Proceedings of the 2016 IEEE
International Conference on Robotics and Biomimetics
Qingdao, China, December 3-7, 2016

A Novel and Effective Moving-objects Detection Method Combined with Stereo Localization and Mapping System

Libo Sun[1], Lei Fan[1] and Long Chen*

Abstract— **In this paper, we propose a novel moving-objects detection method which, in contrast to state-of-the-art moving-objects detection methods, takes static feature points into consideration during detection. It benefits both the tracking and mapping approaches in real-time Simultaneous Localization and Mapping (SLAM) system whose localization depends on static objects primarily. Our method obtains accurate static feature point sets continuously to estimate camera poses. In return, these camera poses are employed to find inliers and outliers using succinct distinction method. More specifically, we estimate camera pose by using static map points extracted from a high accurate global map. The map is generated in a real-time stereo SLAM system, and the camera pose is estimated and optimized by using a 3D-2D projection matching search with local bundle adjustment optimization. Once we have a calculated camera pose, we begin to find inliers and outliers. Finally, combined with superpixel segmentation, we capture the moving objects and then give feedback to the whole SLAM system. Detailed results are demonstrated comparing to other moving-objects detection methods on selected KITTI datasets with moving objects.**

I. Introduction

Real-time Simultaneous Localization and Mapping and 3D reconstruction is becoming increasingly popular in robotics research. Inspired by its great prospects, several SLAM works have been proposed to achieve accurate location and construction results [1-3]. However, one of the biggest problems of SLAM lies in the moving-objects detection. The moving objects which can not be accurately detected will be treated as wrong static objects, being one of the major error for both localization and mapping.

Detection and tracking of moving objects (DATMO) was first introduced into SLAM by Wang et al. [4]. They try to solve this problem by establishing a Bayesian framework fusing laser scanner. Besides lasers, some other sensors are also introduced, such as the inertial measurement unit (IMU) and the radar [4-6]. However, compared with general optical camera, these sensors may fail to provide color information itself, which holds back a better understanding of the environment.

Recently, multiple sensor-fusion methods have been proposed [6]. They argue that fusing various sensors can improve detection accuracy by reducing false positive detection and misclassification. However, combining data from different sensors may bring new problems, such as multiple-sensor calibration, signal synchronization, and information association.

L. Sun, L. Fan and L. Chen are with School of Data and Computer Science, Sun Yat-sen University, Guangzhou, Guangdong, P.R.China.

Fig. 1. The detection results of our algorithm. The left part is the segmentation of the left image which helps us understanding the depth information. The green line denotes hinge relationship, while the red and blue line stands for occlusion. The right part is the results of our method. Moving objects are covered by the disparity.

Stereo cameras have good senses of the color information of environment, and compared with monocular camera, it can provide the disparity of each pixel. With disparities, depth information of environment can be obtained by conversion operation [7]. Most of the current stereo-based methods can be roughly divided into two categories depending on whether using ego-motion knowledge or not.

Those methods without using ego-motion knowledge choose to implement under the drive of highlighting dynamic objects boundaries. Methods proposed in [5, 6] calculated gradients on optical flow images by employing Lucas-Kanade (LK) [7] method algorithm, and along with graph-cuts and Simple Linear Iterative Cluster (SLIC) super pixel segmentation methods, boundaries can be obtained and refined. However, these methods fail to avoid the bad effect of moving-objects shadows. The shadow of moving objects always moves simultaneously, which can not be distinguished by using optical flow.

In [8-11], diversified ego-motion estimation methods such as RAndom SAmple Consensus (RANSAC) are used to estimate relative camera pose between two consecutive frames. After the camera pose is obtained, Residual Image Motion Flow(RIMF) would be used to quantify the difference between optical flow and global optical flow of pixels to get moving-objects segmentation results [11, 12]. Then, the method use Gaussian approximation to propagate uncertainty of ego-motion.

As most of the current advanced moving-objects detection methods takes insufficient consideration of the useful information between several continuous frames, we choose to combine current advanced stereo SLAM with a new pair based moving-objects detection method (sample results are

978-1-5090-4363-7/16/$31.00 ©2016 IEEE

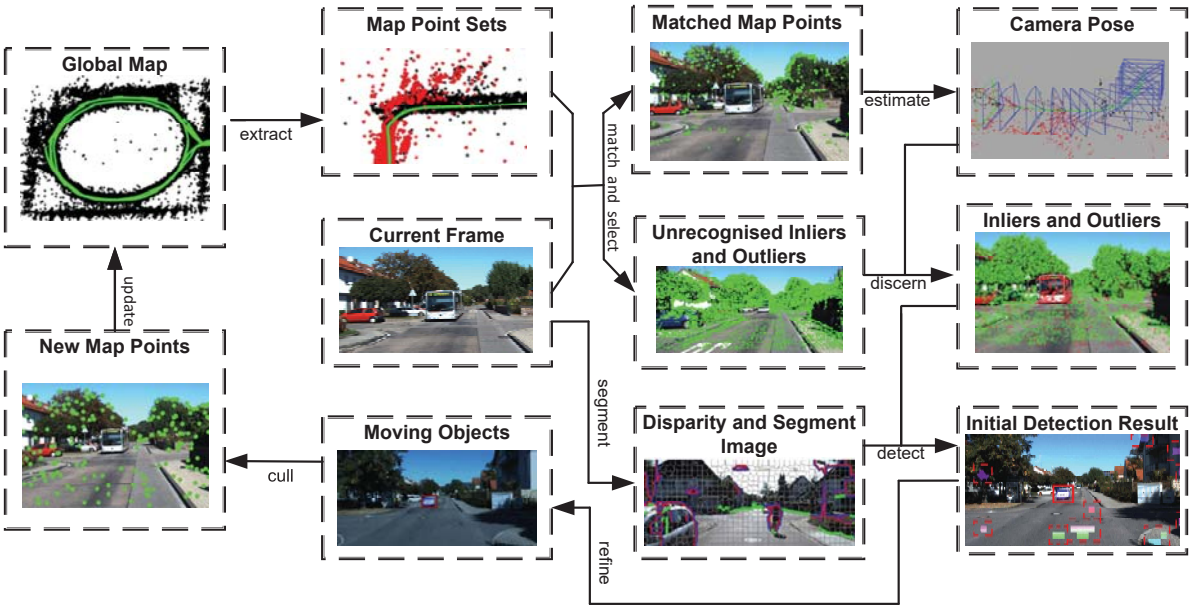


Fig. 2. Overview over our complete method.

demonstrated in Fig. 1) together. This combination can not only detect and cull moving objects to give a robust SLAM system, but also take outcome of the SLAM system into moving-objects detection. In the experiment, we quantitatively evaluate our approach compared with two current advanced methods [11, 13] which confirms higher accuracy obtaining detection results. Our main contributions lie in two aspects: (1) The moving-objects detection method we proposed is combined with current leading SLAM method using stereo camera only. (2) A novel static points concept is introduced into moving-objects detection.

The whole strategy of our method consists of five components which are static feature selection, disparity and segment image obtaining, inlier and outlier distinction, initial detection and final refinement. The rest of this paper are organized as follows: Section II introduces some related works in real-time SLAM and moving-objects detection fields. Section III describes our method, including its five main components. Section IV presents the experiments of our method on public datasets. Section V summarizes the whole process and gives conclusions.

II. Related Work

A. Feature-based Stereo ORB-SLAM

In feature based ORB-SLAM [2], tracking and mapping consist of two separate steps: Firstly, features are extracted from both left and right images and then associated 256 bits ORB descriptors are computed. Descriptors are used to lead a semi-global search between left and right images, after which epipolar match would be performed to obtain the subpixel accuracy disparity. Once the disparities of pixels are determined, depth information can be acquired.

Instead of directly using 3D-3D transform to estimate camera pose [3], a motion model is utilized to get a rough camera pose and then the optimization result will be obtained by using local bundle adjustment [2]. To guide the 3D-2D projection optimization, an initial map should be provided. In the process of initialization, subpixel accuracy matching locations are obtained by epipolar search. All these matched features which pass the related strict checks will be created as initial map points. Once an initial map is obtained, it will be updated when new frames come.

B. SPS-based Image Segmentation

Slanted Plane Smoothing (SPS) algorithm [14] assumes that 3D scenes consist of several piece-wise planes, which is an essential hypothesis for constructing 3D-planes during calculating the disparity image. SPS algorithm can be used for two purposes: (1) Segment the left image into superpixels and then classify the boundary types. (2) Perform dense stereo matching assisted by the boundary types. To get a SPS-based image segmentation, first, the initial semi-dense disparity maps are calculated from stereo pairs using the semi-global matching (SGM) [7] method. During the cost calculation, Hamming distance and the sum of absolute difference (SAD) are combined to measure matching similarities. Then, images are segmented into several superpixels by conducting SLIC [15]. Disparity information is also used as an attribute during segmentation for better clustering.

C. Moving-Objects Detection

Diversified ego-motion estimation methods are utilized in [6, 11] to estimate relative poses between two frames. With the threshold of RIMF difference for each pixel and the gradient of depth image, the work [11] gets the moving-objects detection result. The work [13] builds a 3D voxel-based map for dynamic obstacle detection in urban scenes. By using a flood-fill approach for cluster voxels, they can

Fig. 3. Static feature selection result on KITTI dataset. Green color represents static features.

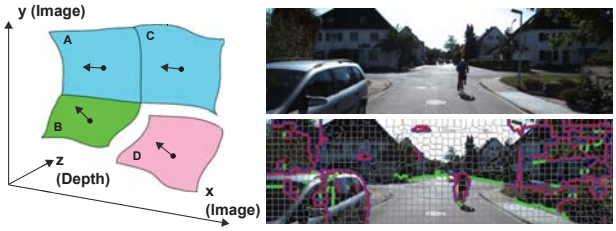


Fig. 4. The left image depicts the relationships between four superpixels, where arrows denote the normal vector. The relations between segment A and C, A and B, B and C are coplanar, hinge and occlusion respectively. The right image shows the segmentation of the left image, whose moving object is surrounded by hinge and occlusion.

label whether the cluster is moving or not combined with ego-motion knowledge.

III. System Overview

The overview of the proposed moving-objects detection method in the whole SLAM sytem is shown in Fig. 2. We choose stereo ORB-SLAM framework to implement visual SLAM, since far from now, it is the most accurate and fast open source SLAM system. SPS algorithm is introduced to segment the stereo pair and classify the boundary types between adjacent superpixels. In order to find inliers and outliers, which will be used to detect moving objects, a thorough but succinct distinction function will be implemented. Further more, we decouple final detection into two seperated steps - initial detection and final refinement. Outliers combined with segment results will be utilized to find an initial detection and then, refinement will be implemented to get final results. Five components of our method will be described in this section.

A. Static Feature Selection and Motion Estimation

One of the approaches proposed in this paper is that we use a set of static feature points to estimate camera pose and detect moving objects at the same time. The main reason of enabling ORB-SLAM to achieve high accuracy is that it can create an accurate set of global map points to estimate camera pose and update it in real-time. Once a feature point is chosen to be created as map point, it will be observed continuously. And if the following continuous several frames can not observe it successfully, it will be culled. This design enable ORB-SLAM system to be robust when facing small amount of dynamic objects. However, it does not try to cull moving objects and features of these moving-objects will be created as map points, which sometimes will not be culled as expected and have a negative effect on accuracy.

Our system, (show in Fig. 2), not only takes the advantage of using map points to estimate camera pose, but also takes these map points into consideration. In practice, we continuously use existing map points to estimate camera pose and detect moving-objects. If moving objects are detected, features of these moving objects will be culled and only these remaining features will be chosen to update the existing map. To get the camera pose, motion-only bundle adjustment (BA) [16] is introduced to estimate and optimize camera pose (i.e.,rotation matrix R and translation matrix t). For more details on this estimation, including its implementation, we refer to the ORB-SLAM work [2].

In fact, we derive a rule that, if a point was observed in several consecutive frames, it can be considered as potential static point, and only features whose observation frequencies are higher than a threshold will be selected as static features. These matched static features will not only be used to estimate camera poses, but also be used in the process of moving object detection. During the process of moving-objects detection, the observation frequency will be considered. We also hold the principle that if a feature has a high observation frequency, its location area has little chance to contain moving objects. After current camera pose and matched static features are obtained, they will be treated as a standard to find inliers and outliers, and then static features will be used to label static objects.

In practice, we can get almost full static feature points (shown in Fig.3), even facing large numbers of dynamic features, it can provide accurate results, since dynamic features can be easily detected and hardly included.

B. Disparity Calculation and Image Segmentation

In order to get image segmentation result, we first use piece-wise planar [14] which is assumed in 3D environment to derive smoothing slanted planes, and then SGM method [17] and SLIC-like superpixel segmentation [15] are performed to get disparity image and segmentation results. The disparity plane $\theta_i = (A_i, B_i, C_i)$ assigned to each segment i is obtained by applying RANSAC with the initial SGM output. Disparity estimation of each pixel can be estimated as:

$$\widehat{d} = A_i p_x + B_i p_y + C_i \tag{1}$$

where (p_x, p_y) represents the coordinate of pixel p.

To obtain segmentation and disparity image, an energy function E^* is defined as:

$$E^* = E(s, \theta, f, o, I, d) \tag{2}$$

where s represents the image segmentation, θ is the disparity plane, f assigns the "outlier flag", o is the line label between adjacent segments, I is the reference image for segmentation, and d is the initial disparity images, which tend to be smoothed. Boundary labels of segments and disparity image will be obtained by minimizing energy function.

We establish three boundary types to classify the relationship between adjacent segments, including hinge, occlusion

Fig. 5. This figure displays the outliers (red) and inliers (green). The outliers are mainly located on the moving bus. For other mismatched outliers, we use the SPS method to reduce the noises during capturing the final result.

and coplanar. Hinge means two neighboring segments' disparity only agree on the boundary, while occlusion stands for the disparities are not the same at all and coplanar represents they are at the same plane roughly. The normal vector of each superpixel is defined as:

$$\vec{n_i} = \frac{(A_i, B_i, -1)}{\sqrt{(A_i^2 + B_i^2 + 1)}} \tag{3}$$

and two important parameters which are utilized for boundary classification are defined as:

$$\alpha_{i,j} = arccos(\frac{\vec{n_i} \cdot \vec{n_j}}{\|\vec{n_i} \cdot \vec{n_j}\|}) \tag{4}$$

$$\delta_{i,j} = \frac{1}{\beta_{i,j}} \sum_{p \in \beta_{i,j}} (\widehat{d}(p, \theta_i) - \widehat{d}(p, \theta_j))^2 \tag{5}$$

where $\alpha_{i,j}$ represents the angle between i, j segment, $\beta_{i,j}$ is a pixel set which contains the boundary between i, j segment, and $\delta_{i,j}$ denotes the difference between the estimated disparities of the boundary pixels from segment i, j. These parameters enable us to determine the category between different segments and the relationship between these three boundary types is depicted in Fig.4.

In practical condition, this superpixel method not only allows to benefit from general moving objects detection, but also effectively reduces the bad effects of shadows of moving objects.

C. Inlier and Outlier Distinction

To keep the efficiency of our method, few features are detected at first to implement SLAM process and the matched static features in the SLAM system will be treated as initial inliers. Then, during the process of moving-objects detection, we detect more features in current frame to find moving-objects. As for these new detected features in the process of moving-objects detectionWe try to find features matching results P_{match} between current and previous frame. The matching work start from finding the best match in previous left for the feature candidates in the current left image with a MxM window at first, and then in previous right image, the current right image and finally the current left image again which form a circle match work.

An inlier and outlier label is adopted to find moving-objects. Inliers are considered as features which locate at static objects, while outliers are regarded as these features which belong to moving-objects. To distinguish inliers and outliers among the new detected features, a RANSAC-based inlier and outlier distinction is introduced.

We hold the view that the static-objects have much more match points than moving objects, which enables us to distinguish inliers and outliers. We use the matched features pair to compute the ego-motion of the camera by minimizing the sum of reprojection errors using Gausing-Newton optimization. The reprojection error function are defined as:

$$\sum_{i=1}^{N} ||x_i^{(l)} - \pi^{(l)}(X_j; r, t)||^2 + ||x_i^{(r)} - \pi^{(r)}(X_j, r, t)||^2 \tag{6}$$

where (r, t) are transformation parameters. $x_i^{(l)}$ and $x_i^{(r)}$ denote current left and right feature location of points. X_j represents 3D position of features in previous coordinate system. Both $\pi^{(l)}$ and $\pi^{(r)}$ are transformation and projection functions which transform and project features 3D position into 2D image plane. During this process, inliers and outliers can be distinguished in the RANSAC process (demonstrated in Fig. 5).

D. Initial Detection

The inliers and outliers vary among each segment, which prevent us from separating these segments by simply basing on their amounts. Before the detection, the selected static features are included as inliers and they will be considered as equal amount of inliers in the same location using their observation frequencies. . To get the satisfied segment evaluation result, both the proportion and amount of outliers are adopted, and we also include outlier cautiously by using the following judgement function:

$$O(i) = \begin{cases} good & \lambda_s min\{D(O(i), O(j)\} > T_s \quad j \in all \\ bad & otherwise \end{cases} \tag{7}$$

where O stands for outlier, λ_s is a constant, and T_s represents the threshold which depends on the total amount of inlier and outlier. Bad outlier will not be used for moving-objects detection. The result shown in Fig.6 (a) is the output of initial detection.

E. Final Refinement

In refinement, noises must be effectively culled and only these superpixels of the moving objects will be reserved. The relationship between each two segment and the boundary labels are quite useful in this refinement process. These noises on the road can be easily wiped off, since its relationships between adjacent segments are coplanar and these neighbor segments are estimated as static. The center of a moving object, which might be short of outliers, can be optimized by comparing with its neighbors. Fig.6 (b) shows as the final refinement result. After we get the final moving-objects detection results, we cull these features which locate on the area of moving-objects. The culling results will give feedback to the whole SLAM system, which can help to improve the robust and accurate of the SLAM system. As these features located on moving-objects are culled, the

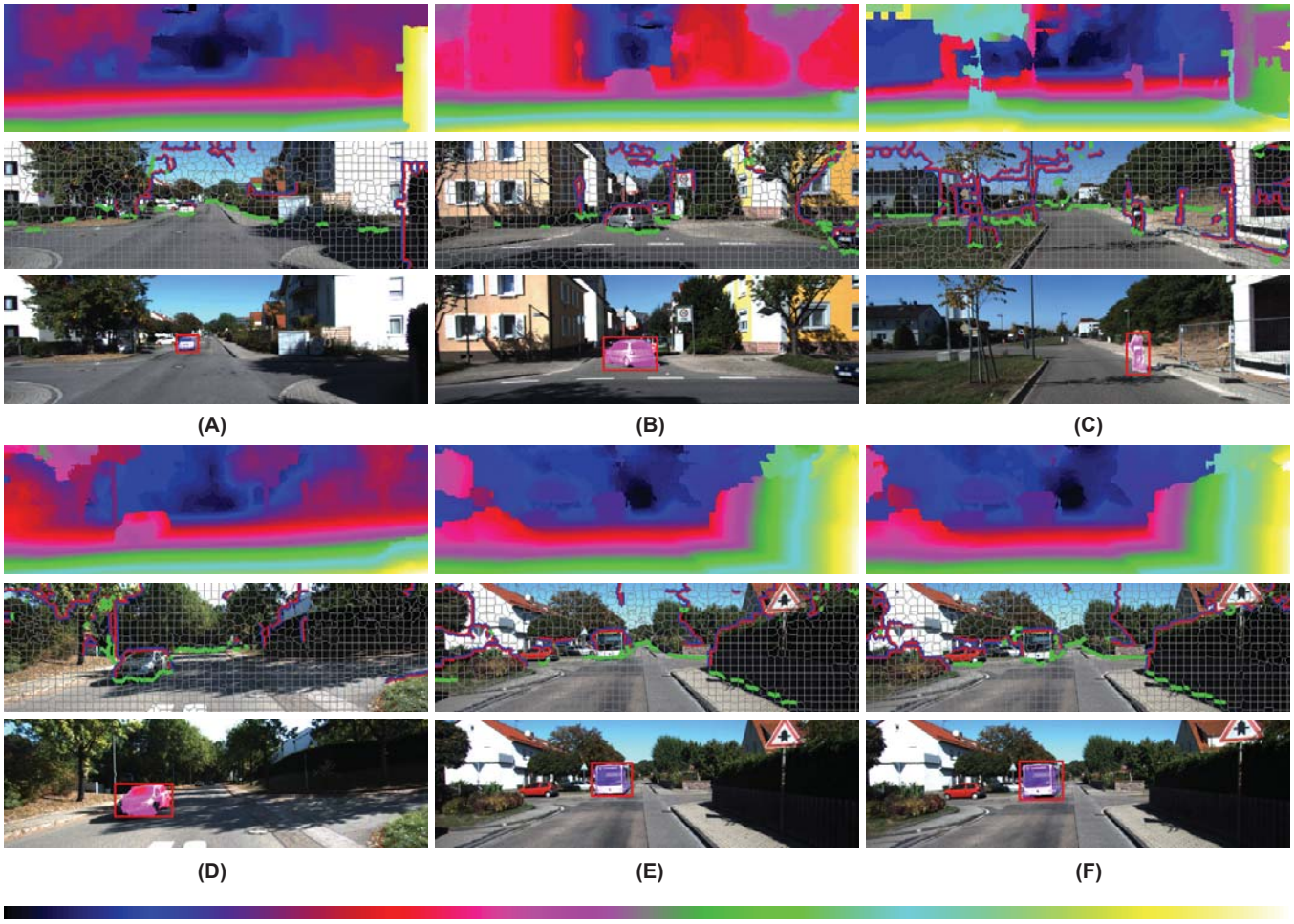


Fig. 7. Each group of results contains a disparity image, a segment image and the final moving-objects detection result. The detected moving objects are covered by the disparity, which helps us determine the position of the moving objects in following step.

TABLE I

AN SUMMARIZE OF PART OF OUR CHOSEN DATASETS FOR EVALUATING OUR ALGORITHM. DUE TO THE REASON THAT OUR ALGORITHM IS COMBINED WITH SLAM WORK, OUR TESTING IMAGES ARE SELECTED FROM KITTI ODOMETRY DATASETS. WE SELECT SEQUENCES WITH APPARENT MOVING OBJECTS, AND MOVING OBJECTS ARE COUNTED REPEATEDLY BETWEEN CONSECUTIVE FRAMES.

Sequence Number	Dataset Index	Frames	Start Index	End Index	Car	Bicycle	Bus
1	1	16	490	505	16	0	0
2	2	12	91	102	12	0	0
3	2	27	983	1009	0	27	0
4	4	14	7	20	28	0	0
5	4	12	117	128	24	0	0
6	5	12	2559	2571	12	0	0
7	5	12	2611	2622	0	0	12
8	5	12	2611	2622	0	0	12
9	5	12	2611	2622	0	0	12
10	5	12	2611	2622	0	0	12

points in the SLAM system are almost full static, which can be also treated as reliable inliers in the moving-objects detection.

IV. EXPERIMENTS

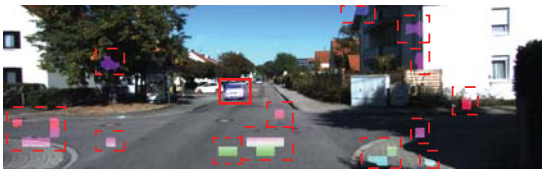
Table I demonstrates part of our selected sequences in KITTI odometry dataset [18]. Due to that majority of odometry dataset does not contain moving objects, we give the start index and end index of chosen sequences. These sequences include different kinds of moving objects, such as car, bicycle and bus. The ground truth of all these moving objects are tagged with bounding boxes manually.

Compared with these methods [11, 13], we embedded

TABLE II

THE RESULTS OF OUR ALGORITHM ON TARGET SEQUENCES, AND THE COMPARISON WITH RESULTS OF [13] AND [11].

Methods	True Moving	False Moving	False Static	True Static	Overlapping	
					20.0	50.0
Ours	211	5	14	n/a	94.78	91.74
Broggi's [13]	204	23	3	n/a	88.69	82.17
Zhou's [11]	192	21	17	n/a	91.30	83.49



(a)

(b)

Fig. 6. (a) The results before using the relations between semgents to refine. Note some noises are inevitably contained,which will be culled in the refinement process. (b) The results after refinement. Noises on the road are excluded for the reason that they are coplaner with other ground.

our moving-objects detection method into a real-time SLAM system to meet real needs and improve the reliability at the same time. Since it is an embedded function in stereo SLAM system, we choose continuous sequence datasets to implement experiments.

We perform our experiments on the selected sequences. Our method has good performance in practical test and detection results are shown in Fig.7. Comparison result with other two outstanding related works [11, 13] is shown in Table II. The proposed method achieves 9.57% and 8.25% higher accuracy when we measure with an overlapping area larger than 50%.

V. CONCLUSION

In this paper, we propose a novel moving-objects detection method for stereo SLAM system, which combine moving-objects detection with real-time SLAM system. Compared with these pair based moving-objects detection methods, our method takes inlier, outlier and static point into consideration, which not only enables us to get outstanding moving-objects detection results, but also makes whole SLAM system to get robust performances when facing moving objects. In our experiments, we show that the detection results of our approach is comparable to related outstanding methods and its performance in SLAM system is also well performed.

REFERENCES

[1] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European Conference on Computer Vision*. Springer, 2014, pp. 834–849.

[2] R. Mur-Artal, J. Montiel, and J. D. Tardós, "Orb-slam: a versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[3] B. Kitt, A. Geiger, and H. Lategahn, "Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme." in *Intelligent Vehicles Symposium*, 2010, pp. 486–492.

[4] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte, "Simultaneous localization, mapping and moving object tracking," *The International Journal of Robotics Research*, vol. 26, no. 9, pp. 889–916, 2007.

[5] L. Montesano, J. Minguez, and L. Montano, "Modeling the static and the dynamic parts of the environment to improve sensor-based navigation," in *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE, 2005, pp. 4556–4562.

[6] R. O. Chavez-Garcia and O. Aycard, "Multiple sensor fusion and classification for moving object detection and tracking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 525–534, 2016.

[7] H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2. IEEE, 2005, pp. 807–814.

[8] J. S. Kim, D. H. Yeom, and Y. H. Joo, "Fast and robust algorithm of tracking multiple moving objects for intelligent video surveillance systems," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 3, pp. 1165–1170, 2011.

[9] J. H. Ko, B. A. Mudassar, and S. Mukhopadhyay, "An energy-efficient wireless video sensor node for moving object surveillance," *IEEE Transactions on Multi-Scale Computing Systems*, vol. 1, no. 1, pp. 7–18, 2015.

[10] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: a systematic survey," *IEEE transactions on image processing*, vol. 14, no. 3, pp. 294–307, 2005.

[11] D. Zhou, V. Frémont, B. Quost, and B. Wang, "On modeling ego-motion uncertainty for moving object detection from a mobile platform," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, pp. 1332–1338.

[12] A. Talukder and L. Matthies, "Real-time detection of moving objects from moving vehicles using dense stereo and optical flow," in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, vol. 4. IEEE, 2004, pp. 3718–3725.

[13] A. Broggi, S. Cattani, M. Patander, M. Sabbatelli, and P. Zani, "A full-3d voxel-based dynamic obstacle detection for urban scenario using stereo vision," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 71–76.

[14] K. Yamaguchi, D. McAllester, and R. Urtasun, "Efficient joint segmentation, occlusion labeling, stereo and flow estimation," in *European Conference on Computer Vision*. Springer, 2014, pp. 756–771.

[15] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[16] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustmenta modern synthesis," in *International workshop on vision algorithms*. Springer, 1999, pp. 298–372.

[17] P. Lenz, J. Ziegler, A. Geiger, and M. Roser, "Sparse scene flow segmentation for moving object detection in urban environments," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 926–932.

[18] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3354–3361.