Towards Dense Moving Object Segmentation based Robust Dense RGB-D SLAM in Dynamic Scenarios

Youbing Wang and Shoudong Huang
Faculty of Engineering and Information Technology
University of Technology, Sydney
Sydney, Australia
Email: youbing.wang@student.uts.edu.au, shoudong.huang@uts.edu.au

***Abstract*—Based on the latest achievements in computer vision and RGB-D SLAM, a practical way for dense moving object segmentation and thus a new framework for robust dense RGB-D SLAM in challenging dynamic scenarios is put forward. As the state-of-the-art method in RGB-D SLAM, dense SLAM is very robust when there are motion blur or featureless regions, while most of those sparse feature-based methods could not handle them. However, it is very susceptible to dynamic elements in the scenarios. To enhance its robustness in dynamic scenarios, we propose to combine dense moving object segmentation with dense SLAM. Since the object segmentation results from the latest available algorithm in computer vision are not satisfactory, we propose some effective measures to improve upon them so that better results can be achieved. After dense segmentation of dynamic objects, dense SLAM can be employed to estimate the camera poses. Quantitative results from the available challenging benchmark dataset have proved the effectiveness of our method.**

***Keywords*—moving object segmentation, motion segmentation, robustness, RGB-D SLAM**

I. INTRODUCTION

As simultaneous localization and mapping (SLAM) matures, more and more researchers are interested in enhancing its robustness thus making it more applicable in various challenging real scenarios [1] [2]. In sparse feature-based SLAM, although RANSAC [3] and robust kernels [4] are being widely used to handle outliers, good results can be obtained only when the outliers constitute a minority of the data. When it comes to dynamic scenarios where moving objects amount to be a larger part of the data, their performance will degrade dramatically [5]. Furthermore, sparse feature-based SLAM cannot handle motion blur and featureless regions. On the other hand, dense SLAM [6], as the latest achievement in RGB-D SLAM, is very robust against motion blur and featureless regions. Nevertheless, it is very susceptible to dynamic elements in the images. Therefore, neither kind of the available approaches in SLAM area is capable of handling scenarios composed of large motions, and how to extend these mature solution frameworks originally designed for doing SLAM in static environments to handle dynamic scenarios gracefully is still an open problem.

On the other hand, in multibody SLAM (also known as multibody structure and motion (MSaM) [7] in computer vision) which aims to describe the structures of both the static and moving objects as well as the objects' motions and camera poses, various motion segmentation methods have been proposed to firstly separate different motion groups from each other. Although, based on feature trajectory analysis, there exist some algebraic methods, such as generalised principal component analysis (GPCA) [8] and local subspace affinity (LSA) [9], that can separate moving objects from each other as long as the objects are all rigid and enough frames have been taken into account, they usually require that the same feature set is available throughout the *n* chosen frames (therefore called n-view based motion segmentation thereafter for convenience), which amounts to be a very tough condition in real scenarios. As the result, most of the applicable motion segmentation methods in multibody SLAM/MSaM are two-view based [7], which only need two frames for analysis, but it also means that across every two consecutive video frames the moving objects may not be separated out completely. Instead, due to the inherent sensor noises, only when those movements go well beyond the noise level can the motion segmentation methods detect them as moving parts confidently.

Nevertheless, as the latest achievement in computer vision for n-view based motion segmentation, [10] tries to get grid of some unrealistic conditions and has proved to be capable of producing good dense segmentation results of non-rigid moving objects for some real data.

In this paper, we are concerned about doing SLAM with some continuous videos taken by a freely-moving RGB-D camera which is the only input and no other prior knowledge available. In addition to motion blur and featureless regions, those videos contain non-rigid objects that may be static, partially moving, or totally moving within large ranges from time to time. To cope with motion blur and featureless regions, dense SLAM [6] is a better choice than those sparse feature-based methods. However, special measures need to be taken to enhance its robustness in dynamic scenarios. Inspired by multibody SLAM, to do dense SLAM in such kinds of environment, we need to densely separate the moving objects out before doing SLAM. And [10] is quite promising for this goal. To substantiate our ideas, firstly, through theoretical analysis, we argue that moving object segmentation is a better choice to enable us to do robust SLAM in dynamic scenarios as well as multibody SLAM when compared with other motion segmentation methods. Then, we propose a practical avenue to achieve this goal: based on the output of [10], we propose practical measures to improve the dense segmentation results, then employ dense SLAM to estimate camera poses. The main characteristics of our method are that it aims to separate different moving objects out before doing SLAM, it can handle both rigid and non-rigid moving objects in a unified manner, and both the segmentation and SLAM are done densely.

The structure of this paper is as follows. Firstly we talk about different motion segmentation methods and RGB-D SLAM algorithms in Section II, showing the advantages of combining n-view based moving object segmentation and dense SLAM to handle general dynamic scenarios. After this, we propose the framework for robust dense RGB-D SLAM in dynamic scenarios, at the centre of which is a practical way for dense moving object segmentation based on [10] in Section III. Then in Section IV, to demonstrate the effectiveness of our approach, we show the moving object segmentation results and the improved SLAM results of our method compared with those of the original dense SLAM as well as sparse feature-based SLAM when possible using some challenging real data. And Section V concludes the paper.

II. Related Work

In this section, we firstly give a brief review of different ways to do motion segmentation. Motion segmentation constitutes a very important step for doing SLAM in dynamic scenarios, but practical ways capable of handling general dynamic scenarios are still missing.

Furthermore, we give a short comparison of the available methods for visual odometry in RGB-D SLAM, showing their advantages and disadvantages.

A. Motion Segmentation: Two-view Versus N-view

Motion segmentation aims to separate the available scenarios into different motion groups without prior knowledge about the moving objects or the camera's motion. Generally speaking, most of the available methods are only suitable for rigid or articulated objects, falling into either the two-view based or n-view based group. The output of the former group are the detected moving groups (corresponding to detectable moving parts beyond a threshold) of different objects; while that of the latter maybe correspond to the moving objects or not, depending on the whether the moving objects are rigid, the motion is big enough and the length n is big enough.

From $2D$ to $3D$, numerous two-view based motion segmentation methods have been proposed [11], [7], [5], [12]. They constitute a quick way for us to detect instant motion and get visual odometry in SLAM and structure and motion [7] [5]. However, intuitively, only when the two frames are discrete, the scenarios only contain rigid moving objects, and the motion between them are big enough (but not too big), two-view base motion segmentation methods can separate different objects into different motion groups. Otherwise, this kind of methods can only tell us those moving parts that have gone beyond the threshold determined by the sensor noise level. Accordingly, because of the inevitable overlapping between different motion groups, the visual odometry and loop-closures constraints obtained by two-view based motion segmentation methods are usually biased or even wrong no matter which threshold we choose, and a theoretically better way capable of avoiding this problem is the n-view based motion segmentation methods.

As a matter of fact, there are also lots of n-view based $3D$ motion segmentation methods available in computer vision, and interested readers can refer to [13] for a detailed review. Nevertheless, some strong assumptions have prevented most of them from finding practical applications in SLAM. Firstly, most of them usually assume that every point has the same trajectory length, which can only be met in controlled experiments. Secondly, most of them are sensitive to non-Gaussian noises and cannot tolerate errors brought forward from feature detection, matching and tracking. Last but not least, most of them can only handle rigid or articulated body while practical SLAM frameworks need to handle rigid as well as various non-rigid moving objects including human beings. Therefore, in SLAM, we need a robust motion segmentation method that can handle noisy continuous data with outliers and can handle non-rigid and rigid bodies in a unified manner.

B. From Motion Segmentation to Moving Object Segmentation

As we can see, depending on the composition of the related scenarios, chosen method and video length, motion segmentation results can be divergent. However, for most applications, separating moving objects apart is more desirable. For example, in the SLAM area, firstly it will enable us to get robust visual odometry results in dynamic scenarios. Secondly, the continuous presence of moving objects can also disable the traditional loop-closure estimation methods [14], and moving object segmentation is the only way out of the dilemma.

Nevertheless, moving object segmentation is even harder than motion segmentation when it comes to handling non-rigid moving objects. Until quite recently, [10] proposes a robust method for moving object segmentation based on long term point trajectory analysis without prior model, making it applicable for some real scenarios. Firstly, it can handle point trajectories of arbitrary length. Secondly, by employing spectral clustering and a model selection process, it can handle noisy data with outliers. Furthermore, for some scenarios that contain more than one non-rigid moving object, it can also get good segmentation results. It is based on this method that we propose a practical way to do dense moving object segmentation for robust dense SLAM in dynamic scenarios.

C. RGB-D SLAM: Sparse Versus Dense?

With the advent of affordable Microsoft Kinect, we are endowed with dense depth information along with RGB data. And with the help of pose-graph SLAM, localization and mapping can be decoupled into two sub-steps. In RGB-D SLAM, maps are naturally dense. However, when calculating camera poses, we can choose to employ sparse feature-based methods or dense methods to get visual odometry and loop-closure relative pose constraints [15]. Initially, sparse feature-based methods are very popular because dense methods are usually very slow. Now with the emergence of dense SLAM

[6], the situation is changing. Dense SLAM is making use of every pixel, both its color and depth information, and it can do so in real time. To cope with motion blur and featureless regions, dense SLAM is a better choice.

Nevertheless, as we will see, dense SLAM is vulnerable to the negative effects of dynamic elements in the images. As the result, if we still want to make use of dense SLAM to handle dynamic scenarios, we firstly need to densely separate the dynamic objects out. In this paper, we propose to combine a practical dense moving object segmentation method to fulfil this purpose, and the improved dense SLAM results after this step have attested the effectiveness of our method.

III. Moving Object Segmentation Based Robust Dense SLAM

A robust moving object segmentation method is proposed to enhance the robustness of dense SLAM in dynamic scenarios, and the overall process of our SLAM framework is illustrated in Fig. 1.

A. *Proposed Robust Moving Object Segmentation Method*

Moving object segmentation, as the pre-requisite of SLAM, need to be robust and adaptive enough to handle various dynamic scenarios. However, the state-of-the-art method may produce under or over-segmentation results, depending on the scenarios. Therefore, we propose further measures solely based on RGB information to overcome this kind of problems.

1) The Original Moving Object Detection and Segmentation Method and Its Results: As shown in Fig. 2, the original moving object segmentation method [10] can be divided into three steps: calculating optical flow, sparse point trajectory clustering and densification. Through this process, we can get good segmentation results for some videos including rigid and non-rigid moving objects. However, for many other videos, over-segmentation or under-segmentation can happen, and examples are shown in the left columns of Fig. 3 and 4, and tuning the related parameters can only help convert these two kinds of results into each other, instead of solving the problem.

2) Robust Measures to Improve Results: Since under-segmentation is harder to tackle, we firstly turn it into the over-segmentation scenario by tuning the related parameters of the original method.

Secondly, we propose to look for those separated regions that always share the same motion model during the process and merge them. According to multi-view geometry [16], for those points belonging to a rigid body or the static environment, if we can get their positions in every two consecutive frames based on [17], we can find a fundamental matrix to describe their motion. On the other hand, for non-rigid human body, we cannot find a fundamental matrix to describe its motion as a whole; instead, each part of it may need one to approximate its motion, and the corresponding fundamental matrices are changing. For those regions whose predominant parts always share the same fundamental matrices across the process, we will merge them into one group. A typical result is illustrated in Fig. 3. As we can see, after this step, over-segmented static groups can be found and combined together.

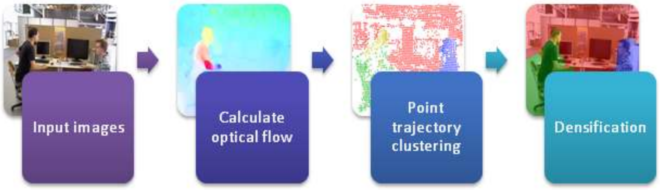


Fig. 2: Flowchart of [10] with images showing the results at different stages.

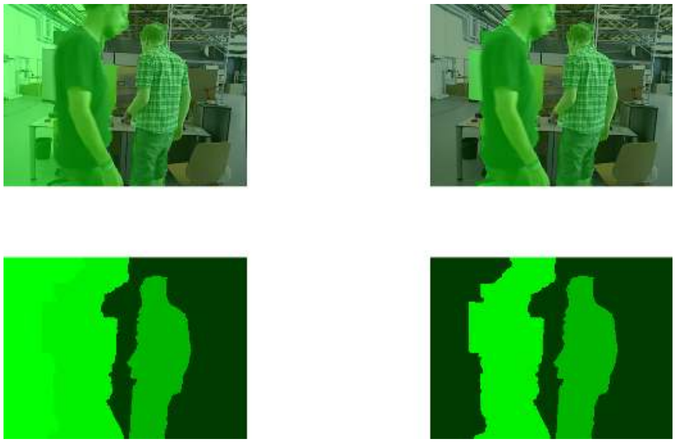


Fig. 3: The left column represents the result before the merging using fundamental matrix, while the right one represents that after merging. The lower images represent the segmented group masks in different color intensities, and the upper images are the combination of the original image and segmentation results. Best viewed in color.

Thirdly, based on the assumption that the largest group corresponds to the static environment, we look for the biggest group in each frame and remove it. And for the available benchmark dataset [18], the aforementioned assumption is valid.

Fourthly, we check the changes of the connection relationship between the remaining neighboring regions across all of the frames of the video. For those regions that appear simultaneously for some period of time and remain connected (the distances between the closest points from different regions are always within 1 pixel) during this process, it is highly possible that they belong to the same object, so we propose to agglomerate them at this step. This step can re-combine the over-segmented parts of human body together as shown in Fig. 4.

Through these steps, we can reasonably combine some segregated regions to produce more elegant segmentation results for the static environment and moving objects. Fig. 5 has shown the final segmentation results after applying our merging procedures.

B. *Dense SLAM*

After densely separating the moving objects out of the images, we can employ dense SLAM to get camera poses only

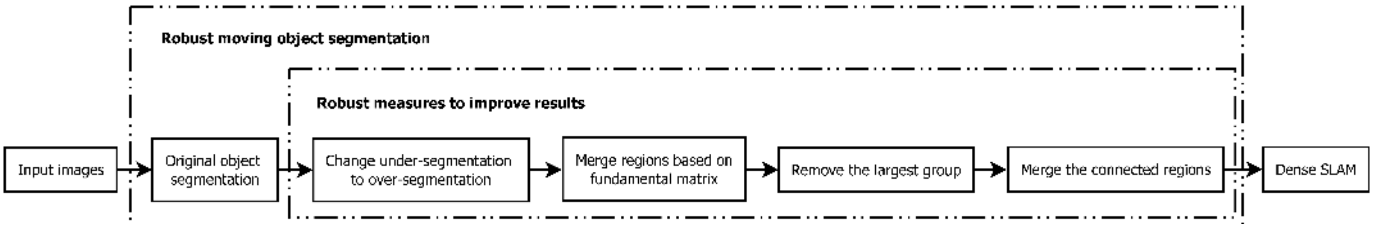

Robust moving object segmentation
Robust measures to improve results
Input images
Original object segmentation
Change under-segmentation to over-segmentation
Merge regions based on fundamental matrix
Remove the largest group
Merge the connected regions
Dense SLAM

Fig. 1: Flowchart of the whole process for robust SLAM

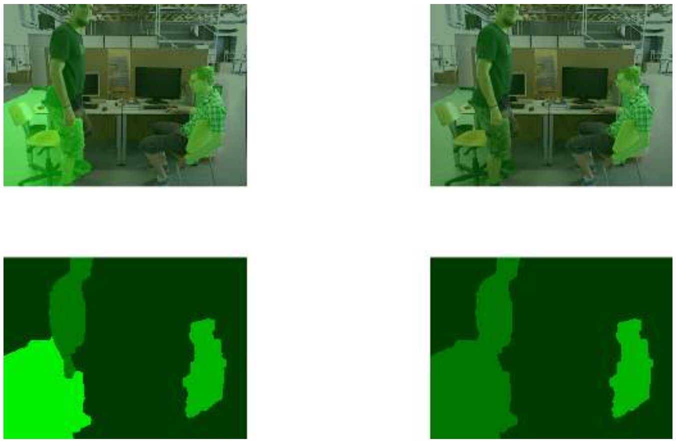


Fig. 4: The left column represents the result before merging the connected neighboring regions, while the right one represents that after merging. The lower images represent the segmented group masks in different color intensities, and the upper images are the combination of the original image and segmentation results. Best viewed in color.

using the remaining static parts.

The overall process of dense SLAM is as follows: after getting visual odometry by using both dense color and depth information, selecting keyframes and detecting loop-closures, dense SLAM makes use of g2o [19] for pose optimization, with pose trajectories as the output. Interested reader can refer to [6] for more details.

There are several practical reasons why we need to densely remove the moving objects before using dense SLAM. Firstly, to improve robustness, dense SLAM proposes a fast dense image registration method based on joint optimization of the color and depth errors of all the available pixels. Although t-distribution has also been employed to deal with large errors, as we will see, those pixels corresponding to the moving objects in the scenarios can impose unavoidable negative effects on the optimization results. Secondly, dense SLAM proposes an entropy-based method for keyframe selection and loop-closure validation to reduce drift. However, without firstly densely removing the moving objects, the entropy value will be spoiled. As the result, unexpected keyframes may be selected and false loop closures may be found. So, to some degree, dense SLAM amounts to be a method specially designed for static scenarios, and very susceptible to moving objects.

To further support our analysis, we have shown the dense SLAM results both before and after dense moving object segmentation in the next section as a comparison.

IV. Experimental Results Using the Benchmark Dataset

To show the effectiveness of our method, we have chosen to compare the SLAM results on some challenging benchmark sequences (the *walking* series) provided by [18].

A. RGB-D SLAM Benchmark and Methods Involved

Soon after the establishment of RGB-D SLAM as an independent research area, [18] provides a large benchmark dataset for it. It is composed of 39 sequences recorded using Kinect in different indoor environments. RGB and depth images are provided along with ground-truth trajectories of the camera obtained from a high-accuracy motion capture system.

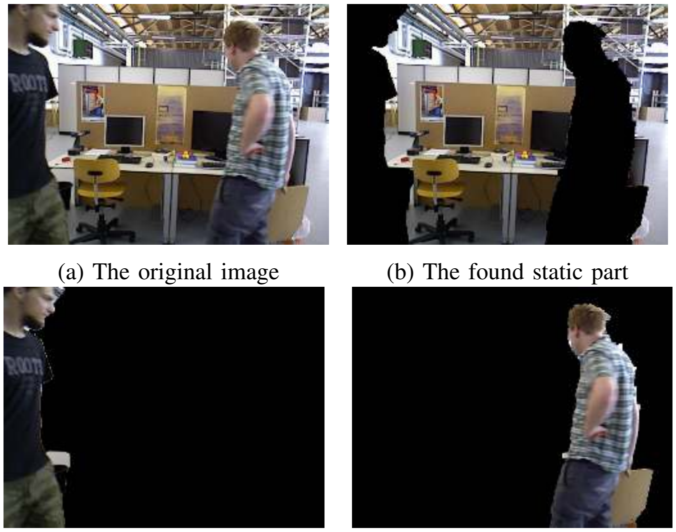
Although many good results have been reported for most of this dataset [6], the *walking* sequences are still among the few most challenging ones that have not been fully solved yet. According to [18], the *walking* sequences are specially designed for evaluating the robustness of visual SLAM and odometry algorithms when there are non-rigid moving objects dynamically occupying large parts of the visible scene. The major difficulties of these sequences lie in that motion blur, featureless zones and large human movements are pervasive in the images.

Recently, we propose a sparse feature-based two-view motion segmentation method [5]. By combining with pose-graph SLAM, we reported a meaningful result for one challenging *walking* sequence. In addition, we have also chosen dense SLAM as another reference. Therefore, including our own method, there are altogether three methods involved in this paper for comparison. Nonetheless, only for one sequence the results of the three methods are presented; while for the rest sequences we mainly compare the results gotten by our method with those obtained by directly applying dense SLAM to the original videos.

B. Comparison of the Experimental Results Using Real Data

1) Experimental sequences: We firstly applied our method and dense SLAM to four challenging *walking* sequences in the TUM dataset, whose details are illustrated as follows:

a) walking_static: In this sequence, the camera is being kept in place manually, so its movement is small, whereas two people are moving around the table with large motions.



(a) The original image

(b) The found static part

(c) Found people1

(d) Found people2

Fig. 5: The original image and segmentation results (from the *walking_static* sequence).

b) walking_halfsphere: In this sequence, the camera is moving on a small half sphere whose diameter is about one meter, while two persons are freely walking around in the office scene.

c) walking_xyz: In this sequence, the camera's movement is small while the two people are moving around the table arbitrarily.

d) walking_rpy: In this sequence, the camera mainly rotates, and two people are walking around the table. Since large parts of the visible scenes are dynamic, it constitutes a very difficult task.

2) Comparison of the results: Following the rules proposed by [1] [6], we have quoted both the root mean square error (RMSE) of relative pose error (RPE) in meters per second and the RMSE of absolute trajectory error (ATE) for comparison.

Among the four sequences, [5] only reports a meaningful result for the first one: the RMSE of RPE is $0.084m/s$, and the RMSE of ATE is $0.161m$. And for parts of the other three sequences, the overall comparison of the results obtained by our method (represented as MS_DSLAM) versus those of dense SLAM is summarized in Table I. In addition, two representative detailed comparison of RPE of the four sequences is shown in Fig. 6, and similar results can be observed in the ATE case.

As we can see, the applicability of the traditional sparse feature-based SLAM to challenging dynamic scenarios is limited [5], nor could dense SLAM [6], which is known as the state-of-the-art method in RGB-D SLAM, produce good results as in static scenarios, while our method has demonstrated its effectiveness and robustness.

3) Discussion: As we know, sparse feature-based methods as a classical choice in SLAM can usually produce satisfactory results. However, as noted in [5], when it comes to blurred images and featureless regions in dynamic scenarios, both the quantity and quality of detectable features decreases dramatically, thus jeopardizing the applicability of this kind of methods in this case. Similarly, the results also prove that dense SLAM is very sensitive to moving objects, as our previous analysis has indicated.

TABLE I: Comparison of the RPE (m/s) & ATE (m) Results of Dense SLAM Versus Those of Our Method (MS_DSLAM) Using the Four Sequences

Seq.	**RPE**		**ATE**	
	DenseSLAM	***MS_DSLAM***	***DenseSLAM***	***MS_DSLAM***
1	0.309	0.022	0.470	0.024
2	0.175	0.080	0.116	0.055
3	0.321	0.055	0.202	0.040
4	0.477	0.088	0.515	0.076

On the contrary, our method has managed to acquire good results comparable to those produced by dense SLAM in static scenarios. It has verified not only our related theoretical analysis, but also the usability and effectiveness of our practical measures in handling some challenging dynamic scenarios.

V. Conclusions

Compared to motion segmentation, moving object segmentation amounts to be a less understood, yet more difficult and more relevant research topic to doing SLAM in dynamic scenarios, semantic understanding of changing environments among many other practical applications.

In this paper, we propose a practical moving object segmentation method that can densely segment rigid and non-rigid moving objects in a unified manner by building upon the latest achievements in computer vision. Combined with dense SLAM, it constitutes a new way for robust dense SLAM in dynamic scenarios as well as multibody SLAM. Results from some challenging real data have shown that this method is quite promising.

On the other hand, in the near future, we aim to further improve the robustness of our method and the accuracy of the dense moving object segmentation results, and build a complete solution for robust SLAM in general dynamic scenarios and multibody SLAM problem.

References

[1] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for rgb-d cameras," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013, Conference Proceedings.

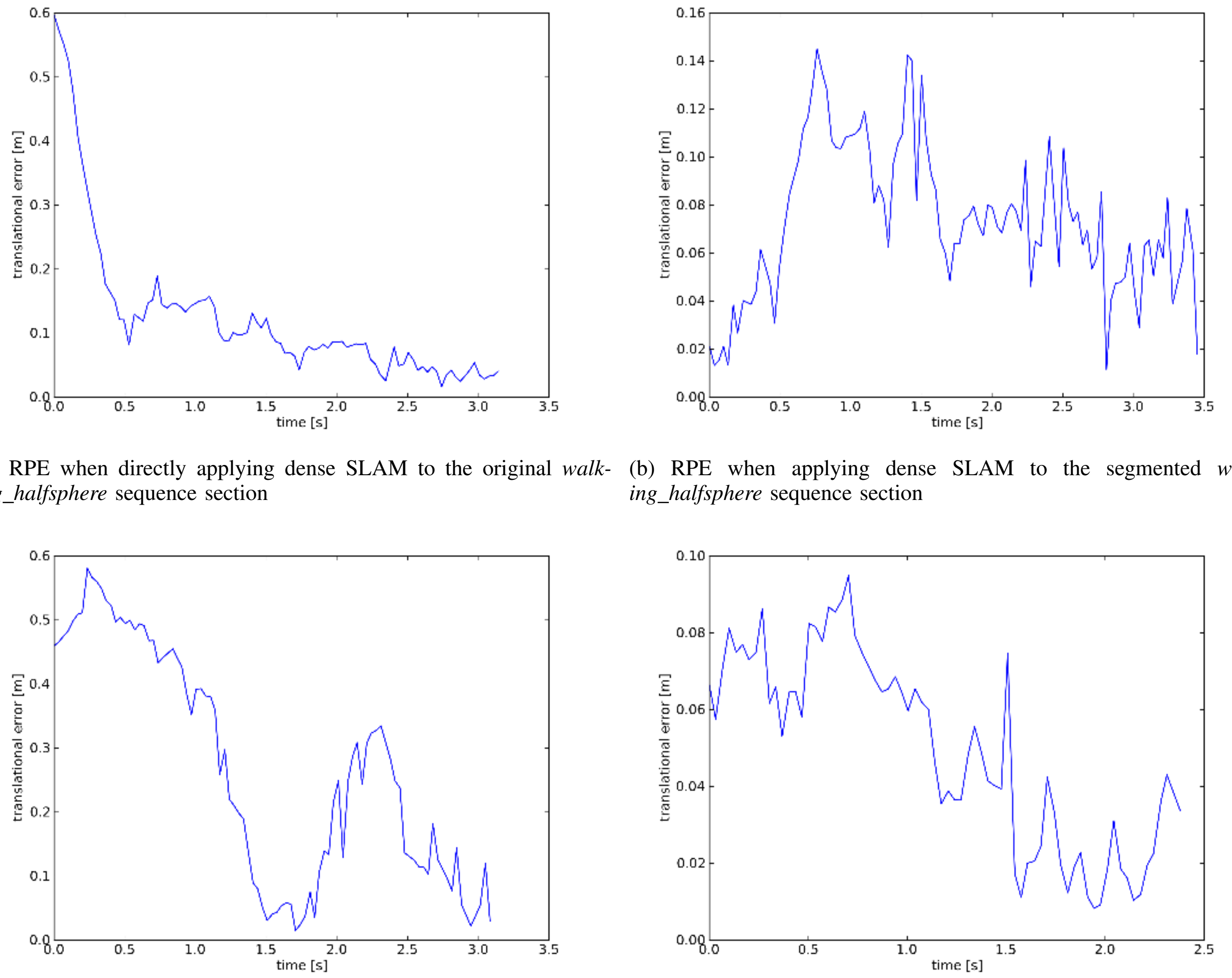
[2] N. Sunderhauf, "Robust optimization for simultaneous localization and mapping," PhD Thesis, 2012.

[3] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[4] P. J. Huber, "Robust estimation of a location parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

[5] Y. Wang and S. Huang, "Motion segmentation based robust rgb-d slam," in *World Congress on Intelligent Control and Automation (WCICA)*, 2014, Conference Proceedings.

[6] C. Kerl, J. Sturm, and D. Cremers, "Dense visual slam for rgb-d cameras," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2013, Conference Proceedings, pp. 2100–2106.



(a) RPE when directly applying dense SLAM to the original *walking_halfsphere* sequence section

(b) RPE when applying dense SLAM to the segmented *walking_halfsphere* sequence section

(c) RPE when directly applying dense SLAM to the original *walking_xyz* sequence section

(d) RPE when applying dense SLAM to the segmented *walking_xyz* sequence section

Fig. 6: Comparison of dense SLAM results before and after moving object segmentation using two of the four sequences.

[7] K. Schindler, U. James, and H. Wang, *Perspective n-view multibody structure-and-motion through model selection*. Springer, 2006, pp. 606–619.

[8] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (gpca)," in *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, vol. 1, 2003, Conference Proceedings, pp. I–621–I–628 vol. 1.

[9] J. Yan and M. Pollefeys, *A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate*. Springer, 2006, pp. 94–106.

[10] P. Ochs, J. Malik, and T. Brox, "Segmentation of moving objects by long term video analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 1, 2013.

[11] S. M. Smith, "Asset-2: Real-time motion segmentation and object tracking," *Real-Time Imaging*, vol. 4, no. 1, pp. 21–40, 1998.

[12] Y. Wang and S. Huang, "An efficient motion segmentation algorithm for multibody rgb-d slam," in *Australasian Conference on Robotics and Automation (ACRA)*, 2013, Conference Proceedings.

[13] R. Tron and R. Vidal, "A benchmark for the comparison of 3-d motion segmentation algorithms," in *Computer Vision and Pattern Recognition, IEEE Conference on*, 2007, Conference Proceedings, pp. 1–8.

[14] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," pp. 647–665, 2008.

[15] P. Henry, M. Krainin, E. Herbst, X. Ren, and D. Fox, "Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 647–663, 2012.

[16] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[17] L. Xu, J. Jia, and Y. Matsushita, "Motion detail preserving optical flow estimation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 9, pp. 1744–1757, 2012.

[18] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of the IEEE Int. Conf. on Intelligent Robot Systems (IROS)*, 2012, Conference Proceedings.

[19] R. Kummerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011, Conference Proceedings, pp. 3607–3613.