5G wireless mobile communications fusion cognitive radio (CR), Millimeter wave, Massive MIMO antenna array, ultra-dense networking, full-duplex communication (FD), and wireless full-duplex to significantly improved system performance. The characteristics of 5G vehicle networking are mainly reflected in low delay and high reliability to compare IEEE 802.11p with spectrum utilization.

The efficient use of spectrum is an important feature of 5G user experience. The application of 5G communication technology will solve the problems of the current vehicle networking resources. The efficient use of spectrum in 5G vehicle networking mainly in the following aspects:

1）D2D wireless communication;

2）full-duplex communication mode;

3）cognitive Radio technology.

5G wireless communication networks are expected to have ultra-high capacity and provide gigabit-per-second data rates for users. A millimeter-wave communication system with a frequency band of 30-300 GHz is proposed to exchange information between 5G terminals or between the base station. The millimeter waves have a very large bandwidth to provide very high data transmission rates. The interference of the environment and the probability of interruption of the connection which between the different terminals will be reduced in millimeter technology. Table II is a comparison of key technical parameters of between 5G vehicle network and IEEE 802.11p vehicle network. The result shows that 5G vehicle network has better wireless link characteristics than IEEE 802.11p vehicle networking.

Short range radar in ultra wideband operation at 24 GHz and at 79 GHz from 2013 at the latest will be used first in premium and later on in upper class models. Main applications will be ACC support, pre-crash detection, parking assistance, and blind spot surveillance. Market introduction of 24 GHz SRR will start in 2005. SRR sensors won't have angular measurement capabilities in the first generation (except the valeo-raytheon sensor), but future generations will also be able to provide angular information.

Although these sensors will be more expensive, they will contribute to the minimization of the total number of sensors and therefore they will reduce overall system costs. 77 GHz ACC systems will be extended to be operational at low speeds including full stop capability. This will provide increased customer benefits and it will contribute significantly to the market success of ACC systems.

In the same manner the 77 GHz sensor will be used not only for comfortable driving (ACC stop & go) but also for predictive and active safety systems.

Active safety systems up to an automatic emergency braking in unavoidable crash situations will be the key for a considerable reduction of the total number of crashes and fatalities.

Planar antennas in combination with digital beam forming provide interesting front end concepts for 77 GHz radar. These techniques might become feasible for high volume production as far as costs of 77 GHz components and powerful digital signal processing units will further decrease.

УДК 621.002

SEMI-DIRECT RGB-D SLAM ALGORITHM FOR MOBILE ROBOT IN DYNAMIC INDOOR ENVIRONMENTS

Tian Rui[1], Zhang Yunzhou[1,2], Gao Chengqiang[1], Deng Yi[1], Jiang Hao[1]

[1]Information Science and Engineering College, Northeastern University, Shenyang, China

[2]Robot Science and Engineering College, Northeastern University, Shenyang, China

Abstract. *In order to solve the problem of accurate navigation of mobile robots in dynamic indoor environment, a semi-direct RGB-D visual SLAM (Simultaneous Localization and Mapping) algorithm based on motion detection algorithm is proposed. The algorithm is mainly divided into three parts: motion detection, camera positioning and dense map construction based*

on TSDF (Truncated Signature Distance Function) model. Firstly, a preliminary estimation of the pose of the camera is achieved by using a sparse image alignment algorithm. Then, a real-time updated Gaussian model based on image patches is established to segment moving objects in the image. Based on this, the local map points projected in the moving area of image are eliminated, and the pose of the camera is further optimized. Finally, the TSDF dense map is constructed by using camera pose and RGB-D camera image information. The dynamic update of the map in real time is achieved by using the image motion detection result and the color change of the map Voxel. The experimental data under TUM dataset show that the proposed algorithm can effectively improve the camera positioning accuracy and real-time update dense map in indoor dynamic environment, which greatly enhances system robustness and environmental information for robot sensing.

Introduction. Autonomous navigation of mobile robot is a hot research field in robotics. In order to realize real-time and accurate positioning of mobile robot, vision-based real-time simultaneous localization and mapping (SLAM) system has been widely used. In particular, the SLAM system [1-4] based on RGB-D cameras can directly use the color and depth information to complete the perception of the camera's positioning and environmental information.

Visual Odometry [5] is a method of estimating pose of a robot by using continuous image sequence output by single or multiple cameras. Most VO systems at this stage assume that the environment in which the camera is located is static. However, in the actual environment, there are inevitably dynamic objects, such as walking pedestrians, moving tables and chairs and so on.

In the dynamic scene, the solution to the odometer problem can be divided into two categories. The first type is The algorithm of RDLSAM [6] based on the a priori adaptive RANSAC algorithm which use the probability model constructed according to the distribution of the map points. Bibby's proposed SLAMIDE algorithm [7] use the expectation maximization algorithm to update the feature point motion model in the scene and introduce the dynamic object into the SLAM, however, the dynamic Map points increase memory consumption and reduce the search speed for map points. The second method is to introduce Motion Object Detection (MOD), which divides the image area into static feature area and dynamic feature area. Wang [8] uses dense optical flow to segment the image sequence, but the algorithm can only be used for the motion segmentation of the image or video sequence. Sun [9] uses frame difference method to realize the segmentation of moving objects, and then uses the quantized depth image to realize the segmentation of dense point cloud map, the algorithm can solve the SLAM problem in dynamic environment, and takes about 0.5s to divide the part.

Most SLAM systems can only build sparse [10] or semi-dense [11] static maps, but they cannot be used for robot navigation. DynamicFusion algorithm proposed by Richard [12] can rebuild a dynamic environment, but it can only be used for smaller environments and needs to be compacted on the GPU. The dynamic change detection algorithm based on the TSDF map proposed by Google's Tango [13] project group can construct a dynamic map, but the algorithm can only be applied to slowly changing scenes and cannot reconstruct a high dynamic environment.

Based on semi-direct visual SALM, this paper proposes a new RGB-D SLAM algorithm which is suitable for indoor dynamic environment based on fusion of motion detection algorithm. In the visual odometer section, the initial pose of the camera is calculated using a semi-direct method. The motion detection algorithm is used to segment the static and dynamic regions of the image, the map points projected on the dynamic feature regions of the image are eliminated, and the pose and map points of the camera are further optimized by minimizing the re-projection error or the closed-loop constraint. Building on the known camera pose and dynamic region of the image, a TSDF dense map that can be dynamically updated in real time is constructed.

System Framework and Pipeline. The framework of system is shown in Figure 1. The system can be divided into four threads according to the implementation process namely tracking, local mapping, loop closure detection and dense mapping.

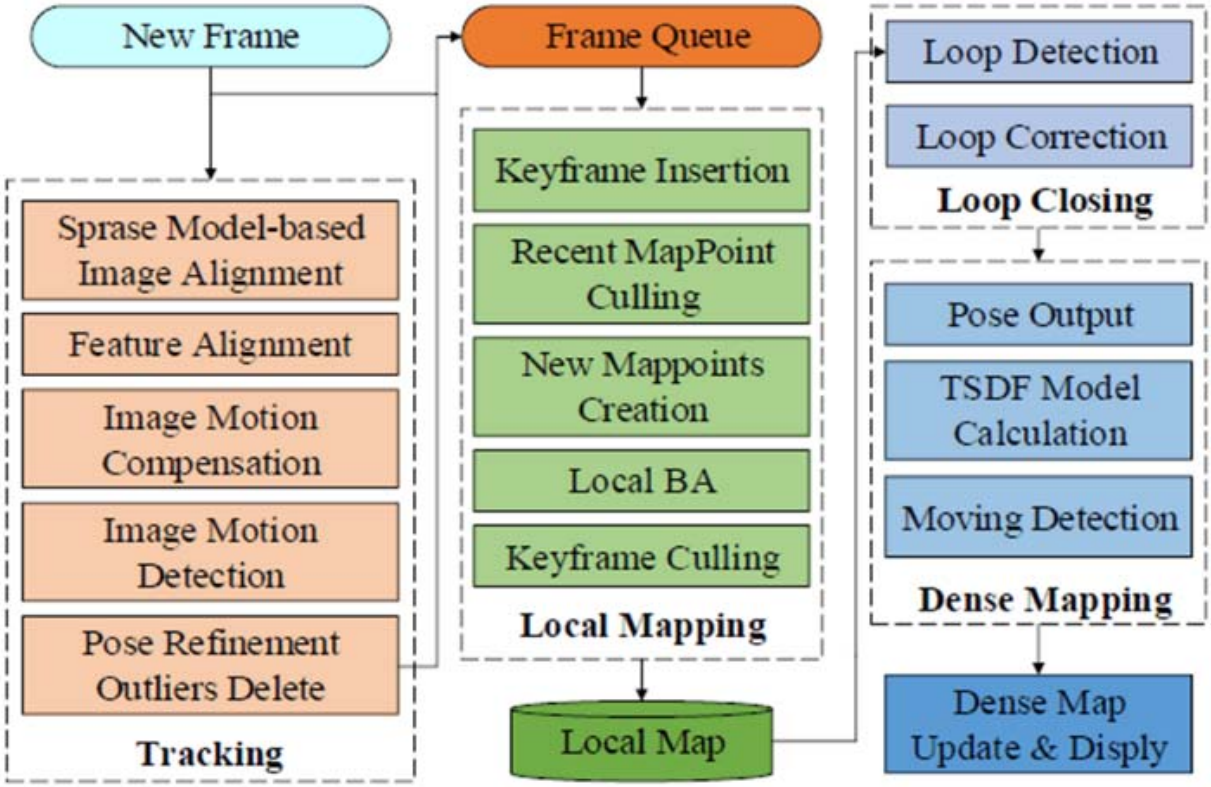

Figure 1 – Overview of our system

Semi-direct VO fused motion detection. In the part of visual odometer, the initial pose of the camera is estimated by minimizing the photometric error, and solve the feature points correspondence based on local map tracking. The motion detection algorithm based on motion compensation is used to segment the image. Remove the dynamic feature points of the image and gain an accurate solution of camera pose.

A. Initial estimate of camera pose. Assuming that a 3D point P_w in the world coordinate system has projection points in image I_k, I_{k-1}, the difference of the pixel grayscale after the projection is

$$\delta I(T_{k,k-1}, P_w) = I_k(\pi(T_{k,w} P_w)) - I_{k-1}(\pi(T_{k-1,w} P_w)) \tag{1}$$

By minimizing the photometric error, it can be converted to the least squares problem

$$T_{k,k-1} = arg\min_{T_{k,k-1}} \frac{1}{2} \sum_{P_w^i \in K} \left\| \delta I(T_{k,k-1}, P_w^i) \right\|^2 \tag{2}$$

In order to avoid repeated computation of the Hessian matrix in the solution of (4), Write (4) as follows using the Inverse Compositional Algorithm [14]

$$T_{k,k-1} = arg\min_{T_{k,k-1}} \frac{1}{2} \sum_{P_{k-1}^i \in M_{k-1}} \left\| I_{k-1}(\pi(T(\xi) \cdot P_{k-1}^i)) - I_k(\pi(T_{k,k-1} \cdot P_{k-1}^i)) \right\|^2 \tag{3}$$

Where, M_{k-1} represents the space point set in time $k-1$ in camera coordinate system, $T(\xi)$ represents the amount of pose update between frames.

Using Gauss Newton algorithm, and update the pose estimation of the camera:

$$\hat{T}_{k,k-1} = \hat{T}_{k,k-1} \cdot T(\xi) \tag{4}$$

B. feature points matching solution. First, the current frame image is divided into patches (10×10 pixels).In the local map, k key frames with the largest number of projection points in the current frame are selected. Then the map points included in the key frames are projected on the current frame image. Each patch includes a few projection points corresponding to the set P_{src} of map points. We select the most frequently map point P_{src} and select the matching point corresponding to the key frame K_j who is closest to the current frame in the observation value of P_i as a target observation point x_i to form a combination $\{P_i, K_j, \mathrm{x}_i\}$ for finding a matching point in the current frame.

After obtaining the initial pose of the camera in the previous step, the map points P_i can be projected onto the current frame as the features of the current frame according to the pose between the current frame and the key frame K_j. We introduced affine matrix A to optimize the positions of features by minimizing photometric error,

$$\mathrm{x}'_i = arg\ \min_{\mathrm{x}'_i} \frac{1}{2} \left\| I_k(\mathrm{x}'_i) - A_i \cdot I_j(\mathrm{x}_i) \right\|^2 \tag{5}$$

I_j is the grayscale image corresponding to the keyframe K_j.In order to compensate the error caused by the different exposure time of the camera at two different times, this paper introduces the gray value compensation γ,

$$\mathrm{x}'_i = arg\ \min_{\mathrm{x}'_i} \frac{1}{2} \left\| I_k(\mathrm{x}'_i) - A_i \cdot I_j(\mathrm{x}_i) + \gamma \right\|^2 \tag{6}$$

In order to simplify the calculation, we use the inverse construction method,

$$\mathrm{x}'_i = arg\ \min_{u'_i} \frac{1}{2} \sum_{\mathrm{x}^k_i \in P_r} \left\| I_j(A(\mathrm{x}^k_i, \Delta\mathrm{x})) - I(\mathrm{x}'^k_i) - \gamma \right\|^2 \tag{7}$$

P_r is the patch of 4×4 pixels around the feature point x_i in image I_r.

C. Detection of Movement. The detection of moving objects in SLAM belongs to the detection of moving objects based on free moving cameras, which is a problem cannot be solved with traditional static background based motion detection methods [15]. A model of gray level changing of image Patch based on SGM (Single Gaussian Model) is introduced in [16]. Background Gaussian Model and Candidate Background Gaussian Model are presented by two SGMs respectively, when a new image inputs, the two SGMs are updated simultaneously to avoid the influence of the foreground on the background model. When the mean value of the gray value changes beyond the threshold, background model and candidate background model will be exchanged. Foreground and background can be distinguished according to the change of variance of image Patch.

Figure 2 shows the motion detection using the dual SGM model after the image motion compensation using the initial pose.

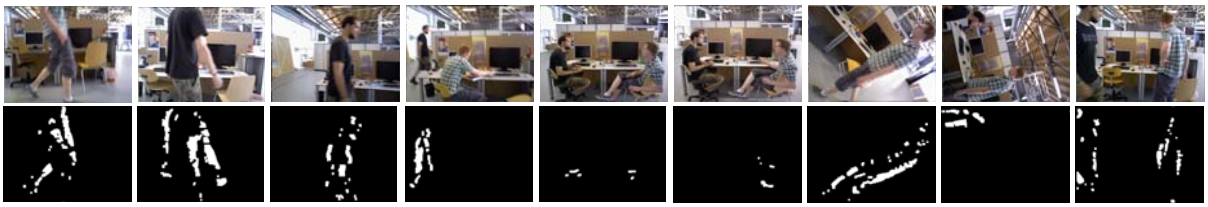

Figure 2 – Moving detection result. All the Image example are taken from "fr3/walking" sequence in TUM RGB-D datasets.
First row: the original RGB image.
Second row: the mask result of moving detection of local body motion

D. Dynamic Point Elimination and Camera Pose Optimization. In the previous step, the dynamic and static regions in the image were successfully segmented. In order to eliminate the outer points introduced by the moving objects in the scene, the feature points falling on the dynamic area of the image in the current frame image need to be removed to obtain the static point set $\boldsymbol{P_s}$.

After obtaining the set of static map points $\boldsymbol{P_s}$ and its matching point set x_s, the pose of camera can be further optimized by minimizing reprojection errors,

$$\hat{T}_{k,w} = arg\min_{\hat{T}_{k,w}} \frac{1}{2} \sum_{\mathrm{x}_i \in \mathrm{x}_s} \left\| \mathrm{x}_i - \pi(T_{k,w} P_i) \right\|^2 \tag{8}$$

Because BA problem of single pose point for constraint of map point is too restrictive, only the pose of camera is optimized, and the optimization of map points is completed in local BA.

In particular, the process will only be in static area when constructing map points by extracting FAST[17] corners from key-frames. Then filter map points and key-frames according to common view relationship between map points and key-frames. At last, select key-frames which have the same view with current key-frame and their observed map points to construct a local BA problem, and further optimize the pose of key-frames and map points.

Experiment Analysis. For the implementation of our method we use the Intel E3-1230 CPU, basic frequency is 3.30HGZ, with 12GB memory without GPU, and test in the Ubuntu14.04. The part of pose calculation is tested in the TUM RGB-D dataset, while mapping part are tested in the actual environment.

Many excellent SLAM algorithms [18,19] facing the dynamic environment also used the TUM RGB-D dataset which contains the dynamic scenes. In this paper we test the algorithm and make contrast with DVO, BaMVO.

A. Visual odometry evaluation. The evaluation of visual odometry mainly is based on RPE (Relative Pose Error). As shown in table 1, in the dynamic environment, the algorithm outperforms well in the "sitting" sequence and "walking" sequence. Both the translational and rotational RPE of this paper are significantly lower than the comparison paper in both low and high dynamic environments. The accuracy of our algorithm in the low dynamic environment increased to 50.6%, and the accuracy in the high dynamic environment increased to 43.1%. Compared with the algorithm which does not include motion elimination in this paper, the accuracy in the low dynamic environment increases to 8.6%, and the accuracy in the high dynamic environment increases to 64.9%.

Table 1 – Comparison of visual odometry RPE

Sequences		RMSE of translation drift [m / s]					RMSE of rotation drift [°/s]				
		DVO [1]	BaMVO [21]	Static weight [20]	Our Method without "RM"	Our Method	DVO [1]	BaMVO [21]	Static weight [20]	Our Method without "RM"	Our Method
static	fr2/desk	0.0296	0.0299	0.0173	0.0115	**0.0111**	1.3920	1.1167	0.7266	**0.5466**	0.5565
	fr3/long-house	0.0231	0.0332	0.0168	**0.0134**	0.0160	1.5689	2.1583	0.8012	**0.5086**	0.5167
low dynamic	fr2/desk-person	0.0354	0.0352	0.0173	0.0098	**0.0094**	1.5368	1.2159	0.8213	0.4685	**0.4674**
	fr3/sitting-static	0.0157	0.0248	0.0231	0.0111	**0.0081**	0.6084	0.6977	0.7228	0.3223	**0.2552**
	fr3/sitting-xyz	0.0453	0.0482	0.0219	0.0137	**0.0131**	1.4980	1.3885	0.8466	0.4941	**0.4941**
	fr3/sitting-rpy	0.1735	0.1872	0.0843	0.0231	**0.0229**	6.0164	5.9834	5.6258	0.7456	**0.6991**
	fr3/sitting-halfphere	0.1005	0.0589	0.0389	0.0340	**0.0263**	4.6490	2.8804	1.8836	0.9493	**0.7838**
high dynamic	fr3/walking-static	0.3818	0.1339	0.0327	0.0278	**0.0102**	6.3502	2.0833	0.8085	0.4903	**0.2525**
	fr3/walking-xyz	0.4360	0.2326	0.0651	0.1184	**0.0320**	7.6669	4.3911	1.6442	2.1494	**0.6869**
	fr3/walking-rpy	0.4038	0.3584	**0.2252**	×	×	7.0662	6.3389	5.6902	×	×
	fr3/walking-halfphere	0.2638	0.1738	0.0527	0.1184	**0.0476**	5.2179	4.2863	2.4048	1.8284	**1.045**

B. SLAM system evaluation. Unlike the assessments of visual odometry, SLAM systems use Absolute Trajectory Error (ATE) metrics. Experiments are still performed on TUM dynamic datasets.

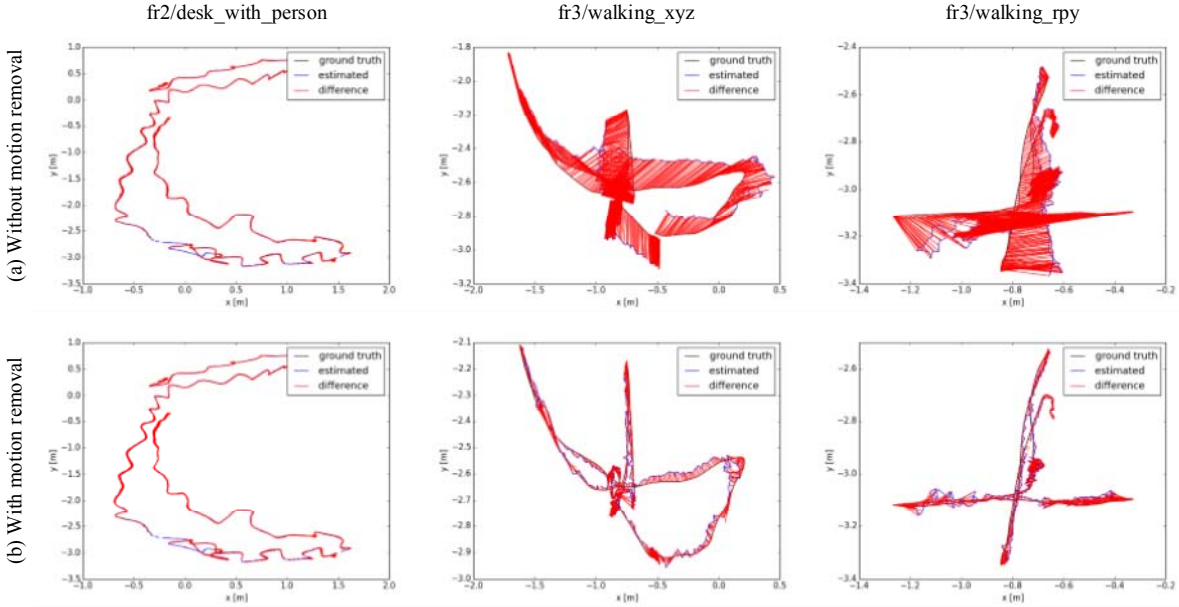

Figure 3 – Examples of estimated trajectories from our SLAM system:
(a) estimated trajectories of the without motion detection and removal SLAM;
(b) estimated trajectories of the complete SLAM system parameter as a reference

In Fig. 3, the trajectories of SLAM algorithm are compared. The blue trajectory represents the ground truth. The black line represents the estimated trajectory, and the red line represents the error. The more the red part is, the larger the error is.

Conclusion. This paper presents a real-time RGB-D SLAM algorithm for indoor dynamic environment. The algorithm uses the semi-direct visual odometry as the SLAM front-end to calculate the pose of the camera. By incorporating the motion detection algorithm, the moving objects and the removal of the dynamic feature points are completed, which

effectively improves the accuracy of the odometry and avoids the impact of dynamic objects in closed-loop detection step. Based on the pose of camera calculated by SLAM, a Mesh-like dense map is constructed based on the TSDF model. The algorithm supposed in this paper is validated in the TUM dataset and the actual environment. The accuracy of camera pose is much higher than the other SLAM algorithm in dynamic environments. The supposed algorithm not only locates the robot accurately for indoor dynamic environment, but also greatly enriches the environment information which is perceived by the robot.

In the future work, IMU (Inertial Measurement Unit) data will be considered in order to add up constraints to solve camera pose, which further improves the accuracy and robustness of the algorithm.

References

1. Kerl C, Sturm J, Cremers D. Robust odometry estimation for RGB-D cameras[C] // Robotics and Automation (ICRA), 2013 IEEE International Conference on. IEEE, 2013: 3748-3754.

2. Mur-Artal R, Tardós J D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras[J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-1262.

3. Newcombe R A, Izadi S, Hilliges O, et al. KinectFusion: Real-time dense surface mapping and tracking[C] // Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on. IEEE, 2011: 127-136.

4. Endres F, Hess J, Sturm J, et al. 3-D mapping with an RGB-D camera[J]. IEEE Transactions on Robotics, 2014, 30(1): 177-187.

5. Nistér D, Naroditsky O, Bergen J. Visual odometry[C] // Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. Ieee, 2004, 1: I-I.

6. Tan W, Liu H, Dong Z, et al. Robust monocular SLAM in dynamic environments[C] // Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on. IEEE, 2013: 209-218.

7. Bibby C, Reid I. Simultaneous localisation and mapping in dynamic environments (SLAMIDE) with reversible data association[C] // Proceedings of Robotics Science and Systems. 2007, 117: 118.

8. Wang Y, Huang S. Motion segmentation based robust rgb-d slam[C] // Intelligent Control and Automation (WCICA), 2014 11th World Congress on. IEEE, 2014: 3122-3127.

9. Sun Y, Liu M, Meng M Q H. Improving RGB-D SLAM in dynamic environments: A motion removal approach[J]. Robotics and Autonomous Systems, 2017, 89: 110-122.

10. Forster C, Pizzoli M, Scaramuzza D. SVO: Fast semi-direct monocular visual odometry[C] // Robotics and Automation (ICRA), 2014 IEEE International Conference on. IEEE, 2014: 15-22.

11. Engel J, Schöps T, Cremers D. LSD-SLAM: Large-scale direct monocular SLAM[C] // European Conference on Computer Vision. Springer, Cham, 2014: 834-849.

12. Newcombe RA,Fox D, Seitz S M. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time[C] // Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 343-352.

13. Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF[C]//Computer Vision (ICCV), 2011 IEEE international conference on. IEEE, 2011: 2564-2571.

14. Baker S, Matthews I. Lucas-kanade 20 years on: A unifying framework [J]. International journal of computer vision, 2004, 56(3): 221-255.

15. Van Droogenbroeck M, Barnich O. ViBe: A disruptive method for background subtraction [J]. Background Modeling and Foreground Detection for Video Surveillance, 2014: 7.1-7.23.

16. Yi K M, Yun K, Kim S W, et al. Detection of moving objects with non-stationary cameras in 5.8 ms: Bringing motion detection to your mobile device[C] // Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on. IEEE, 2013: 27-34.

17. Rosten E, Porter R, Drummond T. Faster and better: A machine learning approach to corner detection[J]. IEEE transactions on pattern analysis and machine intelligence, 2010, 32(1): 105-119.

18. Klingensmith M, Dryanovski I, Srinivasa S, et al. Chisel: Real Time Large Scale 3D Reconstruction Onboard a Mobile Device using Spatially Hashed Signed Distance Fields[C] // Robotics: Science and Systems. 2015, 4.

19. Li S, Lee D. RGB-D SLAM in dynamic environments using static point weighting[J]. IEEE Robotics and Automation Letters, 2017, 2(4): 2263-2270.

20. Kim D H, Kim J H. Effective background model-based RGB-D dense visual odometry in a dynamic environment[J]. IEEE Transactions on Robotics, 2016, 32(6): 1565-1573.