



Semantic SLAM Based on Joint Constraint in Dynamic Environment

Yuliang Tang^{1,2}, Yingchun Fan¹, Shaofeng Liu¹, Xin Jing², Jintao Yao²,
and Hong Han^{1,2}(✉)

¹ School of Artificial Intelligence, Xidian University, Xi'an 710071, China
hanh@mail.xidian.edu.cn

² Shaanxi Key Laboratory of Integrated and Intelligent Navigation,
Xi'an 710071, China

Abstract. In most existing SLAM (Simultaneous localization and mapping) methods, it is always assumed that the scene is static. Lots of errors would occur when the camera enters a highly dynamic environment. In this paper, we present an efficient and robust visual SLAM system which associates dynamic feature points detection with semantic segmentation. We obtain the stable feature points by the proposed depth constraint. Combined with the semantic information provided by BlitzNet, every image in the sequence is divided into environment region and potential dynamic region. Then, using the fundamental matrix obtained from the environment region to construct epipolar line constraint, dynamic feature points in the potential dynamic region can be identified effectively. We estimate the motion of the camera using the stable static feature points obtained by the joint constraints. In the process of constructing environment map, moving objects are removed while static objects are retained in the map with their semantic information. The proposed system is evaluated both on TUM RGB-D dataset and in real scenes. The results demonstrate that the proposed system can obtain high-accuracy camera moving trajectory in dynamic environment, and eliminate the smear effects in the constructed semantic point cloud map effectively.

Keywords: SLAM · Semantic segmentation · Joint constraint · Dynamic objects

1 Introduction

SLAM plays an important role in the field of robot navigation and unmanned driving. Many excellent achievements have been produced in visual SLAM, which are mainly classified into direct method based on photometric error [1, 2] and indirect method based on salient points matching [3]. The main purpose of both methods is to obtain environmental information through sensors to achieve camera pose estimation and map construction. It is a premise for most of the current

Supported by organization the open fund of Shaanxi Key Laboratory of Integrated and Intelligent Navigation (No. SKLIIN-20180102 and No. SKLIIN-20180107).

visual slam systems that the environment is static, which severely limits the application of visual SLAM due to lots of dynamic objects in the environment.

With the development of deep neural networks, target detection and semantic segmentation algorithms have achieved great progress, and many experts are committed to integrating visual SLAM with deep learning. Some studies specify dynamic targets by directly regarding people, cars or animals as dynamic objects, such as [4, 22]. However, it may cause the loss of useful information in the constructed map.

In this work, a robust SLAM system to deal with dynamic objects on RGB-D data is proposed. The image is divided into environment region and potential dynamic region by the semantic information provided by the improved BlitzNet [17]. In order to eliminate the influence of missing values in the depth image and the sudden changes of the depth value in the edge of objects and environment, we proposed a depth constrain to obtain the stable feature points. And the dynamic feature points can be identified effectively by the epipolar line constraint constructed by the environment region. The static feature points are used to estimate the motion trajectory of the camera, while the dynamic feature points are used to determine the motion state of the potential dynamic objects. Finally, the point cloud map with semantic information is built.

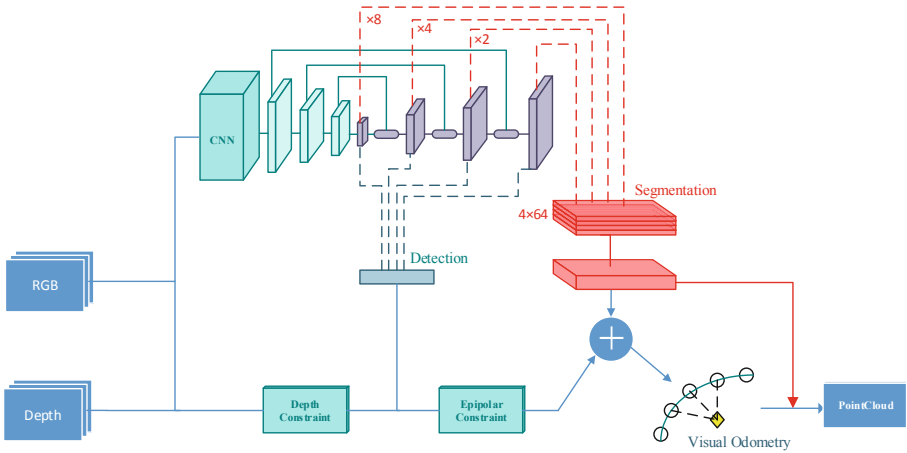


Fig. 1. Overview of the system.

2 Related Work

2.1 Dynamic SLAM

The presence of dynamic objects will seriously affect the mapping results and the estimation of the camera pose. Specified priori dynamic targets are utilized

in [4,5] to handle dynamic environment. Burgard et al. [6] propose a data association technique to incorporate both dynamic and stationary objects directly into camera pose estimation. A different multi-camera combination strategy is introduced to deal with dynamic object effectively in [7]. Whats more, Henri Rebecq et al. [8] propose a method of using a special event camera which can achieve robust performances in highly dynamic environment, however, high cost limits the use of such methods.

2.2 Semantic Segmentation Based on Deep Learning

At present, most of the advanced semantic segmentation techniques based on deep learning are derived from full convolution network (FCN) [9], and different strategies are proposed to improve the segmentation effect. In terms of models, encoder-decoder architecture has been widely used, such as [10,11]. About convolution kernel, authors of [12,13] have done a lot of important work using dilated convolution to enhance receptive field to integrate context information. Starting from multi-scale feature fusion, Zhao et al. [14,15] use spatial pyramid pooling to integrate different scale features to obtain global information. As for instance segmentation, Mask R-CNN can detect objects in an image while simultaneously generating a segmentation mask for each instance, but it lacks real-time performance [16]. In this paper, a real-time semantic segmentation algorithm BlitzNet [17] is used to transform semantic segmentation into instance segmentation.

2.3 Semantic SLAM

Some approaches combine classic SLAM with semantic segmentation to build a more robust semantic map such as [18,19], but both of them do not focus on the localization of camera. Other approaches focus on locating and processing dynamic objects. For instance, Bowman et al. [20] propose probabilistic data association to improve the robustness of localization, and some algorithms [21,22] combine different deep network with moving consistency check to reduce the impact of dynamic objects. However, most of these methods roughly treat certain classes of objects as dynamic objects, even if these objects are static in the images, thus dynamic objects detection is not precise enough.

3 System Description

3.1 Framework of Our System

The overview of our system is presented in Fig. 1. Firstly, the RGB images pass through a CNN (Convolution Neural Network) that performs object detection and pixel-wise segmentation at the same time. The detected information includes some common objects such as people, screens, tables and chairs, etc. As for RGB-D data, we employ depth constraint and epipolar line constraint combined

with object bounding box to determine potential dynamic points. After the instance segmentation result arrives, potential dynamic feature points will be added to the fusion module. Outliers located in the real moving objects can be removed effectively. More accurate camera trajectory can be obtained by the visual odometry. Finally, the constructed point cloud map and semantic information are integrated to obtain a semantic point cloud map.

3.2 Potential Dynamic Point Detection

Dynamic object detection algorithms are generally based on regional features of the image, such as texture, color, grayscale, and so on. In this paper, the potential dynamic points detection is realized by the proposed joint constraints. Finally, the dynamic objects can be detected by fusing the semantic segmentation algorithm.

For two adjacent frames of depth image, there are regions with incomplete depth (the depth value of these regions is 0), and there is a sudden change of depth value at the edge of the object [23]. The most stable feature points are on the surface of certain objects, such as the regions on the desk marked by the red dashed frame as shown in Fig. 2. Using image depth information to obtain stable feature points can effectively reduce the problem of high false alarm rate caused by strong parallax.

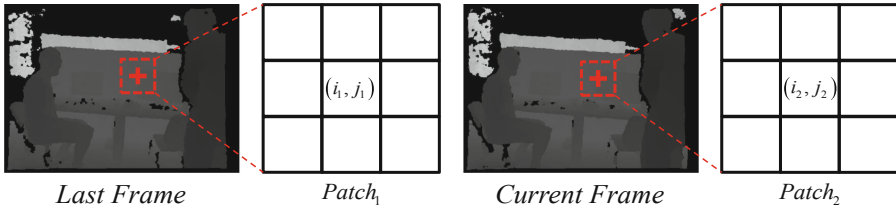


Fig. 2. Patches centered on the integer pixel coordinates of the feature points in two adjacent depth images. (Color figure online)

In order to find the stable feature points on the image, we consider a 3×3 image patch centered on the integer pixel coordinates of the feature points. As shown in Fig. 2, the red crosses represent the locations of the corresponding feature points on the two frames of depth image, where (i_1, j_1) and (i_2, j_2) are the integer pixel coordinates of the feature points on the previous and current frame, respectively. If any depth value on the image patch is 0, the depth value of the feature point is considered missing and the corresponding feature point pair is deleted. The depth value of the feature point is replaced by the average depth of the patch as shown in the following equation:

$$\hat{d} = \frac{1}{9} \sum_{x=1}^3 \sum_{y=1}^3 Patch(x, y) \quad (1)$$

where x, y are the coordinates of pixels in the patch. The Euclidean distance of the average depth of two feature points \hat{d}_1, \hat{d}_2 is used to exclude outliers with greater depth deviation to obtain stable feature points, as shown in Eq. 2.

$$D_d = \sqrt{(\hat{d}_1 - \hat{d}_2)^2} \quad (2)$$

By setting a threshold ξ , we can get the stable matching points P_{s_1}, P_{s_2} as shown in Eq. 3.

$$P_{s_1}, P_{s_2} = \{P_1, P_2 | D_d(\hat{d}_1, \hat{d}_2) < \xi\} \quad (3)$$

Using BlitzNet, the potential moving objects region can be obtained, such as the person region, and other region as the environment region. Therefore, the fundamental matrix F can be calculated by stable matching points in the environment region using RANSAC algorithm. Epipolar geometric describes the constraint relationship between the matching points in different angles of view. P_{m_1}, P_{m_2} denote feature points in the potential moving objects region of the previous frame and current frame, respectively.

$$P_{m_1} = [u_1, v_1, 1]^T, P_{m_2} = [u_2, v_2, 1]^T \quad (4)$$

We can distinguish the dynamic feature points in potential moving region by the epipolar line constraint as follows:

$$D_e = \frac{|P_{m_2}^T F P_{m_1}|}{\sqrt{l_x^2 + l_y^2}} \quad (5)$$

where l_x, l_y represent epipolar lines coordinate. D_e only depends on the epipolar geometry theory and the consistency relationship between the projection of the feature points. The specific algorithm process is described in Algorithm 1.

Algorithm 1. Joint Constraint Detection Algorithm

Input: Previous frame's feature points, P_1 , Previous frame's depth, d_1 ;
Current frame's depth, d_2 ;

Output: The set of potential dynamic points, S ;

1: Calculate current frame's feature points P_2 by optical flow pyramid,

2: Obtain stable matching points P_{s_1}, P_{s_2} by depth constraint

3: Calculate fundamental matrix $F = \text{findFundamentalMat}(P_{s_1}, P_{s_2})$

4: for each matched point pair p_1, p_2 in P_1, P_2 do

5: if p_1 is not in potential moving objects region then

6: continue

7: Calculate epipolar line $L = \text{computeCorrespondEpilines}(p_1, F)$

8: Calculate distance from epipolar line constrain D_e by Eq. 5

9: if $D_e > \xi$ then

10: Append p_2 to S

11: end if

12: end for

3.3 Sematic Segmentation

For scene analysis, BlitzNet, a deep neural network which can complete the object detection and semantic segmentation in one-time forward propagation, is used as the basic network in our experiment, whose backend is changed to meet the requirement of instance level segmentation tasks in our system.

BlitzNet only takes RGB image as input. In this experiment, VOC and COCO datasets are used for joint training and SSD300 is used as the backbone network, moreover, the object detection mAP on the VOC12 verification set can reach up to 83.6, while the semantic division mIOU reaches approximately 75.7. It has a good effect in the general scene, as shown in Fig. 3(a) and (b). The combination of the detection results and the segmentation results obtained by the network can get the desired instance segmentation image as shown in Fig. 3(c).

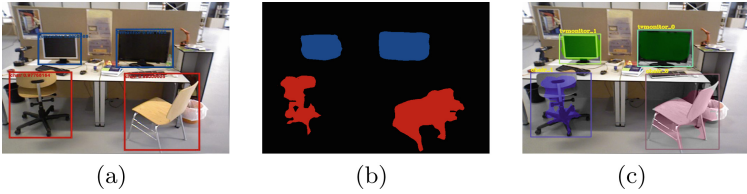


Fig. 3. Results of the improved BlitzNet. (a) Object detection. (b) Semantic segmentation. (c) Instance segmentation.

3.4 Dynamic Object Detection

In Sect. 3.2, the algorithm of detecting potential dynamic feature points is introduced, which can roughly find the dynamic feature points in the image. In this section, we will get more accurate dynamic points to detect dynamic objects in scene. Each segmented target in the image is enclosed by the detection box defined as the influence area. We divide feature points into four sets of points, as shown in Fig. 4: static points in potential moving region $U_s \in \mathbb{R}^{n \times 2}$, potential dynamic points $U_d \in \mathbb{R}^{m \times 2}$, outliers in environment $V_d \in \mathbb{R}^{M \times 2}$, and stable points in environment $V_s \in \mathbb{R}^{N \times 2}$. We propose two proportions, one is region dynamic point ratio τ_d , and the other is region points ratio τ_r , as shown in Eq. 6.

$$\tau_d = \frac{m}{m+n}, \tau_r = \frac{m+n}{m+n+N} \quad (6)$$

The value of threshold τ_d is 0.5 and τ_r is 0.15 in this experiment. Once the results of both equations are greater than their threshold, segmented targets within the detection box will be classified as dynamic targets, like the yellow part in the right figure in Fig. 4. The external parameter matrix to estimate trajectory of the camera can be obtained to estimate trajectory of the camera

by solving the least squares problem shown below:

$$\min_{R,t} \sum_{i=1}^{N+n} \|P_{ai} - (RP_{bi} + t)\|^2 \quad (7)$$

where, $P_b \subseteq U_s \cup V_s$ and P_a is matching points in the previous frame.

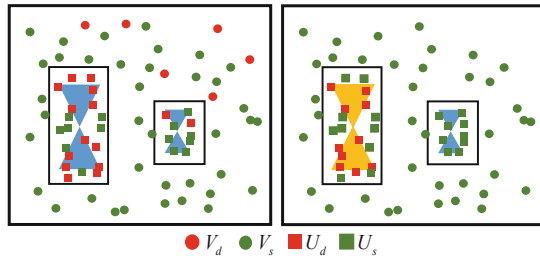


Fig. 4. Four types of points in dynamic objects detection. (Color figure online)

4 Experiments and Results

This section shows the experimental results of the proposed method. We have evaluated our system both on TUM RGB-D dataset [24] and in real-world environment.

4.1 Dynamic Points Detection

TUM datasets provide several image sequences in dynamic environment with accurate ground truth and camera parameters, and it is divided into categories of *walking*, *sitting*, and *desk*. We mainly test the dynamic feature points detection experiment in the *walking* sequence, and the motion amplitude of the dynamic object in this sequence is large.

The process of dynamic points detection and dynamic objects segmentation is shown in Fig. 5. The image can be divided into potential dynamic region and environment region by the semantic information provided by BlitzNet. The approximate distribution of the dynamic points in potential moving region can be obtained by the proposed joint constraint. According to the calculation results of τ_d and τ_r , we can judge that the two people in the bounding boxes are dynamic objects and feature points in the mask of people is regarded as dynamic points. It is obvious that the person is classified as dynamic object in this experiment automatically, and our algorithm retains a lot of static scenarios and removes dynamic part as much as possible.

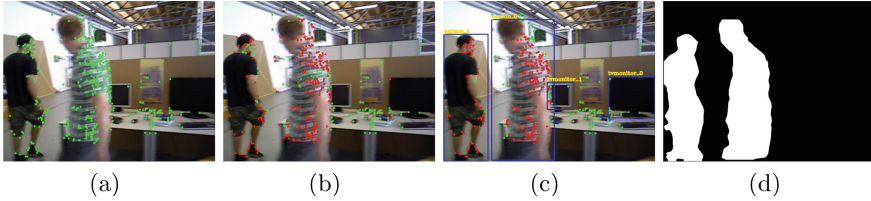


Fig. 5. Combination BlitzNet and joint constraint. (a) Stable feature points obtained by the depth constraint on the current frame. (b) Potential dynamic point detection by the joint constraint. (c) Fusion results. (d) Segmented dynamic objects.

4.2 Evaluation of SLAM System

In this section, we demonstrate the proposed method on TUM RGB-D datasets and adopt ORB-SLAM2 as the global SLAM solution. We select highly dynamic sequence *walking* and weakly dynamic sequence *sitting* to evaluation the SLAM system. Quantitative comparison results are shown in Tables 1, 2 and 3, where *static*, *rpy*, *xyz*, and *half* in the first column stand for four types of camera motions. The proposed dynamic detection thread combined with CNN is added to the system to accomplish the task of localization, thus metrics of absolute trajectory error (ATE) and relative pose error (RPE) are used for evaluation.

As we can see from Table 1, our method can make better performance in most high dynamic sequence such as *fr3/w/rpy*, *fr3/w/xyz* and *fr3/w/half*. Compared with ORB-SLAM2, our algorithm gets an order of magnitude improvement particularly in *walking* sequence, meanwhile, our positioning accuracy is better than DynaSLAM on *rpy*, *xyz*, and *half* camera motions in *walking* sequence.

What Table 2 gives is the relative attitude error under the same datasets, where RMSE (T) is the root mean square error of translation, and RMSE (R) the root mean square error of rotation. It can be seen from the data that our algorithm still has better robustness in relative posture than DynaSLAM and ORB-SLAM2.

Table 1. Results of absolute trajectory error

Sequence	ORB-SLAM2			DynaSLAM			Ours		
	RMSE	Mean	Median	RMSE	Mean	Median	RMSE	Mean	Median
<i>fr3/w/static</i>	0.3194	0.2626	0.3761	0.0068	0.0061	0.0056	0.0078	0.0068	0.0062
<i>fr3/w/rpy</i>	0.5391	0.4884	0.4419	0.0354	0.0302	0.0260	0.0320	0.0260	0.0209
<i>fr3/w/xyz</i>	0.5979	0.5421	0.4707	0.0164	0.0140	0.0121	0.0153	0.0133	0.0118
<i>fr3/w/half</i>	0.4543	0.3777	0.2740	0.0296	0.0251	0.0200	0.0268	0.0228	0.0195
<i>fr3/s/half</i>	0.0185	0.0145	0.0123	0.0229	0.0201	0.0179	0.0235	0.0204	0.0178

For ORB-SLAM2, camera trajectories are more complete because the dynamic targets are not eliminated. Although a large number of frames can

Table 2. Results of relative pose error

Sequence	ORB-SLAM2		DynaSLAM		Ours	
	RMSE (T)	RMSE (R)	RMSE (T)	RMSE (R)	RMSE (T)	RMSE (R)
<i>fr3/w/static</i>	0.1928	3.5992	0.0089	0.2612	0.0103	0.2714
<i>fr3/w/rpy</i>	0.3881	1.5906	0.0448	0.9894	0.0426	0.9483
<i>fr3/w/xyz</i>	0.4090	7.6553	0.0217	0.6284	0.0199	0.6018
<i>fr3/w/half</i>	0.3215	6.6515	0.0284	0.7842	0.0261	0.7394
<i>fr3/s/half</i>	0.0209	0.5614	0.0325	0.8822	0.0276	0.7475

Table 3. Results of successfully tracked trajectory

Sequence	Total	ORB-SLAM2		DynaSLAM		Ours	
		Tracked	Ratio (%)	Tracked	Ratio (%)	Tracked	Ratio (%)
<i>fr3/w/static</i>	717	714	99.6	375	52.3	692	96.5
<i>fr3/w/rpy</i>	866	825	99.8	546	63.0	806	93.1
<i>fr3/w/xyz</i>	827	795	91.8	757	91.5	824	99.6
<i>fr3/w/half</i>	1021	942	92.3	525	51.4	1018	99.7

be ensured to be tracked, the accumulation of errors can eventually lead to failure of the navigation. DynaSLAM achieves a more accurate camera trajectory than the ORB-SLAM2, however, the frames tracked ratio of DynaSLAM without inpainting is not as good as the ORB-SLAM2. As shown in Table 3, our algorithm can keep most of the frames tracked with high accuracy, which provides a guarantee for long-term navigation.

An example of the estimated trajectories of the three systems compared to the ground-truth in *fr3/w/half* are illustrated in Fig. 6. There is a large difference between the trajectory of ORB-SLAM2 and the real trajectory, while DynaSLAM and our system maintain a smaller difference but our trajectory is more complete than DynaSLAM. In addition, the translation error diagram shows that our algorithm has better stability and robustness.

Dynamic object removal can improve the mapping quality effectively. Because of the limitation of computing resources, we adopt the way of off-line mapping. As shown in Fig. 7 ORB-SLAM2 cannot handle the dynamic environment in *fr3/w/xyz* dataset, in which point cloud with smear will be built. DynaSLAM can get a point cloud without semantics because it only identifies people in TUM data, whereas our algorithm can deal with dynamic object effectively and eliminate the drag effect significantly. Furthermore, the semantic information is mapped to the point cloud. It is clear that the static objects such as screens are marked by blue and chairs are marked by red in our results.

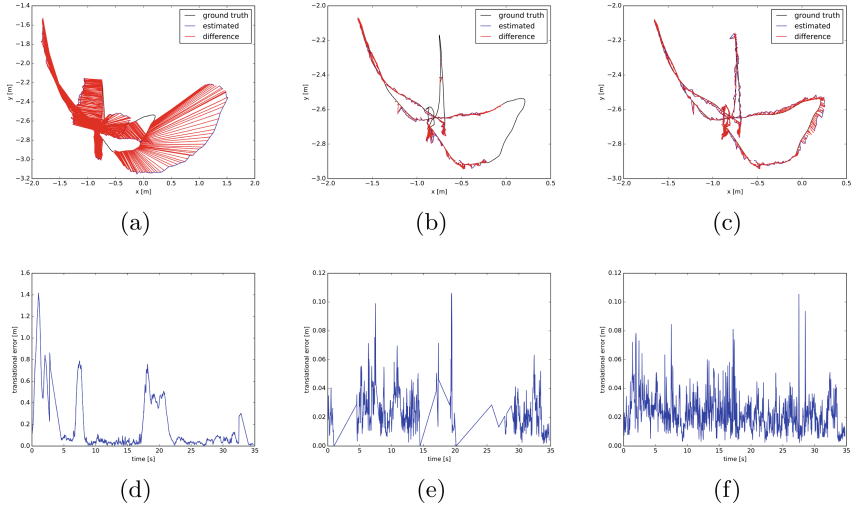


Fig. 6. Results of three algorithms in *fr3/w/half*. (a), (d) from ORB-SLAM2, (b), (e) from DynaSLAM, (c), (f) from our system.

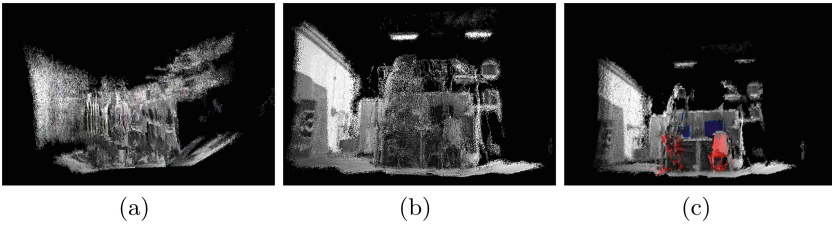


Fig. 7. Point cloud comparison of three algorithms in *fr3/w/xyz*. (a) ORB-SLAM2. (b) DynaSLAM. (c) Our system. (Color figure online)

4.3 Evaluation in Real-World Environment

In order to verify the robustness of moving object detection in dynamic environment, we use Xtion Pro camera to conduct extensive experiments in a laboratory environment. Xtion Pro camera can capture RGB images and depth images with 640×480 resolution. Before testing, we calibrate the camera in detail and use ROS to transmit the image data. The results obtained by the proposed method are shown in Fig. 8. In the experiment, the red points represent the dynamic feature points and the green points are the static feature points.

The first line shows a sequence of images taken in an office, there is a walker and a sitting person, and the sitting person can be regarded as a static target during this period of time. In the second line, most of the correct dynamic points can be constrained within the range of dynamic targets by the proposed joint constraints, but it is still a little insufficient, parts of the dynamic feature points on the walker are judged to be stationary. In the third line, combined with the



Fig. 8. Results in Lab environment. (Color figure online)

semantic information provided by the improved BlitzNet, the walker and the sitting person are distinguished effectively by the bounding boxes with region IDs. In the fourth line, we obtained a pixel-wise segmentation of the walker. In the real-world environment, the proposed algorithm is simple and feasible, and it can effectively identify the motion state of pedestrians.

5 Conclusion

In this paper, a semantic SLAM system based on joint constraint is proposed to detect the dynamic objects in the dynamic scene and accomplish the task of localization and mapping. The experiments on TUM dataset demonstrate the effectiveness and robustness of our system in localization. In addition, our system can obtain a more complete map with semantic information. Finally, we apply our algorithm to the real environment and it still has a notable performance. Future extensions of this work might include, among others, adaptive threshold method, on-line mapping and breaking the restrictions of application scope from semantic segmentation network.

References

1. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: large-scale direct monocular SLAM. In: European Conference on Computer Vision, pp. 834–849 (2014)
2. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(3), 611–625 (2018)

3. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robot.* **33**(5), 1255–1262 (2016)
4. Wolf, D.F., Sukhatme, G.S.: Mobile robot simultaneous localization and mapping in dynamic environments. *Auton. Robots.* **19**(1), 53–65 (2005)
5. Wang, C.C., Thorpe, C., Thrun, S.: Online simultaneous localization and mapping with detection and tracking of moving objects: theory and results from a ground vehicle in crowded urban areas. In: *IEEE International Conference on Robotics and Automation*, pp. 842–849 (2003)
6. Bibby, C., Reid, I.: Simultaneous localisation and mapping in dynamic environments (SLAMIDE) with reversible data association. In: *Proceedings of Robotics: Science and Systems*, pp. 105–112 (2007)
7. Zou, D., Tan, P.: CoSLAM: collaborative visual SLAM in dynamic environments. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(2), 354–366 (2013)
8. Rebecq, H., Horstschaefer, T., Scaramuzza, D.: Real-time visual-inertial odometry for event cameras using keyframe-based nonlinear optimization. In: *British Machine Vision Conference* (2017)
9. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440 (2015)
10. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(12), 2481–2495 (2017)
11. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241 (2015)
12. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions (2015). arXiv preprint [arXiv:1511.07122](https://arxiv.org/abs/1511.07122)
13. Chen, L.C., Papandreou, G., Kokkinos, I.: DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **40**(4), 834–848 (2018)
14. Zhao, H., Shi, J., Qi, X.: Pyramid scene parsing network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2881–2890 (2017)
15. Zhao, H., Qi, X., Shen, X., Shi, J., Jia, J.: ICNet for real-time semantic segmentation on high-resolution images. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) *ECCV 2018*. LNCS, vol. 11207, pp. 418–434. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01219-9_25
16. He, K., Gkioxari, G., Dollár, P.: Mask R-CNN. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969 (2017)
17. Dvornik, N., Shmelkov, K., Mairal, J.: BlitzNet: a real-time deep network for scene understanding. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4154–4162 (2017)
18. McCormac, J., Handa, A., Davison, A.: SemanticFusion: dense 3D semantic mapping with convolutional neural networks. In: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4628–4635 (2017)
19. Li, X., Belaroussi, R.: Semi-dense 3D semantic mapping from monocular SLAM (2016). arXiv preprint [arXiv:1611.04144](https://arxiv.org/abs/1611.04144)
20. Bowman, S.L., Atanasov, N., Daniilidis, K.: Probabilistic data association for semantic SLAM. In: *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1722–1729 (2017)

21. Yu, C., Liu, Z., Liu, X.J.: DS-SLAM: a semantic visual SLAM towards dynamic environments. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1168–1174 (2018)
22. Bescos, B., FÁCil, J.M., Civera, J.: DynaSLAM: tracking, mapping, and inpainting in dynamic scenes. *IEEE Robot. Autom. Lett.* **3**(4), 4076–4083 (2018)
23. Xiang, G., Tao, Z.: Robust RGB-D simultaneous localization and mapping using planar point features. *Robot. Auton. Syst.* **72**, 1–14 (2015)
24. Sturm, J., Engelhard, N., Endres, F.: A benchmark for the evaluation of RGB-D SLAM systems. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 573–580 (2012)