



ARTICLE

Tunnel Mapping in Low-Light Environments: A Synergistic Scheme of Image Enhancement and Multi-Source Factor Graph Optimization

Qilong Wang¹, Ning Wang¹, Shuhan Luo¹, Xiang Gao², Yuqian Lu³ and Min He^{4,*}

¹School of Mechanical Engineering, Xi'an University of Technology, Xi'an, China

²School of Electrical Engineering, Xi'an University of Technology, Xi'an, China

³Faculty of Engineering and Design, The University of Auckland, Auckland, New Zealand

⁴School of Civil Engineering and Architectural, Xi'an University of Technology, Xi'an, China

*Corresponding Author: Min He. Email: hem@xaut.edu.cn

Received: 08 February 2026; Accepted: 20 April 2026; Published: 27 May 2026

ABSTRACT: Tunnel environments often suffer from GPS denial, uneven illumination, and structural uniformity, which lead to feature degradation, loop closure failure, and long-distance drift in SLAM systems. To solve these problems, this study aims to propose a high-precision SLAM method suitable for tunnel structural health monitoring. Firstly, an ABA-CLAHE image enhancement algorithm is proposed, which adopts cascaded processing of nonlinear brightness adjustment in HSV space and CLAHE local contrast optimization to improve low-light image quality and enhance feature stability. Then, SURF feature matching combined with the RANSAC algorithm is used to ensure feature matching accuracy. Finally, a factor graph model is constructed by integrating IMU pre-integration, laser odometry, visual odometry, and loop closure constraints, and iSAM2 incremental optimization is employed to achieve globally consistent mapping. Municipal tunnel tests show that the loop closure error is reduced to 0.096 m and the global reprojection error is 1.10 pixels, and the structural continuity of the constructed dense 3D map is significantly improved. This method provides a technical solution with centimeter-level accuracy for tunnel structural health monitoring, which is demonstrating strong practical potential for engineering applications.

KEYWORDS: SLAM; municipal tunnels; image enhancement; factor graph

1 Introduction

SLAM (Simultaneous Localization and Mapping) technology was first proposed in the 1980s. Since then, it has become a core technology in fields such as mobile robotics, autonomous driving, augmented reality, and virtual reality. Its core goal is to simultaneously determine an autonomous system's position and construct an environmental map in unknown spaces—laying the groundwork for environmental perception and autonomous navigation.

Early SLAM research mainly focused on EKF-based (Extended Kalman Filter-based) 2D laser SLAM systems. Examples include Gmapping, Hector-SLAM, and Cartographer. This work primarily addressed state estimation and uncertainty modeling problems. With advancements in computing power and sensor accuracy, SLAM technology gradually shifted from the filtering framework to the optimization framework [1–4]. The field also expanded from 2D into 3D space. These changes led to several branches. Notable examples include graph optimization SLAM, sparse visual SLAM, and LiDAR-Inertia tightly-coupled SLAM. Representative algorithms within these branches include GTSAM, ORB-SLAM, and Fast-LIO2 [5–8].

In recent years, advances in multi-source sensors—particularly LiDAR, visual sensors, and IMUs (Inertial Measurement Units)—have driven progress in SLAM. Current research focuses on high-precision 3D SLAM, characterized by multi-sensor fusion, semantic mapping, real-time performance, and robustness optimization. Key technical approaches now include laser SLAM, visual SLAM, visual-inertial SLAM, LiDAR-inertial SLAM, and fully fused systems, which find wide use in autonomous driving, service robotics, post-disaster search and rescue, and underground space surveying [9–12].

Municipal tunnels are the backbone of urban underground infrastructure, supporting key functions such as traffic management, pipeline installation, and the construction of comprehensive utility tunnels. The level of intelligence in their construction and maintenance directly impacts urban operational efficiency and public safety.

Traditionally, tunnel construction and inspection depend on manual positioning with tools like total stations and GNSS (Global Navigation Satellite System)—a method plagued by low efficiency and poor environmental adaptability. These limitations become more pronounced during maintenance operations. Manual inspections are not only time-consuming and labor-intensive but also struggle to accurately identify structural surfaces in complex environments.

1.1 Related Works

Using surveying robots for efficient monitoring and measurement of municipal tunnel structures has significant engineering and practical value. It is crucial for timely identifying safety hazards, enabling informatized tunnel management, gaining insights into construction progress, and conducting structural health assessments.

Compared with open scenes, tunnel environments present more challenging conditions for SLAM, including GNSS/GPS denial, repetitive and elongated spatial structures, sparse distinguishable features, rapid illumination variation, and accumulated long-range drift. These factors significantly constrain the stability and accuracy of conventional SLAM systems [13]. Current SLAM research focusing on tunnels and similar corridor-like environments with singular features primarily involves visual odometry systems and multi-sensor fusion SLAM systems.

For visual SLAM, the ORB-SLAM series serves as classic representatives. These frameworks progressively integrate feature point extraction, loop closure detection, and IMU assistance. The result is a unified system supporting monocular, stereo, and RGB-D (Red Green Blue-Depth) cameras. Specifically, ORB-SLAM2 achieved multi-sensor input compatibility. ORB-SLAM3 further incorporated a tightly-coupled visual-inertial fusion mechanism. This significantly enhanced robustness in dynamic scenes [14,15]. PL-SLAM introduced structural lines and geometric edge features. This improved map construction capability in low-texture indoor environments [16]. To address loop closure mismatches, global descriptor methods like NetVLAD (based on image semantic embeddings) are widely adopted. Such approaches greatly enhance global consistency in repetitive-structure environments [17]. The OKVIS system leverages a nonlinear optimization framework. It jointly estimates visual and inertial data within a unified state vector. Suited for high-precision tasks in robotics and UAV (Unmanned Aerial Vehicle) navigation [18]. VINS-Mono implemented a sliding-window optimization strategy. This enabled tight coupling of visual and IMU data. Real-time performance maintained, positioning accuracy and map reuse capabilities substantially improved [19]. Its team later launched VINS-Fusion. This supports multi-camera and multi-IMU fusion, broadening application adaptability [20]. AirSLAM combines CNN-extracted point-line features with a lightweight re-localization module. Effectiveness increased significantly in low-light and texture-poor scenarios [21]. MLINE-VINS integrates Manhattan-world assumptions and line feature constraints into visual-inertial architecture. Notably enhances stability and mapping precision in elongated structures [22].

In multi-sensor fusion SLAM, current research emphasizes integrating LiDAR for high-precision ranging, cameras for rich texture cues, and IMUs for high-rate motion compensation, thereby improving robustness in dynamic environments. Representative studies include MSCKF [23], which pioneered joint visual–inertial state estimation, and V-LOAM [24], which achieved complementary fusion of visual and LiDAR odometry. LVI-SAM [25] adopts factor-graph optimization to tightly couple visual–inertial and LiDAR–inertial subsystems, significantly improving initialization and relocalization performance. Subsequent frameworks, such as R2LIVE and R3LIVE [26,27], further integrate visual, inertial, and LiDAR measurements to enable collaborative reconstruction of geometric structure and texture. In addition, FAST-LIVO [28] addresses cross-modal dimensional mismatch by performing direct image–point-cloud alignment via a sequential update mechanism, leading to improved fusion accuracy. Building upon this work, FAST-LIVO2 [29] further optimizes the sparse-direct fusion pipeline for resource-constrained platforms, introducing efficient memory management and lightweight computation strategies, enabling real-time high-precision state estimation on edge devices while retaining strong robustness in degraded perception conditions. Meanwhile, DaLiTI [30] proposes a degradation-aware LiDAR–thermal–inertial fusion framework, which dynamically adjusts the fusion weights of LiDAR and thermal measurements based on real-time modal quality, effectively addressing the simultaneous failure of visual and LiDAR sensors in extreme degraded scenarios such as fire scenes and chemical plant gas leaks.

1.2 Main Contributions

Conventional state-of-the-art SLAM systems exhibit severe performance degeneration in tunnel scenarios, which are typified by sparse visual textures, uneven low illumination, and repetitive structural layouts, manifesting as critical failures including loop closure detection mismatch, persistent pose drift, and unstable feature-based odometry output owing to the lack of tunnel-specific architectural optimizations and insensitivity to harsh environmental disturbances. Targeting these inherent limitations, this paper presents a tightly coupled image-enhanced visual odometry factor graph optimization SLAM framework that fuses multi-modal data from LiDAR, vision, and IMU, with three core customized innovations that yield non-universal performance gains particularly effective in tunnel environments. First, to mitigate feature instability caused by severe illumination fluctuations, a ABA-CLAHE (Adaptive Brightness Adjustment-Contrast Limited Adaptive Histogram Equalization) image enhancement is tailored to tunnel lighting characteristics, which dynamically calibrates contrast enhancement parameters to suppress noise and amplify weak structural features, significantly improving the robustness and quantity of extractable visual keypoints in low-light tunnel segments; second, to address high mismatch rates and localization failure in texture-sparse tunnel regions, a dual-strategy feature matching mechanism integrating SURF (Speeded Up Robust Features) descriptors and geometry-constrained RANSAC (Random Sample Consensus) filtering is implemented, which harnesses linear structural priors of tunnels to refine feature association and eliminate outliers, enabling high-precision feature correspondence in monotonous tunnel scenes; third, to suppress cumulative drift and reduce loop closure errors, a tightly coupled multi-source factor graph optimization model is constructed that integrates IMU preintegration factors, LiDAR odometry factors, visual odometry factors, and tunnel-adapted loop closure constraints, with dynamic weight tuning for sensor uncertainty in tunnel environments, thereby minimizing false loop constraints and attenuating global pose drift. Collectively, these scene-customized designs address major limitations of generic SLAM systems in tunnels, realizing a high-precision, low-drift, and robust multi-sensor fusion SLAM solution tailored for underground tunnel applications.

The paper is organized as follows: [Section 1](#) explains the research importance and challenges in tunnel environments, and summarizes existing work in vision SLAM. [Section 2](#) details the proposed

method. Section 3 validates the effectiveness of our work through experimental results. Section 4 summarizes the full paper and points out the limitations of current methods.

2 Methods

2.1 Feature Point-Based Visual Odometry System

Given the high requirements for preserving structural and texture information in 3D mapping for municipal tunnels—and the relaxed real-time constraints—the visual odometry system relies on inter-frame motion estimation from visual images to achieve 3D mapping. Inter-frame estimation typically depends on accurate feature point extraction and matching. Therefore, a feature point-based approach is adopted in the visual odometry system. After image enhancement, keypoints are extracted and matched between image frames. Once pairwise-matched keypoints are obtained between images, camera pose is estimated. Then depth information of 3D points is derived via triangulation, generating a sparse point cloud. Finally, BA (Bundle Adjustment) optimization is applied for local refinement, addressing cumulative errors from increasing images. The system block diagram is shown in Fig. 1.

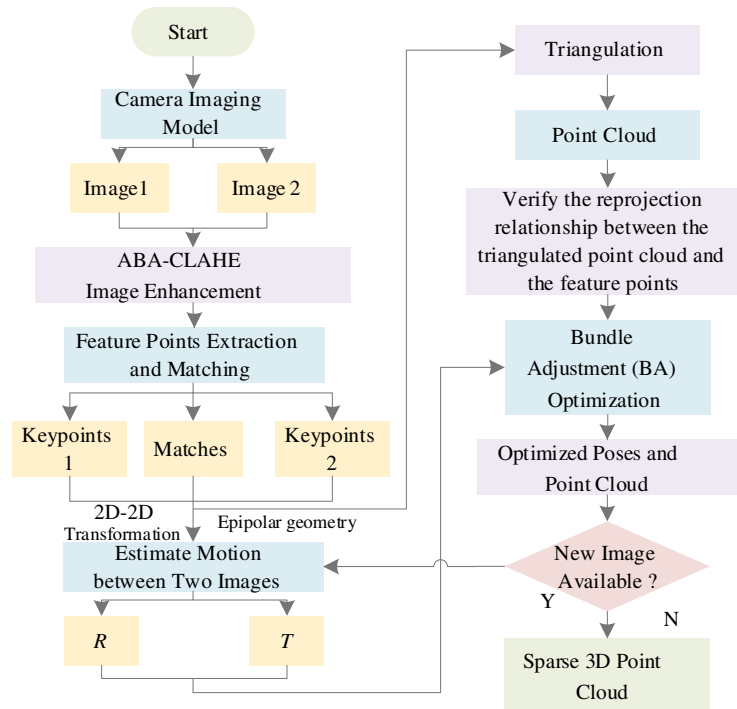


Figure 1: Block diagram of visual odometer system.

2.1.1 ABA-CLAHE Image Enhancement Algorithm

Image preprocessing is fundamental for 3D mapping systems and proves critical in municipal tunnel environments characterized by complex lighting and homogeneous textures. Our primary objective is to enhance image clarity and contrast while improving feature point stability and distribution density to support subsequent 3D mapping. HDR (High Dynamic Range) imaging is not used since motion blur and ghosting occur during robot movement, and HDR may over-smooth fine structural details needed for tunnel inspection. Three common image enhancement algorithms—histogram equalization, gamma transformation, and

CLAHE (Contrast-Limited Adaptive Histogram Equalization)—were applied to municipal tunnel images under significant illumination variations. Results are shown in Fig. 2.

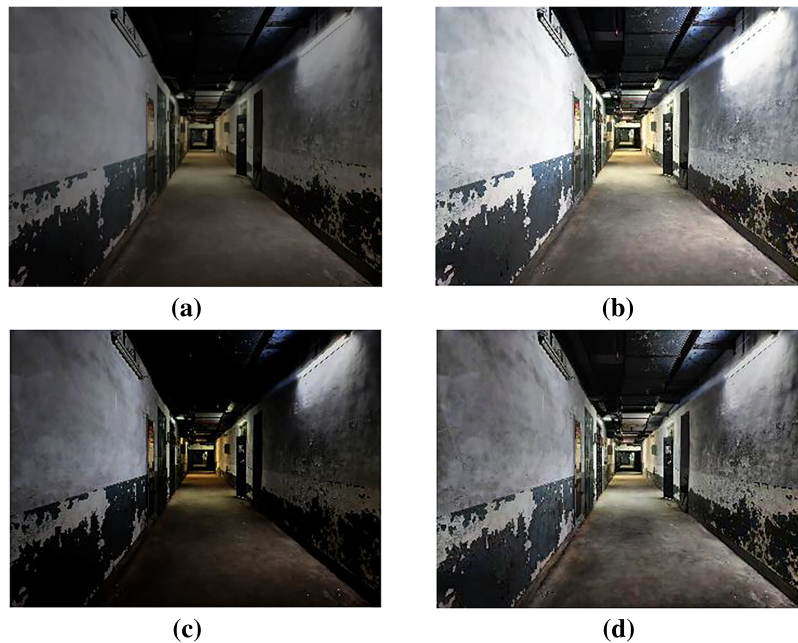


Figure 2: Image enhancement algorithm results. (a) Original image; (b) histogram equalization; (c) gamma transformation; (d) CLAHE.

Analysis of results shows that under low-light conditions, histogram equalization effectively enhances overall image contrast, particularly improving detail visibility in dark regions. However, this method causes over-enhancement in bright areas, leading to overexposure and loss of image details. Gamma transformation moderately improves global contrast and sharpness but delivers suboptimal enhancement under non-uniform illumination with unnatural brightness handling. In comparison, CLAHE demonstrates superior performance in local contrast enhancement, significantly accentuating textures and details to improve feature discernibility in low-light environments. Nevertheless, CLAHE-processed images may exhibit noticeable intensity boundaries and artifacts in localized regions.

To address these limitations, a two-stage enhancement strategy is implemented:

(1) Adaptive Brightness Adjustment: A preliminary smoothing function is applied to regulate overall luminance distribution, mitigating unnatural transitions caused by abrupt brightness variations.

(2) CLAHE-Based contrast Enhancement: The normalized image undergoes CLAHE processing to optimize local contrast while preserving critical details.

This integrated ABA-CLAHE (Adaptive Brightness Adjustment—CLAHE) method the image preprocessing approach for subsequent feature extraction and 3D mapping, enhancing system robustness and accuracy in low-light environments. The algorithm flowchart is shown in Fig. 3.

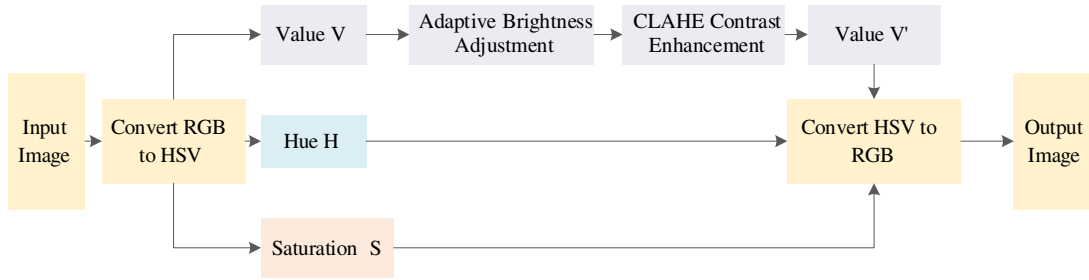


Figure 3: The ABA-CLAHE image enhancement algorithm flowchart.

The HSV (Hue, Saturation, Value) color space characterizes images through three dimensions: Hue (H), Saturation (S), and Value (V). The Value component is independent of the Hue and Saturation components. Processing the Value channel does not affect the other two components. Therefore, image brightness can be corrected by adaptively adjusting the extracted Value component.

For nonlinear brightness adjustment targeting low-light and overexposed conditions, the following adaptive brightness adjustment function is proposed:

$$V_{mid} = \frac{\sum_{x=1}^n \sum_{y=1}^m V(x, y)}{m * n}, \quad (1)$$

$$V(x, y) = V_{mid} * V(x, y)^\alpha + (1 - V_{mid}) * V(x, y)^\beta, \quad (2)$$

where:

n, m —Image width and height,

V_{mid} —mean brightness value of the entire image,

$V(x, y)$ — V -component of point (x, y) ,

α, β —low-light and high-light brightness correction coefficients, through controlled variable experiments conducted specifically in low-illumination and uneven lighting conditions of municipal tunnels, the optimal combination of image enhancement and SLAM feature performance is selected as $\alpha = 0.25, \beta = 0.4$

Fig. 4 shows the adaptive brightness adjustment results of the Value (V) channel for Fig. 2a.



Figure 4: Adaptive brightness adjustment results of the value (V) channel. (a) HSV image; (b) value (V) channel; (c) adaptive value (V) adjustment.

As seen in Fig. 4, the brightness adjustment function imposes certain limitations on the original image's luminance range during mapping, resulting in uneven grayscale distribution and consequently affecting the enhancement of image contrast. To address this, the CLAHE algorithm is used to perform contrast enhancement and brightness adjustment on the original image, thereby improving its overall visual quality. Fig. 5 shows the comparative results after processing the original image (Fig. 2a) with proposed method and other algorithms, which include CLAHE, Detail preserving noise-aware retinex model, ABC (Artificial Bee Colony) optimized image enhancement and Optimized Bézier curve-based intensity mapping. The results indicate that our method effectively enhances texture details in the ceiling areas of municipal tunnel environments while maintaining balanced overall brightness distribution.

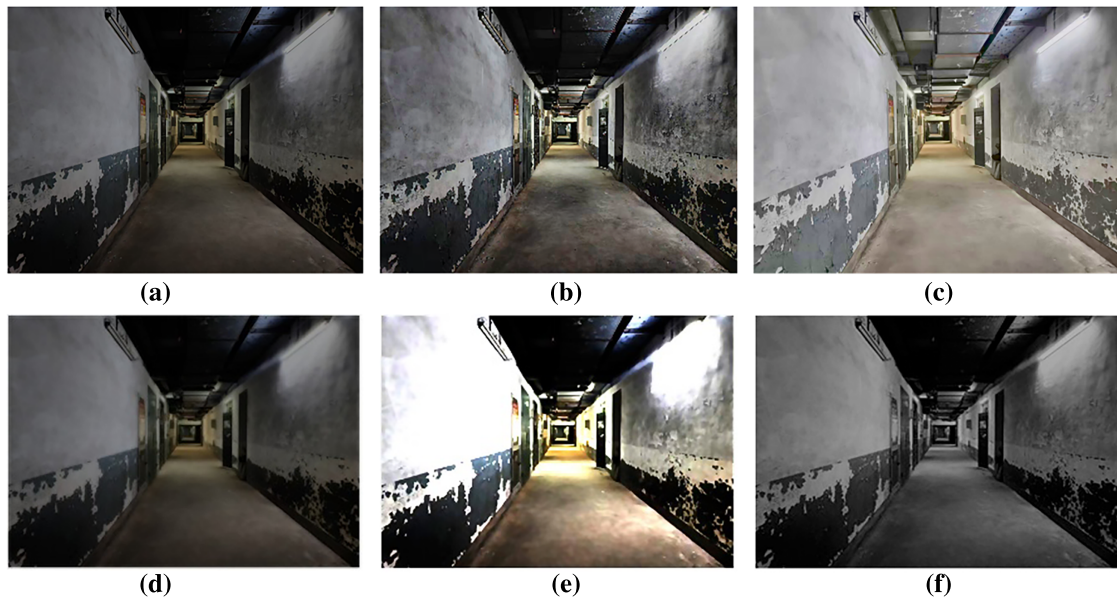


Figure 5: Image enhancement processing results (a) original image; (b) CLAHE; (c) ABA-CLAHE; (d) detail preserving noise-aware retinex model; (e) ABC optimized image enhancement; (f) optimized Bézier curve-based intensity mapping.

Subsequently, the same positions of the results in Fig. 5 are magnified. The magnification region (based on the original image) is located at $x = 138$, $y = 10$ with a width of 100 and a height of 77, and the magnification factor is set to 3.

As shown in Fig. 6a–f, Fig. 6a is generally dark with insufficient details and contrast. Fig. 6b shows limited enhancement, incomplete recovery of dark-region details, and weak scene layering. Fig. 6d achieves a moderate brightness improvement, but the identifiability of key structures such as ceiling pipelines remains insufficient. Fig. 6e suffers from excessive enhancement, resulting in severe overexposure in the right region, complete loss of details, and introduction of obvious noise and artifacts, which noticeably degrades realism of the scene. Fig. 6f brightens dark regions but still lacks adequate overall contrast and detail clarity. Fig. 6c represents the best-performing result. It greatly improves brightness while substantially restores ceiling pipelines and wall textures, enhances the three-dimensional sense of the scene without overexposure or color shift, and achieves the best balance among detail recovery, contrast improvement, and realism preservation.

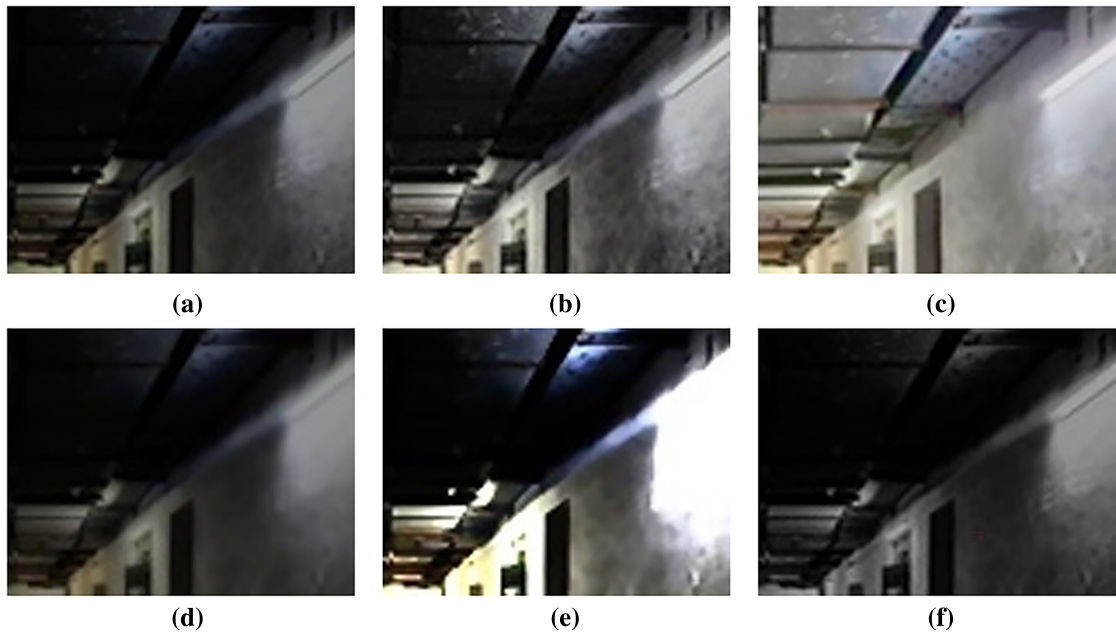


Figure 6: Local magnified views of the results for each enhancement algorithm (a) original image; (b) CLAHE; (c) ABA-CLAHE; (d) detail preserving noise-aware retinex model; (e) ABC optimized image enhancement; (f) optimized Bézier curve-based intensity mapping.

PSNR (Peak Signal-to-Noise Ratio), SSIM (Structural Similarity Index), Entropy and STD (Standard Deviation) are used as objective evaluation indicators to measure the distortion degree, structure retention, detail richness and contrast improvement effect of enhanced images respectively. Among them, the higher the PSNR, the closer the image is to the original image and the smaller the distortion. In many image-processing studies, the effect is good when it is higher than 25 dB, and there is almost no distortion when it is higher than 30 dB. SSIM is used to measure the similarity of structure, brightness, contrast and human visual perception and values closer to 1 indicate better structural similarity. Entropy is used to measure the richness of image details. The larger the entropy after enhancement, the richer the details. STD is used to measure image contrast. The larger the value, the better the contrast and the clearer the visual effect. The results are presented in [Table 1](#).

Table 1: Comparison of PSNR, SSIM, Entropy, and STD of various algorithms in low-light scenes.

	PSNR (dB)	SSIM	Entropy	STD
CLAHE	30.03	0.7550	7.1659	41.9088
ABA-CLAHE	38.93	0.8336	7.5705	58.2504
Detail Preserving Noise-Aware Retinex Model	32.58	0.9677	7.1624	47.6289
Artificial Bee Colony Optimized Image Enhancement	8.34	0.5996	6.9110	96.5895
Optimized Bézier Curve-Based Intensity Mapping	35.02	0.9858	7.0316	51.1372

As shown in [Table 1](#), by comparing five image enhancement algorithms, the ABA-CLAHE algorithm performs best in fidelity (PSNR = 38.93 dB), detail richness (Entropy = 7.5705) and contrast (STD = 58.2504), while maintaining good structural similarity (SSIM = 0.8336), achieving the the most balanced performance. The Optimized Bézier Curve-Based Intensity Mapping algorithm is superior in structural similarity (SSIM = 0.9858), which is suitable for scenes with high requirements for structural integrity. The Artificial Bee Colony Optimized Image Enhancement algorithm causes severe distortion due to excessive contrast improvement, and thus has limited practical application value.

Additionally, these algorithms are compared along three dimensions: detail retention mechanism, noise control strategy and computational complexity. First, Retinex-based models enhance details through illumination decomposition, which tends to over-smooth tiny textures (such as fine cracks) in tunnel images. ABA-CLAHE first realizes global brightness equalization through adaptive nonlinear adjustment of the V component in HSV space, and then highlights texture details via local contrast optimization of CLAHE. It is more targeted in preserving fine textures of tunnel structures and meets the requirements of SLAM feature extraction. Bionic optimization algorithms (ABC, Bézier curve) realize brightness mapping through global optimization, which is easy to lose the relative contrast of local textures. The global-local cascade strategy of ABA-CLAHE can balance overall brightness and local details. Second, although the noise-aware Retinex model introduces a noise suppression module, it still amplifies inherent image noise under very low-light conditions (e.g., illuminance < 50 lx). ABA-CLAHE introduces brightness smoothing constraints in the adaptive brightness adjustment stage to avoid excessive enhancement of dark-region noise. Meanwhile, with a CLAHE clip limit of 2.0, CLAHE can effectively suppress noise amplification in local regions, showing better noise control performance in low-light tunnel scenes. Bionic optimization algorithms have no special noise control mechanism and tend to enhance noise as “details” during optimization, resulting in false feature points in subsequent feature extraction. Finally, Retinex-based models involve complex steps such as illumination decomposition, Gaussian filtering and noise estimation, with a time complexity of $O(W \times H \times k)$ (k is the size of the filtering kernel). Bionic optimization algorithms such as ABC and Bézier curve involve iterative optimization, with a time complexity of $O(N \times W \times H)$ (N is the number of iterations) and high computational cost. ABA-CLAHE adopts pure pixel-level operations without iteration or complex filtering, with a time complexity of $O(W \times H)$. Its computational efficiency is much higher than the above advanced algorithms, making it more suitable as an image preprocessing module for the SLAM front end to achieve efficient connection with subsequent feature extraction and matching steps.

2.1.2 Dual-Strategy Feature Point Extraction and Matching

In municipal tunnel environments, feature point extraction and matching in images encounter significant challenges due to factors such as uneven illumination and complex textures. To address demanding conditions including low-light situations and texture scarcity, it is essential to employ feature extraction algorithms with enhanced robustness and accuracy. This ensures reliable feature detection and stable matching, thereby improving both positioning and mapping precision of visual 3D mapping systems in complex environments while increasing their adaptability in dynamic and uncertain conditions.

For comparative analysis, three classical feature-based methods are evaluated: SIFT (Scale-Invariant Feature Transform), ORB (Oriented FAST and Rotated BRIEF), and SURF. Based on [Fig. 5](#), this study analyzes the feature point extraction and matching results under normal illumination, uneven illumination, and uneven illumination after ABA-CLAHE enhancement. A comparison of the number of extracted feature points and processing time for each algorithm is presented in [Table 2](#).

Table 2: The number of extracted feature points and processing time for each algorithm under different conditions.

Algorithm	Conditions	Points	Time/s
SIFT	Illumination	733	0.4886
	Uneven Illumination	370	0.6936
	Uneven Illumination after ABA-CLAHE	559	0.5911
ORB	Illumination	500	0.0470
	Uneven Illumination	500	0.1054
	Uneven Illumination after ABA-CLAHE	481	0.0761
SURF	Illumination	2344	0.5392
	Uneven Illumination	1202	0.2312
	Uneven Illumination after ABA-CLAHE	1877	0.3827

A comparative analysis of SIFT, ORB, and SURF algorithms under controlled matching point conditions (fixed at 400 points) are conducted, evaluating both matching time and accuracy rates across varying thresholds. The test results under three conditions are presented in the [Tables 3–5](#).

Table 3: Matching time and accuracy of different thresholds of feature points under sufficient illumination.

Threshold	Metrics	SIFT	ORB	SURF
100	Number of Correct Matches	327	331	372
	Accuracy Rate (%)	81.75	82.75	93
	Matching Time (s)	0.004	0.002	0.009
150	Number of Correct Matches	344	340	385
	Accuracy Rate (%)	86	85	96.25
	Matching Time (s)	0.004	0.001	0.009

Table 4: Matching time and accuracy of different thresholds of feature points under uneven illumination.

Threshold	Metrics	SIFT	ORB	SURF
100	Number of Correct Matches	261	315	361
	Accuracy Rate (%)	70.35	78.75	90.25
	Matching Time (s)	0.0001	0.0001	0.0004
150	Number of Correct Matches	273	323	373
	Accuracy Rate (%)	73.58	80.75	93.25
	Matching Time (s)	0.0001	0.0001	0.0003

Table 5: Matching time and accuracy of different thresholds under uneven illumination & ABA-CLAHE.

Threshold	Metrics	SIFT	ORB	SURF
100	Number of Correct Matches	311	324	365
	Accuracy Rate (%)	77.75	81.00	91.25
	Matching Time (s)	0.0001	0.0001	0.0004
150	Number of Correct Matches	323	335	378
	Accuracy Rate (%)	80.75	83.75	94.50
	Matching Time (s)	0.0002	0.0001	0.0003

The experimental results show that uneven illumination can significantly impact both feature point extraction and matching procedures in feature matching algorithms. Under uneven illumination conditions, the performance of subsequent feature point extraction and matching with the ABA-CLAHE algorithm is significantly better than that of images without the ABA-CLAHE algorithm, and is close to the results under sufficient illumination.

From the experimental data, it can be observed that the SURF algorithm, with a variety of thresholds, will extract the most feature points while still managing to maintain relatively good matching accuracy. This makes it particularly suitable for structural environments such as municipal tunnels that have cracks and fissures. SURF feature detection based on the Hessian matrix exhibits much stronger robustness to blur and illumination variations than binary descriptors such as ORB. In contrast, learning-based descriptors require a large amount of labeled data in tunnel scenarios, resulting in poor generalization across different tunnel engineering deployments. Moreover, as the 3D mapping of the tunnel environment does not need real-time performance, the SURF algorithm—though it features higher computational cost in feature extraction and matching—can achieve stable performance under shaky conditions while also providing strong robustness.

Lighting conditions in municipal tunnels are generally poor, with particularly pronounced uneven illumination. To further improve the accuracy of feature matching, a mismatched feature rejection strategy is proposed that combines SURF-based coarse matching with the RANSAC algorithm. This approach enhances the stability and precision of matching results by eliminating incorrect matching points.

The RANSAC algorithm estimates optimal model parameters through iterative computations in the presence of outlier data. Inliers are data points that conform to the optimal model, while outliers are points that do not fit the model, such as noise, abnormal values with large matching errors, or outliers in curve estimation. Therefore, the RANSAC algorithm also functions as an outlier detection technique.

The SURF-RANSAC algorithm has the following key steps:

- (1) Random Sampling: Select a minimal random subset from the data to initialize model fitting.
- (2) Model Estimation: Compute preliminary model parameters using this subset.
- (3) Inlier Identification: Evaluate all data points against the model, classifying points with errors below a defined threshold as inliers.
- (4) Model Evaluation: Record both the quantity of inliers and their distribution quality for the current model.
- (5) Iterative Refinement: Repeat the process until either reaching maximum iterations or achieving a satisfactory model.
- (6) Optimal Model Selection: Finalize the model demonstrating the highest inlier consensus.

RANSAC is adopted to remove outliers from SURF feature matches, thereby improving matching accuracy. The resulting high-precision inlier matches provide reliable constraints for computing epipolar geometry, leading to a more accurate fundamental matrix and relative camera pose. This, in turn, refines the triangulation of 3D scene points and enhances the localization and mapping accuracy of visual odometry, which is critical for stabilizing pose estimation in weakly textured tunnel environments. The total number of feature points remains 400, and the quality of these feature points is presented in [Table 6](#).

Table 6: Quality of image feature points before and after the SURF-RANSAC.

Threshold	Metrics	SURF	SURF-RANSAC
100	Number of Correct Matches	365	343
	Accuracy Rate (%)	91.25	85.75
	Inlier Ratio	–	94.00
	Matching Time (s)	0.0004	0.0005
150	Number of Correct Matches	378	355
	Accuracy Rate (%)	94.50	88.75
	Inlier Ratio	–	94.00
	Matching Time (s)	0.0003	0.0004

After obtaining pairwise matched feature points between the captured images, the rotation and translation matrices between the two images are computed by applying the principles of epipolar geometry. After determining the relative pose between two adjacent image frames, triangulation is utilized to recover the three-dimensional depth information of the scene.

2.2 Backend Graph Optimization for Multi-Sensor 3D Mapping

3D mapping systems typically employ a frontend-backend architecture. Sensor data is processed, features are extracted, and initial state estimation as well as local map construction are performed by the frontend. Conversely, the backend is primarily responsible for trajectory optimization, loop closure processing, and the enforcement of global consistency. Traditional filtering methods often prove inadequate in complex environments, particularly when large-scale datasets or repetitive structures such as municipal tunnels are being handled. This inadequacy stems from their limited capacity to maintain global consistency. In contrast, factor graph optimization excels in these challenging scenarios by efficiently managing large constraint sets and preserving spatial coherence across long-distance mapping operations, making it particularly suitable for infrastructure inspection and large-scale surveying applications where precise global alignment is critical.

Our system employs a factor graph that fuses multi-sensor observations from LiDAR, monocular cameras, and IMUs to estimate robot trajectories. The multi-sensor fusion framework of our factor graph is depicted in [Fig. 7](#).

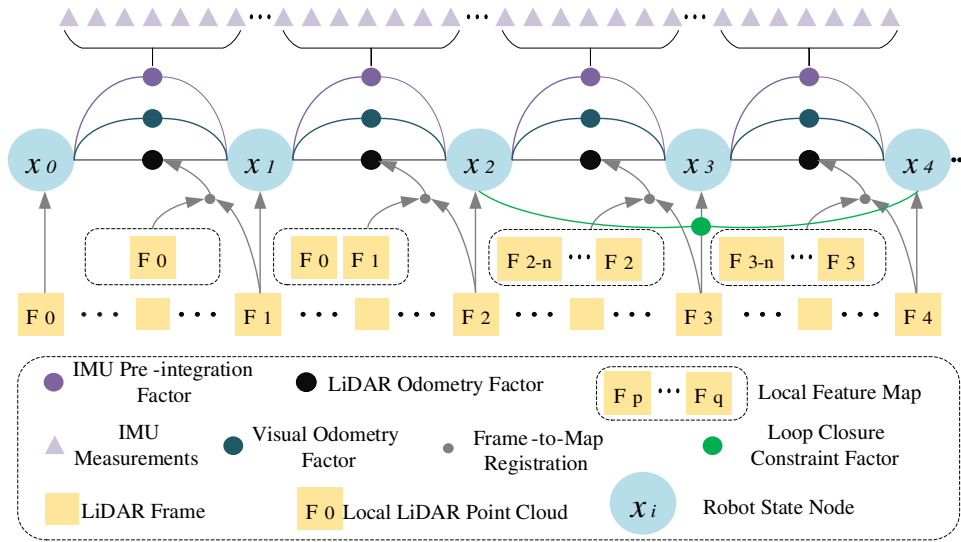


Figure 7: Multisensor fusion factor graph framework.

Our 3D mapping system employs factor graph optimization to address the scalability challenges posed by continuous robot navigation in municipal tunnels. As the robot moves, accumulating nodes and factors create progressively larger optimization problems. To maintain computational efficiency at scale while ensuring global consistency, specialized sparse matrix computation techniques are implemented.

The framework’s core innovation lies in its systematic construction of factor nodes that precisely encode multi-sensor constraints. These nodes integrate heterogeneous measurements including IMU readings, LiDAR point cloud matches, and inter-frame pose variations. Using sensor-specific nonlinear models, the system computes observation-prediction residuals and rigorously quantifies uncertainties through covariance matrices.

The optimization backbone comprises four specialized factor types working in concert: IMU pre-integration factors maintain inertial constraints, LiDAR odometry factors handle point cloud alignment, visual odometry factors process camera-based positioning, and loop closure factors enforce global consistency. Together, these components form a robust objective function that drives the factor graph optimization process.

(1) IMU pre-integration factors

The IMU provides accelerometer and gyroscope measurements, enabling the derivation of state changes between two adjacent poses. Due to the high frequency of the IMU, directly optimizing the state at each moment is impractical. Therefore, pre-integration theory is employed to reduce computational complexity. The IMU pre-integration factor connects the states of two adjacent moments, i and j , including poses \mathbf{T}_i and \mathbf{T}_j , as well as velocities \mathbf{v}_i and \mathbf{v}_j .

Based on the IMU continuous-time measurement model, the observed values obtained after pre-integration are:

$$\Delta \mathbf{R}_{ij} = \prod_{k=i}^{j-1} e^{((\boldsymbol{\omega}_k - \mathbf{b}_g) \Delta t_k)}$$

$$\Delta \mathbf{v}_{ij} = \sum_{k=i}^{j-1} \mathbf{R}_k (\mathbf{a}_k - \mathbf{b}_a) \Delta t_k$$

$$\Delta \mathbf{p}_{ij} = \sum_{k=i}^{j-1} \mathbf{v}_k \Delta t_k + \frac{1}{2} \mathbf{R}_k (\mathbf{a}_k - \mathbf{b}_a) \Delta t_k^2, \quad (3)$$

where $\Delta \mathbf{R}_{ij}$, $\Delta \mathbf{v}_{ij}$ and $\Delta \mathbf{p}_{ij}$ represent the changes in rotation, velocity, and position. Δt_k is the time interval between the k -th sampling and the $(k + 1)$ -th sampling.

The error function is:

$$\mathbf{e}_{\text{IMU}} = \begin{bmatrix} \log \left(\Delta \mathbf{R}_{ij}^T \mathbf{R}_i^T \mathbf{R}_j \right) \\ \mathbf{v}_j - \left(\mathbf{v}_i + \mathbf{g} \Delta t + \mathbf{R}_i \Delta \mathbf{v}_{ij} \right) \\ \mathbf{p}_j - \left(\mathbf{p}_i + \mathbf{v}_i \Delta t + \frac{1}{2} \mathbf{g} \Delta t^2 + \mathbf{R}_i \Delta \mathbf{p}_{ij} \right) \end{bmatrix}, \quad (4)$$

where Δt is the time interval between two frames.

(2) LiDAR odometry factors

The constraint on the laser pose transformation $\mathbf{T}_{ij}^{\text{Lidar}}$ between adjacent frames is defined, with the observation model as:

$$\mathbf{T}_{ij}^{\text{Lidar}} = \mathbf{T}_i^{-1} \cdot \mathbf{T}_j. \quad (5)$$

The error function is:

$$\mathbf{e}_{\text{Lidar}} = \log \left(\left(\mathbf{T}_{ij}^{\text{Lidar}} \right)^{-1} \cdot \mathbf{T}_i^{-1} \cdot \mathbf{T}_j \right). \quad (6)$$

(3) visual odometry factors

The relative transformation $\mathbf{T}_{ij}^{\text{Visual}}$ between keyframes is constrained using visual feature matching results, with the observation model as:

$$\mathbf{T}_{ij}^{\text{Visual}} = \mathbf{EstimateFromFeatures} (i, j). \quad (7)$$

The error function is:

$$\mathbf{e}_{\text{Visual}} = \log \left(\left(\mathbf{T}_{ij}^{\text{Visual}} \right)^{-1} \cdot \mathbf{T}_i^{-1} \cdot \mathbf{T}_j \right). \quad (8)$$

(4) loop closure factors

The constraint between historical frames and the current frame is established for global error correction. Loop closure detection identifies loop frame k and obtains its relative pose, with the observation model as:

$$\mathbf{T}_{ik}^{\text{Loop}} = \mathbf{ComputeRelativePose} (i, k). \quad (9)$$

The error function is:

$$\mathbf{e}_{\text{Loop}} = \log \left(\left(\mathbf{T}_{ik}^{\text{Loop}} \right)^{-1} \cdot \mathbf{T}_i^{-1} \cdot \mathbf{T}_k \right). \quad (10)$$

The ultimate goal of factor graph optimization is to minimize the weighted sum of squares of all error terms, with the objective function as

$$\min_{\{\mathbf{T}_i, \mathbf{v}_i, \mathbf{b}_i\}} \sum \|\mathbf{e}_{\text{IMU}}\|_{\Sigma_{\text{IMU}}}^2 + \sum \|\mathbf{e}_{\text{LiDAR}}\|_{\Sigma_{\text{LiDAR}}}^2 + \sum \|\mathbf{e}_{\text{vision}}\|_{\Sigma_{\text{vision}}}^2 + \sum \|\mathbf{e}_{\text{loop}}\|_{\Sigma_{\text{loop}}}^2. \quad (11)$$

In multi-sensor fusion-based 3D mapping systems, the continuous accumulation of sensor data leads to rapid expansion of the factor graph's scale. Traditional batch-based global optimization methods (e.g., Gauss-Newton or Levenberg-Marquardt algorithms) would incur excessive computational resource consumption, failing to meet real-time requirements. To solve this problem, the iSAM2 [31] optimizer from GTSAM is employed as the core backend optimization module.

In our proposed 3D mapping system, the iSAM2 optimizer incrementally processes factors from three sensor subsystems: visual, LiDAR, and IMU. The backend optimization module integrates optimized pose nodes with IMU pre-integration factors into the factor graph, leveraging both visual local optimization and LiDAR-IMU loop closure detection. Through a sliding window mechanism, the system maintains real-time operation while preserving map consistency and precision.

As shown in Fig. 8, a simple factor graph structure is presented, while Fig. 9 demonstrates its variable elimination process.

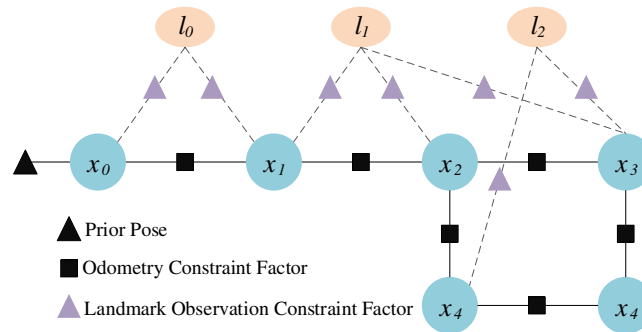


Figure 8: Simple factor graph structure.

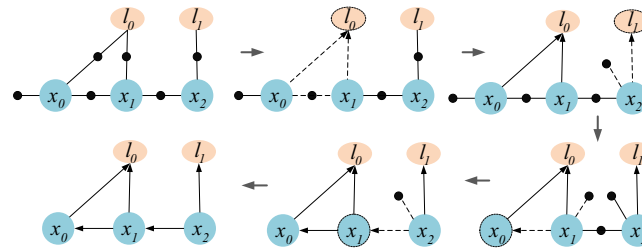


Figure 9: Schematic diagram of factor graph elimination.

The variable elimination procedure converts the factor graph into a Bayesian network by removing all connected factor nodes, enabling joint probability density estimation. Through sequential application of the chain rule, conditional probability distributions are derived, reformulating the problem as nonlinear least-squares minimization. The system then integrates the computed pose transformations by incrementally registering point cloud frames to the global map, achieving consistent 3D reconstruction.

3 Experiments and Analysis

Our multi-sensor 3D mapping system is developed under Ubuntu 20.04 with ROS Noetic, to achieve a unified framework for multi-source heterogeneous sensor data acquisition, transmission, and management. The experimental platform, as shown in Fig. 10, is a tunnel inspection robot equipped with three key sensors:

a 16-line LiDAR, an industrial camera, and a high-precision IMU. No wheel odometry is used. The sensor and mainboard models are listed in [Table 7](#).

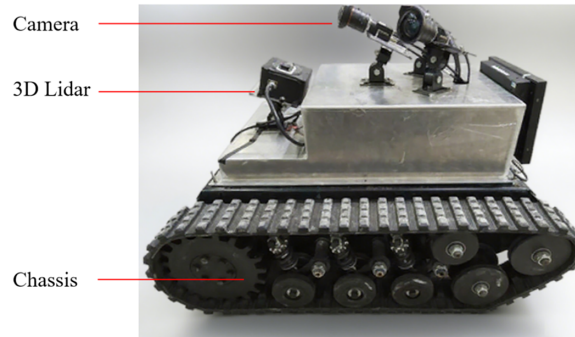


Figure 10: Physical prototype of the municipal tunnel inspection robot.

Table 7: Sensors and mainboard models equipped on the mobile robot.

Parts	Model
Camera	Hikvision MV-CS050-10UC
LiDAR	LIVOX HAP
IMU	WitMotion HWT901B
Mainboard	NVIDIA Jetson Orin NX 16 GB

The system employs a software-based synchronization scheme to ensure a consistent temporal reference for multi-sensor data. All sensor drivers run on the Ubuntu 20.04 operating system, and the time synchronization mechanism provided by the ROS framework is used to assign unified timestamps to the collected data. Specifically, the LiDAR driver automatically attaches a ROS timestamp when parsing each frame of point cloud data to mark the generation time of the data; the IMU module continuously reads information such as acceleration, angular velocity, and quaternions through the serial communication protocol, and embeds the current ROS system timestamp during the reading process; The camera driver acquires buffered image frames via the MVS SDK and parses frame metadata. Accurate timestamps are then assigned to each frame to align image data with other sensor streams.

Experiments are conducted in a municipal utility tunnel located in Xi'an, China. The tunnel is a long, straight, enclosed underground corridor with low illumination, no GPS signal, and homogeneous concrete structures.

In municipal tunnel environments, monocular camera images are first used to generate sparse point clouds, which are then fused with LiDAR and IMU data to produce dense 3D maps. [Fig. 11](#) demonstrates the sparse reconstruction results under low-light tunnel conditions, where green points represent the camera trajectory.

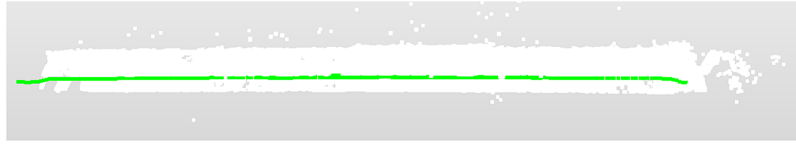


Figure 11: Sparse 3D mapping results in low-light municipal tunnel environments.

The results of 3D mapping using multi-sensor fusion in municipal tunnel environments are shown in Fig. 12.

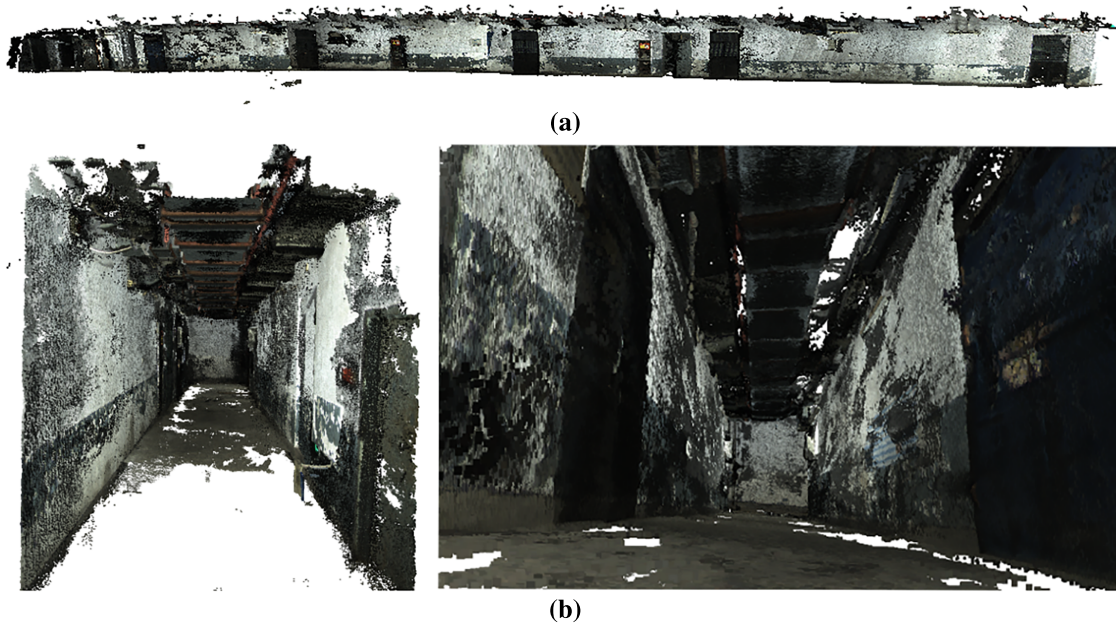


Figure 12: 3D mapping results using multi-sensor fusion. (a) Side view; (b) internal details.

A Leica TS16 total station is used to measure the tunnel environment under sufficient lighting conditions. The measurement accuracy of the Leica TS16 within its measuring range is at the millimeter level; therefore, the trajectory measured by the total station is regarded as the ground truth in this study. Subsequently, the robot is driven along the same route and tested using three algorithms, namely R2LIVE, LVI-SAM, and the proposed method, in the above environment. In addition, ablation studies are conducted by respectively removing the ABA-CLAHE algorithm and the SURF-RANSAC algorithm for further comparison. The test results, including the 3D trajectories and trajectories along the main motion direction, are shown in Fig. 13.

As shown in Fig. 13, the trajectories of R2LIVE, LVI-SAM and the ablation model without the ABA-CLAHE module show larger fluctuations than the proposed method. In the low-light tunnel environment, the lack of image enhancement leads to poor image quality, fewer effective feature points and weakened visual constraints, so the system depends more on IMU and LiDAR and produces obvious trajectory jitter. For the ablation experiment without the SURF-RANSAC module, the cumulative error is small in the early stage but gradually increases with time, resulting in severe trajectory drift in the later stage, which makes the estimated trajectory deviate significantly from the ground truth and the actual travel distance.

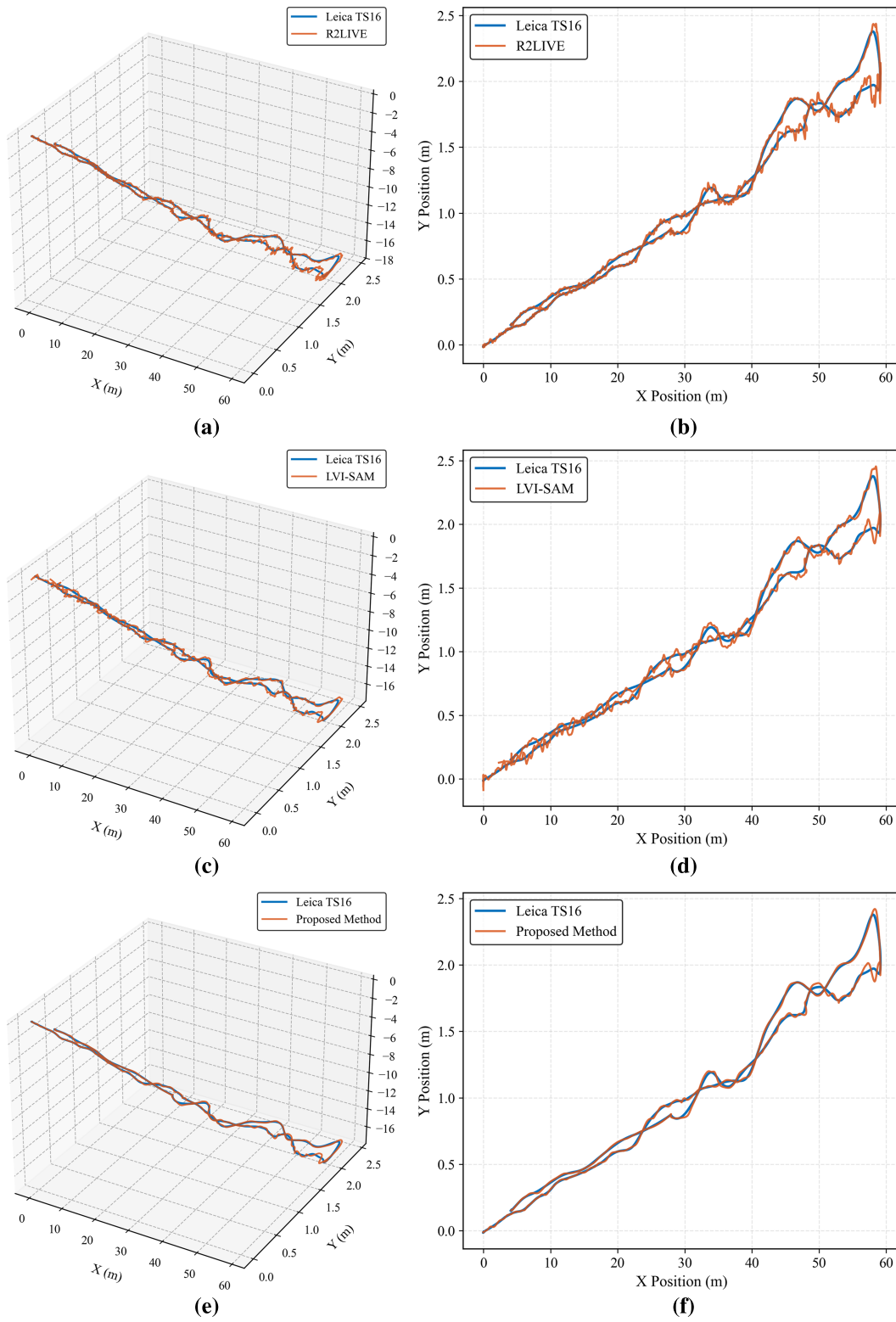


Figure 13: (Continued)

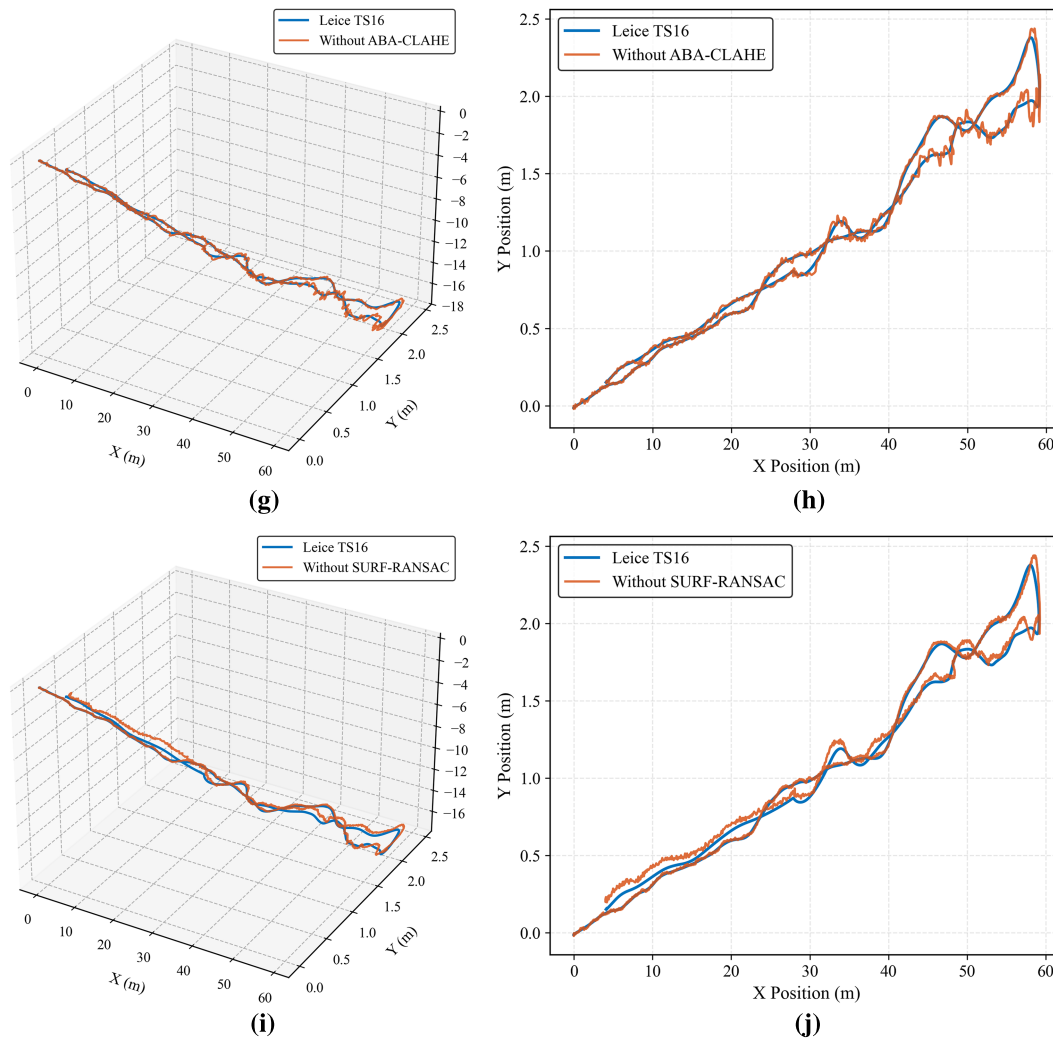


Figure 13: Comparison chart of each algorithm trajectory. (a) Global trajectory of R2LIVE; (b) XY-plane trajectory of R2LIVE; (c) global trajectory of LVI-SAM; (d) XY-plane trajectory of LVI-SAM; (e) global trajectory of proposed method; (f) XY-plane trajectory of proposed method; (g) global trajectory in the ablation study without ABA-CLAHE; (h) XY-plane trajectory in the ablation study without ABA-CLAHE; (i) global trajectory in the ablation study without SURF-RANSAC; (j) XY-plane trajectory Global trajectory in the ablation study without SURF-RANSAC.

To quantitatively evaluate the performance of the proposed algorithm, a series of widely accepted metrics are adopted in this work. For the trajectory accuracy assessment in SLAM, APE (Absolute Pose Error) and RPE (Relative Pose Error) are utilized to measure the global consistency and local drift of the estimated trajectory, respectively. The RMSE (Root Mean Square Error) is employed as the core statistical indicator to quantify the overall deviation between the estimated poses and the ground truth. The calculated APE and RPE values of each algorithm are listed in [Table 8](#).

Table 8: APE and RPE values of each algorithm trajectory.

	R2LIVE		LVI-SAM		Proposed Method		Remove ABA-CLAHE		Remove SURF-RANSAC	
	APE	PRE	APE	PRE	APE	PRE	APE	PRE	APE	PRE
Max	0.0496	0.1765	1.0379	0.4445	0.3525	0.0463	1.8379	0.0640	1.9067	0.4576
Mean	0.0951	0.0239	0.0851	0.0124	0.0645	0.0080	0.0851	0.0104	0.0596	0.0109
Median	0.0633	0.0188	0.0561	0.0092	0.0396	0.0054	0.0561	0.0078	0.0452	0.0082
RMSE	0.1290	0.0299	0.1188	0.0208	0.0920	0.0109	0.1122	0.0218	0.0891	0.0198
STD	0.0872	0.0179	0.0878	0.0167	0.0657	0.0074	0.0778	0.0117	0.0663	0.0165

Each algorithm is run five times in the same environment. The mean of total trajectory length, loop closure error, and loop closure error ratio for each algorithm are shown in [Table 9](#).

Table 9: Mean loopback error.

Algorithm	Total Trajectory Length (m)	Loop Closure Error (m)	Loop Closure Error Ratio
R2LIVE	60.651	0.121	0.203%
LVI-SAM	60.419	0.116	0.185%
Proposed Method	60.375	0.096	0.159%
Without ABA-CLAHE	60.528	0.110	0.182%
Without SURF-RANSAC	60.895	0.168	0.276%

The above data demonstrates that in terms of APE and RPE, the RMSE (Root Mean Square Error) values between R2LIVE, LVI-SAM and proposed method are relatively small, indicating that our proposed multi-sensor fusion method achieves good localization and 3D mapping accuracy in real-world municipal tunnel experiments. However, the trajectory comparison reveals that our multi-sensor fusion solution exhibits smoother trends. Combined with the loop closure error data (0.096 m for fusion, with a lower error ratio), this improvement stems from the enhanced perceptual capability enabled by incorporating LiDAR and camera information. Thus, the proposed multi-sensor fusion solution not only performs well in localization and mapping precision but also demonstrates advantages in loop closure detection and optimization, making it suitable for environments with features similar to municipal tunnels.

Ablation experimental results show that after removing the ABA-CLAHE image enhancement module, the total trajectory length increases slightly, and the loop closure error rises from 0.096 to 0.110 m, indicating that this module effectively improves feature quality and reduces cumulative drift. When the SURF-RANSAC mismatch rejection module is removed, the trajectory length is significantly stretched and the loop closure error surges to 0.168 m, demonstrating that this module is critical for suppressing false matches and ensuring positioning accuracy.

The accuracy of the entire multi-sensor fusion system can be evaluated using residual histograms and reprojection residual histograms from input images. These histograms statistically represent error distributions, providing intuitive insights into model reliability and optimization effectiveness. Below is a detailed explanation:

(1) Residual Histogram

Residuals represent the difference between observed values (e.g., image feature points) and model-predicted values (e.g., projected 3D points). This histogram displays the error distribution of all matched

points, with the x-axis indicating pixel-level error and the y-axis showing frequency. Key evaluation criteria include:

(a) Matching Quality: If most residuals cluster within 0–1 pixels, matching accuracy is high.

(b) Outlier Detection: A high frequency of residuals >2 pixels suggests mismatches or dynamic interference.

(c) Robustness: Residuals consistently within 0.3–0.8 pixels and an RMSE of ~1.1 pixels indicate stable error control.

(2) Reprojection Residual Histogram

Reprojection residuals measure the error when optimized 3D points are reprojected onto images, reflecting final model-image consistency. Evaluations focus on:

(a) Mapping Accuracy: Residuals concentrated within 0–1 pixels imply high alignment between the 3D model and camera parameters.

(b) Optimization Efficacy: If >90% of residuals are <1 pixel (with few >3 pixels), bundle adjustment (BA) is effective.

(c) Anomaly Detection: Isolated high residuals may stem from occlusions, dynamic objects, or feature mismatches.

(3) Differences and Relationships

Residual histograms aggregate errors from matching, initial pose estimation, etc., while reprojection residuals solely assess final model precision.

Together, they evaluate mapping quality:

(a) If reprojection errors are significantly smaller than initial residuals, optimization succeeds.

(b) If both remain high, inspect input data or parameter configurations.

During the 3D mapping process, global statistical metrics for all input images are output as shown in [Table 10](#).

Table 10: Global statistics related indicators.

Metric	Global Value	Without ABA-CLAHE	Without SURF-RANSAC
Total Input Images	222	222	222
Successfully Calibrated Images	210	182	197
3D Point Cloud Count	41,980	29,630	76,240
Total Residual Count	576,558	412,360	1,248,930
Trajectory Length (Min)	2	2	2
Trajectory Length (Mean)	13	8	10
Trajectory Length (Max)	84	67	77
Global Reprojection Error	1.10058	1.87426	3.24671

The residual histogram and reprojection residual histogram are shown in Fig. 14.

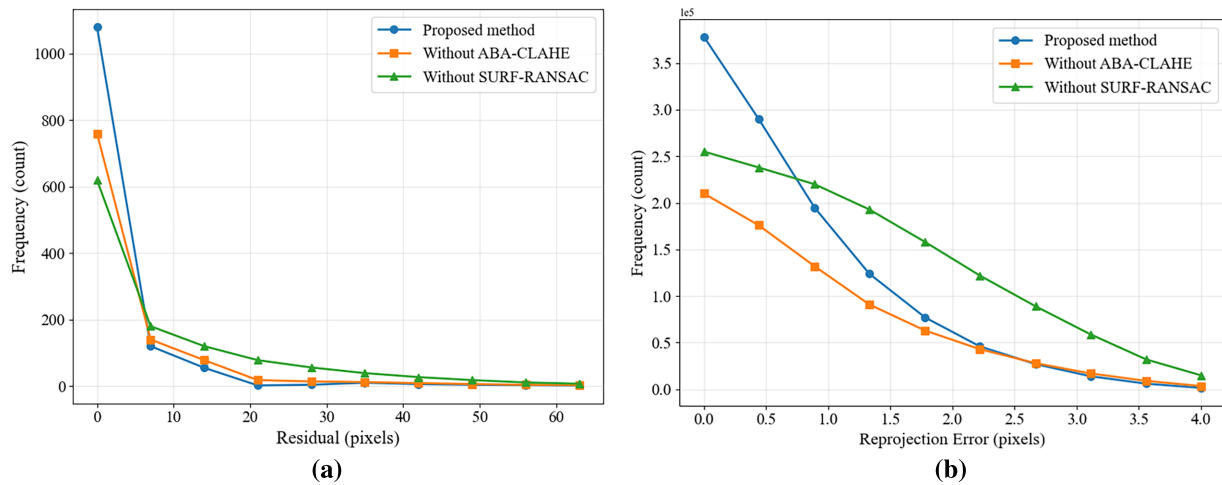


Figure 14: The residual and reprojection residual histogram. (a) The residual histogram; (b) the reprojection residual histogram.

The experimental results demonstrate that the proposed multi-sensor fusion system achieved high-precision 3D reconstruction performance in the municipal tunnel environment. Specifically, among the 222 input images, 210 are successfully calibrated (94.6% success rate), generating 41,980 3D points with a global reprojection error of 1.10058 pixels, indicating excellent overall accuracy. Statistical analysis reveals that over 90% of images maintained a validation rate exceeding 90%, with median single-image residuals concentrated between 0.3–0.8 pixels and mean values ranging from 0.3–1.0 pixels. Notably, more than 90% of residuals are distributed within the 0–1.0 pixel range, confirming the stability and reliability of the feature matching algorithm. The mapping consistency is further supported by trajectory metrics, showing an average of 13 observations per 3D point and a maximum continuous trajectory length of 84 observations. While minor issues are observed (12 uncalibrated images and localized high-residual areas potentially caused by occlusions or dynamic objects), the system maintained robust performance with an RMSE of 1.10 pixels, demonstrating both high precision and strong adaptability for structured environments like municipal tunnels. According to the statistical results of the comparative ablation experiments, after removing ABA-CLAHE, the image quality under low-light conditions deteriorates, resulting in a significant reduction in the number of successfully calibrated images and valid observations. Consequently, the low-error peak of the residual distribution is weakened, and the reprojection error distribution shifts toward higher-error regions as a whole. In contrast, after removing SURF-RANSAC, although the number of matching points and residuals involved in optimization increases significantly, the absence of an effective outlier rejection mechanism leads to a prominent rise in the frequency of medium and high-error intervals, and a more pronounced long-tailed distribution of reprojection errors. This indicates that more low-quality matches are retained in the system, thereby exacerbating trajectory drift and degrading global consistency in the subsequent optimization. These results validate the effectiveness of the multi-sensor fusion approach in challenging underground scenarios.

A total of 222 frames of tunnel scene images were processed in this experiment, among which 12 frames fail to complete effective calibration, accounting for 5.4%. Meanwhile, high residual distributions with residuals higher than 2 pixels appear in local areas. Combined with the actual tunnel scene and sensor-collected data, typical failure cases and cause analysis are as follows:

(1) Among the 12 uncalibrated images, 6 frames stem from local extreme underexposure (illuminance < 30 lx) caused by tunnel lighting equipment faults, 4 frames result from strong light overexposure at tunnel entrances and exits, and the remaining 2 frames suffer from local light spot interference induced by tunnel lamp reflection. Although the ABA-CLAHE algorithm realizes image enhancement in conventional low-light and uneven illumination scenes, it still fails to fully recover effective texture features in extreme illumination scenes with an illumination difference > 200 lx, which causes the number of extracted feature points to be less than 50 and makes camera pose estimation and calibration impossible.

(2) In the flat concrete sections of tunnels without cracks, pipelines or structural edges, the scene shows strong homogeneity. Even after image enhancement, fewer than 80 effective feature points are extractable from a single frame, and the inlier rate of feature matching drops below 75%. This leads to an increase in the pose estimation error of visual odometry, and the local reprojection residual rises to 2–3 pixels. The positioning accuracy needs to be compensated by laser odometry and IMU preintegration, which becomes a performance bottleneck of the algorithm.

(3) Slight vibration occurs during the travel of the tunnel inspection robot, which causes motion blur in some images. Although SURF feature detection shows better robustness to slight blur than ORB and SIFT, it still reduces the repeatability of feature points by 10%–15% and indirectly affects the stability of feature matching and pose estimation.

Based on the above failure cases, the core limitations of the algorithm in this paper are summarized as follows: insufficient feature enhancement capability in extreme illumination scenes, excessive dependence on laser/IMU in weak-texture areas, and limited robustness to sensor motion blur. The above limitations also clarify the core direction for the subsequent optimization of the algorithm.

4 Conclusion

A novel SLAM method specifically designed for municipal tunnel environments is presented, which integrates image-enhanced visual odometry with factor graph optimization techniques. The proposed ABA-CLAHE image enhancement algorithm significantly improves image quality and feature stability under low-light conditions. By employing a SURF feature-based coarse matching strategy combined with RANSAC outlier rejection, the method achieves high-precision feature matching even in uneven illumination scenarios. The system constructs a multi-source factor graph model incorporating IMU pre-integration, LiDAR odometry, visual odometry, and loop closure constraints, with global consistency optimization implemented through the iSAM2 incremental smoothing algorithm. Experimental results demonstrate that the proposed method reduces loop closure error to 0.096 m and maintains a global reprojection error of 1.10 pixels in tunnel environments, while significantly improving the structural continuity of dense 3D maps. These advancements provide high-precision technical support for tunnel structural health monitoring.

Combined with the experimental performance and limitations of the proposed algorithm, the proposed method can be further improved in the following directions to enhance the accuracy, robustness and engineering adaptability of tunnel SLAM:

(1) To address the problem of missing features in extreme underexposure or overexposure scenes, a lightweight low-light enhancement network (such as a lightweight version of Retinex-Net) can be integrated with the ABA-CLAHE algorithm to realize joint optimization of pixel-level image enhancement and feature-level texture completion, while ensuring computational efficiency for embedded inspection robot platforms.

(2) Structural prior information of tunnels is exploited to fuse point features with line and plane features. Effective feature constraints are supplemented via structural priors in weak-texture regions, reducing dependence on LiDAR/IMU and improving the independent positioning capability of visual odometry.

(3) To mitigate the interference of sensor time synchronization error and motion blur, a joint error model of synchronization error and motion blur is established. The error compensation term is incorporated into the constraint function of the factor graph to realize joint optimization of synchronization error, motion blur and pose estimation, thus improving the algorithm robustness against sensor acquisition errors.

The proposed method provides a feasible solution for high-precision mapping in tunnel structural health monitoring. Further research will be carried out around the above directions to continuously improve the robustness and engineering adaptability of the algorithm in complex tunnel scenarios, and provide technical support for the inspection and monitoring of urban underground infrastructure.

Acknowledgement: None.

Funding Statement: This research is funded by the National Key R&D Program of China, grant number 2022YFB2602203; the Natural Science Basic Research Program of Shaanxi, grant numbers 2025JC-YBMS-699 and 2024JC-YBQN-0495).

Author Contributions: Methodology and writing—original draft preparation, Qilong Wang; data curation, Ning Wang; formal analysis, Shuhan Luo; supervision, Xiang Gao; writing—review and editing, Yuqian Lu and Min He. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author, Min He, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Montemerlo M, Thrun S. Simultaneous localization and mapping with unknown data association using FastSLAM. In: 2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422); 2003 Sep 14–19; Taipei, Taiwan. p. 1985–91. doi:10.1109/ROBOT.2003.1241885.
2. Widjiantoro BL, Indriawati K, Alexander Buyung TSN, Wahyuadnyana KD. Experimental validation: perception and localization systems for autonomous vehicles using the extended Kalman filter algorithm. *Int J Smart Sens Intell Syst.* 2024;17(1):20240002. doi:10.2478/ijssis-2024-0002.
3. Zhao J, Li J, Zhou J. Research on two-round self-balancing robot SLAM based on the gmapping algorithm. *Sensors.* 2023;23(5):2489. doi:10.3390/s23052489.
4. Can A, Price J, Montazeri A. A nonlinear discrete-time sliding mode controller for autonomous navigation of an aerial vehicle using hector SLAM. *IFAC-PapersOnLine.* 2022;55(10):2653–8. doi:10.1016/j.ifacol.2022.10.110.
5. Xu J, Wang D, Liao M, Shen W. Research of cartographer graph optimization algorithm based on indoor mobile robot. *J Phys Conf Ser.* 2020;1651(1):012120. doi:10.1088/1742-6596/1651/1/012120.
6. Cvišić I, Ćesić J, Marković I, Petrović I. SOFT-SLAM: computationally efficient stereo visual simultaneous localization and mapping for autonomous unmanned aerial vehicles. *J Field Robot.* 2018;35(4):578–95. doi:10.1002/rob.21762.
7. Mur-Artal R, Montiel JMM, Tardos JD. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Trans Robot.* 2015;31(5):1147–63. doi:10.1109/tro.2015.2463671.
8. Xu W, Cai Y, He D, Lin J, Zhang F. FAST-LIO2: fast direct LiDAR-inertial odometry. *IEEE Trans Robot.* 2022;38(4):2053–73. doi:10.1109/tro.2022.3141876.
9. Chen P, Zhao X, Zeng L, Liu L, Liu S, Sun L, et al. A review of research on SLAM technology based on the fusion of LiDAR and vision. *Sensors.* 2025;25(5):1447. doi:10.3390/s25051447.
10. Song B. Review on multisensor SLAM datasets for advanced perception and mapping technologies. *Appl Comput Eng.* 2024;97(1):170–4. doi:10.54254/2755-2721/97/20241270.

11. Zhang Y, Shi P, Li J. 3D LiDAR SLAM: a survey. *Photogramm Rec.* 2024;39(186):457–517. doi:10.1111/phor.12497.
12. Chen K, Xiao J, Liu J, Tong Q, Zhang H, Liu R, et al. Semantic visual simultaneous localization and mapping: a survey. *IEEE Trans Intell Transp Syst.* 2025;26(6):7426–49. doi:10.1109/TITS.2025.3556928.
13. Ghadimzadeh Alamdari A, Zade FA, Ebrahimkhanlou A. A review of simultaneous localization and mapping for the robotic-based nondestructive evaluation of infrastructures. *Sensors.* 2025;25(3):712. doi:10.3390/s25030712.
14. Mur-Artal R, Tardós JD. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans Robot.* 2017;33(5):1255–62. doi:10.1109/TRO.2017.2705103.
15. Campos C, Elvira R, Rodriguez JGG, Montiel JMM, Tardos JD. ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multimap SLAM. *IEEE Trans Robot.* 2021;37(6):1874–90. doi:10.1109/tro.2021.3075644.
16. Pumarola A, Vakhitov A, Agudo A, Sanfeliu A, Moreno-Noguer F. PL-SLAM: real-time monocular visual SLAM with points and lines. In: 2017 IEEE International Conference on Robotics and Automation (ICRA); 2017 May 29–Jun 3; Singapore. p. 4503–8. doi:10.1109/ICRA.2017.7989522.
17. Arandjelovic R, Gronat P, Torii A, Pajdla T, Sivic J. NetVLAD: CNN architecture for weakly supervised place recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. p. 5297–307. doi:10.1109/CVPR.2016.572.
18. Leutenegger S, Lynen S, Bosse M, Siegwart R, Furgale P. Keyframe-based visual-inertial odometry using nonlinear optimization. *Int J Robot Res.* 2015;34(3):314–34. doi:10.1177/0278364914554813.
19. Qin T, Li P, Shen S. VINS-mono: a robust and versatile monocular visual-inertial state estimator. *IEEE Trans Robot.* 2018;34(4):1004–20. doi:10.1109/TRO.2018.2853729.
20. Qin T, Pan J, Cao S, Shen S. A general optimization-based framework for local odometry estimation with multiple sensors. arXiv:1901.03638. 2019. Available from: <https://arxiv.org/abs/1901.03638>.
21. Xu K, Hao Y, Yuan S, Wang C, Xie L. AirSLAM: an efficient and illumination-robust point-line visual SLAM system. *IEEE Trans Robot.* 2025;41(5):1673–92. doi:10.1109/TRO.2025.3539171.
22. Ye C, Li H, Lin W, Yang X. MLINE-VINS: robust monocular visual-inertial SLAM with flow Manhattan and line features. *IEEE Trans Instrum Meas.* 2025;74(72):5041213. doi:10.1109/TIM.2025.3595235.
23. Mourikis AI, Roumeliotis SI. A multi-state constraint Kalman filter for vision-aided inertial navigation. In: Proceedings 2007 IEEE International Conference on Robotics and Automation; 2007 Apr 10–14; Rome, Italy. p. 3565–72. doi:10.1109/ROBOT.2007.364024.
24. Zhang J, Singh S. Visual-lidar odometry and mapping: low-drift, robust, and fast. In: 2015 IEEE International Conference on Robotics and Automation (ICRA); 2015 May 26–30; Seattle, WA, USA. p. 2174–81. doi:10.1109/ICRA.2015.7139486.
25. Shan T, Englot B, Ratti C, Rus D. LVI-SAM: tightly-coupled lidar-visual-inertial odometry via smoothing and mapping. In: 2021 IEEE International Conference on Robotics and Automation (ICRA); 2021 May 30–Jun 5; Xi'an, China. p. 5692–8. doi:10.1109/icra48506.2021.9561996.
26. Lin J, Zheng C, Xu W, Zhang F. R²LIVE: a robust, real-time, LiDAR-inertial-visual tightly-coupled state estimator and mapping. *IEEE Robot Autom Lett.* 2021;6(4):7469–76. doi:10.1109/LRA.2021.3095515.
27. Lin J, Zhang F. R3LIVE: a robust, real-time, RGB-colored, LiDAR-inertial-visual tightly-coupled state estimation and mapping package. In: 2022 International Conference on Robotics and Automation (ICRA); 2022 May 23–27; Philadelphia, PA, USA. p. 10672–8. doi:10.1109/ICRA46639.2022.9811935.
28. Zheng C, Zhu Q, Xu W, Liu X, Guo Q, Zhang F. FAST-LIVO: fast and tightly-coupled sparse-direct LiDAR-inertial-visual odometry. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2022 Oct 23–27; Kyoto, Japan. p. 4003–9. doi:10.1109/IROS47612.2022.9981107.
29. Zhou B, Zheng C, Wang Z, Zhu F, Cai Y, Zhang F. FAST-LIVO2 on resource-constrained platforms: LiDAR-inertial-visual odometry with efficient memory and computation. *IEEE Robot Autom Lett.* 2025;10(8):7931–8. doi:10.1109/LRA.2025.3581125.
30. Wang Y, Liu Y, Chen L, Chen H, Zhang S. Degradation-aware LiDAR-thermal-inertial SLAM. *IEEE Robot Autom Lett.* 2025;10(8):8035–42. doi:10.1109/LRA.2025.3581127.
31. Jang W, Kim TW. iSAM2 using CUR matrix decomposition for data compression and analysis. *J Comput Des Eng.* 2021;8(3):855–70. doi:10.1093/jcde/qwab019.