Transactions on Computational and Scientific Methods | Vo2. 1, No. 1, 2022
ISSN: 2998-8780
https://pspress.org/index.php/tcsm
Pinnacle Science Press

Advancements and Challenges in Visual SLAM: Process, Types, and Future Directions

Elijah Anderson[1], **Mia Taylor**[2]

Florida Atlantic University, Boca Raton, USA[1], Florida Atlantic University, Boca Raton, USA[2]

Elijah.a8763@gmail.com[1], miamia9@gmail.com[2]

Abstract: Simultaneous Localization and Mapping (SLAM) has been a prominent research area in computer vision and mobile robotics for over two decades. Visual SLAM, which utilizes a camera as the sole external sensor, aims to create a map of an unknown environment while simultaneously determining the position of the sensors within that map. This paper provides an overview and synthesis of the visual SLAM process, key research findings in the field, and the available public datasets for visual SLAM. The paper also explores future directions in visual SLAM, including SLAM in dynamic environments, SLAM with multi-feature fusion, SLAM with multi-sensor integration, SLAM with multi-robot collaboration, and SLAM incorporating deep learning techniques.

Keywords: Visual SLAM process, dynamic scene, multi-sensor fusion, multi-feature fusion, multi- robot cooperation, deep learning.

1. Introduction

Autonomous navigation is a key to the intelligentization of mobile robots. To achieve autonomous navigation, mobile robots will face three key issues: Where are I, where do I go, and how do I go? “Where am I?” The mobile robot needs to solve the problem of positioning itself. “Where do I go?” and “How to go?” is the path planning problem that mobile robots needto solve. The positioning problem is the basis of the path planning problem, and the positioning requires the robot to "familiar" with the surrounding environment. So the robot's first task is to perceive the surrounding environment and describe it in some way, and the result of the description is even the so-called map. Simultaneous localization and mapping (SLAM) is a hot research topic in the field of computer vision and robotics for nearly two decades. As a technology, its purpose is to construct an environmental map in an unknown environment andlocate sensors in the map. In SLAM, the sensors used are diverse, including laser radar, sonar, and cameras. While in different sensor modes, although the camera is relatively inexpensive, itprovides rich environmental information and allows for robust and accurate position recognition. Therefore, the SLAM research with the camera as the main sensor has become thefocus of attention, which is the visual SLAM.

Depending on the visual sensor, visual SLAM can be divided into three categories. The first category is a monocular visual SLAM with a monocular camera as the only external sensor. The sensor settings used by monocular visual SLAM are the cheapest and smallest. However, since the depth of the pixel cannot be obtained from only one monocular camera, the map constructed by the monocular visual SLAM cannot know its scale, and its estimated trajectory is unknown. In addition, multi-view or filtering techniques are required to generate the initial map during the initial phase of SLAM system startup because it cannot be triangulated from thefirst frame. Last but not least, there is a scale drift in the monocular SLAM, and there must be no rotation only during initialization, and there must be a certain degree of translation. That is to say, if there is no translation, the monocular will not be initialized, which will cause the SLAM to fail. The second type is a stereo visual SLAM that uses multiple monocular cameras as external sensors, where only two monocular cameras are used, which is called stereo visual SLAM. Stereo visual SLAM uses the principle of outer-line geometry constraint to match the characteristics of the left and right cameras, so that the complete feature data can be directly extracted under the current frame rate, so it is widely used, which directly solves the monocular visual SLAM. Map initialization issues in the system. However, due to the complexity of the system design, the system cost is relatively high, and its viewing angle range is limited, and it isimpossible to obtain a distant scene, so that reliable measurement can be performed only within a certain scale range, thereby lacking flexibility. The third type is an RGB-D visual SLAM based on a depth sensor based on a monocular camera and an infrared sensor. The depth camera obtains the corresponding depth image while obtaining the color image, so that the depth information of the pixel can be conveniently obtained. However, since depth cameras use infrared light for depth measurement, they are easily interfered with by infrared light emitted by sunlight or other sensors, so they cannot be used outdoors, and they interfere with each other when used at the same time. For objects with a transmissive material, the position of these points cannot be measured because the reflected light is not received. In addition, the RGB-D camera has some disadvantages in terms of cost and power consumption.

SLAM is essentially a state estimation problem. The current SLAM problem solving methods can be divided into a filter-based method and a graph-based optimization method. The filter- based method mainly uses the recursive Bayesian estimation principle to determine the system state (including the current pose of the robot and all map feature positions) under the assumption that the observation information from 0 to t and the control information are known.The posterior probability is estimated. There are various filter-based methods depending on the way the posterior probability is expressed. Commonly used include Extended Kalman Filter (EKF) method[1], Extended Information Filter (EIF) method[2], and Particle Filter (PF) method[3]. To emphasize its incremental nature, the filter-based SLAM method is also commonly referred to as online SLAM (on-line SLAM). It is worth noting that the filter-based method has problems such as linearization and update efficiency, which makes it difficult to apply to map creation in large-scale environments[4,5]. Different from the filtering method, only the current pose of the robot is considered. The graph optimization method estimates the complete motion trajectory and map of the robot through all the observation information, so it is also called the full SLAM method (full SLAM). Since the map features can be transformed into pose constraints by the marginalization method, it is simplified to the estimation of the pose sequence. Such methods can be visually described in the form of a graph, and the resulting graph is called a pose graph. The nodes in the figure correspond to the position and attitude of the robot at different times, while the edges describe the spatial constraint between the pose and the pose. This constraint can be obtained by registration of an odometry or observation information. After the graph is constructed, the position of the node in the graph (in the pose space) is optimized to best satisfy the constraint relationship represented by the edge, and the optimized result corresponds to the motion trajectory of the robot.

The organizational structure of this paper is as follows: In chapter 2, the process of visual SLAM, the main research results of visual SLAM and the public data set of visual SLAM are introduced and summarized. In chapter 3, the development trend of visual SLAM is discussed according to five research directions of visual SLAM, such as SLAM in dynamic environment, SLAM in multi-feature fusion, SLAM in multi-sensor fusion, SLAM in multi-robot collaboration, and SLAM in deep learning. Finally, a summary of the full text is summarized.

2. Visual SLAM Process

The classic framework of visual SLAM consists of four parts: visual odometry, back-end optimization, loop detection and mapping. The following four parts are introduced separately.

2.1. Visual Odometry

Visual Odometry (VO), also known as the visual front end, is simply referred to as the front end, which estimates the camera motion based on adjacent image frames, and provides a good initial value to the back end. Depending on whether image features are used in the implementation of a visual odometry, the front end can be divided into a feature-based front end and a direct method based front end. Feature-based front end is considered to be the mainstream method of visual odometry. It is not sensitive to dynamic objects and illumination, and has stable operation, which is a better solution at present.However, although the feature method dominates the visual odometry, researchers have realized that it has at least the following shortcomings: (1) the calculation of feature extraction and description is very time consuming;

(2) the use of features, ignoring the elimination All information except features; (3) Cameras sometimes move to places where features are missing, and often there is no obvious texture information. The direct method exists to overcome these shortcomings of the feature law. The direct method estimates the motion of the camera based on the luminance information of the pixel, and can completely ignore the feature extraction and description, thus avoiding the calculation time of the feature and avoiding the feature missing. As long as there are light and dark changes in the scene (which can be gradients without local image gradients), the direct method works. Of course, the direct method also has its shortcomings (1) non-convexity. The camera pose in the direct method is calculated by searching the gradient to reduce the objective function. Since the image function is strongly non-convex, the objective function needs to take the gray value of the pixel, which makes the optimization algorithm easy to enter very small, and thus the direct method can be successful only when the motion of the camera is small. (2) There is no distinction between individual pixels. It's too much like it, so we either calculate the image block or calculate the complex correlation. Since each pixel is inconsistent with the "opinion" of changing camera motion, only a few can obey the majority, replacing the quality by quantity. (3) A strong assumption when the gray value is constant. If the camera is auto- exposure, it will make the image as bright or dark as it adjusts the exposure parameters. This can also happen when the light changes. The feature point method has a certain tolerance to illumination, while the direct method calculates the gray level difference, and the overall gray level change will destroy the gray level invariant hypothesis, which makes the algorithm fail. In response to this, the current direct method begins to calibrate the camera with a more detailed photometric model so that the direct method can be operated when the exposure time changes.A feature-based visual odometry is described below.

Feature extraction and matching of the image data input by the sensor, and then estimating the camera motion based on the matched feature pairs, which is the workflow of the feature-based visual odometry. Image features can be generally divided into point features, line features, and edges, contour features, and surface features. Since features such as lines, edges, contours, and faces are processed in high-dimensional space, the amount of calculation is large, and the point features are occluded. Relatively robust, and the extraction speed is fast and the recognition is good. Therefore, in the feature-based SLAM method, the image features used are mostly point features. Historically, researchers have proposed many image features. Among them, the most famous ones are SIFT[6], SURF[7], ORB[8] and other feature points.

SIFT algorithm is a method to detect the significant features in the image. It can determine the position of the points with significant features in the image, and also give a floating point feature descriptor of the point, which contains the position, scale and direction. SIFT first obtains a multi-scale representation of the image, and then searches for significant feature points in the multi-scale representation of the image. Difference of gaussian (DoG) operator is utilized for this purpose. In this way, the location and scale of the salient feature points are determined. Then, the gradient parameters of the pixels in the neighborhood of the significant feature pointsare used to determine the direction parameters of each point. After the direction of each pointis obtained, the directions of the pixels in the neighborhood can be combined to obtain the direction of the significant feature points. The SIFT descriptor has invariance to the scale, rotation and illumination changes of the image, and also has certain stability to the affine transformation, the change of the angle of view, the truth of the local form, and the noise interference. But it also has its shortcomings, that is, its computational load is a lot, which leads to the operation speed is not fast enough, and the real-time demand of SLAM can be satisfied only when GPU acceleration is needed.The SURF algorithm is an improvement of the SIFT algorithm, using the DoH-based speckle feature detection method. In the description of the feature points, the SURF algorithm uses the Harr wavelet template in two directions to calculate the gradient through the integral map, and then the neighborhood points in the neighborhood. The gradient direction is counted in a fan shape to obtain the main direction of the feature points. The SURF algorithm is fast, stable, and widely used. The ORB algorithm proposed by EthanRublee in 2011 uses the improved FAST feature point detection algorithm[9], and the ORB feature descriptor uses the improved binary string feature descriptor BRIEF[10]. Thanks to the extremely fast binary descriptor, the ORB greatly accelerates the extraction of the entire image feature.

After feature extraction and matching, if enough paired feature points can be obtained, the visual odometry can solve the camera motion between adjacent frames according to whether the feature point pair is 2D or 3D. If the feature points are both 2D, that is, the matching relationship is 2D to 2D, the camera motion can be solved by the pole constraint. If the 3D position of one of the feature points of the feature Point pair can be determined by triangulation or the depth map of the RGB-D camera, that is, the matching relationship is 3D to 2D, then the camera motion can be estimated using PnP (Perspective-n-Point). The PnP method is a very important attitude estimation method without using the polar constraint and obtaining better motion estimation in few matching points. There are many methods for solving PnP problems, such as direct linear transformation (DLT), P3P[11], EPnP (Efficient PnP)[12], UPnP[13], etc., which estimate poses with 3 pairs of points. In addition, the least squares problem can be constructed and solved iteratively in a nonlinear optimization manner, namely Bundle Adjustment[14]. If the feature point pairs are all 3D, that is, the matching relationship is 3D to 3D, then it can be solved using the Iterative Closest Point (ICP). Similar to PnP, the ICP solution is also divided into two ways: the solution using linear algebra (mainly SVD), and the solution using nonlinear optimization (similar to Bundle Adjustment).

2.2. Back-end Optimization

Since the sensors are all noisy, the camera motion calculated from the "less accurate" image data is also noisy, and the problem of back-end optimization is to estimate the entire system based on the image data with noise. The state (the trajectory of the robot itself, and the map), and the uncertainty of this state estimate - the maximum a posteriori probability estimate (Maximum-a-Posteriori, MAP). In visual SLAM, visual odometry is more related to computer vision, such as image feature extraction and matching. Back-end optimization is mainly filtering and nonlinear optimization algorithm. The filtering-based method only considers the pose of the robot at the current moment, and the graph-based optimization method estimates the entire pose sequence, which is the main difference between the filtering method and the graphoptimization method.

The SLAM back-end optimization method based on graph optimization is mainly divided into: nonlinear least squares method + sparse structure, relaxation technique, stochastic gradient descent method, and popular optimization.

2.2.1 Nonlinear Least Squares Method+Sparse Structure

Lu and Milios[15] For the first time, convert the SLAM problem into a least squares problem. Dellaert and Kaess [16] proposed a method for decomposing information matrix into square root (SAM), and discussed the SAM method from three aspects: batch, linear incremental and nonlinear incremental. The proposed method has the advantages of high precision and short processing time. In 2008, Kaess [17] proposed the iSAM method. From the incremental point of view, this method solves the problem of node ordering in the closed-loop detection plagued SAM method by combining linearization and heuristic node sorting. It makes full use of the sparsity of the matrix and obtains a more accurate map than the SAM method.In 2012, Kaess etal. [18] combined new data processing methods, improved iSAM, and proposed iSAM2. In the data processing, the Bayesian tree processing method is introduced to improve the optimization efficiency. In 2011, Giorgio Grisetti et al. [19] proposed a method to avoid rotatingsingular values, which combines least squares and manifold optimization, and describes the back-end optimization process in detail, and the proposed method improves. The accuracy of the results. In 2012, David M. Rosen et al. [20] proposed a robust incremental least squares estimation (RISE) method based on the Dog-leg method and the iSAM framework, which improves stability while ensuring certain efficiency. In 2016, Jingshan Zhang et al. [21] improved the Gauss-Newton method and proposed a nonlinear optimization algorithm based on phase retrieval and source recovery.

2.2.2 Relaxation Technique

In 2000, Duckett et al. [22] proposed a relaxation technique. The idea of relaxation technologyis to move the node to the place where its "neighbor" thinks it is, that is, to recalculate and update the location information of the current node according to the positional relationship between the current node and its neighboring nodes and related constraints, and iterate through all the nodes in each iteration. With relaxation technology optimization, three sources of information are needed: an external location identification system, a global direction from the sensor, and a local ranging of the odometry. When the node of the fully connected graph is n, the algorithm has a complexity of in the worst case. However, as the size of the map increases, the number of nodes in the map does not increase, so the complexity is O(n) or linear. On the basis of Frese [23] and others, in 2005, a multi-stage relaxation technique was proposed, and the multi-grid method was used to solve differential equations, which improved the optimization efficiency of nodes when loop closure occurred. In 2014, Carlone et al. [24] proposed a measurement-based relaxation technique. The article describes a method of aberrant exclusion in planar pose graph optimization based on linear technology, which allows for a fast and global solution and improves the robustness of backend optimization.

2.2.3 Stochastic Gradient Descent

Olson [25] was the first to apply the stochastic gradient descent method to the SLAM backend optimization. The stochastic gradient descent seeks the optimal value by selecting a constraintand by moving a set of nodes. Olson also improved the SGD at the same time, and even if the initial value is poor, it can quickly converge to the optimal value to avoid falling into local optimum. Grisetti et al. [26] introduced the parameterization method of trees into SLAM back- end optimization, improved node parameterization, optimized by SGD method, improved optimization efficiency, and expanded the application range of optimization algorithm. Later, Grisetti et al. [27] based on previous work, improved the optimization method, and distributedthe rotation error to each node, which provided an efficient solution for the robot to learn the three-dimensional maximum likelihood map in a non-flat environment. Gao et al. [38] proposed an MSGD for magnetic sequence SLAM, improved SGD, and maintained high efficiency and scalability when processing larger data.

2.2.4 Manifold Optimization

Manifold optimization refers to optimization in a manifold space, not in a traditional Europeanspace. In the European space, the rotational component of the robot pose may be singular during optimization, resulting in divergence of the optimization results. There are two ways to avoid singular values. The first is to use a quaternion or matrix to represent the rotational component of the robot pose. But this approach creates unnecessary errors. The secondapproach is optimized in the manifold space proposed by Grisetti [28]. Moreover, Grisetti layered the poses and optimized them separately. Researchers have developed a generic open source tool for graph optimization based on manifold optimization (g2o). Based on this, Kümmerle et al. [29] proposed the g2o framework to improve development efficiency.

2.3. Loop Detection

Loop detection is of great significance to the SLAM system, which is related to the accuracy of the estimated trajectories and maps of the SLAM system over time. In the process of feature matching, pose estimation, etc. in the visual odometry, there is inevitably an error. As the camera moves for a long time, the cumulative error will become larger and larger, and the introduction of loop closing detection can be very good. Eliminate accumulated errors and ensure the consistency of trajectories and maps. In addition, since the loop closing detection can provide the correlation between the current data and all the previous data, after the tracking algorithm in the visual odometryr fails, the current pose cannot be located, and the current pose can be relocated by the loop closing detection.

The purpose of loop closing detection is to determine whether the robot has arrived at a place that has been visited. Commonly used judgment methods are: based on geometric distance and image-based appearance. The method based on geometric distance refers to whether the current position of the robot is in the vicinity of the previous position by the pose estimation. The method cannot detect the loop when the error is accumulated. The method based on the appearance of an image refers to judging whether it is a loop closing by calculating the similarityof two images. In visual SLAM, the similarity of two images can be calculated by the word bag model, which is a tree structure composed of words, each of which can generate a corresponding word vector according to the word bag model. The similarity between images iscalculated by comparing word vectors.

At present, most of the word bag models are realized by point features. In 2008, Newman et al. [30] proposed a word bag model based on SIFT features and Chou-Liu tree. In 2012, Juan et al. [31] proposed a word bag model based on the binary descriptor of the BRIEF descriptor, whichhas the advantages of fast speed, low storage and so on. In 2013, Lee et al. [32] proposed a wordbag model based on MSLD line feature descriptors, which achieved good results in experiments.In 2015, Yang et al. [33] proposed a word bag model based on the BRIEF point featuredescriptor and the LBD line feature descriptor based on the research of Lee et al., which has stronger robustness in loop detection.

2.4. Mapping

The map is an abstract description of the environment around the robot. This description will vary depending on the application direction of the SLAM, ie it is not fixed. For a camera, it has six degrees of freedom of motion, and we need at least a three-dimensional map. Sometimes, we want a beautiful reconstruction result, not only a set of spatial points, but also a textured triangular patch. At other times, we don't care about the way the map looks. We just need to know things like "A to B can pass, and B to C can't." Even sometimes, we don't need a map, or the map can be provided by others, for example, a moving vehicle can often get a local map that has already been drawn. For maps, we have too many ideas and needs. Therefore, compared to the aforementioned visual odometry, loop closing and backend optimization, there is no fixed form and algorithm for mapping. A collection of spatial points can also be called maps. A beautiful 3D model is also a map. A picture that marks cities, villages, railways, and rivers is also a map. The form of the map depends on the application of the SLAM. In general, they can be divided into metric maps and topological maps.

2.4.1 Metric Map

Metric maps emphasize the precise representation of the positional relationship of objects in the map. Usually we classify them with Sparse and Dense. Sparse maps are abstracted to a certain degree and do not need to express all objects. For example, if we choose a part of the representative meaning, called Landmark, then a sparse map is a map made up of road signs, not part of the road signs can be ignored. In contrast, dense maps focus on modeling everythingyou see. For positioning, a sparse roadmap map is sufficient. When used for navigation, we oftenneed dense maps (otherwise what happens if we hit the wall between two road signs?). Densemaps are usually composed of many small blocks at a certain resolution. A two-dimensional metric map is a number of small grids (Grid), and three dimensions are many small squares (Voxel). Generally, a small block contains three states of occupancy, idle, and unknown to express whether there is an object in the cell. When we query a spatial location, the map can give information about whether the location can pass. Many navigation algorithms such as A* and D* can use such maps.However, in many application scenarios, many details of such a mapare not needed. Therefore, storing the status information of each lattice point leads to a large amount of wasted storage space. On the other hand, large-scale metric maps sometimes have consistency problems. A small steering error may cause the walls of the two rooms to overlap,causing the map to fail.

2.4.2 Topological Map

Topological maps emphasize the relationship between map elements compared to the accuracyof metric maps. A topology map is a graph consisting of nodes and edges. Only the connectivity between nodes is considered. For example, A and B are connected, regardless of how to reach point B from point A. Topological maps relax the need for accurate location and remove the details of the map, which makes topological maps generally not used to describe maps with relatively complex environment structure. How to segment the map to form nodes and edges, and how to use topology maps for navigation and path planning is still a problem to be studied.

2.5. Visual SLAM Dataset

The TUM dataset is a public RGB-D dataset from the Technical University of Munich (TUM) that contains many RGB-D videos that can be used as experimental data for RGB-D or monocular SLAM. It also provides an accurate trajectory measured with a motion capture system that can be used as a standard trajectory to calibrate the SLAM system.

The KITTI data set is a collaboration between KIT and TTIC and is currently the largest computer vision algorithm evaluation data set in the world for autonomous driving scenarios.KITTI collected data in a variety of traffic environments, and measured the precise trajectory of the data by using the data collection vehicle loaded with color cameras, black and white cameras, lidar and GPS/IMU positioning system and other sensors as ground truth.KITTI dataset can be used to test a variety of tasks in vehicular environment, such as visual odometry, object detection, tracking, road and lane detection, stereo, optical flow.

The EuRoC data set is the visual inertia data set collected on the micro air vehicle (MAV).The first data sets were collected in an industrial environment, providing the ground truth of the 6D pose collected by the laser tracking system and used primarily to evaluate visual inertial positioning algorithms.The second batch of data was collected in a room equipped with a motion capture system, which provides an accurate description of the 3d environment and is mainly used for 3d environment reconstruction.To enable researchers to thoroughly evaluate their algorithm, EuRoC provides image data ranging from good vision to motion blur to dark light.

2.6. The Main Research Results of Visual SLAM

MonoSLAM [34], proposed by Professor Davison of Imperial College London in 2007, is the first truly visual SLAM system based on a monocular camera and is real-time. The Kalman filter (KF)-based SLAM algorithm is a typical representative of the probabilistic framework method. Before the nonlinear optimization algorithm is applied to the theoretical maturity of SLAM, the extended Kalman filter (EKF)-based optimization algorithm is mainly used in SLAM. . MonoSLAM is a representative of an EKF-based visual SLAM system.

In the same year, Klein, a researcher in the field of augmented reality, proposed PTAM (Parallel Tracking and Mapping) [35], which also played a milestone in the development of visual SLAM.The significance of PTAM is mainly reflected in two points: (1) PTAM proposes and implements the parallelization of tracking and mapping process; (2) PTAM is the first to use nonlinear optimization instead of using traditional filter as the backend.

RGBD-SLAM-V2 [36] is a system proposed by F. Endres in 2014 to calculate SLAM using a depth camera. The RGBD-SLAM-V2 front end extracts the image features of the 3-D points, matches them, renders the point cloud, builds the pose graph at the back end and optimizes with g2o, and finally outputs the map. RGBD-SLAM-V2 uses only the depth camera - RGBD camera, and uses the current popular technology such as image feature extraction, loop detection, point cloud, and graph optimization in the SLAM field. The effect is good, but feature point extraction and point cloud rendering. It is a time-consuming link, and the real-time performance of the algorithm needs to be improved.

LSD-SLAM [37] is a visual SLAM scheme proposed by J. Engle et al. Its implementation makes the direct method based on single object applied to the visual SLAM in real sense and has achieved good results. LSD-SLAM does not calculate feature points, but only directly for pixels,the so-called "direct method". The feature-based visual SLAM can only construct sparse maps, while the direct method based LSD-SLAM can construct semi-dense Map, this is an advantage of LSD-SLAM over feature point based SLAM schemes.

SVO is the abbreviation of Semi-direct Visual Odoemtry [38]. It is a visual odometry based on the sparse direct method proposed by Forster et al. in 2014. Due to the sparse direct method, it does not have to work hard to calculate descriptors, and does not have to deal with as much information as dense and semi-dense, so SVO is fast and can meet real-time performance, which is one of its advantages. SVO can reach more than 100 frames per second on the PC platform. In the subsequent SVO 2.0, the speed reached an astonishing 400 frames per second. This makes it ideal for applications where computing platforms are limited, such as the positioning of drones, handheld AR/VR devices.

ORB-SLAM [39] is a SLAM system proposed in 2015, representing a peak of the mainstream feature point SLAM. In the subsequent ORB-SLAM2, the author extended the ORB-SLAM to support not only monocular cameras, but also binocular cameras and RGB-D depth cameras. Compared to previous work, ORB-SLAM has the following obvious advantages: (1) not only supports monocular cameras, but also supports binocular cameras, and RGB-D depth cameras, which makes ORB-SLAM have a good pan (2) The whole process of the system is based on the ORB feature, so that the ORB-SLAM can meet the real-time performance on the CPU; (3) the loop closing detection of the ORB is its highlight; (4) after the double-threaded structure of the PTAM ORB-SLAM proposed a three-threaded structure, and achieved very good tracking and mapping effects, and ensured the global consistency of the trajectory and the map.

Direct Sparse Odometry (DSO) [41] is a visual odometry method based on novel, high-precision sparse direct structures and motion formulas. It combines a completely straightforward probability model (minimized luminosity error) with consistent joint optimization of all model parameters, including geometry and camera motion expressed as inverse depth in the reference frame. The experimental results show that DSO is obviously due to the direct and indirect methods in terms of tracking accuracy and robustness.

3. Future Development Trend of Visual SLAM

3.1. SLAM in Dynamic Scenes

At present, many visual SLAM systems can work normally in static and rigid bodies. The illumination changes in these places are not obvious, and there is no human interference in occlusion and motion blur. Undoubtedly, this apparently overly idealized static environment assumption limits the scope of use of visual SLAM applications. After all, the actual environment is dynamic and variable, and visual sensors are usually installed on dynamic platforms. Therefore, in order to apply SLAM to the actual scene as soon as possible, it is necessary to study SLAM [42,43,44] in the dynamic scene.

The literature [45] proposes a motion removal method based on RGB-D data and integrates it into the RGB-D SLAM front end. The motion removal method acts as a preprocessing stage to filter out data related to moving objects.

The literature [46] proposes a new dynamic scene generation method—generalized motion SLAM. The method is based on a probability hypothesis density filter that anchors the observer state in a probabilistic manner by fusing the inferred observer information and the observer motion report.The literature deduces the general GEM-SLAM theoretical framework and proves that it summarizes the existing SLAM algorithm based on probability hypothesis density (PHD). Simulation results using a distance sensor and multiple moving objects to achieve a specific model show that GEM-SLAM achieves significant improvements over the three benchmark algorithms.

The literature [47] combines two-dimensional object detection and three-dimensional geometric segmentation to achieve real-time computing performance of semantic instance segmentation. In addition, the literature also proposes a method for detecting and segmenting the motion of semantically unknown objects, thereby further improving the accuracy of camera tracking and map reconstruction. The results show that this method is basically consistent with the previous method in terms of positioning and target reconstruction accuracy, and even better. The literature [48] proposes a workflow for accurately segmenting objects and marking them as potential dynamic object regions based on semantic information. A new motion detection and motion removal method is proposed.

3.2. Multi-Feature Fusion SLAM

For geometric computer vision algorithms that rely on the correspondence of points, especially visual SLAM, low-texture scenes are one of the main weaknesses. However, in many environments, line and plane-based geometric primitives can be reliably estimated despite the low texture, such as in urban and indoor scenes, or in so-called "Manhattan World", structured edges and planes. Dominant. In addition, the SLAM method based on the point feature can easily fail when the feature points temporarily disappear due to motion blur or the like. To this end, researchers have gradually focused on image features outside of point features.

The literature [49] proposed a positioning and mapping (SLAM) algorithm for handheld 3D sensors based on point and surface features. The algorithm uses the minimum set of primitives in the RANSAC framework for robust calculations and estimates the pose of the sensor. The algorithm has the following advantages: (1) due to the small number of planar features, the corresponding search and registration speed is faster; (2) the planar model is more compact than the point model; (3) as a global registration algorithm, it does not exist Locally small or any initialization problem.

[50] proposed a simultaneous localization and mapping system for RGB-D sensors based on point features and planar features. In order to achieve fast planar segmentation, a line primitive segmentation method based on distance image is proposed. Due to the limitations of RGB-D sensors, global flat maps often contain duplicate landmarks. For better mapping performance, a refinement step was designed to merge the repeated planar landmarks. The experimental results of the data set and the handheld RGB-D sensor verify the robustness and effectiveness of the SLAM system.

The literature [51] is based on ORB-SLAM and extends its content to handle the correspondence of points and lines simultaneously. The literature proposes a solution that allows the SLAM system to work when most of the points disappear from the input image. The literature [52] proposes a stereo visual SLAM system combining points and line segments, which can work stably in a wider range of scenarios, especially In the case where the point features are sparse or unevenly distributed. PL-SLAM utilizes points and segments in all instances of the process: visual metrics, keyframe selection, Bundle Adjustment, and more.

The literature [53] developed and tested a planar-based simultaneous localization and mapping algorithm, which can deal with the problem of uneven sampling density of the speed-measuring scanning lidar sensor in real time. The algorithm uses an efficient planar detector to quickly provide stable features for both positioning and landmarks in graph-based SLAMs. When the plane cannot be detected or the positioning support is insufficient, a new constraint tracking algorithm selects a set of minimum supplemental point features to provide to the location solver. The experimental results of the algorithm were compared with the two most advanced algorithms, GICP and LOAM, which showed an order of magnitude faster speed and higher accuracy on all data sets.

3.3. Multi-Sensor Fusion SLAM

Whether it is an actual robot or a hardware device, it usually does not carry only one type of sensor, but often incorporates multiple sensors. Researchers in academia love Big Clean Problems, such as the visual SLAM implemented by a single camera. But industry players are more focused on making algorithms more practical and have to deal with complex and trivial scenarios. In this context, the SLAM that combines the camera with other sensors has become a hot spot[56,57,58].

[54] Using adaptive Kalman filter (KF) fusion of landmark sensors and strap-down inertial measurement unit data, three-dimensional simultaneous localization and mapping (SLAM) and Its observability analysis was studied. In addition to the state of the vehicle and the location of the landmark, the self-tuning filter estimates the covariance of the IMU calibration parameters and the measured noise. Examining the observability of the 3D SLAM system leads to the conclusion that the system is still observable, and at least one of the two conditions must be met. i) Acceleration observations of the non-collinear vectors of the two known landmarks ii) The known landmarks are not placed in a straight line. The literature [55] proposes and compares two methods for estimating unknown scale parameters in the monocular SLAM framework. Directly related to the scale is an estimate of the absolute velocity and position of the object in three dimensions. The first method is based on the spline fitting task of Jung and Taylor, and the second method is extended Kalman filtering. The literature embeds an online multi-rate extended Kalman filter and an inertial sensor in the parallel tracking and mapping (PTAM) algorithm of Klein and Murray. In this inertial/monocular SLAM framework, the literature presents a real-time, robust, fast-converging scale estimate.

The observer of the [59] only needs a monocular camera and an inertial measurement unit (IMU) whose contribution is to improve the convergence speed of the nonlinear observer. The iterative process and the process of maintaining the origin of the inertia within the field of view.

[60] proposes a new underwater camera-inertial measurement unit (IMU) calibration model, which requires calibration once and then between the two cameras Parameters and external parameters and the IMU can automatically calculate the index based on the environment. It seems that this is the first time to consider calibrating the underwater camera-IMU through environmental indicators.

3.4. Multi-Robot Collaborative SLAM

There are already many solutions for the visual SLAM system of a single robot. However, the multi-robot visual SLAM field has yet to be studied in terms of communication topology, mission planning and map fusion[61,63].

The literature [62] proposes a large-scale SLAM algorithm to process multiple robots, where the global map maintains a series of local subgraphs established by different robots. The relative relationship. The key problem in dealing with multiple robots is to find the connectionsbetween them and integrate them to maintain overall geometric consistency; detailing the events that introduce these links into the global map. The literature [64] proposed a multi- mobile robot intelligent self-localization algorithm based on simultaneous localization and mapping (SLAM), and proposed a multi-resolution based on multi-resolution An intelligent self- localization method for maps and evolutionary computations based on the relative positions ofother robots within the perceived range. Experimental results show the effectiveness of the method. The literature [65] proposed a multi-robot SLAM hybrid algorithm combining particlefiltering and map fusion. The algorithm does not rely on the intersection point, and calculates the unknown relative pose according to the local map of the robot. The literature [66] proposeda new multi-robot collaborative visual SLAM system CORB-SLAM, which has the functions of map fusion and map sharing. Experimental results on public data sets demonstrate the performance of CORB-SLAM.

3.5. Deep Learning SLAM

With the great success of deep learning in the field of computer vision, many researchers havegreat interest in the application of deep learning in the field of robotics. As a large SLAM systemwith many sub-modules, there are many sub-modules, the feature matching in the visual frontend, and the position recognition in the loop detection can be used to obtain better results by applying deep learning.

The literature [67] introduces SLAM to optimize the estimated odometry information in the feature learning process, which makes the feature effect learned by deep neural network moresignificant. The literature [68] employs a slightly different approach, using a deconvolution network to learn low-dimensional global representation vectors. The proposed 12-layer deconvolution network encodes and decodes the image itself, and in the process learns the representation of the image in the reduced feature space and then uses it to compare the images to identify the loop closing. Experimental results show that this method will have less perceptual aliasing than other methods of the same type. In [69], an unsupervised learning framework is proposed, which not only uses image reconstruction for supervision, but also usesthe pose estimation method to enhance the supervised signal, adding training constraints for monocular depth and camera motion estimation tasks. The experimental results show that the depth estimation task performed by the unsupervised learning framework proposed in the literature on the KITTI dataset is comparable to the supervised method, which is 13.5% higher than the existing method. In addition, it can also assist the initialization process of the ORB- SLAM system to improve the robustness of the system in effectively improving the strong illumination and weak texture scenarios.

4. Conclusion

Since 2007, the first real-time monocular visual SLAM system MonoSLAM has been launched, visual SLAM has undoubtedly achieved amazing development in more than a decade, however, whether it is sensor limitation, program application range, or system real-time and accuracy. Interms of visual SLAM, there are still many defects. To this end, mobile robots have improved their autonomous navigation capabilities, and there is still a long way to go, and researchers must pay more effort and sweat. Throughout the full text, we begin with an overview of the visual SLAM process from a visual odometry, and then introduce the more well-known public data sets in the field. Finally, the future development of visual slam is discussed in five aspects: dynamic scene, multi-feature fusion, multi-sensor fusion, multi-robot cooperation and deep learning.

References

[1] Smith R C, Cheeseman P. On the representation and estimation of spatial uncertainty[J]. International Journal of Robotics Research, 1986, 5(4): 56-68.

[2] Thrun S, Liu Y F, Koller D, et al. Simultaneous localization and mapping with sparse extended information filters[J]. International Journal of Robotics Research, 2004, 23(7/8): 693-716.

[3] Montemerlo M, Thrun S, Koller D, et al. FastSLAM: A factored solution to the simultaneous localization and mapping problem[C]//Proceedings of the National Conference on Artificial Intelligence. Menlo Park, USA: AAAI, 2002: 593-598.

[4] Huang S D, Dissanayake G. Convergence and consistency analysis for extended Kalman filter based SLAM[J]. IEEE Transactions on Robotics, 2007, 23(5): 1036-1049.

[5] Thrun S, Burgard W, Fox D. Probabilistic robotics[M]. Cambridge, USA: MIT Press, 2005.

[6] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International journal of computer vision,2004, 60(2): 91–110.

[7] Bayh, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF)[J]. Computer vision and image understanding, 2008, 110(3): 346–359.

[8] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF[C]//Proceedingsof 2011 IEEE International Conference on Computer Vision. Barcelona, Spain, 2011: 2564–2571.

[9] Rosten E , Porter R , Drummond T . Faster and Better: A Machine Learning Approach to Corner Detection[J]. 2010, 32(1):0-119.

[10] Calonder M, Lepetit V, Strecha C, et al. BRIEF: binary robust independent elementary features[C]//Proceedings of the 11th European Conference on Computer Vision. Heraklion, Crete, Greece, 2010: 778–792.

[11] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, “Complete solution classifcation for the perspective-three-point problem,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, pp. 930–943, Aug 2003.

[12] V. Lepetit, F. Moreno-Noguer, and P. Fua, “Epnp: An accurate o(n) solution to the pnp problem,” International Journal of Computer Vision, vol. 81, no. 2, pp. 155–166, 2008.

[13] A. Penate-Sanchez, J. Andrade-Cetto, and F. Moreno-Noguer, “Exhaustive linearization for robust camera pose and focal length estimation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 10, pp. 2387–2400, 2013.

[14] Triggs B . Bundle Adjustment - A Modern Synthesis[M]// Vision Algorithms: Theory and Practice. Springer Berlin Heidelberg, 1999.

[15] Lu F, Milios E. Globally Consistent range scan alignment for environment mapping [J]. Autonomous Robots, 1997, 4 (4): 333-349.

[16] Dellaert F, Kaess M. Square root SAM: simultaneous localization and mapping via square root information smoothing [M]. [S. l.] Sage Publications, Inc. 2006.

[17] Kaess M, Ranganathan A, Dellaert F. iSAM: incremental smoothing and mapping [J]. IEEE Trans on Robotics, 2008, 24 (6): 1365-1378.

[18] Kaess M, Johannsson H, Roberts R, et al. iSAM2: incremental smoothing and mapping using the Bayes tree [J]. International Journal of Robotics Research, 2012, 31 (2): 216-235.

[19] Grisetti G, Kummerle R, Stachniss C, et al. A tutorial on graph-based SLAM [J]. IEEE Intelligent Transportation Systems Magazine, 2011, 2 (4): 31-43.

[20] Rosen D M, Kaess M, Leonard J J. An incremental trust-region method for Robust online sparse least-squares estimation [C]// Proc of IEEE International Conference on Robotics and Automation. [S. l.] IEEE Press, 2012: 1262-1269.

[21] Zhong J, Tian L, Varma P, et al. Nonlinear optimization algorithm for partially coherent phase retrieval and source recovery [J]. IEEE Trans on Computational Imaging, 2016, 2 (3): 310-322.

[22] Duckett T, Marsland S, Shapiro J. Learning globally consistent maps by relaxation [C]// Proc of International Conference on Robotics and Automation. [S. l.] IEEE Press, 2000: 3841-3846 vol. 4.

[23] Frese U, Larsson P, Duckett T. A multilevel relaxation algorithm for simultaneous localization and mapping [M]. [S. l.] IEEE Press, 2005.

[24] Carlone L, Censi A, Dellaert F. Selecting good measurements via ℓ 1, relaxation: A convex approachfor robust estimation over graphs [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. [S. l.] IEEE Press, 2014: 2667-2674.

[25] Olson E, Leonard J, Teller S. Fast iterative alignment of pose graphs with poor initial estimates [C]//Proc of IEEE International Conference on Robotics and Automation. [S. l.] IEEE Press, 2006: 2262-2269.

[26] Grisetti G, Grzonka S, Stachniss C, et al. Efficient estimation of accurate maximum likelihood maps in 3D [C]// Proc of IEEE//RSJ International Conference on Intelligent Robots and Systems. [S. l.] IEEE Press, 2007: 3472-3478.

[27] Gao C, Harle R. MSGD: Scalable back-end for indoor magnetic field-based GraphSLAM [C]// Proc of IEEE International Conference on Robotics and Automation. [S. l.] IEEE Press, 2017.

[28] Grisetti G, Kummerle R, Stachniss C, et al. Hierarchical optimization on manifolds for online 2D and 3D mapping [C]// Proc of IEEE International Conference on Robotics and Automation. [S. l.] IEEE Press, 2010: 273-278.

[29] Kümmerle R, Grisetti G, Strasdat H, et al. G2o: a general framework for graph optimization [C]// Proc of IEEE International Conference on Robotics and Automation. [S. l.] IEEE Press, 2011: 3607-3613.

[30] Cummins M , Newman P . FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance[M]. Sage Publications, Inc. 2008.

[31] Galvez-Lo?Pez D , Tardos J D . Bags of Binary Words for Fast Place Recognition in Image Sequences[J]. IEEE Transactions on Robotics, 2012, 28(5):1188-1197.

[32] Lee J H , Zhang G , Lim J , et al. Place recognition using straight lines for vision-based SLAM[C]// Robotics and Automation (ICRA), 2013 IEEE International Conference on. IEEE, 2013.

[33] Yang S. Place Recognition using Multiple Feature Types[J]. Advances in Robotics & Automation, 2015, 01(s2).

[34] Davison A J, Reid I D, Molton N D, et al. MonoSLAM: Real-Time Single Camera SLAM[J]. IEEETransactions on Pattern Analysis & Machine Intelligence, 2007, 29(6):1052-1067.

[35] Klein G, Murray D. Parallel Tracking and Mapping for Small AR Workspaces[C]// IEEE and ACM International Symposium on Mixed and Augmented Reality. IEEE Computer Society, 2007:1-10.

[36] 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.

[37] Engel J, Schöps T, Cremers D. LSD-SLAM: Large-Scale Direct Monocular SLAM[M]// Computer Vision – ECCV 2014. Springer International Publishing, 2014:834-849.

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

[39] Mur-Artal R , Montiel J M M , Tardos J D . ORB-SLAM: a Versatile and Accurate Monocular SLAM System[J]. IEEE Transactions on Robotics, 2015, 31(5):1147-1163.

[40] Mur-Artal R , Tardos J D . ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras[J]. 2016.

[41] Engel J , Koltun V , Cremers D . Direct Sparse Odometry[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017:1-1.

[42] Saarinen J , Andreasson H , Stoyanov T , et al. Normal Distributions Transform Occupancy Maps: Application to large-scale online 3D mapping[C]// Robotics and Automation (ICRA), 2013 IEEE International Conference on. IEEE, 2013.

[43] Maddern W , Milford M , Wyeth G . CAT-SLAM: probabilistic localisation and mapping using a continuous appearance-based trajectory[J]. The International Journal of Robotics Research, 2012, 31(4):429-451.

[44] Wang, H.M, Hou, Z G, Cheng, and M.Tan. Online mapping with a mobile robot in dynamic and unknown environments[J]. International Journal of Modelling Identification & Control, 4(4):415.

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

[46] C. Evers and P. A. Naylor, "Optimized Self-Localization for SLAM in Dynamic Scenes Using Probability Hypothesis Density Filters," in IEEE Transactions on Signal Processing, vol. 66, no. 4, pp. 863-878, 15 Feb.15, 2018.

[47] Hachiuma R , Pirchheim C , Schmalstieg D , et al. DetectFusion: Detecting and Segmenting Both Known and Unknown Dynamic Objects in Real-time SLAM[J]. 2019.

[48] L. Zhao, Z. Liu, J. Chen, W. Cai, W. Wang and L. Zeng, "A Compatible Framework for RGB-D SLAM in Dynamic Scenes," in IEEE Access, vol. 7, pp. 75604-75614, 2019.

[49] Y. Taguchi, Y. Jian, S. Ramalingam and C. Feng, "Point-plane SLAM for hand-held 3D sensors," 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, 2013, pp. 5182-5189.

[50] L. Zhang, D. Chen and W. Liu, "Point-plane SLAM based on line-based plane segmentation approach," 2016 IEEE International Conference on Robotics and Biomimetics (ROBIO), Qingdao, 2016, pp. 1287-1292.

[51] A.Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu and F. Moreno-Noguer, "PL-SLAM: Real-time monocular visual SLAM with points and lines," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017, pp. 4503-4508.

[52] R. Gomez-Ojeda, F. Moreno, D. Zuñiga-Noël, D. Scaramuzza and J. Gonzalez-Jimenez, "PL-SLAM: A Stereo SLAM System Through the Combination of Points and Line Segments," in IEEE Transactionson Robotics, vol. 35, no. 3, pp. 734-746, June 2019.

[53] Shane G W , Voorhies R C , Laurent I . Efficient Velodyne SLAM with point and plane features[J]. Autonomous Robots, 2018.

[54] F. Aghili, "Integrating IMU and landmark sensors for 3D SLAM and the observability analysis," 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, 2010, pp. 2025-2032.

[55] Gabriel Nützi, Weiss S , Scaramuzza D , et al. Fusion of IMU and Vision for Absolute Scale Estimationin Monocular SLAM[J]. Journal of Intelligent & Robotic Systems, 2011, 61(1-4):287-299.

[56] Petersen A . A Specialized Kalman Filter Framework for IMU Aided Stereo SLAM[J]. 2014.

[57] A.Barrau and S. Bonnabel, "Invariant filtering for Pose EKF-SLAM aided by an IMU," 2015 54th IEEE Conference on Decision and Control (CDC), Osaka, 2015, pp. 2133-2138.

[58] Triputen, Sergey & Schreve, Kristiaan & Tkachev, Viktor & Rätsch, Matthias. (2017). Closed-form Solution for IMU based LSD-SLAM Point Cloud Conversion into the Scaled 3D World Environment.

[59] J. Nielsen and R. Beard, "Ground Target Tracking Using a Monocular Camera and IMU in a Nonlinear Observer SLAM Framework," 2018 Annual American Control Conference (ACC), Milwaukee, WI, 2018, pp. 6457-6462.

[60] C. Gu, Y. Cong and G. Sun, "Environment Driven Underwater Camera-IMU Calibration for Monocular Visual-Inertial SLAM," 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 2019, pp. 2405-2411.

[61] Gil A , óscar Reinoso, Mónica Ballesta, et al. Multi-robot visual SLAM using a Rao-Blackwellized particle filter[J]. Robotics and Autonomous Systems, 2010, 58(1):68-80.

[62] Vidal-Calleja T A , Berger C , Joan Solà, et al. Large scale multiple robot visual mapping with heterogeneous landmarks in semi-structured terrain[J]. Robotics & Autonomous Systems, 2011, 59(9):654-674.

[63] Zou D , Tan P . CoSLAM: Collaborative Visual SLAM in Dynamic Environments[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 35(2):354-366.

[64] Toda Y , Suzuki S , Kubota N . Evolutionary Computation for Intelligent Self-localization in Multiple Mobile Robots Based on SLAM[C]// Proceedings of the 5th international conference on Intelligent Robotics and Applications - Volume Part I. Springer-Verlag, 2012.

[65] S. Saeedi, M. Trentini and H. Li, "A hybrid approach for multiple-robot SLAM with particle filtering," 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, 2015, pp. 3421-3426.

[66] Li F., Yang S., Yi X., Yang X. (2018) CORB-SLAM: A Collaborative Visual SLAM System for Multiple Robots.

[67] P. Agrawal, J. Carreira and J. Malik, "Learning to See by Moving," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, pp. 37-45.

[68] Mukherjee A , Chakraborty S , Saha S K . Learning Deep Representation for Place Recognition in SLAM[J]. 2017.

[69] Geng M , Shang S , Ding B , et al. Unsupervised Learning-based Depth Estimation aided Visual SLAM Approach[J]. Circuits Systems and Signal Processing, 2019.