A Survey of the Simultaneous Localization and Mapping (Slam) Based on Rgb-D Camera

Zhifan Zhang[a], Mengna Liu[b], Chen Diao[c,*], and Shengyong Chen[d]

Tianjin University of Technology, Tianjin 300384, China

[a]13116150096@163.com, [b]lomoula@yeah.net, [c]diaochen@yeah.net, [d]csy@tjut.edu.cn

*Corresponding author

Keywords: Survey, Computer Vision, RGB-D Camera, SLAM, Visual Odometry.

Abstract: In recent years, the simultaneous localization and mapping (slam) have received increasing attention from computer vision and robotics, and multitudinous of results have been proposed. This paper gives a review of the slam framework base on rgb-d camera. Then, the paper provides insight into the developments on slam issues, such as visual odometry, back-end optimization and, loop closing, to address the major limitations still facing the rgb-d slam. Some latest results on the slam based on the rgb-d camera are also summarized. Finally, some conclusions are drawn, and several future research hot spots highlighted.

1. Introduction

The SLAM is a technique for estimating the motion state of the sensor while constructing a map in the unknown environment at the same time. The original SLAM is mainly used in robot autonomous robot [1]. It is widely utilized in 3D real-time modeling, autopilot on temporary field roads [2] and rescue tasks for complex environments [3-5]. The Visual SLAM (vSLAM) is the technique which used the camera as the only external sensor [6]. Although there are still many challenges facing camera-based sensors, it is expected that such solutions will eventually offer significant advantages over other types of sensors. In early vSLAM techniques, the Monocular camera and the Binocular camera are employed as the exteroceptive sensor. In recent years, the RGB-D camera has received increasing attention from civilian and military. Compared with traditional vSLAM, the main advantages of SLAM base on RGB-D are low cost, simple structure and easier to get the depth information.

Although multitudinous surveys for the vSLAM technique have been proposed [7-10], most are preferring to introduce the SLAM technology based on monocular or binocular, only a few of them handle RGB-D SLAM in an exclusive manner. [7] proposed the survey on vSLAM but did not thoroughly describe the different implementation details of each vSLAM solution. [10] also published a review on visual SLAM, but it is too early to summarize the latest trends and research hotspots. Therefore, the SLAM technique base on RGB-D camera is categorized and summarized as a survey paper. This paper is unique in that it systematically discusses the different components of SLAM technique bases on RGB-D cameras while highlighting the nuances that exist in their different algorithm.

This paper is organized as follows. Section 2 introduced the RGB-D camera and SLAM framework base on RGB-D camera. Section 3 gave a review of the different methods to solve the SLAM problem of the RGB-D camera and the weaknesses and the strengths of each one are discussed. Section 4 summarized some key technology on Visual Odometry, Back-End Optimization, and Loop Closing. The summarizes of the future research hot spots and the development trend of SLAM are provided in Section 5 and finally, Section 6 concludes the paper.

2. RGB-D SLAM framework

2.1 The universal RGB-D camera

The first RGB-D camera in the world is the Kinect camera produced by the Microsoft Corp in 2010. The structured light technique has chosen to be a technical scheme in Kinect camera. It consists of 3D Depth Sensors,Color Camera,Motorized Tilt and Microphone Array[11](see Fig 1).

The middle part of the Kinect camera is a RGB Lens, The two sides are infrared emitters and CMOS Image Sensors, both of which form a Depth Sensor.

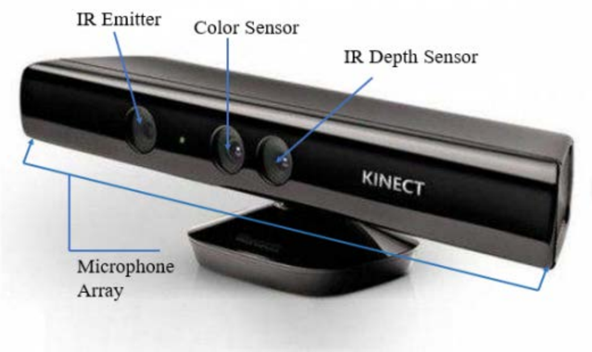

Fig.1 Microsoft Kinect

Finally, ASUS also produced a somatosensory camera Xtion Pro Live (see Fig 2). Its configuration of RGB Sensor and Depth Sensor are very similar to Kinect [11]. It consists of RGB camera and two, Color Sensor, IR Emitter and a pair of audio.

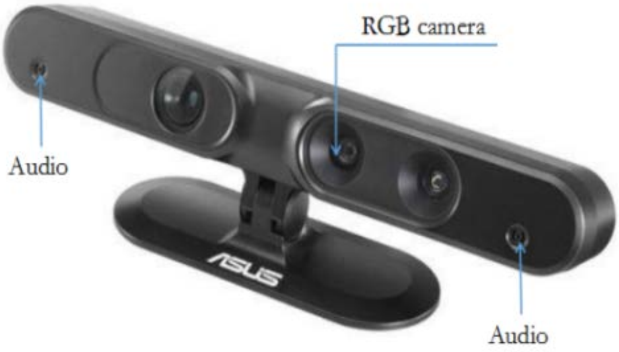

Fig.2 Xtion Pro Live

After that, Microsoft Corp produced Kinect V2 somatosensory equipment again; The TOF technique has chosen to be a technical scheme in Kinect V2. Then Intel produced Realsense Somatosensory Device. In 2017, ASUS produced Xtion V2 depth of field camera; they have been greatly improved in their quality. Because of its low price and simple principle, the RGB-D camera is widely welcomed in the field of visual SLAM.

2.2 RGB-D SLAM framework

The RGB-D SLAM framework and algorithm have been proposed in the past decades. These algorithms are provided in the Robot Library and the Visual Library (see Fig 3).

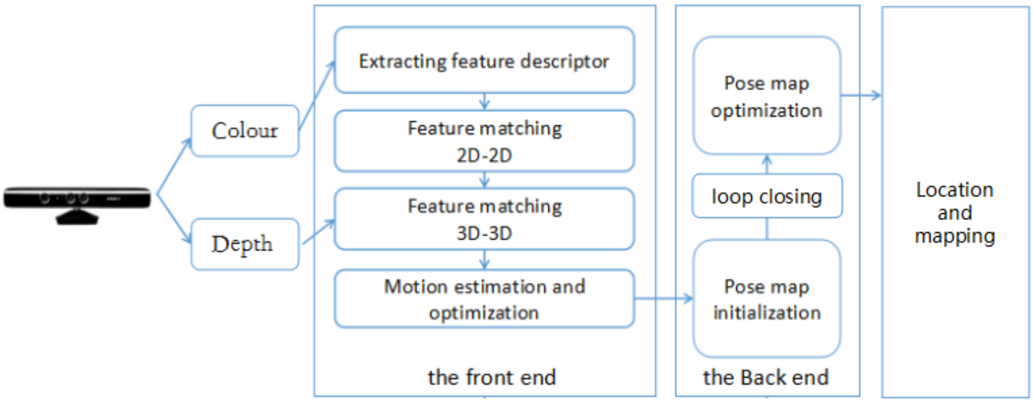

Fig.3 RGB-D SLAM flow chart

The RGB-D SLAM process includes the following steps:

Reading the information from the sensor. In the RGB-D SLAM, it aims to read and preprocess the RGB and depth information of the camera.

Visual Odometry (VO). The visual odometry is also called the front-end. The front-end receives the RGB image and the Depth image from the RGB-D camera to solve the problem of camera movement in adjacent images. In the vSLAM, the front-end is classified by the feature methods and the direct methods. Because the features are expected to be distinctive and invariant to viewpoint and illumination changes, as well as resilient to blur and noise [13]. Most of the RGB-D SLAM uses the feature method.

In the front-end algorithm, the RGB and Depth images of two adjacent frames are obtained firstly, then the RGB images are detected and the descriptors are extracted respectively. Matching the feature descriptors in two adjacent frames to obtain 2D feature matching points set. The coordinate information of two feature points in the corresponding RGB images and the depth information in the depth image are extracted respectively, and the coordinates of the 3D feature points in the space are calculated. When a coordinate 3D matching point is formed, it is added to the set of 3D coordinates matching points. The rotation and translation matrix between two adjacent frames can be calculated from the matched 3D points [12]. Then repeat the process until there is no new RGB image and Depth image input.

Optimization. Because it is connected to the front-end, it is usually referred to as the Back-End. The main task of the back-end algorithm is to estimate the state of the whole system from noisy data and give the Maximum a Posterior (MAP) of the state. Get the motion transformation between initialization pose graphs from the front-end. The constraint condition is continuously added through the loop closing. Then, utilizing the nonlinear error function to optimize the pose of the camera in the pose map, and ultimately get the global optimal pose and trajectory of the camera.

Loop Closing. Loop closing is an algorithm for detecting the similarity of observation data. Once the loop closing is successful, the information will be sent to the back-end to adjust the trajectory and map of the robot, and the cumulative error can be effectively reduced through loop closing.

Mapping. At this stage, building a map that can describe the environment according to the estimated trajectory, but the way of description is not fixed, it depends on the needs of SLAM tasks. At present, the map is classified by the Metric Map and the Topological Map.

3. RGB-D SLAM landmark achievements

[12] is the first method to reconstruct the indoor environment using the RGB-D camera. This algorithm extracts the SIFT features from color images and finds the depth information in depth images. Then the RANSAC method is used to match the 3D feature points, and then as the initial value of ICP (iterative closest point) to obtain more accurate pose. Richard A. Newcombe,Andrew J. Davison proposed KinectFusion [17] in 2011. It is the first RGB-D SLAM based on the GPU real-time construction of dense 3D map algorithm, KinectFusion uses the TDSF model for depth

data fusion and uses the ray projection algorithm to calculate the surface of the scene that can be seen in the current perspective. This algorithm used a voxel block hashing in the mapping process to reduce a computational cost. But Kinect fusion is affected by amount data and in [18], the data has been greatly reduced by unifying coplanar points [19]. RTAB-MAP [20] (Real-Time Appearance-Based Mapping RGB-D SLAM) provides a more complete solution. In particular, a loop closing based on Bag of words (Bow) is provided. RTAB-MAP not only supports Kinect, but also supports binocular sensors, but because of the upper integration level, it is difficult to develop two times on the basis of it. Raul Mur-Artal,Juan Domingo Tardos proposed ORB-SLAM [21] in 2015 is an algorithm for constructing sparse maps; they proposed a more complete ORB-SLAM2 algorithm [22] in 2016. ORB-SLAM2 algorithm supports three forms: monocular, binocular and RGB-D. It uses three threads to complete the algorithm: real-time tracking feature points thread, Local Mapping thread and Loop Closing thread. Because the ORB feature that does not consume time, it makes the whole algorithm could real-time operation on CPU. In addition, ORB-SLAM2 uses ORB dictionary file to ensure that loop back detection can effectively reduce accumulative error. Sala-Moreno et al. proposed SLAM++ algorithm [23] is an “object oriented” concept RGB-D SLAM. In this algorithm, the information of some 3D objects has been recorded in advance in the database. Because the 3D object has been re-identified, the point cloud map estimated in the algorithm is replaced by more precise object map, and the data volume of the algorithm will also be greatly reduced.Because the range of the Field of vision (FOV) and depth measurement of RGB-D camera is limited, there are still some problems in the registration of distant frames. Monocular SLAM systems can be extended to wide-angle cameras to increase the FOV range, but can not obtain depth information, so it is still unstable for lack of texture scene. Khalid Yousif, Yuichi Taguchi, Srikumar Ramalingam proposed MonoRGBD-SLAM in 2017 [24]. The SLAM system uses RGB-D cameras and wide-angle cameras to combine the advantages of both. The system extracts 3D point features from RGB-D cameras, extracts 2D point features from monocular images, and then uses these feature points to realize registration from RGB-D to RGB-D and from RGB-D to monocular images. Compared with a single RGBD SLAM, this algorithm is more robust. Ming H, Westman E, Zhang G, et al. proposed KDP-SLAM [25] uses only CPU, hand-held RGB-D sensors to reconstruct large indoor environments in real time, and it is the first dense planar SLAM to run on a CPU in real time (30 fps). Although the dense visual SLAM method can estimate the dense reconstruction of the environment, it lacks certain robustness in the tracking part, especially when the optimization process is not initialized properly. [26] proposed inertial RGBD-SLAM system combined with map deformation information, is more robust to fast motions and periods of low textureand and low geometric variation than the related RGB-D only SLAM system. [27] proposed combining events, images, and IMU for Robust Visual SLAM in HDR and High-Speed Scenarios in 2018. This algorithm combines the complementary advantages of the two sensors, and proposes a state estimation method which combines event, standard frame and inertial measurement tightly for the first time. And this hybrid method improves the state estimation accuracy by 130% compared with the pure event camera and 85% compared with the standard visual inertial system. In addition, Google launched the Project Tango in 2014 to move SLAM technology to mobile phones, the Project Tango uses motion tracking, area learning and depth perception technology for space exploration and self-location [28].

4. Key technology

4.1 Visual Odometry

The Visual Odometry of vSLAM is mainly categorized as being the feature methods and the direct methods [29]. The feature method is the mainstream solution for the current vSLAM because of its stable operation and insensitivity to light and dynamic objects.

4.1.1 Feature methods

The feature method is the mainstream of VO now: for the two images, firstly features of the image are extracted, and then the transform matrix to calculate the camera according to the matching characteristics of two images. The most widely use of this is point matching, such as SIFT [14], SURF [15], and ORB [16].

The SIFT (scale invariant feature transform) feature extraction algorithm proposed by David G. Lowe in 1999, the SIFT feature is discriminable and its descriptors are represented by 128-dimensional vectors. In addition, the SIFT feature is rotational invariant, scale invariant, radiative invariant, and robust to noise and illumination changes. The algorithm was improved in 2004 [14]. In [25], the SIFT algorithm is applied to feature extraction and descriptor computation for two adjacent RGB images. However, because the SIFT algorithm is more intricate, the vector dimension of the SIFT feature is too high so that it has high time complexity. Compared with the SIFT features, the time complexity of the SURF (speeded up robust features) is relatively low. Herbert Bay proposed the SURF algorithm in 2006 [15]. SURF features have scale rotation invariance, and compared with the SIFT feature, the speed of the using SURF algorithm is increased by 3~7 times [31-33]. In [34], used SURF algorithm to detect feature and extract descriptors in RGB images collected, which greatly reduced the time complexity and ultimately built 3D environment model in real time. Ethan Rublee proposed ORB (oriented FAST and rotated BRIEF) in 2011 [16], the calculation speed is accelerated further, the calculation speed of ORB is 100 times that of SIFT features, and is 10 times of the SURF feature, which combines the FAST [35] feature detection operator and BRIEF [36] descriptors and made some improvements on this basis. ORB has rotation invariance but does not have the scale invariance. ORB-SLAM2 [22], as an improved algorithm for ORB-SLAM, supports monocular, binocular, and RGB-D cameras.

4.1.2 Direct methods

The method that does not use the feature is called the direct method. The direct method directly writes all the pixels in the image into a pose estimation equation to get the relative motion between the frames. In this method, all the information within the image can be taken advantage to help the subsequent use of the map. And it has the good robustness to the environment with fewer characteristics. Nevertheless, the direct method needs more computation than the feature method, and it is also susceptible to failure when scene illumination changes as the minimization of the photometric error between two frames relies on the underlying assumption of the brightness consistency constraint. The camera positioning method proposed in [37] relies on each pixel point of the image and builds a dense 3D map. [17] get the depth image from Kinect, and get all the pixels in every frame to minimize the distance and get the pose of the camera, then merge the deep image and get the global map information at last. A robust RGB-D SLAM scheme is proposed by [38], which combines depth error and pixel strength error as an error function and minimizes the cost function to get the optimal pose (see Fig 4).

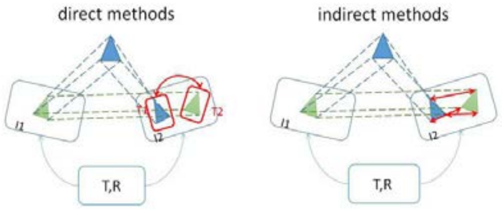

Fig.4 The direct methods utilize all information of the triangle to match to the image and the indirect methods utilize the features of the triangle to match to the features of the image

4.2 Back-end optimization

The Back-end optimization of vSLAM is mainly classified by the filter methods and the non-filter methods. In the early twenty-first Century, the filter methods have been the mainstream of vSLAM, and the current mainstream method is the non-filter methods.

4.2.1 Filter methods

The filtering method mainly uses Bayes's theorem to make a priori estimation of the position and posture of the robot, then the observation information is used to estimate the position and the map of the robot. The main use of the filtering method is the extended Calman filter (EKF) and the particle filter (PF). In [39-41], R. Smith etc. used EKF for the first time to solve SLAM problems. But EKF still has many shortcomings: for example, assuming the noise Gauss distribution and linearization error. For this reason, nonlinear optimization is proposed [38].

4.2.2 Non-filter methods

After twenty-first Century, SLAM researchers began to learn from the methods of SFM (Structure from Motion) problem and put the Bundle Adjustment (BA) into SLAM. The non-filter methods and the filter methods are different fundamentally. It is not an iterative process, but it takes into account the information in all the past frames. The error is divided into every observation by optimizing. The Bundle Adjustment in SLAM is often given in the form of graphs, so the researchers also call it the Graph Optimization.

4.3 Loop Closing

In the SLAM algorithm, the process of map reconstruction will produce the accumulation of errors, and it is prone to drifts in camera pose estimates. After an exploration phase, return to a right pose may not yield the same camera pose measurement, as it was at the start of the run. Those issues maybe lead the system to erroneous measurements or great failure in the end. The realization of closed loop is an effective way to eliminate accumulated errors. An important task of loop closing is how to correctly and effectively judge whether the camera has passed the same place.

Loop closing method mostly uses the Bag of Words (Bow) method, the Bow method refers to the technique of converting the content of an image into a digital vector by using the visual dictionary tree. First extracts feature from a large number of training images and classify these features. The leaf nodes are called dictionaries, and then an image can be described as a vector under the dictionary according to whether the corresponding words appear.

[43] released a K-means extended K forked tree to express a dictionary. Feature extraction is performed on the training image set, and the feature descriptor space is discretized into a cluster by K-means method. Thus, the first node layer of the dictionary tree is created. The following layer is obtained by repeatedly executing this operation for each cluster until a total layer is obtained. Finally get W leaf nodes. [45] uses a dictionary method of SIFT features to constantly search for places that have been visited. [46-47] uses a dictionary method based on SURF descriptors to detect SURF features in loop closing. [48] proposed a FAST based feature detection and BRIEF descriptor binary dictionary, and added the direct index so that the matching points between images can be effectively acquired. [21] uses a dictionary method based on ORB features to select the candidate closed loop first, and then the geometric verification of the closed loop through similarity calculation.

5. The future research hotspot and development trend of visual slam

5.1 Semantic SLAM

The current visual SLAM technology is still in the feature point or pixel level, resulting in the difference of characteristics may be too weak. In addition, the point cloud map built with current technology does not distinguish between different objects, it leads to the lack of information contained in the map and cannot continue to be used. So, the construction of semantic map has become a research hot spots in the future vision of the SLAM. The meaning of semantic SLAM is that the SLAM system not only gets the geometric structure information from the environment but also recognizes the independent information in the environment in order to get its location, posture and attributes and other semantic information, so as to cope with the more complex environment [49]. The general idea of semantic SLAM is to combine semantic map with SLAM to generate a 3D semantic model. In the past, similar work has been done by random forests and 3D reconstruction, and in recent years (2016), there has been a growing trend to combine deep learning with SLAM.

Semantic Fusion proposed by McCormac is a 3D dense semantic map construction algorithm based on Convolution Neural Network(CNN) [50]. It relies on Elastic Fusion SLAM algorithm to provide interframe pose estimation for indoor RGB-D video. And it uses CNN to predict pixel-level object class labels. Finally, it combines Bayesian upgrade strategy and the Conditional Random Field (CRF) model upgrades the probability of the CNN predictions from different perspectives, and finally generates a 3D dense semantic map containing semantic information. The CNN in Semantic Fusion chooses to add depth channels to the deconvolution semantics segmentation network structure in the framework of caffe, so that it can input four-channel RGB-D images and transmit them, then a dense pixel level semantic probability map is obtained. In [51-53], object recognition and visual SLAM are combined to build maps with object labels. [54] proposed use label information to bundle adjustment or optimize the objective functions to optimize the location and label information of feature points. [55] proposed a single and semi dense 3D semantic mapping construction method based on CNN and LSD-SLAM.

The advantages of semantic SLAM are as follows [56]:1. The classical SLAM methods generally assume that the environment is static. However, semantic SLAM can predict mobile attributes (human or robot). 2. In semantic SLAM, the knowledge of similar objects can be shared, and the extensibility of the SLAM system can be improved with maintaining a shared knowledge database. 3. Compared to the classic SLAM system, the path planning of the semantic SLAM is more intelligent because it can better plan the optimal path.

5.2 Multisensor fusion

The camera can capture the abundant details of the scene, and the inertial measurement unit (IMU) have a high frame rate and an accurate short time estimation can be obtained. The two sensors are complementary [57-58], which can be used together with better results, so using vision sensor and inertial navigation integration has become a hot research topic in the SLAM. At present, visual Inertial Odometry (VIO) is mainly divided into loosely coupled and tightly coupled [59], loosely coupled means after the camera and IMU self motion estimation of pose respectively then combine them. In contrast to the loose coupling, the tightly coupling is to combine the state of the camera and the IMU to perform the state estimation. One of the mainstream techniques of combining visual sensors with IMU to estimate pose is a method base on the filter. [60] proposed Real-time technology of monocular vision and IMU fusion based on EKF. [61] put fusion problem into two threads to process, and the inertial measurement and feature tracking between consecutive images on the first thread to handle in order to get a relatively high frequency of position estimation, the second thread contains a BA to reduce the effect of linear error estimation. The technique base on optimization is another mainstream technology for the combination of visual sensors and IMU. [62] makes the IMU error in the form of full probability to fuse the projection error of the road mark to form a joint nonlinear error function that will be optimized. Although the optimization method has become the mainstream in the classical visual SLAM, but in VIO, because of the high data frequency of IMU, the optimization of the state needs more computation. Therefore, it is still in the coexistence stage of filtering and optimization [63-64].

Besides, SLAM based on line feature or surface feature [65-67] and SLAM based on multi-robot [68-71] will gradually become the latest research focus of SLAM in the future.

6. Conclusion

During the course of this survey, we have outlined the essential building blocks of a generic RGB-D SLAM system; including Visual Odometry, Back-end Optimization and Loop Closing.We have also discussed the details of the latest open-source state of the art systems in RGB-D SLAM including KinectFusion, RTAB-MAP, ORB-SLAM, SLAM++ and MonoRGBD-SLAM,etc. Finally, we compiled and summarized what added information closed-source RGB-D SLAM systems have to offer.

Over the past decade, visual SLAM has gone through three great times: asking questions, searching for algorithms, and improving algorithms. We are now at third times[71], how to make the mature algorithm framework constantly perfected and improved, how to make the actual process of environmental applications in improving the robustness of the algorithm to deal with more complex environment, how to make the robot more intelligent object recognition and path planning, are all desired properties that unfortunately most states of the art systems lack and remain challenging topics in the field.

References

[1] Chatila R and Laumond J,Position referencing and consistent world modeling for mobile robots. IEEE International Conference on Robotics and Automation. Proceedings, IEEE. 138–145, (1985).

[2] Thrun S, Montemerlo M and Dahlkamp H, et al, Stanley. The Robot That Won the DARPA Grand Challenge. The 2005 DARPA Grand Challenge. 1–43,(2006).

[3] Thrun S, Hahnel D and Ferguson D, et al. A system for volumetric robotic mapping of abandoned mines. In: IEEE International Conference on Robotics and Automation.4270–4275 vol. (2003).

[4] Pinies P, Tardos J D and Neira J. Localization of avalanche victims using robocentric SLAM. In: Ieee/rsj International Conference on Intelligent Robots and Systems, IEEE. 3074–3079, (2006).

[5] Fuentes-Pacheco J, Ruiz-Ascencio J and Rendón-Mancha J M, “Visual simultaneous localization and mapping: a survey,” Artificial Intelligence Review 43, 55–81, (2015).

[6] Se S, Lowe D G and Little J J, “Vision-based global localization and mapping for mobile robots,” IEEE Transactions on Robotics 21, 364–375, (2005).

[7] Fuentes-Pacheco J, “Visual simultaneous localization and mapping: a survey,” Kluwer Academic Publishers (2015).

[8] Bailey T and Durrant-Whyte H, “Simultaneous localization and mapping (SLAM): Part II,” IEEE Robotics&Automation Magazine 13, 108–117, (2006).

[9] Aulinas J, Petillot Y R, and Salvi J, et al, “The SLAM problem: a survey,” CCIA184 (1), 363–371, (2008).

[10] Coulter A. “Big four slam Which? Survey,” Travel Trade Gazette Uk & Ireland (2003).

[11] Jieqiong Ding. “A Study on SLAM Algorithm Based on RGB-D.” XIDIAN UNIVERSITY (2014).

[12] Henry P, Krainin M and Herbst E, et al, “RGB-D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments Experimental Robotics,” Springer Berlin Heidelberg, 647–663, (2014).

[13] Younes G, Asmar D and Shammas E, “A survey on non-filter-based monocular Visual SLAM systems.” (2016).

[14] Lowe D G, “Distinctive Image Features from Scale-Invariant Keypoints,” International Journal of Computer Vision 60, 91–110, (2004).

[15] Bay H, Ess A and Tuytelaars T, et al,” Speeded-Up Robust Features,” Computer Vision & Image Understanding. 110, 404–417, (2008).

[16] Rublee E, Rabaud V and Konolige K, et al, “ORB: An efficient alternative to SIFT or SURF. ” IEEE International Conference on Computer Vision, IEEE. 2564–2571, (2012).

[17] Newcombe R A, Izadi S and Hilliges O, et al, “ KinectFusion: Real-time dense surface mapping and tracking,” IEEE International Symposium on Mixed and Augmented Reality. IEEE Computer Society, 127–136, (2011).

[18] Salas-Moreno R F, Glocken B, and Kelly P H J, et al, “Dense planar SLAM,” IEEE

International Symposium on Mixed and Augmented Reality, IEEE. 157–164, (2014).

[19] Taketomi T, Uchiyama H and Ikeda S. “Visual SLAM algorithms: a survey from 2010 to 2016,” Ipsj Transactions on Computer Vision & Applications. 9, 16 (2017).

[20] Labbé M, and Michaud F. “Online global loop closure detection for large-scale multi-session graph-based SLAM,” Ieee/rsj International Conference on Intelligent Robots and Systems, IEEE. 2661–2666, (2014).

[21] Mur-Artal R, Montiel J M M, and Tardós J D, “ ORB-SLAM: A Versatile and Accurate Monocular SLAM System.” IEEE Transactions on Robotics. 31, 1147–1163, (2015).

[22] Mur-Artal R and Tardós J D, “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras,” IEEE Transactions on Robotics. 33, 1255–1262, (2016).

[23] Salas-Moreno R F, Newcombe R A and Strasdat H, et al. “SLAM++: Simultaneous Localisation and Mapping at the Level of Objects,” Computer Vision and Pattern Recognition, IEEE.1352–1359, (2013).

[24] Yousif K, Taguchi Y, Ramalingam S. “MonoRGBD-SLAM: Simultaneous localization and mapping using both monocular and RGBD cameras,” IEEE International Conference on Robotics and Automation, IEEE. 4495-4502, (2017).

[25] Ming H, Westman E, Zhang G, et al, “Keyframe-based dense planar SLAM,” IEEE International Conference on Robotics and Automation, IEEE. 5110-5117, (2017).

[26] Laidlow T, Bloesch M, Li W, et al, “Dense RGB-D-inertial SLAM with map deformations,” Ieee/rsj International Conference on Intelligent Robots and Systems, IEEE. 6741-6748, (2017).

[27] Vidal A R, Rebecq H, Horstschaefer T, et al, “Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High-Speed Scenarios,” IEEE Robotics & Automation Letters, 994-1001, (2018).

[28] McCue, “3D Scan Your World - Google Project Tango.” Forbes. N.p., Forbes. Web,(2014).

[29] Jan Stühmer, Gumhold S and Cremers D, “Real-Time Dense Geometry from a Handheld Camera,” Dagm Conference on Pattern Recognition, Springer-Verlag. 11–20, (2010).

[30] Henry P, Krainin M and Herbst E, et al, “RGB-D Mapping: Using Kinect-Style Depth Cameras for Dense 3D Modeling of Indoor Environments,” International Journal of Robotics Research. 31, 647–663, (2012).

[31] Zhang Z, Huang Y and Li C, et al, “Monocular vision simultaneous localization and mapping using SURF,” Intelligent Control and Automation. Wcica 2008. World Congress on, IEEE. 1651–1656, (2008).

[32] Ye Y. “The Research of SLAM Monocular Vision Based on The Improved SURF Feather,” International Conference on Computational Intelligence and Communication Networks, IEEE. 344–348, (2015).

[33] Wang Y T and Feng Y C, “Data association and map management for robot SLAM using local invariant features,” IEEE International Conference on Mechatronics and Automation, IEEE.1102–1107, (2013).

[34] Engelhard N, Endres F and Hess J, et al, “Real-time 3D visual SLAM with a hand-held RGB-D camera.” The RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum (2011).

[35] Rosten E, and Drummond T, “Machine Learning for High-Speed Corner Detection,” European Conference on Computer Vision. 430–443, (2006).

[36] Calonder M, Lepetit V and Strecha C, et al, “BRIEF: Binary Robust Independent Elementary

Features,” Computer Vision-Eccv 2010, Pt Iv, 6314.778–792, (2010).

[37] Jan Stühmer, Gumhold S and Cremers D, “Real-Time Dense Geometry from a Handheld Camera,” Dagm Conference on Pattern Recognition, Springer-Verlag. 11–20, (2010).

[38] Kerl C, Sturm J and Cremers D, “Dense visual SLAM for RGB-D cameras,” Ieee/rsj International Conference on Intelligent Robots and Systems, IEEE. 2100–2106, (2014).

[39] Davison A J, Reid ID, and Molton ND, et al. “MonoSLAM: real-time single camera SLAM,” IEEE transactions on pattern analysis and machine intelligence. 29, 1052, (2007).

[40] Davison A J, “Real-Time Simultaneous Localisation and Mapping with a Single Camera.” IEEE International Conference on Computer Vision, IEEE Computer Society. 1403,(2003).

[41] Davison, A J, “SLAM with a Single Camera,” SLAMCML Workshop at ICRA. 2002,(2002).

[42] Hauke Strasdat, J.M.M. Montiel, and Andrew J. Davison, “Visual SLAM: Why filter? ☆,” Image & Vision Computing. 30, 65–77, (2012).

[43] Galvez-López D and Tardos J D, “Bags of Binary Words for Fast Place Recognition in Image Sequences,” IEEE Transactions on Robotics. 28, 1188–1197, (2012).

[44] Arthur D and Vassilvitskii S, “k-means++: the advantages of careful seeding,” Eighteenth Acm-Siam Symposium on Discrete Algorithms. Society for Industrial and Applied Mathematics, 1027-1035, (2007).

[45] Eade E and Drummond T, “Unified Loop Closing and Recovery for Real Time Monocular SLAM,” British Machine Vision Conference, 1–10, (2008).

[46] Gálvez-López D and Tardós J D, “Real-time loop detection with bags of binary words,” Ieee/rsj International Conference on Intelligent Robots and Systems, IEEE. 51–58, (2011).

[47] Cummins M and Newman P M, “Appearance-only SLAM at large scale with FAB-MAP 2.0,” International Journal of Robotics Research. 30, 1100–1123, (2011).

[48] Galvez-López D and Tardos J D. Bags of Binary Words for Fast Place Recognition in Image Sequences. IEEE Transactions on Robotics. 28, 1188–1197 (2012).

[49] ZHAO Yang, LIU Guoliang, and TIAN Guohui, et al, “A Survey of Visual SLAM Based on Deep Learning,” ROBOT Vol.39, 889–896, (2017).

[50] Mccormac J, Handa A, Davison A, et al, “SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks,” 4628-4635, (2016).

[51] Nüchter A and Hertzberg J, “Towards semantic maps for mobile robots. Robotics & Autonomous Systems” 56, 915–926, (2008).

[52] Civera J, Gálvez-López D and Riazuelo L, et al, “Towards semantic SLAM using a monocular camera,” Ieee/rsj International Conference on Intelligent Robots and Systems, IEEE. 1277–1284, (2011).

[53] Koppula H S, Anand A and Joachims T, et al, “Semantic labeling of 3D point clouds for indoor scenes,” International Conference on Neural Information Processing Systems. Curran Associates Inc. 244–252, (2011).

[54] Fioraio N and Stefano L D. Joint Detection, “Tracking and Mapping by Semantic Bundle Adjustment,” Computer Vision and Pattern Recognition, IEEE. 1538–1545, (2013).

[55] Li X and Belaroussi R,“Semi-Dense 3D Semantic Mapping from Monocular SLAM” (2016).

[56] Salas-Moreno and Renato F, “Dense semantic SLAM,” London, UK. Imperial College (2014).

[57] QUAN Meixiang and PIAO Songhao, LI Guo, “ An overview of visual SLAM,” CAAI Transactions on Intelligent Systems 11, 768–776, (2016).

[58] Gui J, Gu D and Wang S, et al, “A review of visual inertial odometry from filtering and optimization perspectives,” Advanced Robotics 29, 1289–1301, (2015).

[59] Agostino Martinelli, “Closed-Form Solution of Visual-Inertial Structure from Motion,” International Journal of Computer Vision 106, 138–152, (2014).

[60] Mourikis A I and Roumeliotis S I, “A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation,” IEEE International Conference on Robotics and Automation, IEEE. 3565–3572, (2007).

[61] Mourikis A I and Roumeliotis S I, “A dual-layer estimator architecture for long-term localization.” Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on, IEEE. 1–8, (2008).

[62] Leutenegger S, Furgale P and Rabaud V, et al, “Keyframe-Based Visual-Inertial SLAM using Nonlinear Optimization,” Robotics: Science and Systems. 789–795, (2013).

[63] Usenko V, Engel J and Stückler J, et al, “Direct visual-inertial odometry with stereo cameras” IEEE International Conference on Robotics and Automation, IEEE. 1885–1892, (2016).

[64] Tkocz M and Janschek K, “Towards Consistent State and Covariance Initialization for Monocular SLAM Filters,” Journal of Intelligent & Robotic Systems. 80, 1–15, (2015).

[65] Saarinen J P, Andreasson H and Stoyanov T, et al, “3D Normal Distributions Transform Occupancy Maps: An Efficient Representation for Mapping in Dynamic Environments,” International Journal of Robotics Research. 32, 1627–1644, (2013).

[66] Maddern W, Milford M and Wyeth G, “CAT-SLAM: probabilistic localisation and mapping using a continuous appearance-based trajectory,” International Journal of Robotics Research. 31, 429–451, (2012).

[67] Wang H M 1, “Online mapping with a mobile robot in dynamic and unknown environments.” International Journal of Modelling Identification & Control. 4, 415–423, (2008).

[68] Su-YongAn, Jeong-GwanKang and Lae-KyoungLee, et al, “Line Segment–Based Indoor Mapping with Salient Line Feature Extraction,” Advanced Robotics. 26, 437–460, (2012).

[69] Zhou H, Zou D and Pei L, et al, “StructSLAM: Visual SLAM with Building Structure Lines,” IEEE Transactions on Vehicular Technology. 64, 1364–1375, (2015).

[70] Benedettelli D, Garulli A and Giannitrapani A, “Cooperative SLAM using M-Space representation of linear features,” Robotics & Autonomous Systems. 60, 1267–1278, (2012).

[71] Cadena C, Carlone L and Carrillo H, et al, “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age,” IEEE Transactions on Robotics. 32, 1309–1332, (2017).