ISSN: 2664-9640

Application Status and Development Trends of Visual SLAM Technology based on Computer Vision

Zehua Hua, Lu Dong*, Xiaoyue Peng, Tingyu Zhu

Civil Aviation Flight University of China, Guanghan 618300, Sichuan, China

*Corresponding author

Abstract

In the field of future technology, artificial intelligence technologies such as autonomous driving, unmanned mobile carrier platforms, and service robots will become the next hotspots, among which simultaneous localization and mapping (SLAM) technology is one of the key technologies. Visual SLAM technology has emerged as a research focus due to its advantages of low economic cost, rich information content, and excellent mapping performance. This paper first introduces the application background, research platforms, and underlying mathematical theories of visual SLAM technology, including computer image processing technology, multi-view geometry, Lie groups and Lie algebras in algebra, and state estimation in probability theory. Subsequently, it elaborates on the specific process of visual SLAM technology for processing video and image data, covering four main components: front-end visual odometry, back-end optimization, loop closure detection, and map construction. Finally, the current application status and technical obstacles of visual SLAM technology are summarized, and prospects for its future optimization and development directions are presented.

Keywords

Visual SLAM; Computer Image Processing Technology; State Estimation; Loop Closure Detection; Mapping.

1. Introduction

With the rapid development of artificial intelligence technology, an increasing number of researchers worldwide have embarked on the development of intelligent robots [1-2]. These robot products equipped with human-like intelligence are expected to drive significant progress in industrial production, transportation, daily life, and other fields [3-4]. Although intelligent robots have initially acquired human-like logical thinking and external perception capabilities, there are substantial differences between the internal information processing methods of robots and the way humans understand external information for survival. Regarding robot visual perception technology, its core lies in simultaneous localization and mapping (SLAM) technology implemented through camera lenses, abbreviated as Visual SLAM [5]. Simultaneous localization and mapping technology refers to estimating and predicting the motion process of a carrier equipped with various sensors in an unknown environment while constructing a 2D or 3D environmental structure map using computer technology [6-7]. The equipped sensors can be inertial sensors, sonar, lidar, or cameras, and the carriers can be land unmanned vehicles, aerial drones, surface unmanned boats, underwater autonomous vehicles, etc [8]. Therefore, SLAM technology has a very wide range of applications, as shown in Figure 1.

ISSN: 2664-9640

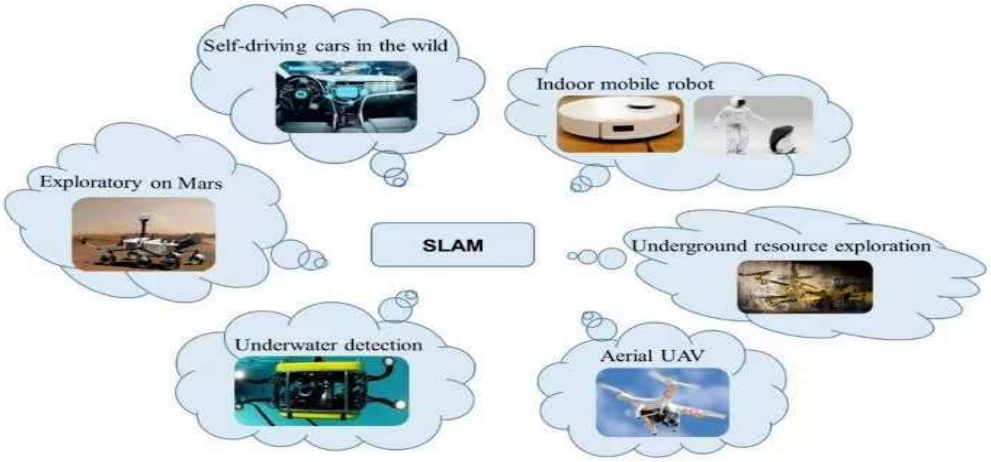

Figure 1. Application Scenarios of Visual SLAM Technology

Visual SLAM technology involves an extensive range of mathematical background knowledge, such as projective geometry, computer vision, image processing, state estimation theory, Lie groups and Lie algebras, and nonlinear optimization problems. Its physical hardware mainly includes visual sensors, motion control circuit boards, microcomputers, and carrier platforms, while its software support mainly includes Linux/Ubuntu operating systems, C++ programming environments, and algorithm libraries such as OpenCV, Sophus, and Pangolin[9].

Visual simultaneous localization and mapping is an important part of robot autonomous navigation and environmental understanding[10]. Its core goal is to simultaneously achieve camera pose estimation and map construction in an unknown environment using monocular, binocular, or RGB-D visual sensors. Compared with lidar SLAM, visual SLAM has the advantages of low equipment cost, rich information content, and good applicability, thus being widely used in fields such as drone navigation, service robots, augmented reality (AR), and virtual reality (VR)[10].

2. Relevant Principles of Visual SLAM Technology

2.1. 3D Stereo Geometry

Since the physical space inhabited by humans is three-dimensional, we are accustomed to dealing with motion problems in the 3D physical world. Motion in 3D space consists of 3 coordinate axes and 6 degrees of freedom. If only the motion of a single point in space is considered, there is no rotation involved, only translation, which can be represented by a 3-dimensional vector coordinate. However, what we describe here is the motion of a camera mounted on a carrier platform, i.e., the motion of a rigid body in space. To describe its attitude, not only a translation vector but also a rotation matrix is required.

Using linear algebra theory for research, in a certain linear space, it is necessary to find a set of bases for that space, and then a vector α will have a coordinate under this set of bases:

$$\alpha = (e_1, e_2, e_3)\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = a_1 e_1 + a_2 e_2 + a_3 e_3 \tag{1}$$

Here, $(a_1', a_2', a_3')^T$ is called the coordinate of α under this basis.

ISSN: 2664-9640

The so-called rigid body motion refers to a motion composed of a rotation and a translation between two coordinate systems. A camera is a rigid body, and its motion is a rigid body motion. There is an Euclidean transformation between the camera's own coordinate system and the world coordinate system. Suppose a set of orthonormal bases (e_1, e_2, e_3) is transformed into (e_1', e_2', e_3') after a rotation. For the same vector, the vector does not move with the rotation of the coordinate system, so its coordinates under the two coordinate systems are $(a_1, a_2, a_3)^T$ and $(a_1', a_2', a_3')^T$, as shown in Figure 2. Therefore, from Formula (1), we can obtain:

$$(e_1, e_2, e_3)\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = (e_1', e_2', e_3')\begin{pmatrix} a_1' \\ a_2' \\ a_3' \end{pmatrix} \tag{2}$$

To describe the relationship between the two coordinates, multiply both sides of the above equation by $\begin{pmatrix} e_1^T \\ e_2^T \\ e_3^T \end{pmatrix}$ on the left. Then the coefficient on the left becomes the identity matrix, i.e.:

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} e_1^T e_1, e_1^T e_2, e_1^T e_3 \\ e_2^T e_1, e_2^T e_2, e_2^T e_3 \\ e_3^T e_1, e_3^T e_2, e_3^T e_3 \end{pmatrix}\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \overset{def}{=} Ra \tag{3}$$

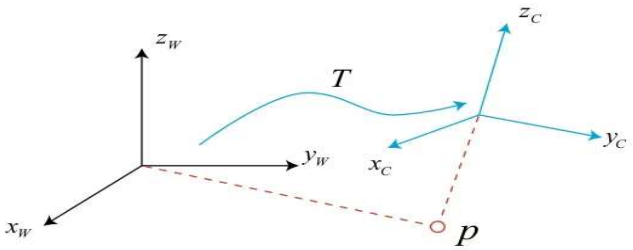

Figure 2. Schematic Diagram of Rotation Relationship of the Same Point Under Two Coordinate Systems

The rotation matrix R has some favorable properties that can be derived using linear algebra knowledge. The rotation matrix is an orthogonal matrix with a determinant equal to 1, i.e., a unit orthogonal matrix. Conversely, an orthogonal matrix with a determinant equal to 1 is also a rotation matrix. Therefore, the set of n-dimensional rotation matrices can be defined as follows:

$$so(n) = \left\{ R \in R^{n \times n} \middle| RR^T = I, \det(R) = 1 \right\} \tag{4}$$

where $so(n)$ denotes the special orthogonal group.

ISSN: 2664-9640

2.2. Fundamentals of Lie Groups and Lie Algebras

A group is an algebraic structure consisting of a set and an operation. Denoting the set as A and the operation as •, the group can be written as G = (A, •). A Lie group refers to a group with continuous and smooth properties. Each Lie group has a corresponding Lie algebra. The Lie algebra describes the local properties of the Lie group, specifically the tangent space near the identity element. The general definition of a Lie algebra is as follows: A Lie algebra consists of a set V, a number field F, and a binary operation [•,•]. If the following properties are satisfied, (V, F, [•,•]) is called a Lie algebra, denoted as g;

(1) Closure: The result of the specified [•,•] operation on any two distinct elements in set V still belongs to V;

(2) Bilinearity: For any three elements X, Y, Z in set V and any two values a, b in number field F, we have

$$[aX+bY,Z]=a[X,Z],[Z,aX+bY]=a[Z,X]+b[Z,Y] \tag{5}$$

(3) Anticommutativity: For any element X in set V, the operation [X, X] = 0 holds;

(4) Jacobi identity: For any three elements X, Y, Z in set V, the following relation holds:

$$[X,[Y,Z]]+[Z,[X,Y]]+[Y,[Z,X]]=0 \tag{6}$$

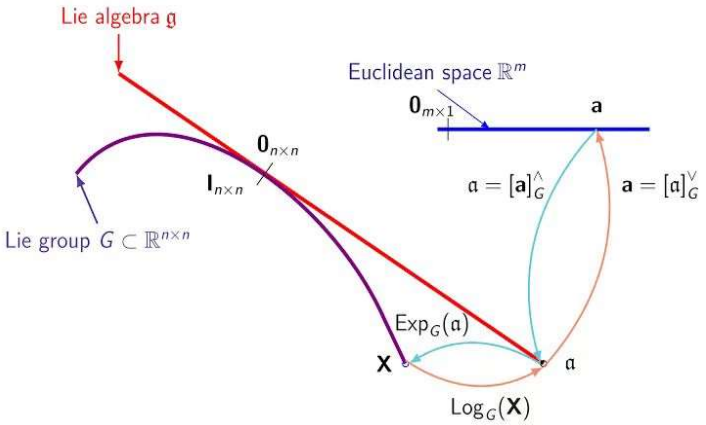

Figure 3. Correspondence Diagram of Lie Groups and Lie Algebras

The Lie algebra corresponding to $so(3)$ is a vector defined on R^3, which can be denoted as φ. Each vector φ can generate a skew-symmetric matrix $\hat{\varphi}$, as shown in Figure 3. Its relationship with $so(3)$ can be corresponding through the exponential map, i.e.:

$$R=\exp(\hat{\varphi}) \tag{7}$$

2.3. Probability Theory and State Estimation

The calculation process of visual SLAM is described by the motion equation and the observation equation, as shown in Figure 4.

ISSN: 2664-9640

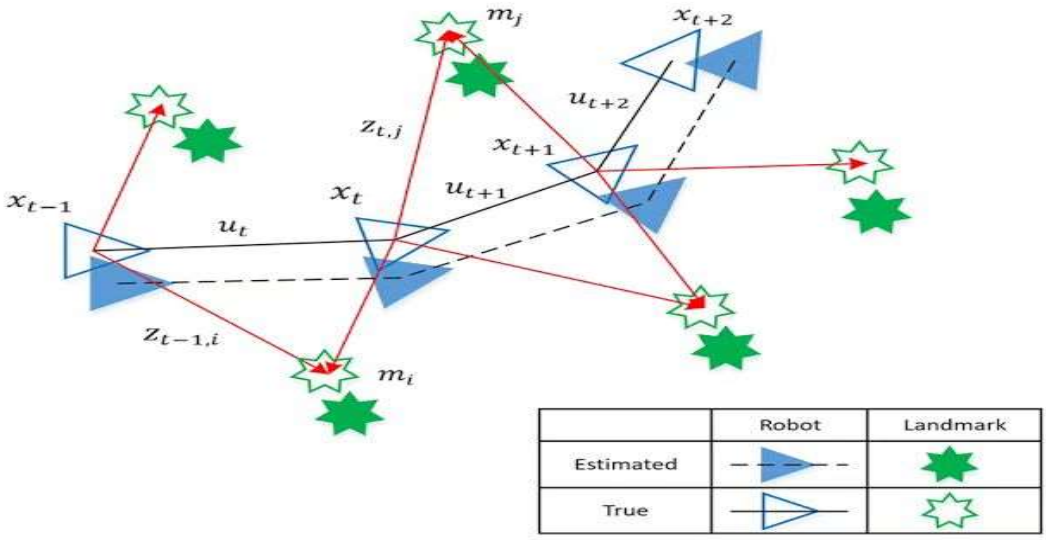

Figure 4. Robot State Estimation Diagram

Combining the above derivations, the motion equation is:

$$\begin{cases} x_k = f(x_{k-1}, u_k) + \omega_k \\ z_{k,j} = h(y_j, x_k) + v_{k,j} \end{cases} \tag{8}$$

where x_k represents the camera's own pose at each moment, the landmark point is y_j , u_k represents the sensor reading or input, and ω_k represents the noise introduced by external interference.

In the observation equation, when the camera detects a target point y_j at position x_k, it will generate observation data $z_{k,j}$, and thus a noise interference $v_{k,j}$ will occur in this observation.

Each equation is affected by noise during the actual operation. Specifically, the input data is always subject to external noise interference, leading to the gradual accumulation of errors, which results in an increasing variance in position estimation. Using the maximum likelihood estimation method, the batch state estimation problem can be transformed into a maximum likelihood estimation problem, which is solved using the least squares method.

3. Basic Process and Key Steps of Visual SLAM Technology

As shown in Figure 5, the entire visual SLAM algorithm process mainly includes the following 4 parts: Firstly, the front-end visual odometry part estimates the camera motion between adjacent images and the local map after reading and preprocessing the camera images. Secondly, the back-end receives the camera poses measured by the visual odometry at different times and the information obtained from loop closure detection, optimizes them, and obtains a globally consistent trajectory and map. Thirdly, loop closure detection can determine whether the robot has reached a previous position, and send the information to the back-end for processing by detecting loops. Fourthly, map construction builds a map corresponding to the task requirements based on the estimated trajectory.

ISSN: 2664-9640

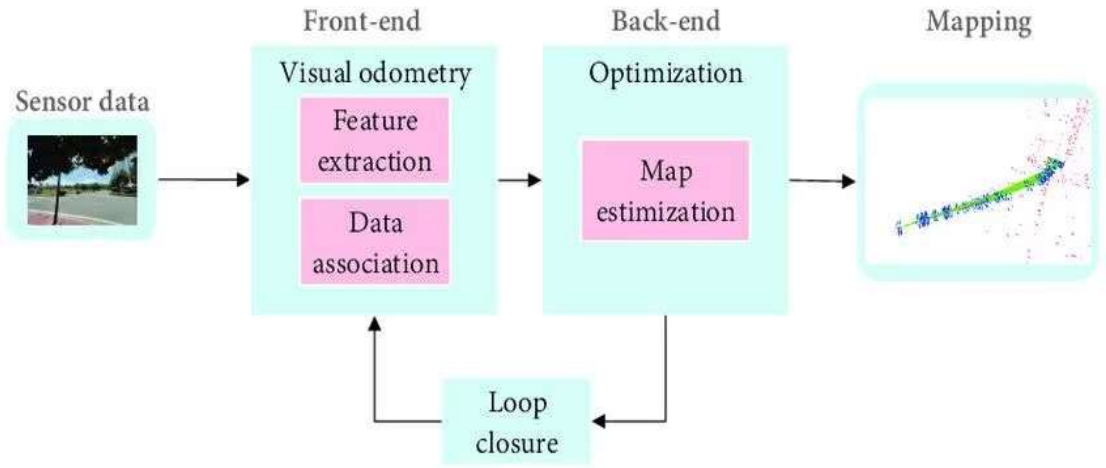

Figure 5. Basic Flowchart of Visual SLAM Technology

3.1. Front-end Visual Odometry

Front-end Visual Odometry (VO) is a core component of the visual SLAM system responsible for estimating the continuous pose changes of the camera. Its main goal is to recover the camera's motion trajectory from image sequences in an unknown environment. As a key module of the SLAM front-end, the output quality of VO directly affects back-end optimization, map construction, as well as the robustness and stability of the entire system[11].

The typical process of VO includes: image preprocessing, feature extraction and tracking, feature matching, outlier rejection (e.g., RANSAC), pose solving (e.g., PnP, triangulation, homography/fundamental matrix decomposition), etc. In recent years, deep learning has been widely used in feature description and matching, such as SuperPoint and SuperGlue, which have further improved the robustness and generalization ability of the front-end VO.

Visual odometry is generally divided into two categories: Feature-based methods and Direct methods. Feature-based methods estimate the relative pose of the camera between two frames by extracting and matching salient feature points in images (such as ORB, SIFT, AKAZE) using feature correspondences. These methods are robust to changes in illumination and exposure, and easy to implement in engineering, which are also the foundation of mainstream systems such as the ORB-SLAM series. In contrast, direct methods solve for camera motion by minimizing pixel brightness errors without the need for feature points. They can maintain good performance in low-texture scenes but are sensitive to illumination changes and require more computing resources. As shown in Figure 6,

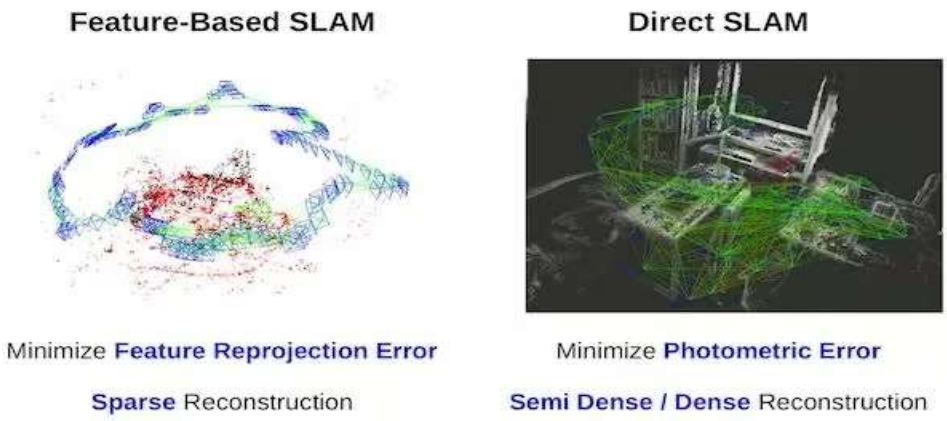

Figure 6. Effect Diagrams of Feature-based Method and Direct Method

ISSN: 2664-9640

In summary, the front-end visual odometry is responsible for providing real-time and preliminary camera motion estimation, which is the foundation for the stable operation of the visual SLAM system. Its accuracy and robustness have a decisive impact on the overall SLAM performance.

3.2. Back-end Optimization

Back-end optimization is a key module in the visual SLAM system for improving the overall trajectory consistency and map accuracy. Its main function is to perform global or local nonlinear least squares optimization on camera poses and 3D map points using constraint relationships between multi-frame observations. The SLAM front-end usually provides preliminary camera pose estimates and a small number of inaccurate feature correspondences, but these estimates often have cumulative errors due to noise, occlusion, feature mismatches, and other factors. By introducing graph optimization methods, back-end optimization effectively suppresses drift and improves global consistency, which is the core guarantee for the accuracy of modern SLAM systems.

Back-end optimization is generally divided into two levels: Local Bundle Adjustment (Local BA) and Global Bundle Adjustment (Global BA). Local BA optimizes the latest keyframes and map points within a sliding window, with a small amount of computation and the ability to run in real-time; Global BA is executed after loop closure is detected, achieving global consistency correction by fusing the constraint relationships between the two previous and subsequent trajectories. As shown in Figure 7,

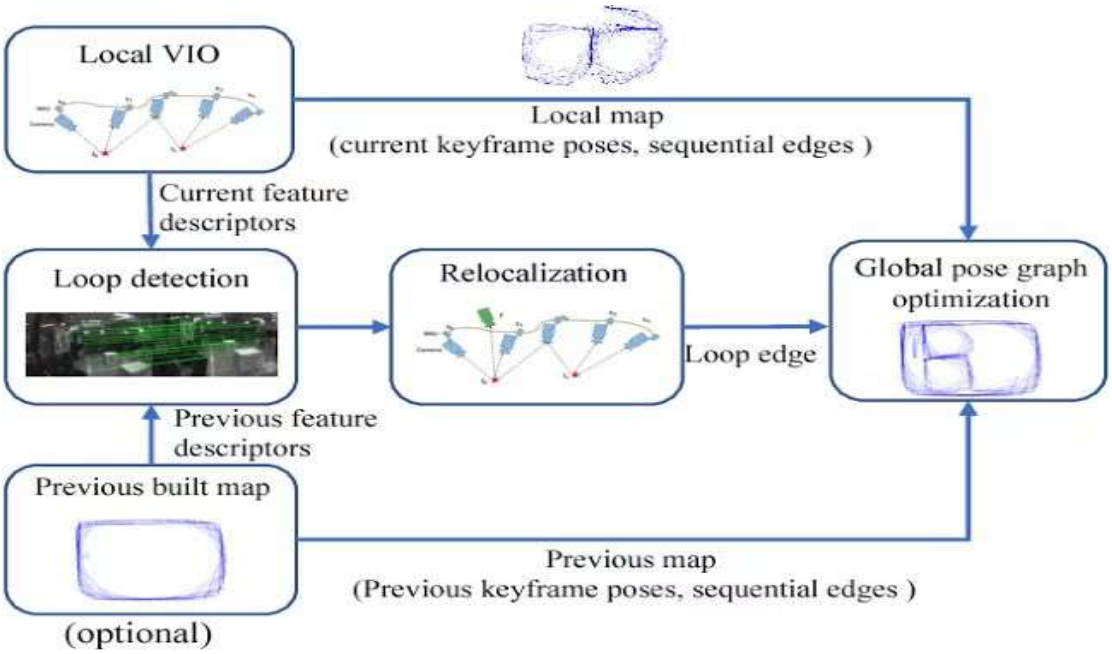

Figure 7. Flowchart of Back-end Optimization in Visual SLAM Technology

Back-end optimization directly affects the upper limit of SLAM accuracy and is a core link in building robust, adaptive, and high-precision SLAM systems. With the development of deep learning, sparse linear algebra, and parallel computing technologies, back-end optimization will evolve towards a more efficient, stable, and intelligent direction in the future.

3.3. Loop Closure Detection

Loop closure detection is an important mechanism for visual SLAM systems to ensure long-term consistency. Its core goal is to automatically identify the scene similarity between the current frame and historical keyframes when the robot revisits a previously visited location,

ISSN: 2664-9640

thereby correcting cumulative drift and improving the global consistency of the map. Since relying solely on front-end visual odometry will accumulate errors with motion, the trajectory will inevitably drift during long-term operation. Therefore, loop closure detection is regarded as a key step of "loop constraints" in SLAM mapping. As shown in Figure 8,

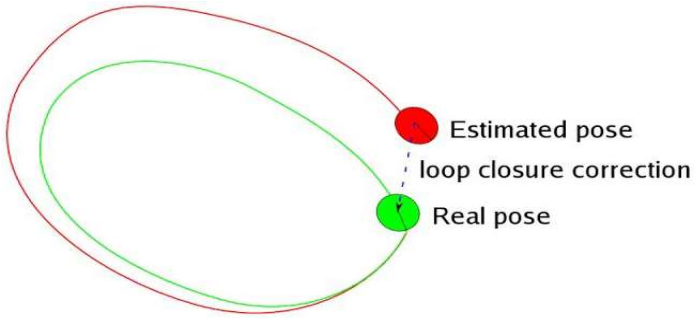

Figure 8. Schematic Diagram of Loop Closure Detection in Visual SLAM Technology

In recent years, loop closure detection methods based on deep learning (such as NetVLAD, SuperGlue) have further improved robustness under environmental changes, occlusions, and illumination variations, making loop closure detection more accurate and reliable. In summary, loop closure detection is a key link to maintain the long-term stable operation of SLAM, and has a decisive impact on the global consistency of the system and the final map accuracy.

3.4. Mapping

The Mapping module aims to construct a consistent, sparse, and locatable 3D map using keyframe and feature point information from the front-end visual odometry[12]. Map construction is one of the core tasks of SLAM, and its quality directly affects tracking stability, loop closure detection accuracy, and back-end optimization results. Current mainstream visual SLAM systems (such as the ORB-SLAM series) mostly adopt sparse feature point maps, forming a complete map construction process including keyframe insertion, map point creation, local optimization, map maintenance, and other links. As shown in Figure 9,

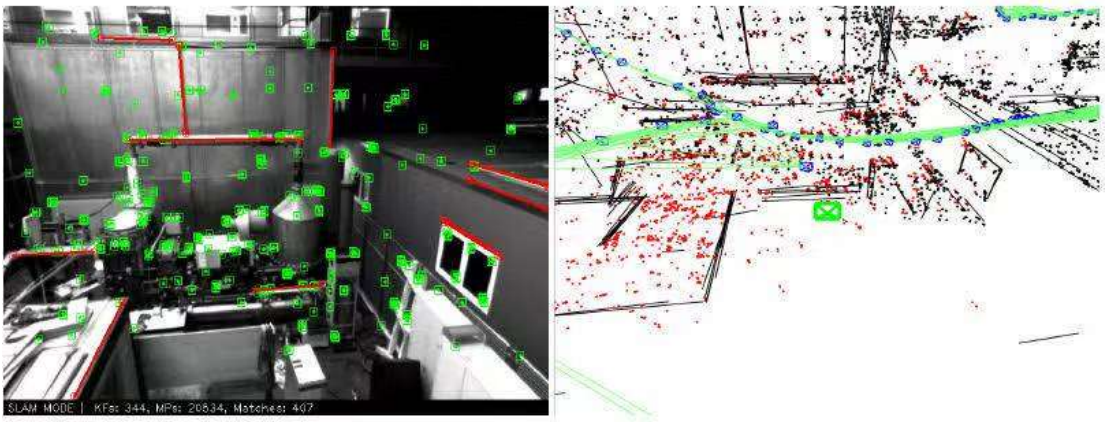

Figure 9. Map Creation Effect Diagram in Visual SLAM Technology

The first step of map construction is completed by the keyframe selection strategy. SLAM does not use every frame of image but inserts keyframes when the parallax, tracking quality, or time threshold conditions are met to control the system scale and improve efficiency. A keyframe contains camera attitude, image features, pyramid layer information, etc., and is the foundation for subsequent map construction.

ISSN: 2664-9640

Subsequently, the system performs Triangulation on the matched features between the new keyframe and adjacent keyframes to generate new 3D map points. Triangulation is usually based on geometric constraints, such as epipolar constraints or fundamental matrix estimation, to recover depth information. For existing map points, observation updates are performed using the current frame, including depth filtering, direction consistency checking, and visibility detection.

4. Conclusion

Early visual SLAM systems mainly adopted feature-based methods, such as MonoSLAM, PTAM, and the ORB-SLAM series. They achieved robust pose estimation by extracting and matching local features, combined with filtering or nonlinear optimization. In recent years, with the development of deep learning, graph optimization, and multi-sensor fusion technologies, visual SLAM has made significant progress in robustness, accuracy, and real-time performance. In particular, systems such as ORB-SLAM2, ORB-SLAM3, VINS-Mono, DSO, and LDSO have achieved high-performance results with engineering practical value in different scenarios.

However, visual SLAM still faces many challenges, such as illumination changes, motion blur, dynamic object interference, low-texture environments, long-term stability, and multi-sensor synchronization. In addition, factors such as the stability of front-end feature extraction and matching, the high computational cost of back-end optimization, and the reliability of loop closure detection all affect the overall system performance. Therefore, systematically sorting out the development context, key modules, typical systems, and shortcomings of existing visual SLAM technologies is of great significance for future research.

Acknowledgments

This work was supported by the project "Standards for Emerging Engineering Majors and Construction of Characteristic Teaching Experiment Platforms" (Civil Aviation Education Talent Programs in China), the open project "Research on the Color Recognition and Appearance Defect Detection System for Solar Cells" [Key Laboratory of Solar Energy Technology Integration and Application Promotion in Sichuan Provincial Universities, Grant No. TYNSYS-2018-Y-06], the project "Research on SLAM for Intelligent Civil Aviation Transportation Robots Based on Stereo Vision" [Civil Aviation Flight University of China, Grant No. 25CAFUC04024], and the project "Design of a Single-Star Simulation System for Calibration of Aviation Attitude Determination Star Sensors" [Chengdu Aeronautic Industry and Cultural Development Research Center, Grant No. CAIACDRCXM2024-04].

References

[1] OBAIGBENA A, LOTTU O A, UGWUANYI E D, et al. AI and human-robot interaction: a review of recent advances and challenges[J].GSC Advanced Research and Reviews,2024,18(2),321-330.

[2] LI J, GAO W, WU Y, et al. High-quality indoor scene 3D reconstruction with RGB-D cameras: a brief review[J]. Computational Visual Media, 2022, 8(3): 369-393

[3] QIN L, WU C, KONG X, et al. BVT-SLAM: a binocular visible-thermal sensors SLAM system in low-light environments[J]. IEEE Sensors Journal,2024,24(7):11599-11609.

[4] ZHANG G, LUO J, XU H, et al. An improved UKF algorithm for extracting weak signals based on RBF neural network[J]. IEEE Transactions on Instrumentation and Measurement, 2022, 71: 1-14.

[5] NAM J, HYEON S, JOO Y, et al. Spectral trade-off for measurement sparsification of pose-graph SLAM[J]. IEEE Robotics and Automation Letters, 2023, 9(1): 723-730.

[6] LI C H, GUO G H, YI P, et al. Distributed pose-graph optimization with multi-level partitioning for multirobot SLAM[J]. IEEE Robotics and Automation Letters, 2024,9(6):4926-4933.

ISSN: 2664-9640

[7] SU P, LUO S, HUANG X. Real-time dynamic SLAM algorithm based on deep learning[J]. IEEE Access, 2022, 10: 87754-87766

[8] YANG C, CHEN Q, YANG Y, et al. SDF-SLAM: a deep learning based highly accurate SLAM using monocular camera aiming at indoor map reconstruction with semantic and depth fusion[J]. IEEE Access, 2022, 10: 10259-10272.

[9] WU H, ZHAO J, XU K, et al. Semantic SLAM based on deep learning in endocavity environment[J]. Symmetry, 2022, 14(3): 614.

[10] Abaspur Kazerouni I, Fitzgerald L, Dooly G, et al. A survey of state-of-the-art on visual SLAM[J]. Expert Systems with Applications, 2022, 205: 117734.

[11] Keetha N, Karhade J, Jatavallabhula K M, et al. Splatam: Splat track & map 3d gaussians for dense rgb-d slam[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 21357-21366.

[12] Teed Z, Deng J. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras[J]. Advances in neural information processing systems, 2021, 34: 16558- 16569.