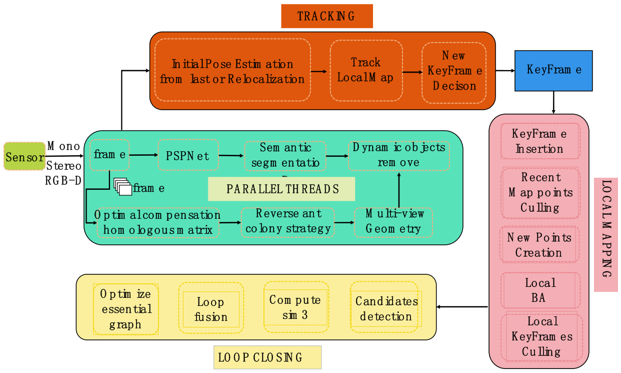


FIGURE 1. The PSPNet-SLAM system is built on the ORB-SLAM2 framework, and we proposed parallel thread before the tracking module. we use the semantic thread introduced in PSPNet and OCMulti-View Geometry thread to cooperate to remove dynamic objects in the scene.

In the monocular and stereo cases, the images were segmented by the PSPNet so that key points in a priori dynamic object are neither tracked nor mapped. All the different stages are described in-depth in the next subsections.

A. SEGMENTATION DYNAMIC CONTENT WITH PSPNet

1) PSPNet FRAMEWORK

The basis of our semantic labeling stream is Compress Pyramid Scene Parsing Network (PSPNet) which is compressed by PSPNet [34]. In traditional semantic SLAM Fully Convolution Network was used to detect the dynamic objects. There were several problems with this approach, such as lack of ability to infer from the context, failure of the association of labels through the relationship between categories; The model might ignore the small things, while the large things might exceed the FCN acceptance range, leading to discontinuous predictions. In summary, FCN does not handle the relationships between scenarios and global information well.

PSPNet is composed of four modules as illustrated in figure.2. Given an input in the figure.2(a), it used a pre-trained ResNet [35] model with the dilated network strategy [36], [37] to extract the feature map. The size of final feature map is 1/8 of the input image, as shown in the figure. 2(b). Next is the most important pyramid pooling module for PSPNet shown in 2(c). A 4-level pyramid was used to collect context information with the pooling kernels covering the whole, half of, and small portions of the image, which were fused as the global prior. Then concatenated the prior with the original feature map in the final part of 2(c). It was followed by a convolution layer to generate the final prediction map in 2(d).

It can be seen in Figure 2 that the feature map with a pyramid structure is transmitted by ResNet in 2(a). In traditional networks, as the much increase of network, it might introduce difficulty of additional optimization for image classification. ResNet solved this problem with skipped connection in each block. The latter layers of deep ResNet mainly learned residues based on previous ones.

In ResNet, a single image X_0 was passed through a convolutional network. In the L-layer network, each layer implemented a nonlinear transformation $H_l(\cdot)$, where l represented the number of layers. $H_l(\cdot)$ a compound function, shows Batch Normalization, ReLU, Conv, or Pooling. The output of the Lth layer was defined as X_l. The simple structure was shown in figure 3:

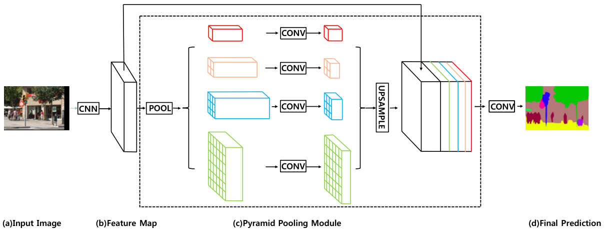


FIGURE 2. The PSPNet framework [34] consists of four parts. In the semantic thread, PSPNet performs semantic segmentation on objects such as pedestrians, books, tables and chairs in key frames. If the semantic information of the object moves relative to the position in consecutive frames, the object is determined to be a dynamic object.

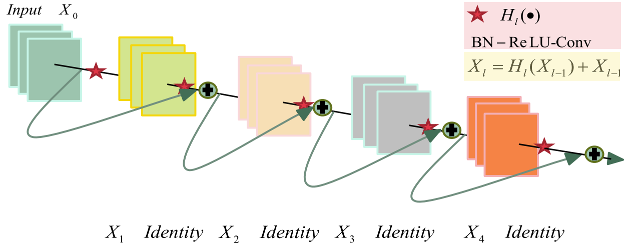


FIGURE 3. ResNet uses the residual function to solve the problem of training accuracy degradation caused by regularization initialization in the process of network layer deepening, and improves the accuracy of network prediction.

The traditional feedforward convolutional network directly used the output of the l layer as the input of the $l+1$ layer to obtain this transferring function: $X_l = H_l(X_{l-1})$. While ResNet added skip-connection when performing nonlinear conversion, the following conversion equation was obtained: $X_l = H_l(X_{l-1}) + X_{l-1}$. The structure of ResNet solved the degradation problem of CNN, is easier to learn than the original features. For the reason that, when the residual was 0, the accumulation layer only performed identity mapping, and the network performance would not decrease. However, in the actual process, the residual error will not be 0, which will also make the stacking layer learn new features based on the input features, thereby leading better performance with excellent object recognition, Classification, cutting performance in the semantic thread in our SLAM system.

2) IMPLEMENT DETAIL

The data sets used in PSPNet's official manual are ADE20K, Cityscapes, PASCAL VOC2012 and PASCAL VOC2012 enhanced data sets, and ours ultimately implements PSPNet-SLAM in TUM data. Therefore, we need to retrain our weight file. In the reference DynaSLAM [6], MASK-RCNN is used as the semantic thread, and the weight file mask_rcnn.h5 of the coco model is used. So, we use pspnet_resnet50.h5 based on the VOC enhanced data set as the pre-training model in our system, and then train our own weight files in the TUM data set. During the training parameter setting process, because the laboratory only has two GPUs, the batch size is set to 8, which cannot reach the original 16. Input size is set to 640*480. The initial learning rate is set to 0.0001, which is multiplied by 0.1 every 30 cycles and decreased by 0.1 times. The weight attenuation coefficient is 0.005.

B. OPTIMAL ERROR COMPENSATION HOMOLOGOUS MATRIX

For tracking mode, we also considered the real-time performance of system, and used DynaSLAM as lightweight low-cost pose estimation, to continue to extract dynamic point in the use of multiple view geometry method. We featured points from the previous frame projection transformation to the current frame. Considering the real-time system and the process of simplicity, we should adopt single matrix methods described before and after the two feature points in the mapping relationship. Meanwhile, With the influence such as noise as the reason of camera movement, we should put forward the optimal error compensation of single matrix method to optimize the projection point position, so as to improve robustness of the system.

1) HOMOGRAPHY MATRIX

In three-dimensional space, there existed any point A, which formed β plane together with the optical centers O_1 and O_2. The plane intersected the lines α_1 and α_2 with l_1 and l_2, as shown in figure 4.

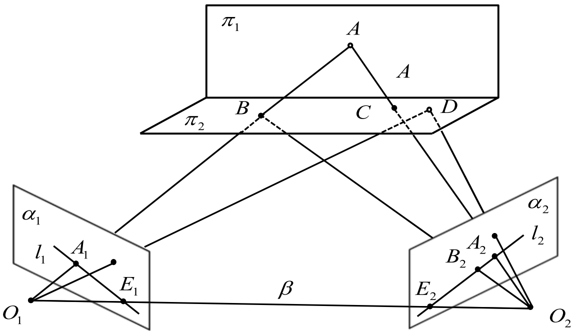


FIGURE 4. **The homography transformation can be simply understood as used to describe the position mapping relationship between the world coordinate system and the pixel coordinate system. The corresponding transformation matrix is called the homography matrix *H*, which can be derived using Projection Plane.**

If B is an point in the π_2 plane, the image of point B on the plane α_1 and α_2 is A_1 and B_2, respectively. B_2 must be located on the intersection line l_2 between plane β and α_2, and the cross product of two points on l_2 can be obtained as follows:

$$X_{l_2} = X_{E_2} \times X_{B_2} = \left[X_{E_2}\right] \cdot X_{B_2} \tag{1}$$

where, $[X]_X$ represents the anti-symmetric matrix of vector X, and the cross product of two vectors can be converted to the anti-symmetric matrix. If $X = (x, y, t)^T$, the anti-symmetric matrix is constructed in the form of (2):

$$[X]_X = \begin{bmatrix} 0 & -t & y \\ t & 0 & -x \\ -y & x & 0 \end{bmatrix} \tag{2}$$

However, the relationship between A_1 and B_2 can be expressed by plane π_2, and the projection formula between any point is as follows:

$$\lambda \cdot \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K \cdot \left(R \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} + T \right) \tag{3}$$

Assumption the three-dimensional coordinate of point B is C, then there is (4)

$$\lambda_{A_1} \cdot X_{A_1} = K \cdot (R \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} + T) \tag{4}$$

The world coordinates are established on plane π_2, and the components on the z-axis of the point on the plane are all 0, then formula (4) is modified to (5).

$$\lambda_{A_1} \cdot X_{A_1} = K \cdot (R \cdot \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} + T) \tag{5}$$

At the same time, the column vector of R is extracted, and the expansion is (6):

$$\lambda_{A_1} \cdot X_{A_1} = K \cdot (R \cdot \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} + T) = K \cdot \begin{bmatrix} r_1 & r_2 & T \end{bmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{6}$$

Let $H_1 = K\,[r_1, r_2, T]$, then (6) can be simplified as:

$$\lambda_{A_1} \cdot X_{A_2} = H_1 \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{7}$$

Similarly, point B_2 on plane α_2 also satisfies the following formula:

$$\lambda_{B_2} \cdot X_{B_2} = H_2 \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{8}$$

where $H_2 = K\left[r_1', r_2', T'\right]$, can be obtained (9) from (7) and (8):

$$\begin{pmatrix} \lambda_{B_2} \\ \lambda_{A_1} \end{pmatrix} \cdot X_{B_2} = H_2 \cdot H_1^{-1} \cdot X_{A_1} = H \cdot X_{A_1} \tag{9}$$

2) OPTIMAL COMPENSATION HOMOGRAPHY MATRIX

When both the object and the camera are moving, the virtual image in the same coordination with the current frame is obtained by using the rotation transformation matrix H. The corresponding relationship between the two frames of images is as follows: $I_i = HI_{t-1}$ in (10), (x_{t-1}, y_{t-1}) and $(x_{t'}, y_{t'})$, the

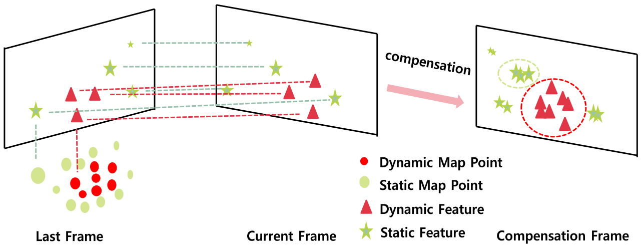


FIGURE 5. In the projection transformation, when the triangular and circular feature points in the previous frame are projected into the current frame through the homography matrix, the projection positions of the feature points in the current frame may be offset due to factors such as system errors and noise. Therefore, optimal compensation homography matrix is proposed to compensate and obtain the compensation frame to improve the performance of dynamic object removal.

image coordinates of the object in the previous frame and the rotated image of the previous frame.

$$\begin{bmatrix} x_{t'} \\ x_{t'} \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ 1 \end{bmatrix} \tag{10}$$

When the homography matrix H is obtained, only 4 pairs of matching point pairs are needed theoretically to obtain the homography. However, in practice, in order to obtain more accurate results, the information of the above four pairs of matching points is often used for results by combining the method of reprojection error optimization and random and sample consensus (RANSAC). In (11), ε is the heavy projection error of the corresponding pixel points between two images. When solving the H matrix, Levenberg-Marquardt method is used to continuously optimize the. When ε is minimum, the updated H matrix is the optimal transformation matrix, and then the optimal transformation matrix can be used to compensate for camera motion. The optimal transformation matrix R is used to convert the coordinates of all pixel points in the image frame at the previous time into a new image with the same resolution as the original image, which is used as the camera motion compensation frame of the image at the current time.

$$\begin{aligned} \varepsilon = \sum_i \Bigg(& \left(x_{t'} - \frac{H_{11}x_{t-1} + H_{12}y_{t-1} + H_{13}}{H_{31}x_{t-1} + H_{32}y_{t-1} + H_{33}} \right) \\ & + \left(y_{t'} - \frac{H_{21}x_{t-1} + H_{22}y_{t-1} + H_{23}}{H_{31}x_{t-1} + H_{32}y_{t-1} + H_{33}} \right) \Bigg) \end{aligned} \tag{11}$$

Through the homography matrix optimized by minimum error ε, the feature points in the previous frame can be projected into the current frame in the form of error compensation, and the original projection points with errors can be modified into projection points more in line with the actual environment, as shown in figure 5.

C. FAST DYNAMIC POINT DETERMINATION UNDER REVERSE ANT COLONY STRATEGY

A large number of x_i' obtained after the projection transformation of the previous frame will be received in the current frame image after the projection transformation of the optimal error homography matrix, and each projection point will be traversed to determine whether the point is a static feature point or a dynamic point. In the feature extraction process, the number of feature points varies from hundreds to hundreds of thousands. If each projection point is judged as a static point, the real-time performance of the SLAM system will be affected to some extent. Considering that the static and dynamic points in the image are distributed in a swarm rather than scattered in a single image, we reversely introduced the theory of ant colony theory. By finding the optimal path of the dynamic point group, the number of feature points can be traversed as little as possible, so as to improve the real-time performance of the SLAM system.

1) ANT COLONY PRINCIPLE

Ant Colony Principle [39] (ACP) algorithm is an artificial intelligence optimization algorithm used to simulate the behavior of natural ant colonies in searching for food. The ant colony optimization algorithm shows that ant can choose the route according to the pheromone secreted by them in the past, and the probability of the route to the food source is proportional to the intensity of pheromone secreted on the route. Therefore, an information feedback phenomenon will be formed in the path of the ants. That is, the more ants choose a certain path, the more pheromones will be left on the path, and the more likely the ants behind will choose this path, so as to find the shortest path.

Suppose if there are m ants, and ants all start from the specified starting point, assuming that they reach the way of food distribution of n nodes, $\tau_{ij}(t)$ said pheromone concentration on the path between nodes i and j at time t, $\eta_{ij}(t)$ is the path $i \rightarrow j$ corresponding heuristic information function. For a certain ant k, the probability of crawling from node i to the next node j is:

$$P_{\mathrm{ij}}^{k} = \begin{cases} \dfrac{(\tau_{ij}(t))^{\alpha}(\eta_{ij}(t))^{\beta}}{\sum\limits_{s \in allowed_k} (\tau_{is}(t))^{\alpha}(\eta_{is}(t))^{\beta}}, & j \in allowed_k \\ 0, & otherwise \end{cases} \tag{12}$$

where, $P_{ij}^{k}(t)$ represents the state transition probability of ant k from node i to j at time t, α is the ant pheromone heuristic factor, β is the expected heuristic factor, and $allowed_k$ represents the node-set that ant k has not yet visited. The greater the β is, the greater the influence of the path distance information on the decision-making of the ants, and the greedier the ants are for the current effect. τ_{ij} is the pheromone concentration of path (i, j), η_{ij} is the heuristic function, and d_{ij} is the Euclidean distance between the current node i and the node j to be selected. The smaller d_{ij} is, the larger η_{ij} is, and the larger P_{ij}^{k} is.

$$n_{ij} = \frac{1}{d_{ij}} \tag{13}$$

Put the node that ant k has passed into the $tabu_k$ table. According to equation (13), ants prefer to choose the node with a short distance from the current node and a high concentration of pheromones. Each ant will update the pheromone in the path immediately after passing a certain path. The update formula is as follows:

$$\tau_{\mathrm{ij}}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{14}$$

$$\tau_{\mathrm{ij}}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) \tag{15}$$

where ρ represents the pheromone volatilization factor, $\rho \in [0, 1)$; m is the number of ants; $\Delta\tau_{ij}(t)$ is the pheromone increment on the path $i \rightarrow j$ at time t; $\Delta\tau_{ij}(t)$ is the pheromone increment released by k on path $i \rightarrow j$ at time t.

2) REVERSE ANT COLONY SEARCH STRATEGY

Due to the error compensation of the projection with feature points of the previous frame in the current frame, the number of feature points will increase, and these feature points are irregularly distributed. Under normal circumstances, we directly use the multi-view geometry method in DynaSLAM [6] to determine whether all feature points are dynamic feature points one by one. However, considering the increase in the number of feature points, the real-time performance of the system is affected. Therefore, we propose a reverse ant colony strategy to reduce the time consumption of the SLAM system in multi-view geometry threads by selectively judging feature points.

In the ant colony strategy, it starts from the starting point and avoids obstacles on the way to the end point, so as to find the optimal way. And our dynamic distribution is in groups, as the distribution of the fixed points, so we put forward a reverse ant colony strategy. Fix an optimal search path from the beginning to the end, and in turn search. In the entire search process, when a dynamic point is found, the search path will shift to the dynamic point group area, until the dynamic point in the area is detected, then return to the offset point and continue to search for the next dynamic point community, specific search scheme in figure 6.

After the feature points of the current frame are projected into the current frame through error compensation, we do not need to spend a lot of time to perform dynamic point judgement on all points because the distribution of dynamic feature points is distributed in the state of the community, According to the density of feature points, a path L is designed to meander through the image to point B starting from A. The ant colony moves continuously from point $K_i = 0$ to the next $K_i (i = 0, 1, 2, \cdots, n)$ point until it moves to the end point B. During the movement of the colony, every point K will use this point as the origin, and R is the dynamic point in the radius of the search area. When the dynamic point is found in the circle, the geometry of the discrete points is calculated by the convex hull, and the search bandwidth of ΔZ is extended outward with the geometric edge as the boundary. Whenever a new one is found in the ΔZ bandwidth after the dynamic point, it continues to expand ΔZ outward until no new dynamic point is found in the extended area, returns to the L path and moves to the next K point to search for dynamic points in the new area.

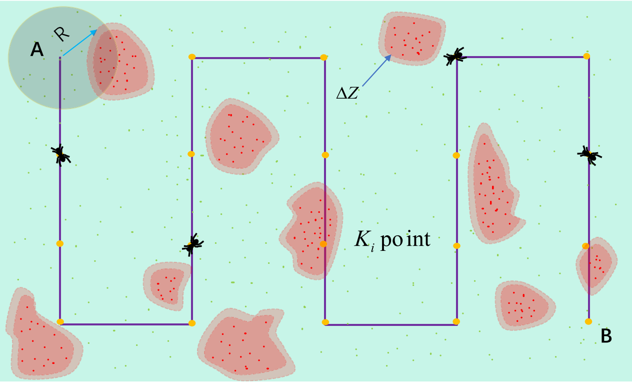


FIGURE 6. In the reverse ant colony strategy, the black ants gradually search from point A to point B with a circle of radius R. When a dynamic point is found at node K_i, it is offset to the center of its dynamic point community and the distance is ΔZ Expand outwards until there is no dynamic feature point in the ΔZ range of the current dynamic point cluster, return to the AB line and continue to search for the next node K_{i+1}.

When we use Ant colony optimization strategy to determine whether the feature points are dynamic one it is no longer necessary to detect all the existing points in the image, so as to improve the real-time performance of the system to a certain extent.

IV. EXPERIMENT

In this section, we have carried out an experiment of our PSPNet-SLAM in the TUM RGB-D dataset to evaluate its performance in a dynamic environment. First, we will use PSPNet-SLAM and pure ORB-SLAM2 system, DynaSLAM system which uses the Mask R-CNN as the semantic thread to verify the improvement of our system. The Semantics of pyramid structure SLAM system is based on ORB-SLAM2 as the basic framework. PSPNet semantic thread, and multi-view geometric analysis thread with direction ant colony algorithm are combined to dynamically move. The detection performance of objects has been significantly improved. Besides, we run both our system and other excellent SLAM systems in a dynamic environment to analyze their accuracy and time-consuming mapping in the dynamic environment. From the comparison, it is demonstrated that the performance of our system in a laboratory environment is better than other existing ones.

A. EVALUATION ON TUM RGB-D DATASET

The TUM data set is an excellent data set for evaluating the positioning accuracy of the camera, because it provides accurate ground realism for the sequence. It contains 7 sequences recorded by RGB-D cameras at 30fps with a resolution of 640 x 480. At the same time, the TUM RGB-D video data set is composed of 39 sequences recorded by Microsoft Kinect sensors in different indoor environments. According to the purpose of our experiment, we choose the data sequence containing dynamic factors for the experiment,

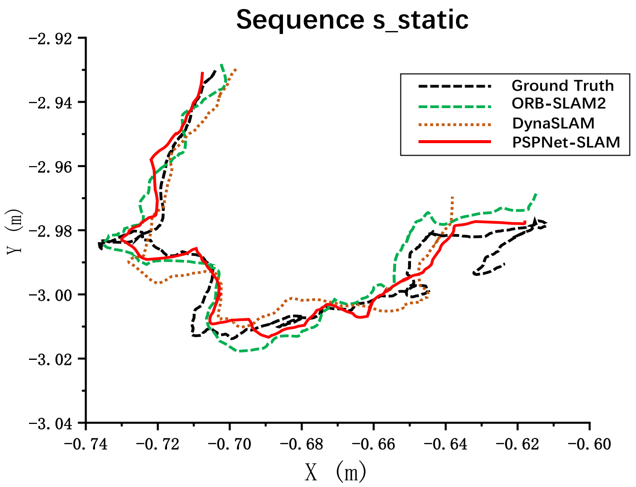
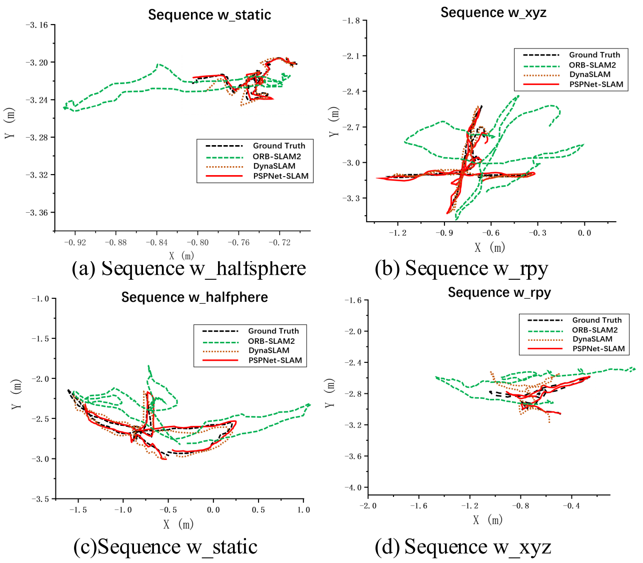


FIGURE 7. ORB-SALM2, DynaSLAM, PSPNet-SLAM trajectory comparison in low-dynamic sequence.



(a) Sequence w_halfsphere (b) Sequence w_rpy

(c)Sequence w_static (d) Sequence w_xyz

FIGURE 8. Trajectory comparison in high-dynamic sequence.

including s_static, w_halfphere, w_rpy, w_static, w_xyz, where s_static is a static sequence, and the rest are dynamic sequences. In the data set we use, pedestrians are the main elemental objects, for they show both static and dynamic behaviors. The word before the underline of the sequence name denotes the state of people in the scene with the initial letter "s" for "setting" and "w" for walking. Meanwhile, the word after the underscore of the sequence name indicates the movement state of the camera at this time.

We run ORB-SLAM2, DynaSLAM, and our own PSPNet-SLAM in the same TUM data environment. It was found that the camera trajectories estimated by these three systems are plotted together with ground truth in one figure. Meanwhile, we converted the three-dimensional space tracking trajectory into a 2D plane trajectory, displayed the three SLAM systems in the same plane to show their performance, and conducted an intuitive comparative analysis. We took the fit between the estimated 2D trajectory and the real trajectory as the basic standard of the evaluation system and analyzed the operation results of the system. In Figure 7 and 8, we can see that in a static environment, sequence s_static, the trajectories of three systems are all very close to the ground truth.

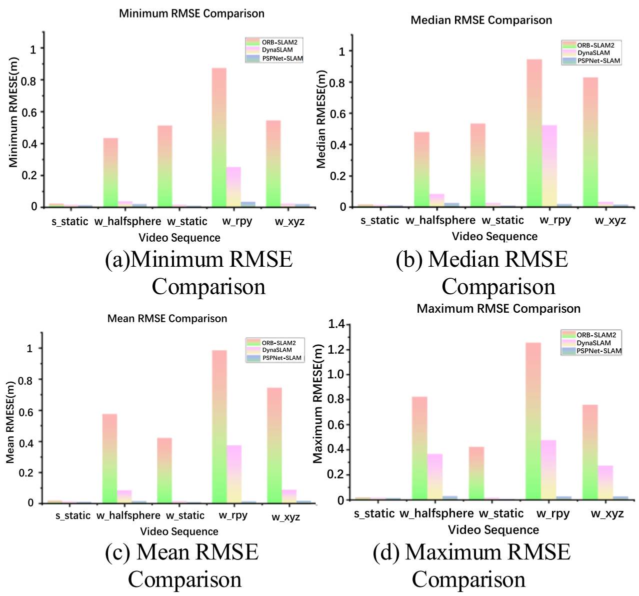
In highly dynamic environments, pure ORB-SLAM2 is affected by dynamic objects in the video sequence. The estimated running trajectory has a large error with the ground truth, and even generates erroneous trajectories in some areas, without a degree of fit with the ground truth. However, when two SLAM systems with semantic parallel threads are introduced, faced with dynamic objects, the PSPNet-SLAM and DynaSLAM systems have excellent performance on ground truth trajectory estimation, for they can accurately estimate the true trajectory. It is indeed that in the low-dynamic environment pure ORB-SLAM2 can filter and classify the objects in the scene through the RANSAC algorithm, and optimize the correction trajectory through the back end. However, the filter of pure ORB-SLAM2 in the environment cannot be applied, and it cannot effectively distinguish whether the object in the scene is a static object or a dynamic one. In contrast, the SLAM system that combines the parallel line semantic process of the pyramid structure and the multi-view geometry of reverse ant colony search can efficiently and quickly remove dynamic object points.

Further qualitative comparison of these three systems was carried out to verify the effectiveness of the PSPNet-SLAM. Each sequence is processed 5 times, and we get median, mean, minimum and maximum RMSE (Root Mean Square Error) results of ATE (Absolute Trajectory Error) to judge its localization Accuracy, while RMSE is computed by:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (X_{obs,i} - X_{\mathrm{model},i})^2}{n}} \tag{16}$$

where n means the number of observations, i denotes the i^{th} observation. $X_{obs,i}$ is the ground truth of the i^{th} observation, while $X_{\mathrm{mod}el,i}$ is the computation result of the i^{th} observation.

According to the results shown in Figure 9, we can see that in low-dynamic sequence s_static, the results of the three approaches are actually very close. Our method and DynaSLAM outperforms ORB-SLAM2 in highly dynamic scenarios, reaching an error similar to that of the original pure ORB-SLAM2 system in static scenarios. The SLAM system introduces semantic threads, whether it is median, mean, minimum, and maximum while the RMSE value is decreasing rapidly. At the same time, our system is superior to DynaSLAM in the decline of the above four values for our system uses a pyramid-structured convolutional network that is more capable of linking context information rather than a fully convolutional neural network, which can combine the contextual semantic information to segment some dynamic objects that DynaSLAM cannot do. Another reason is that the optimal compensation homography matrix is used in the geometric pose estimation, so as to improve the accuracy of



(a)Minimum RMSE Comparison

(b) Median RMSE Comparison

(c) Mean RMSE Comparison

(d) Maximum RMSE Comparison

FIGURE 9. Comparison of median, mean, minimum and maximum RMSE in the intuitive form of bar chart for ORB-SLAM2, DynaSLAM and PSPNet-SLAM.

the projection of the dynamic point of the previous frame to the current frame and omit the probability of filtering of dynamic object points.

We not only take our SLAM system to make a contrastive analysis with original ORB-SLAM2 and DynaSLAM, but also compare with other state-of-the-art SLAM in the dynamic scene, such as DS-SLAM, Detect-SLAM, SOF-SLAM, to analyze the same comparison parameters as in Table 1. The four SLAM systems are all based on ORB-SLAM2, and introduce semantic parallel threads to semantically segment objects in the scene. Among them, Detect-SLAM, SOF-SLAM and PSPNet-SLAM have proposed their own solutions for dynamic objects in the scene, eliminating the impact of dynamic objects on material estimation and mapping. From the experimental test data, we can find that although these four systems are based on the ORB-SLAM2 framework-derived system, considering that each of them will make corresponding modifications when designing the system, it results in a criterion of during cross-evaluation. Since the difference of evaluation details of RMSE or some other difference of experiment condition may exist during the experiment, therefore, in order to verify the effectiveness of our system objectively, we refer to the literature [31], using relative RMSE reduction (i.e. relative accuracy improvement) of each system with respect to the original ORBSLAM2 as the evaluation metric. The relative metric is more reasonable as it can eliminate the accuracy difference caused by other factors which are not related to the dynamic features processing algorithm. The comparative analysis results are shown in Table 1. From the table, we can find that in the low dynamic sequence, several systems have roughly been in the same degree of improvement in accuracy, and our system is in the forefront. In high-dynamic sequence, our accuracy improvement effect is higher than that of several other systems, because in a high-dynamic environment, there are a large number of dynamically moving objects, and some objects do not have moving attributes themselves, but are moved by the movement of other objects. (For example, books, tables, and chairs without movement attributes but driven by people). Although DynaSLAM and SOF-SLAM also remove dynamic objects in the scene, they do not consider the context information in the semantic thread and the geometry thread. The projection error of the feature point from the previous frame to the current frame, results in a system with slightly lower accuracy than our system.

TABLE 1. Comparisons of RMSE [m] for our system against the state-of-the-art in dynamic sequences of TUM RGB-D dataset.

Sequence	DS-SLAM	Detect-SLAM	L.Zhang et al	SOF-SLAM	PSPNet-SLAM
s_static	0.0065	0.0052	-	0.010	0.012
w_halfphere	0.0303	0.0473	0.0636	0.029	0.022
w_rpy	0.0442	0.2959	-	0.027	0.025
w_static	0.0081	0.0075	-	0.007	0.008
w_xyz	0.0247	0.0421	0.0336	0.018	0.016

TABLE 2. Comparison of the time consumption of the two SLAM systems in Geometry thread of the video sequence.

Sequence	DynaSLAM[ms]	PSPNet-SLAM [ms]
s_static	175.42	143.11
w_halfphere	342.47	247.52
w_rpy	229.81	198.69
w_static	237.24	158.52
w_xyz	353.95	263.78

B. SYSTEMS TIME CONSUMPTION

The time consumption of dynamic feature point removal module an element that the On-line SLAM system needs to consider. In Table 2, our system and DynaSLAM system run the five video sequences in the same hardware environment, and calculate the running time of geometric threads in real time. It is noted that the dynamic environment is not optimized for real-time operation. However, its ability to create a lifetime mapping of static scene content is also related to running in offline mode. From the running time-consuming results, we can find that whether our system is in a static environment or a dynamic environment, the geometric thread time-consuming is less than DynaSLAM. It shows that PSPNet-SLAM has made some progress in real-time performance.

V. CONCLUSION

In this work, we have presented a PSPNet-SLAM system that introduces a PSPNet as parallel semantic thread, and builds based on ORB-SLAM2. In semantic thread, we use PSPNet to get pixel-wise semantic segmentation. Due to its pyramid-shaped network structure, more contextual information can be obtained, and the interrelationship between objects in pixels can be found, which is more effective in the detection and removal of dynamic feature points than the full convolutional neural network of other structures. In the geometry thread, we proposed optimal error compensation homography matrix first to compensate for the feature point shift phenomenon caused by the projection transformation error of the feature points of the previous frame. Second, we proposed a reverse ant colony search strategy, which used the characteristics of dynamic point community distribution to selectively detect feature points, thereby improving the robust and real-time performance of geometric thread dynamic feature point detection. In order to verify the performance of our system, we conducted comparative experiments on the TUM dataset with other excellent SLAM systems. The final experimental results also show that our system has improved localization accuracy and real-time performance compared with other slam frameworks.

Although some progress has been made in robustness and real-time performance, there are still many tasks we need to do. On the one hand, the real-time performance of the system is still a problem we will face. In the next work, we will research the real-time processing of image frames in the SLAM system. On the other hand, we need to improve the applicability of the system in different scenarios. In future work, we need to put PSPNet-SLAM in different data sets for experiments, and continue to tune them to improve the system's ability to remove dynamic objects.

REFERENCES

[1] A. J. Davison, "MonoSLAM: Real-time single camera SLAM," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[2] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[3] R. Mur-Artal and J. D. Tardos, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[4] J. Engel, T. Schöps, and D. Cremers, *LSD-SLAM: Large-Scale Direct Monocular SLAM*. vol. 8690. 2014, pp. 834–849.

[5] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 15–22.

[6] B. Bescos, J. M. Facil, J. Civera, and J. Neira, "DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 4076–4083, Oct. 2018.

[7] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 34–46, Feb. 2007, doi: 10.1109/TRO.2006.889486.

[8] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 1271–1278, doi: 10.1109/ICRA.2016.7487258.

[9] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proc. 6th IEEE ACM Int. Symp. Mixed Augmented Reality*, Nov. 2007, pp. 225–234.

[10] A. Kundu, K. M. Krishna, and J. Sivaswamy, "Moving object detection by multi-view geometric techniques from a single camera mounted robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2009, pp. 4306–4312.

[11] D. Zou and P. Tan, "CoSLAM: Collaborative visual SLAM in dynamic environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 2, pp. 354–366, Feb. 2013.

[12] M. Piaggio, R. Fornaro, A. Piombo, L. Sanna, and R. Zaccaria, "An optical-flow person following behaviour," in *Proc. IEEE Int. Symp. Intell. Control (ISIC) Held Jointly IEEE Int. Symp. Comput. Intell. Robot. Autom. (CIRA) Intell. Syst. Semiotics (ISAS)*, Sep. 1998, pp. 4078–4083.

[13] G. Chivilo, F. Mezzaro, A. Sgorbissa, and R. Zaccaria, "Follow-the-leader behaviour through optical flow minimization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep./Oct. 2004, pp. 3182–3187.

[14] A. Handa, J. Sivaswamy, K. M. Krishna, and S. Singh, "Person following with a mobile robot using a modified optical flow," in *Advances in Mobile Robotics*, L. Marques, M. de Almeida, and M. O. Tokhi, Eds. Singapore: World Scientific, 2008.

[15] W. Tan, H. Liu, Z. Dong, G. Zhang, and H. Bao, "Robust monocular SLAM in dynamic environments," in *Proc. IEEE Int. Symp. Mixed Augmented Reality (ISMAR)*, Oct. 2013, pp. 209–218.

[16] S. Wangsiripitak and D. W. Murray, "Avoiding moving outliers in visual SLAM by tracking moving objects," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 375–380.

[17] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "SemanticFusion: Dense 3D semantic mapping with convolutional neural networks," 2016, *arXiv:1609.05130*. [Online]. Available: http://arxiv.org/abs/1609.05130

[18] P. Kaveti, S. Katt, and H. Singh, "Removing dynamic objects for static scene reconstruction using light fields," 2020, *arXiv:2003.11076*. [Online]. Available: https://arxiv.org/abs/2003.11076

[19] J. Cheng, Y. Sun, and M. Q.-H. Meng, "A dense semantic mapping system based on CRF-RNN network," in *Proc. 18th Int. Conf. Adv. Robot. (ICAR)*, Jul. 2017, pp. 589–594.

[20] L. Ma, J. Stuckler, C. Kerl, and D. Cremers, "Multi-view deep learning for consistent semantic mapping with RGB-D cameras," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Vancouver, BC, Canada, Sep. 2017, pp. 598–605.

[21] Y. Sun, W. Zuo, and M. Liu, "RTFNet: RGB-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2576–2583, Jul. 2019, doi: 10.1109/LRA.2019.2904733.

[22] F. Zhong, S. Wang, Z. Zhang, C. Chen, and Y. Wang, "Detect-slam: Making object detection and slam mutually beneficial," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1001–1010.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," 2015, *arXiv:1512.02325*. [Online]. Available: http://arxiv.org/abs/1512.02325

[24] I. A. Barsan, P. Liu, M. Pollefeys, and A. Geiger, "Robust dense mapping for large-scale dynamic environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 7510–7517.

[25] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss, "ReFusion: 3D reconstruction in dynamic environments for RGB-D cameras exploiting residuals," 2019, *arXiv:1905.02082*. [Online]. Available: http://arxiv.org/abs/1905.02082

[26] C. Yu, Z. Liu, X.-J. Liu, F. Xie, Y. Yang, Q. Wei, and Q. Fei, "DS-SLAM: A semantic visual SLAM towards dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1168–1174.

[27] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[28] L. Zhang, L. Wei, P. Shen, W. Wei, G. Zhu, and J. Song, "Semantic SLAM based on object detection and improved octomap," *IEEE Access*, vol. 6, pp. 75545–75559, 2018, doi: 10.1109/ACCESS.2018.2873617.

[29] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[30] K. He, G. Gkioxari, P. Dollar, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2961–2969.

[31] L. Cui and C. Ma, "SOF-SLAM: A semantic visual SLAM for dynamic environments," *IEEE Access*, vol. 7, pp. 166528–166539, 2019, doi: 10.1109/ACCESS.2019.2952161.

[32] Y. Sun, M. Liu, and M. Q.-H. Meng, "Motion removal for reliable RGB-D SLAM in dynamic environments," *Robot. Auto. Syst.*, vol. 108, pp. 115–128, Oct. 2018.

[33] J. Vincent, M. Labbé, and J. S. Lauzon, "Dynamic object tracking and masking for visual SLAM," 2020, *arXiv:2008.00072*. [Online]. Available: https://arxiv.org/abs/2008.00072

[34] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[36] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," 2014, *arXiv:1412.7062*. [Online]. Available: http://arxiv.org/abs/1412.7062

[37] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*. [Online]. Available: http://arxiv.org/abs/1511.07122

[38] G. Chen, H. Cao, J. Conradt, H. Tang, F. Rohrbein, and A. Knoll, "Event-based neuromorphic vision for autonomous driving: A paradigm shift for bio-inspired visual sensing and perception," *IEEE Signal Process. Mag.*, vol. 37, no. 4, pp. 34–49, Jul. 2020, doi: 10.1109/MSP.2020.2985815.

[39] X. Dai, S. Long, Z. Zhang, and D. Gong, "Mobile robot path planning based on ant colony algorithm with A* heuristic method," *Frontiers Neurorobot.*, vol. 13, p. 15, Apr. 2019.

XUDONG LONG received the B.E. degree from Panzhihua University, Panzhihua, China, in 2018. He is currently pursuing the master's degree with the Shanghai University of Engineering Science, Shanghai, China. His research direction focuses on the computer vision. His current research interests include V-SLAM and dynamic object detect.

WEIWEI ZHANG received the Ph.D. degree in mechanical engineering from Hunan University, in 2015. He is currently a Lecturer with the Shanghai University of Engineering Science. His research direction is the technology of intelligent vehicle. His team currently undertakes several major projects from renowned Chinese companies. His current research interests include the technology of image processing, intelligent vehicle, and power train of vehicle.

BO ZHAO received the master's degree from Xidian University, Shaanxi, China, in 1988. She is currently an Associate Professor with the Shanghai University of Engineering Science, Shanghai. She currently focuses on vehicle chassis design optimization.