Robust RGB-D SLAM in Dynamic Environment Using Faster R-CNN

Sifan Yang[1], Jinnan Wang[1,2], Guijin Wang[2], Xiaowei Hu[2], Maomin Zhou[2], Qingmin Liao[1]
[1] Department of Electronic Engineering, Graduate School at Shenzhen, Tsinghua University, China
[2] Department of Electronic Engineering, Tsinghua University, China
e-mail: wangguijin@tsinghua.edu.cn, e-mail: wangjinnan23@gmail.com

***Abstract*—Recent approaches for simultaneous localization and mapping (SLAM) have shown great success for static environment reconstruction. However, these methods suffer from performance degeneration in dynamic environment due to the unstable and ambiguous data associations. In this paper, we propose a robust and efficient method for SLAM in dynamic environment. To eliminate the influence of dynamic objects, we compute the difference between the current frame and keyframe by semantic and geometric information to detect the dynamic object. Then we pick up the reliable data association and get more accurate transform matrix. The experiments on the public dataset demonstrate that our proposed method outperforms the original method in dynamic environment.**

Keywords-component; SLAM; data association; dynamic environment; semantic

I. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) estimates the location of the on-board sensors and construct the map of the unknown environment perceived by the sensors. This is a fundamental problem in mobile robotics for navigation.

In recent years, researchers have been concentrating on the reconstruction of static environment [1]. Davison et al. [2] propose MonoSLAM to reconstruct the stationary scene successfully by a single camera. They choose Extended Kalman Filter (EKF) approach for SLAM and build a sparse but persistent map of landmarks within this probabilistic framework. However, it's inevitable to be inaccuracy and complicated as the landmarks grow due to linearization of EKF. Bundle adjustment [3] is well used for estimating higher accuracy of camera pose and larger sparse geometrical reconstruction [4,5] Klein and Murray [6] use Parallel Tracking and Mapping (PTAM) system to split tracking and mapping into two separate tasks, which makes track respond more quick and makes use of bundle adjustment to attain more precise accuracy independently. But it can't deal with the large, invariant or more occlusions environment. Mur-Artal et al. [7] propose ORB-SLAM2 which build on the ideas of PTAM, adds place recognition and loop closing modules. They use ORB features invariant to different viewpoint and illumination to track, map and do loop closing robustly. The experiment shows reliable solution to large stationary environment reconstruction.

All above of the methods works well in the stationary environment. However, these can't be applied directly to cope with the dynamic environment reconstruction. In dynamic environment reconstruction, the big problem is data association. When the outliers are not so many, we may bear the errors using RANSAC or ICP algorithm [8][9][10] to remove them. But when the number of outliers is too much, it's inevitable to corrupt the whole maps and brings catastrophic damage to the whole construction.

Some approaches [11,12,13,14] have been proposed for dynamic environment better in some aspect. Wolf and Sukhatme [11] propose a method to differentiate between static and dynamic parts using two distinct occupancy grid maps to represent the static and dynamic parts with EKF. But this method is limited by the reconstruction size and some special occasion where the dynamics are a lot or moving slowly. In 2015, Newcombe et al. [12] propose DynamicFusion. They remove the static scene and reconstruct the dense dynamic scene by generalizing the extendedly projective and volumetric TSDF fusion approach originally introduced by [13]. It provides a good idea for rebuilding the non-rigid scenes in real time. But this method may not deal well with more than one moving object and it's not aimed at the stationary environment reconstruction with dynamic object. In 2017, Sun et al. [14] propose a motion removal approach in dynamic environment. They use homography matrix to compute the frame difference between the current and last frame and filter out the region related to the only one motion cluster using Maximum-a-posterior (MAP). This result will degenerate when parallax between consecutive frames is large and is not suitable for many moving objects.

In dynamic environment reconstruction, we use the high-level and robust perception which is similar to [15] beyond basic geometry reconstruction. We first check whether there are objects moving by the threshold which represents consistency of matching and identify every potential candidate of the dynamic object. If the dynamic existence is confirmed, we compute the dynamic region and figure out the dynamic object efficiently. Then we will cull the wrong data association in dynamic region and add more new data association we don't have in static region. In this way, we refine efficient data association to get rid of the limitation in dynamic environment with semantic information. We attain accurate camera pose with pose graph and build 3D mapping in dynamic environment. Experiments on the public dataset demonstrate the robustness of our approach.

II. FRAMEWORK

The overview of our approach is illustrated in Fig. 1. It has four main steps. The first step is semantic mapping that uses Faster-RCNN to detect and identify the potential candidate of the dynamic object whose category will be

978-1-5090-6352-9/17/$31.00 ©2017 IEEE

labeled. Second we will distinguish the stationary from the dynamic environment and refine the data association by removing the mismatching related to the dynamics. Third we estimate the camera pose with better data association and the optimization of the graph. Last but not least, with our accurate pose estimation, we can reconstruct the dynamic environment successfully.

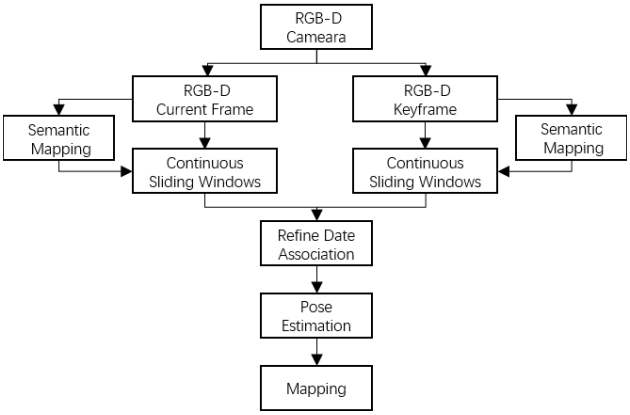

Figure 1. Overview of our approach

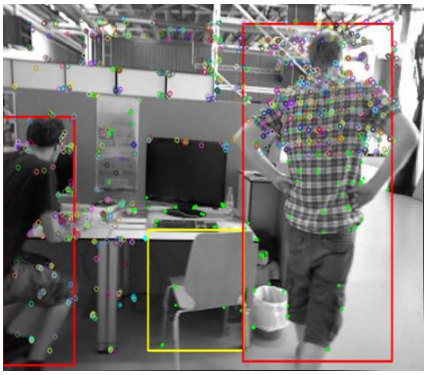

Figure 2. Image with orb features and semantic information

Our Robust SLAM in dynamic environment is mainly based on the two-fold adapted units; a real-time SLAM system and a Faster-RCNN [16]. We implement our SLAM algorithm based on the Faster-RCNN and ORB-SLAM2 [7] framework. The original real-time SLAM system can provide accurate data association in stationary situation and precise corresponding between frames in real time. However, the original SLAM system can't deal with highly dynamic environment reconstruction. Then due to the approach of the method they extract ORB features [17], the ORB features meet a homogeneous distribution. Even when the dynamic part isn't conspicuous, it may also produce enough corresponding relationship to generate the wrong data association relatively.

We precisely refine the corresponding data association of the static in dynamic environment by recognizing the dynamic and the stationary. It's practical to solve the problem of the reconstruction in the dynamic environment by removing the wrong data association of dynamic parts. In this way, we can finally degenerate the dynamic SLAM problem to the typically static SLAM.

A. Semantic Mapping

At the stage of Semantic Mapping, we take advantages of Faster-RCNN which has a good ability to identify different kinds of objects in real time to build a semantic mapping which involves the attributes of the object such as object category (e.g., chair, person), region in the picture, status (e.g., stationary, dynamic) and etc. In the initial frame, we assume that all objects are stationary. Then in next frames the status of objects will change into moving status from the state of rest, only when undistorted points of the region can't match with the points of the corresponding region in the frame. More details about the object's status transform will be discussed in refining the data association.

If the dynamic object can be detected by Faster-RCNN, we can utilize the semantic information which can be seen in Fig. 2 to predict the potentially moving candidate in dynamic environment. In the Fig. 2 the red rectangle represents the person and yellow rectangle represents the chair.

In our work, we detect every frame with Faster-RCNN. In most situation, it works fast and well. However, there're sometimes wrong detection or none detection in my tested dataset due to the blurred, occluded and other cases. We raise the threshold of matching and take the latest three sliding windows as the candidates of the current frame, thinking of the continuous movements of the dynamic object and camera, if the object of the current frame disappear or have a sudden change of its region. This approach alleviates the drawback of these special occasions.

Figure 3. A real example for outlier rejection

B. Refining Data Association

Our performance of refining data association is shown in Fig. 3. The upper picture has obvious wrong matches and dynamic points matches with the original approach. The most cause is the disability to distinguish the dynamic from the still for ORB-SLAM2. But the lower picture with our approach can recognize the dynamic and stationary object

from complicated environment. It shows good matches between stationary scene including stationary chair and don't make matches between dynamic objects such as moving people.

When we finish semantic mapping, we take the feature points which don't include the already dynamic part into account of data association and check whether the residual error of the corresponding pairs is over than the threshold. If so, we think there is something newly moving in the environment. Then we drop out the data association between the object regions which include dynamic and stationary part and compute the transform matrix $T_{h,k}$ start from scratch which is the rigid body transform from frame k to frame h. We take X_i^h and X_i^k as the ith corresponding vertex in the object region. We define $R_{h.k}$, $t_{h.k}$ is the rotation matrix and translation vector of the transform matrix $T_{h,k}$. At this time, we compute the consistency of the points J in corresponding object regions.

$$J = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left\|X_i^h - (R_{h,k}\cdot X_i^k + t_{h.k})\right\|^2} \quad (1)$$

It means that we project the feature points in the stationary region of the current frame k into the corresponding region of the keyframe h with the estimated camera pose $T_{h,k}$. Then we compute the similarity between these. If the J is over the threshold, we label the region of object as dynamic status from stationary status and update the data association by filtering out the data associations in the moving region. Once the status is dynamic, there's no chance for status to turn back. If the J is within the threshold, we label the region as stationary state and reserve previous data association. No matter which status the object is, we will get more precise and robust data association. The detected dynamic object is reserved for describing the environment's change and the detected stationary landmarks can be used for localization and navigation. At this time, the whole data association dataflow has accomplished. We can discriminate the dynamic and stationary part from the scene with geometric match and semantic information successfully.

C. Camera Pose Estimation

Before camera pose estimation, we first get the pyramid images with eight levels and extract about 1000 ORB features which are uniformly distributed in the pictures. Meanwhile we run the Faster-RCNN to detect and identify the object. Then we use our refined data association method to match features with keyframe. We estimate the camera pose $T_{1,k}$ using bundle adjustment. The local visible map for local bundle adjustment grows when we get new keyframe and more matches. If we detect a loop, we will campaign a global bundle adjustment to eliminate the accumulated drift in the loop and keep global consistency.

In the work, after attaining better data association, we apply the back-end of ORB-SLAM2 for more accurate camera pose estimate. Fig. 4 demonstrates that with our algorithm we achieve more impressive and robust result compared with the original approach in the experiment of location. We put a red ball in the valid point's position of the first frame which often represents the start of global map. As the camera moves, the location of the ball from different views depends the accuracy and robustness of camera pose you estimate. The upper image sequence shows the huge instability when people walk in different direction. However, the lower image sequence demonstrates that with our approach even people walk and move the chair accidently, the result of location of the ball is satisfied and robust.

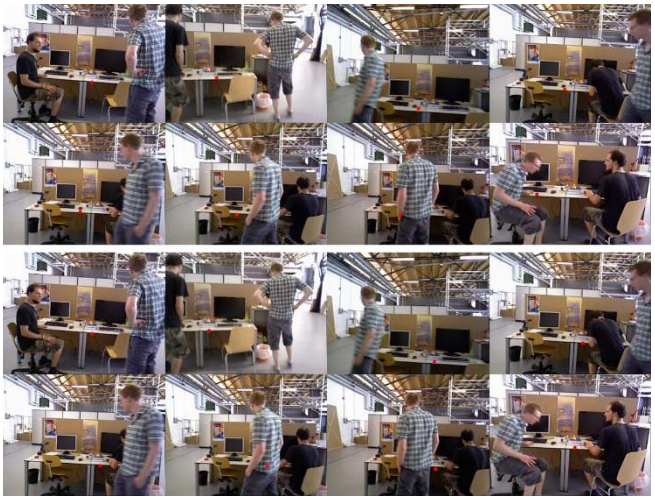

Figure 4. Verifying the robust of localization in the dynamic environment. The upper image sequence represents the result of original method. The lower image consequence represents our approach.

D. Reconstruction

When we have estimated the camera pose successfully, we can project the corresponding RGB-D frame into the common unified 3D map. As for each frame, we project the pixel with valid depth into the whole 3D map where we take the first frame's coordinate system as the global coordinate system. This process is the inverse operation of the experiment of location. As for the ith vertex in frame k, we can project the vertex X_i^k into the whole 3D map with the transform matrix $T_{1,k}$.

The $R_{1.k}$, $t_{1.k}$ is the rotation matrix and translation vector of the transform matrix $T_{1,k}$. The X_1^k is the corresponding vertex in the global map for X_i^k.

TABLE I. RESLUTS: RMSE OF ABSOLUTE TRAJECTORY ERROR

	Keyframes number	**RSME of ATE**	**RSME of x-axis**	**RSME of y-axis**	**RSME of z- axis**
ORB-SLAM2	204	0.7099	0.5381	0.2357	0.3986
Our approach	**69**	**0.2499**	**0.0127**	**0.1716**	**0.1812**
Improvements	66.17%	64.79%	97.64%	27.20%	54.54%

Figure 5. Part of the reconstruction result in dynamic environment. The top is the result of the original approach. The middle is result of our approach. The bottom is the result of ground truth.

III. EXPERIMENTS

We implemented our algorithm in C++ using Faster-RCNN and ORB-SLAM2 framework. The public dataset for high dynamic environment we choose is TUM RGB-D fr3/walking/xyz sequence. It contains about eight hundred frames for RGB-D.

In our experiment, we test two approaches in the public dataset which shown in Table I. The first is the original ORB-SLAM2 system. The second is our approach. Besides, we also record the RMSE of the different directions of the translational drift. The original ORB-SLAM2 method will get lost due to the mismatch between the frame and keyframe ignoring the dynamic object moving. The mismatch makes numbers of keyframe increase. Because people walk mainly in the direction of x-axis, so the RMSE of x-axis is large compared to other directions. Our approach culls the data association within dynamic region and adds data association within static regions. This can be found in Fig. 3. Meanwhile, we add the continuous sliding windows to protect us from blurred pictures which exist a few in the dataset. In this situation, we can make sure that the result is robust not to be disturbed by the walking man. As we can see from Table I., RMSE improvement value for high dynamic sequences are 64.79%. The number of keyframe has reduce by 66.17%. Especially, it improves 97.64% value in the RMSE of x-axis direction. It also improves a lot in other directions. The less redundancy and more accurate camera estimation demonstrate the effectiveness and robustness of our approach.

From the top pictures of Fig.5, we can see that reconstruction with the original method is terrible. You can see the fogged and overlapping reconstruction of rough scene. When we watch the details about the table support, it's obvious that number of table support even has increased surprisingly. As for the middle pictures, the reconstruction is good, we can see the clear scene. And even when we watch the detailed table support, we can find the width of table support is even narrower than the bottom pictures. In a sense, the reconstruction of our approach is good as the result of ground truth in terms of the table support.

At last, with our approach, the experiment shows great robust performance in dynamic environment. We keep all the semantic information of the dynamic and stationary object and can manage the whole environment with dynamic and stationary part. The dynamic shows the change of the environment and the stationary can be used for navigation and other tasks. This can be applied for other use.

IV. CONCLUSIONS AND FUTURE WORKS

We have retrieved more precise estimation of the camera pose and achieved better 3D environment construction results even in the situation where there exist dynamic objects from public dataset. Besides, with the addition of the semantic mapping the data association have become more convincing and accomplished better performance. The advantages of our algorithm can choose the time when we need the object detection wisely. It is not necessary to do this in every frame which is time consuming. It achieves higher accuracy, higher speed, and more robustness. During the pose optimization, we drop the ambiguous data association which consists of the similar but not consistent features. Because the experiments show us that if you just remove the dynamic object bounding box region, the result can be good but not so good. The surrounding location of the dynamic object region can be changed permanently such as the chair.

In future work, we plan to expand our algorithm to retrieve the dynamic object in the still environment. We can both get better dynamic object models not just lots of confusing points and the whole stationary environment.

ACKNOWLEDGMENT

This work is by Tsinghua University Initiative Scientific Research Program (2016SZ0306), supported in part by the National Natural Science Foundation of China under Grant 61671266, 61327902, in part by the Research Project of Tsinghua University under Grant 20161080084, and in part by National High-tech Research and Development Plan under Grant 2015AA042306.

REFERENCES

[1] Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D.westlyeira, J., Reid, I. D., and Leonard, J. J. (2016). "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," IEEE Transactions on Robotics, 32(6), 1309-1332.

[2] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[3] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, “Bundle adjustment a modern synthesis,” in Vision algorithms: theory and practice, 2000, pp. 298–372.

[4] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski. “Building rome in a day.” In ICCV, pages 72 –79, 2009.

[5] B. Klingner, D. Martin, and J. Roseborough. “Street view motion-from-structure-from-motion.” In IEEE International Conference on Computer Vision (ICCV), pages 953–960, 2013.

[6] G. Klein and D. Murray, “Parallel Tracking and Mapping for Small AR Workspace,” in Mixed and Augmented Reality (ISMAR) 2007. 6th IEEE and ACM International Symposium on, pp. 225-234, 2007.

[7] Raúl Mur-Artal and Juan D. Tardós, “ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras,” IEEE Transactions on Robotics, vol. 33, no. 5, pp. 1255-1262, 2017.

[8] P. J. Besl and N. D. McKay, “Method for registration of 3-D shapes,” in Proc. Robotics-DL Tentative, 1992, pp. 586–606.

[9] Q. Miao, G. Wang, and X. Lin, “Kernel based image registration incorporating with both feature and intensity matching,” IEICE TRANSACTIONS on Information and Systems, 93(5), 2010,pp. 1317-1320.

[10] B. He, G. Wang, X. Lin, C. Shi, and C. Liu, “High-accuracy sub-pixel registration for noisy images based on phase correlation, ” IEICE TRANSACTIONS on Information and Systems 94.12 ,2011, pp.2541-2544.

[11] D. Wolf and G. Sukhatme, “Online simultaneous localization and mapping in dynamic environments,” In ICRA, 2004.

[12] R. A. Newcombe, D. Fox, and S. M. Seitz. “Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time.” In IEEE Conference on Computer Vision and Pattern Recognition, pages 343–352, 2015.

[13] B. L. Curless. “New Methods for Surface Reconstruction from Range Images.” PhD thesis, Stanford University, 1997.

[14] Yuxiang Sun, Ming Liu, and Q. H. Meng. Improving rgb-d slam in dynamic environments: A motion removal approach. Robotics & Autonomous Systems, 89:110122, 2017.

[15] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison, “SLAM++: Simultaneous localisation and mapping at the level of objects,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2013, pp. 1352–1359.

[16] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real time object detection with region proposal networks,” in Proc. Neural Inf. Process. Syst., 2015, pp. 91–99.

[17] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: an efficientalternative to SIFT or SURF,” in IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, November 2011, pp. 2564–2571.