

Article

SERail-SLAM: Semantic-Enhanced Railway LiDAR SLAM

Weiwei Song ^{1,2} , Shiqi Zheng ³ , Xinye Dai ^{1,*} , Xiao Wang ³, Yusheng Wang ¹ , Zihao Wang ¹, Shujie Zhou ¹, Wenlei Liu ³ and Yidong Lou ¹ 

¹ GNSS Research Center, Wuhan University, Wuhan 430079, China; sww@whu.edu.cn (W.S.); yushengwhu@whu.edu.cn (Y.W.); wzihao@whu.edu.cn (Z.W.); sjzhou2020@whu.edu.cn (S.Z.); ydlou@whu.edu.cn (Y.L.)

² Hubei LuoJia Laboratory, Wuhan University, Wuhan 430079, China

³ School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China; shiqizheng@whu.edu.cn (S.Z.); kingxiao0902@whu.edu.cn (X.W.); wenleiliu@whu.edu.cn (W.L.)

* Correspondence: xinyed@whu.edu.cn

Abstract

Reliable state estimation in railway environments presents significant challenges due to geometric degeneracy resulting from repetitive structural layouts and point cloud sparsity caused by high-speed motion. Conventional LiDAR-based SLAM systems frequently suffer from longitudinal drift and mapping artifacts when operating in such feature-scarce and dynamically complex scenarios. To address these limitations, this paper proposes SERail-SLAM, a robust semantic-enhanced multi-sensor fusion framework that tightly couples LiDAR odometry, inertial pre-integration, and GNSS constraints. Unlike traditional approaches that rely on rigid voxel grids or binary semantic masking, we introduce a Semantic-Enhanced Adaptive Voxel Map. By leveraging eigen-decomposition of local point distributions, this mapping strategy dynamically preserves fine-grained stable structures while compressing redundant planar surfaces, thereby enhancing spatial descriptiveness. Furthermore, to mitigate the impact of environmental noise and segmentation uncertainty, a confidence-aware filtering mechanism is developed. This method utilizes raw segmentation probabilities to adaptively weight input measurements, effectively distinguishing reliable landmarks from clutter. Finally, a category-weighted joint optimization scheme is implemented, where feature associations are constrained by semantic stability priors, ensuring globally consistent localization. Extensive experiments in real-world railway datasets demonstrate that the proposed system achieves superior accuracy and robustness compared to state-of-the-art geometric and semantic SLAM methods.

Keywords: SLAM; railway; semantic



Academic Editor: Yaoxiang Yu

Received: 29 November 2025

Revised: 2 January 2026

Accepted: 5 January 2026

Published: 7 January 2026

Corrected: 20 May 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Robust and precise state estimation is the cornerstone of autonomous railway operation and infrastructure maintenance [1]. In Global Navigation Satellite Systems (GNSSs), denied environments such as tunnels, valleys, and urban canyons, establishing a consistent global trajectory remains a formidable challenge. While multi-sensor fusion systems based on GNSSs and Inertial Measurement Units (IMUs) are the industry standard, they suffer from drift during prolonged GNSS outages. Consequently, LiDAR-based Simultaneous Localization and Mapping (SLAM) has emerged as a promising solution to provide continuous ego-motion estimation and environmental mapping [2].

Despite the success of LiDAR SLAM in automotive [3] and robotics domains [4], directly transplanting these frameworks to railway environments presents unique and

severe challenges. First, the specific geometric structure of railway corridors often leads to degeneracy. Unlike urban roads rich in distinct geometric features, railway tracks are characterized by repetitive linear structures and feature-scarce surroundings, making frame-to-frame matching prone to longitudinal drift. Second, the high operational speed of trains exacerbates point cloud sparsity. Conventional SLAM front-ends typically employ uniform voxel-grid filtering to downsample data. However, in high-speed railway scenarios, this rigid downsampling inadvertently discards thin but structurally critical features—such as overhead contact system (OCS) poles and rails—while retaining bulky but unstable vegetation, thereby compromising registration accuracy. Third, environmental noise significantly impacts robustness. Railway distinctively features a mix of highly stable static objects (rails, fences) and semi-static or dynamic interference (bushes, shrubs) [5]. Purely geometric methods, which treat all points equally during optimization, fail to distinguish between reliable landmarks and unstable clutter.

To overcome these limitations, integrating semantic information into the SLAM pipeline has become a vital trend [6]. Semantic SLAM can theoretically filter out dynamic objects and improve loop closure. However, most existing semantic approaches rely on discrete labels, ignoring the variable reliability of semantic segmentation networks and the varying structural stability of different static categories [7,8].

Motivated by these observations, this paper proposes a Semantic-Enhanced LiDAR-SLAM framework tailored for the challenging railway environment. Unlike traditional approaches that rely solely on geometric constraints or binary semantic masking, our system deeply couples semantic uncertainty with geometric attributes throughout the mapping and estimation process. We introduce a novel semantic-enhanced adaptive voxel map that utilizes eigen-analysis to partition space based on local geometric distribution rather than fixed grids. This ensures that fine-grained structures like rails are preserved, while redundant planar surfaces are compressed. Furthermore, addressing the noise issue, we propose a confidence-aware filtering strategy. Instead of blindly trusting semantic labels, we leverage the raw probability outputs from the segmentation network to adaptively weight input points, effectively suppressing vegetation noise while preserving stable features. Finally, in the back-end optimization, a category-weighted constraint is formulated, where feature associations are weighted according to the structural reliability of their semantic class, thereby ensuring a tightly coupled optimization of IMU, LiDAR, and GNSS data.

The main contributions of this work are summarized as follows:

- We propose a tightly coupled multi-sensor fusion framework specifically designed to handle the geometric degeneracy and high-speed sparsity of railway environments.
- We develop a Semantic-Enhanced Adaptive Voxel Map that dynamically subdivides space based on geometric eigenvalues and semantic consistency, significantly improving the descriptiveness of local features while maintaining retrieval efficiency.
- We introduce a probabilistic filtering and category-weighted matching strategy. By incorporating segmentation confidence and semantic stability priors into the optimization cost function, the system effectively mitigates the impact of environmental noise and segmentation errors.

2. Related Work

2.1. Train Localization Solution

Accurate train localization is the prerequisite for modern Automatic Train Control Systems (ATCS) to ensure operational safety. Traditionally, railway localization relies on discrete trackside sensors such as balises [9]. This method partitions the railway line into sections, using balises as physical location references to reset the odometer error. While reliable, this infrastructure-heavy approach suffers from high deployment costs, complex

maintenance, and discrete update rates, which limits its ability to support continuous high-precision applications.

To achieve continuous positioning, recent trends have shifted toward integrating onboard sensors. Multi-sensor fusion systems combining GNSS and IMU have become the industry standard for determining absolute position and attitude [10–12]. In open-sky environments, GNSS/INS provides reliable georeferencing; however, relying solely on satellite signals poses challenges in achieving the centimeter-level trajectory smoothness required for high-fidelity environmental reconstruction, especially in areas with signal multipath or intermittent outages (e.g., urban canyons, stations, and tunnels) [13]. Therefore, supplementing GNSS with exteroceptive sensors to form a tightly coupled state estimator is crucial for robust operation.

Simultaneously, the demand for digitized railway maintenance is growing. The concept of the “Digital Twin” requires high-precision, dense 3D maps of railway corridors to automate the inspection of tracks, power lines, and structural clearances [14,15]. While LiDAR has been widely used for static scanning, applying LiDAR-based SLAM on moving trains offers a dual advantage. First, SLAM utilizes environmental geometric constraints to refine the vehicle trajectory, effectively compensating for GNSS/INS drift and ensuring local consistency. Second, and more importantly for infrastructure maintenance, the precise pose estimation from SLAM enables the generation of globally consistent, distortion-free point cloud maps without external infrastructure [16].

Unlike traditional approaches that view LiDAR merely as a backup for GNSS-denied zones, this work leverages LiDAR SLAM as a core component for high-precision mapping [17,18]. By tightly fusing LiDAR’s geometric constraints with the absolute references from GNSS and the high-frequency updates from IMU, our system aims to provide not only robust state estimation across diverse scenarios but also high-quality mapping outputs suitable for autonomous inspection and railway digitalization.

2.2. LiDAR SLAM

LiDAR is unaffected by illumination variations and offers precise ranging capabilities, enabling active acquisition of environmental data that provides a rich representation of geometric structure. Commonly used LiDAR-based pose estimation methods involve extracting geometric features such as edge and planar points from the environment to construct residual constraints for localization [19–23]. Among these, the LOAM series of algorithms is representative [19]. LOAM extracts edge and planar features by calculating point cloud smoothness and builds point-to-line and point-to-plane residual constraints against a local map for pose estimation. To enhance robustness in dynamically cluttered environments, LeGO-LOAM introduces ground point extraction to filter out noise and employs a two-step Levenberg–Marquardt (LM) optimization to improve estimation accuracy and computational efficiency [20]. Subsequently, methods such as LIO-SAM tightly integrate IMU pre-integration, LiDAR odometry, and GNSS observations using factor graph optimization to achieve multi-sensor tightly-coupled pose estimation [22].

However, in railway environments, these methods relying on low-level geometric features face significant challenges. Due to the highly repetitive and structurally similar nature of railway scenes—such as continuously similar sleepers, ballast beds, and linear infrastructure—traditional geometric features like edges and planes lack distinctiveness, which can lead to matching ambiguities and tracking failures. Moreover, such approaches do not effectively leverage semantic information from the environment, such as the topological structure of tracks, signaling equipment, or catenary poles, which possess well-defined semantic meanings. This limitation restricts their adaptability and robustness in long-distance, structured railway settings.

Therefore, introducing semantically meaningful feature representations is expected to enhance feature distinctiveness and stability, thereby improving both localization accuracy and system reliability of LiDAR-based methods in railway applications.

2.3. Map Management

In LiDAR-based SLAM, pose estimation typically involves the registration of the current point cloud frame with a local map, a process that necessitates efficient construction of the local map and search for nearest neighbors [24]. Therefore, effective management of the local map has been a topic of ongoing research. Early methods often employed the KD-Tree for point cloud map storage, which partitions the point cloud using axis-aligned hyperplanes to form a tree structure, offering higher search efficiency compared to sequential scanning. However, conventional KD-Tree is typically static; as new point clouds are continuously inserted, the tree requires frequent rebuilding to maintain its balance and query efficiency, which imposes significant computational overhead and hinders real-time performance [25].

To address this issue, FAST-LIO2 introduced an incremental k-d tree (ikd-tree), which supports dynamic updates and balanced maintenance through a strategy of tagging points for deletion, together with batch insertion and parallel reconstruction, thereby significantly improving operational efficiency. While such methods perform well in most scenarios, they still exhibit limitations in structured and highly repetitive environments such as railways [26]. Due to the highly regular geometry of railway tracks, traditional nearest-neighbor-based registration is prone to mismatches—for instance, incorrectly associating points on the rail surface with those on the ground—leading to degraded localization accuracy.

Therefore, in railway environments, it is necessary to adopt a point cloud representation and storage mechanism capable of preserving structural information, such as voxel-based representations [27,28]. By discretizing the space into volumetric elements and retaining intra-voxel features, such an approach helps enhance distinctiveness and robustness in registration, thereby laying the foundation for long-term reliable operation.

3. Methodology

In this section, we propose a semantic-enhanced SLAM framework designed for railway environments. As illustrated in Figure 1, the system integrates multi-source data from an IMU, LiDAR, and GNSS. The IMU measurements are first pre-integrated to provide prior state information. The point clouds are then processed by a deep learning-based semantic segmentation network, where each point is assigned a semantic label. Leveraging this semantic information, adaptive filtering is performed on the point cloud, followed by a feature matching strategy that incorporates category-based grouping and adaptive weight assignment. Finally, accurate state estimation of the railway vehicle is achieved through a joint optimization that tightly couples LiDAR odometry, inertial pre-integration, and GNSS constraints.

3.1. IMU Preintegration Factor

The IMU is widely used in the field of state estimation, capable of providing prior poses in a passive manner. However, its measurements are inevitably affected by sensor bias and stochastic noise. The angular velocity $\hat{\omega}_t$ and acceleration $\hat{\mathbf{a}}_t$ measured by the IMU at time t can be modeled as:

$$\hat{\omega}_t = \omega_t + \mathbf{b}_{\omega,t} + \mathbf{n}_{\omega,t} \quad (1)$$

$$\hat{\mathbf{a}}_t = \mathbf{R}_t^T (\mathbf{a}_t - \mathbf{g}) + \mathbf{b}_{\mathbf{a},t} + \mathbf{n}_{\mathbf{a},t} \quad (2)$$

where $\hat{\omega}_t$ and $\hat{\mathbf{a}}_t$ are defined as the raw measurements in the body frame $\mathbb{B}$, whereas ω_t and $\mathbf{a}_t$ denote the corresponding true values. The terms $\mathbf{b}_{\omega,t}$ and $\mathbf{b}_{\mathbf{a},t}$ represent the gyroscope and accelerometer biases, respectively, which evolve slowly over time. The noise components $\mathbf{n}_{\omega,t}$ and $\mathbf{n}_{\mathbf{a},t}$ are modeled as Gaussian white noise, with $\mathbf{n}_{\mathbf{a},t} \sim N(0, \sigma_a^2)$, $\mathbf{n}_{\omega,t} \sim N(0, \sigma_\omega^2)$. The rotation matrix $\mathbf{R}_t \in SO(3)$ represents the orientation of the body frame $\mathbb{B}$ with respect to the world frame $\mathbb{W}$ (i.e., $\mathbf{R}_{WB}^t$).

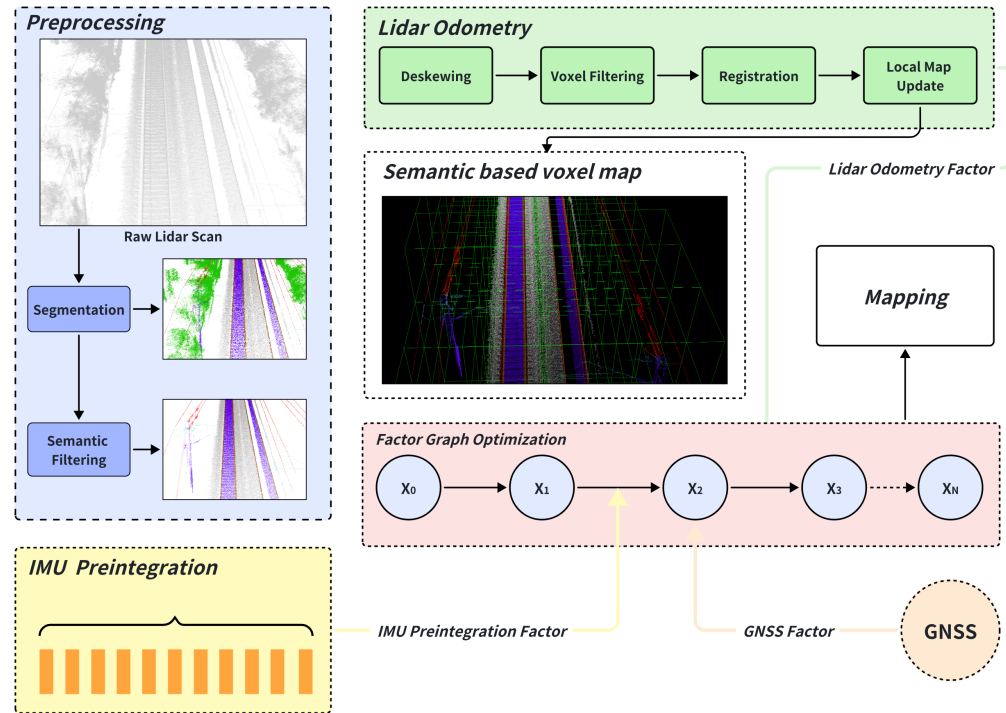


Figure 1. System Overview.

The velocity, position, and orientation of the vehicles at time $t + \Delta t$ can be obtained by integrating the IMU measurements. It is assumed that the angular velocity and acceleration remain constant during the integration interval Δt .

$$\mathbf{v}_{t+\Delta t} = \mathbf{v}_t + \mathbf{g}\Delta t + \mathbf{R}_t(\hat{\mathbf{a}}_t - \mathbf{b}_{\mathbf{a},t} - \mathbf{n}_{\mathbf{a},t})\Delta t \quad (3)$$

$$\mathbf{p}_{t+\Delta t} = \mathbf{p}_t + \mathbf{v}_t\Delta t + \frac{1}{2}\mathbf{g}\Delta t^2 + \frac{1}{2}\mathbf{R}_t(\hat{\mathbf{a}}_t - \mathbf{b}_{\mathbf{a},t} - \mathbf{n}_{\mathbf{a},t})\Delta t^2 \quad (4)$$

$$\mathbf{R}_{t+\Delta t} = \mathbf{R}_t \exp((\hat{\omega}_t - \mathbf{b}_{\omega,t} - \mathbf{n}_{\omega,t})\Delta t) \quad (5)$$

where $\mathbf{R}^t = \mathbf{R}_{WB}^t = (\mathbf{R}_{BW}^t)^T$ is the rotation matrix from the body frame $\mathbb{B}$ to the world frame $\mathbb{W}$.

Directly using these equations within a factor graph optimization framework is computationally inefficient. Each time the state estimate at time t is updated during optimization, all subsequent IMU measurements must be re-integrated. To circumvent this issue, we employ the IMU pre-integration technique [29]. This method compounds the IMU measurements between two consecutive keyframes, k and $k + 1$, into a single relative motion constraint that is independent of the initial state.

The pre-integrated measurements of rotation $\Delta\mathbf{R}_k$, velocity $\Delta\mathbf{v}_k$, and position $\Delta\mathbf{p}_k$ are computed in the local frame of keyframe k :

$$\Delta\hat{\mathbf{R}}_k = \mathbf{R}_k^T \mathbf{R}_{k+1} \quad (6)$$

$$\Delta \hat{\mathbf{v}}_k = \mathbf{R}_k^T (\mathbf{v}_{k+1} - \mathbf{v}_k - \mathbf{g} \Delta t_{k,k+1}) \quad (7)$$

$$\Delta \hat{\mathbf{p}}_k = \mathbf{R}_k^T (\mathbf{p}_{k+1} - \mathbf{p}_k - \mathbf{v}_k \Delta t_{k,k+1} - \frac{1}{2} \mathbf{g} \Delta t_{k,k+1}^2) \quad (8)$$

These pre-integrated terms depend only on the IMU measurements and the bias estimates at keyframe k .

To constrain the optimization problem, we formulate an IMU factor based on the residual between the pre-integrated measurements and the relative motion predicted by the current state estimates of keyframes k and $k + 1$. The residual vector $\mathbf{r}_{\mathcal{I}_k}$ for the measurements between these consecutive keyframes is defined as:

$$\mathbf{r}_{\mathcal{I}_k} = \begin{bmatrix} \mathbf{r}_{\Delta \mathbf{p}} \\ \mathbf{r}_{\Delta \mathbf{v}} \\ \mathbf{r}_{\Delta \mathbf{R}} \\ \mathbf{r}_{\mathbf{b}_a} \\ \mathbf{r}_{\mathbf{b}_\omega} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_k^T (\mathbf{p}_{k+1} - \mathbf{p}_k - \mathbf{v}_k \Delta t_{k,k+1} - \frac{1}{2} \mathbf{g} \Delta t_{k,k+1}^2) - \Delta \hat{\mathbf{p}}_k \\ \mathbf{R}_k^T (\mathbf{v}_{k+1} - \mathbf{v}_k - \mathbf{g} \Delta t_{k,k+1}) - \Delta \hat{\mathbf{v}}_k \\ \text{Log}((\Delta \hat{\mathbf{R}}_k)^T \mathbf{R}_k^T \mathbf{R}_{k+1}) \\ \mathbf{b}_{a,k+1} - \mathbf{b}_{a,k} \\ \mathbf{b}_{\omega,k+1} - \mathbf{b}_{\omega,k} \end{bmatrix} \quad (9)$$

The rotation residual is defined using the logarithm map $\text{Log}(\cdot)$, which maps an $SO(3)$ element to its corresponding vector in $\mathfrak{so}(3)$. The bias residuals model the random walk assumption. This residual vector, weighted by its associated covariance matrix, is incorporated into the overall non-linear least-squares optimization problem as an IMU factor.

3.2. Semantic-Enhanced Adaptive Voxel Map

This paper adopts a frame-to-map registration strategy, which estimates the incremental relative pose by aligning the de-skewed and filtered current point cloud frame with a locally built point cloud map. This approach has been widely demonstrated to offer superior stability compared to frame-to-frame methods. To support efficient point cloud retrieval and matching, a voxel-based method is employed for local map management.

Conventional voxel maps typically partition the point cloud space into uniform grids with side length v . Although straightforward to implement, fixed-resolution voxels often fail to adequately represent complex geometric structures in the environment, thereby limiting the descriptive capability of local features. To address this issue, we propose a semantic-enhanced adaptive voxel map representation that improves the expressiveness of environmental structures while maintaining high retrieval efficiency. Here, geometric reliability refers to the degree of local geometric observability provided by a voxel, which is quantitatively characterized by the eigenvalue distribution of its covariance matrix. Figure 2 provides a conceptual illustration of this rail-to-ground mismatching phenomenon and its impact on point-to-plane residual construction.

The proposed method proceeds as follows, as summarized in Algorithm 1. First, the input point cloud is initially segmented into voxels of side length v . For each voxel, it contains point cloud $p_i (i = 1, \dots, N)$, the centroid of the points within it is computed, followed by the calculation of the covariance matrix $\mathbf{A}$.

$$\bar{\mathbf{p}} = \frac{1}{N} \sum_{i=1}^N \mathbf{p}_i \quad (10)$$

$$\mathbf{A} = \frac{1}{N} \sum_{i=1}^N (\mathbf{p}_i - \bar{\mathbf{p}})(\mathbf{p}_i - \bar{\mathbf{p}})^T \quad (11)$$

Eigen-decomposition of $\mathbf{A}$ yields three eigenvalues $\lambda_1, \lambda_2, \lambda_3$ and their corresponding eigenvectors. The distribution of these eigenvalues reveals the geometric structure of the points inside the voxel: in the case of a planar structure, the two larger eigenvalues are significantly greater than the smallest one; for a linear structure, the largest eigenvalue is

much greater than the other two; if the points are scattered without a dominant structure, the three eigenvalues exhibit no significant difference. Figure 3 provides a schematic illustration of these three typical voxel structure cases.

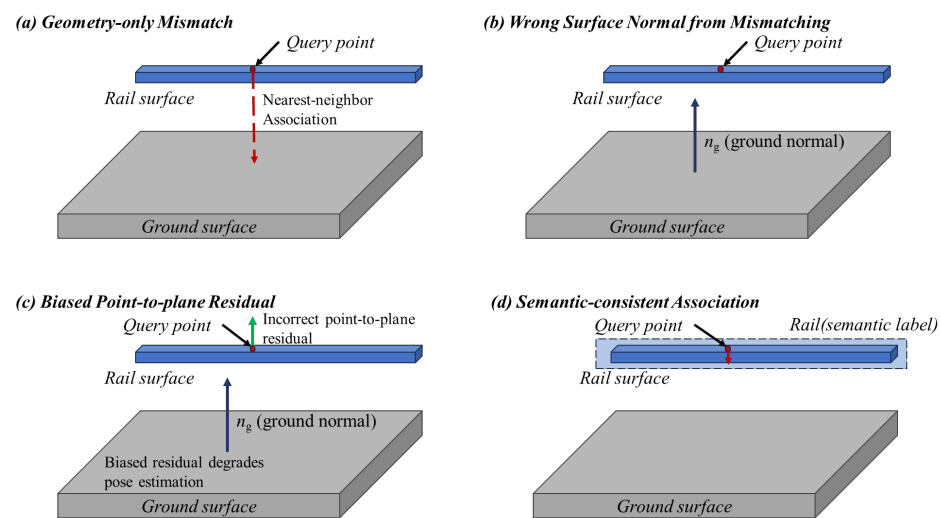


Figure 2. Conceptual illustration of rail-to-ground mismatching and its impact on residual construction in railway environments. (a) Geometry-only mismatch: a query point originating from the rail surface may be incorrectly associated with a nearby ground surface when only geometry-based nearest-neighbor association is applied, due to local geometric similarity in repetitive railway scenes. (b) Wrong surface normal selection: once such mismatching occurs, the point-to-plane residual is constructed with respect to the ground surface, and the corresponding ground normal is used instead of the true rail surface normal. (c) Biased residual in optimization: the residual computed using the incorrect surface normal introduces a systematic bias in the optimization process, which continuously degrades pose estimation rather than acting as zero-mean noise. (d) Semantic-consistent association: by enforcing semantic consistency, the association is restricted within reliable rail structures, preventing rail-to-ground mismatching and enabling correct residual construction.

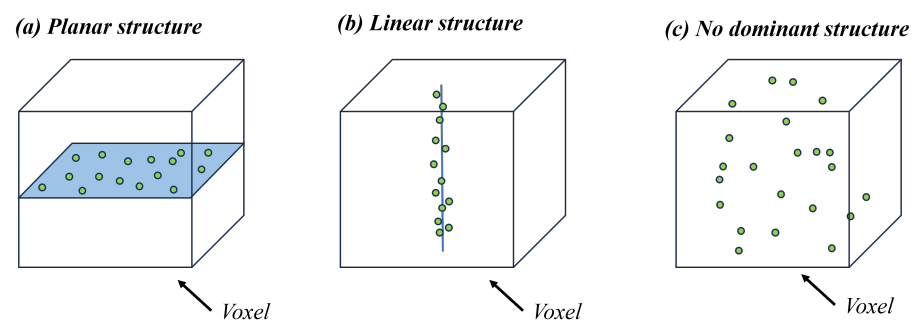


Figure 3. Schematic illustration of three typical voxel structure cases used in the proposed semantic-enhanced adaptive voxel map.

Based on the above criterion, an adaptive splitting rule is applied: if the points inside a voxel exhibit a distinct linear or planar structure, the voxel is retained; otherwise, it is subdivided into eight child voxels, and the evaluation is repeated recursively. If the maximum recursion depth is reached and no dominant structure is identified, the points within that voxel are discarded, as they are considered geometrically unreliable and provide limited observability for pose estimation.

Algorithm 1: Semantic-Enhanced Adaptive Voxel Map Construction**Input:** Point Cloud $\mathcal{P}$, Base voxel size v_{size} , Max depth d_{max} , Semantic threshold τ_{sem} .**Output:** Adaptive Voxel Map $\mathcal{M}$.

```

1 Initialize map  $\mathcal{M} \leftarrow \emptyset$ ;
2 Partition  $\mathcal{P}$  into initial grid voxels  $\mathcal{V}_{init}$  with size  $v_{size}$ ;
3 foreach  $V \in \mathcal{V}_{init}$  do
4   RecursiveBuild( $V, 0$ );

```

```

5 Function RecursiveBuild( $V, depth$ ):
6   Extract point set  $\mathcal{P}_V = \{\mathbf{p}_1, \dots, \mathbf{p}_N\}$  from  $V$ ;
7   if  $|\mathcal{P}_V| < N_{min}$  then
8     return;
9   Compute centroid  $\bar{\mathbf{p}}$  and covariance  $\mathbf{A}$  via Equations (10) and (11);
10  Eigen-decomposition of  $\mathbf{A} \rightarrow \lambda_1 \geq \lambda_2 \geq \lambda_3$  and eigenvectors  $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ ;
    /* Geometric Structure Analysis */
11   $isPlanar \leftarrow (\lambda_2 \gg \lambda_3)$ ;
12   $isLinear \leftarrow (\lambda_1 \gg \lambda_2)$ ;
13  if  $isPlanar \vee isLinear$  then
    /* Check Semantic Consistency */
14    Compute semantic distribution for  $\mathcal{P}_V$ ;
15    Identify dominant label  $l_{dom}$  and probability  $prob$ ;
16    if  $prob > \tau_{sem}$  then
17      if  $isPlanar$  then
18         $\mathbf{n} \leftarrow \mathbf{e}_3$ ; // Surface Normal
19         $Type \leftarrow PLANAR$ ;
20         $Attr \leftarrow \mathbf{n}$ ;
21      else if  $isLinear$  then
22         $\mathbf{v} \leftarrow \mathbf{e}_1$ ; // Line Direction
23         $Type \leftarrow LINEAR$ ;
24         $Attr \leftarrow \mathbf{v}$ ;
    /* Store centroid and geometry for direct matching */
25    Create voxel unit  $U = \{\bar{\mathbf{p}}, Attr, Type, l_{dom}\}$ ;
26    Add  $U$  to  $\mathcal{M}$ ;
27  else
    /* No dominant structure: subdivide or discard */
28    if  $depth < d_{max}$  then
29      Subdivide  $V$  into 8 child voxels  $\{V_c^1, \dots, V_c^8\}$ ;
30      foreach  $V_c \in \{V_c^k\}$  do
31        RecursiveBuild( $V_c, depth + 1$ );
32    else
33      Discard  $\mathcal{P}_V$  (Geometric Clutter / Noise);

```

To incorporate semantic information, semantic labels associated with each point are utilized to compute the distribution of semantic categories inside each voxel. The category with the highest proportion is assigned as the dominant semantic label, and its corresponding probability is recorded. To enhance registration reliability, a probability threshold

τ_{sem} is applied: only voxels whose dominant semantic probability exceeds a set threshold are considered valid semantic units. Voxels with a uniform label distribution—indicating classification ambiguity—are likely to introduce noise during matching and are therefore excluded. During nearest-neighbor search and data association, correspondences are preferentially established between voxels sharing the same semantic type and having high probability, thereby improving matching robustness.

Upon successful subdivision, the geometric attributes of the leaf voxels are cached to accelerate the matching process. Specifically, for planar voxels, the eigenvector corresponding to the smallest eigenvalue is stored as the surface normal $\mathbf{n}$; for linear voxels, the eigenvector corresponding to the largest eigenvalue is stored as the line direction $\mathbf{v}$. These pre-computed attributes allow for direct calculation of geometric residuals (Equations (14) and (15)) without repeated eigen-decomposition during the registration phase.

3.3. LiDAR Odometry Factors

Due to the axial ranging errors observed at both short and long distances, points that are excessively close or overly distant are removed from the LiDAR point cloud as a preprocessing step.

3.3.1. Semantic-Aware Filtering

In conventional LiDAR-based SLAM frameworks, voxel-grid filtering is widely adopted to mitigate computational burdens. However, in high-speed railway scenarios, standard voxelization treats all points equally within a grid, often discarding sparse but critical geometric features while retaining noise from volumetric vegetation. This indiscriminate downsampling compromises the signal-to-noise ratio required for robust registration. To resolve this, we propose an adaptive semantic-probabilistic sampling strategy that prioritizes structurally stable features. Let $\mathcal{P} = \{(p_i, l_i, s_i)\}$ denote the input point cloud, where each point p_i is associated with a semantic label $l_i \in \mathcal{C}$ and a segmentation confidence score $s_i \in [0, 1]$. We introduce a stability coefficient $\beta(c)$ for each semantic class c , which serves as a semantic filtering prior reflecting its long-term geometric consistency and localization value in railway environments. In our implementation, vegetation points are assigned a lower stability weight ($\beta = 0.5$) compared to other categories ($\beta = 1.5$), reflecting their lower geometric stability and weaker pose observability in railway environments (Table 1). An importance weight $w(p_i)$ is formulated to integrate both semantic stability and segmentation uncertainty:

$$w(p_i) = \beta(l_i) \cdot s_i \quad (12)$$

Unlike simple thresholding, we utilize this weight to guide the downsampling process. Points are filtered based on an adaptive threshold τ , ensuring that only high-confidence measurements from stable structures are preserved:

$$\mathcal{P}' = \{p_i \in \mathcal{P} \mid w(p_i) > \tau_{filter}\} \quad (13)$$

where $\tau_{filter} \in (0, 1)$ is a tunable hyperparameter. This strategy effectively suppresses unstable measurements from dynamic objects or vegetation while preserving the sparsity integrity of critical railway infrastructure.

Table 1. The key parameters list of the SERail-SLAM approach.

Name	Value	Description
A. Semantic-Enhanced Adaptive Voxel Map Parameters		
v_{size}	1.5 m	Base Voxel Size
d_{max}	3	Max Recursion Depth
N_{min}	5	Min Points
ϵ_{geo}	0.1	Geometric Threshold
τ_{sem}	0.7	Semantic Consistency
B. Semantic Filtering and Weighting Parameters		
τ_{filter}	0.8	Semantic Filtering Threshold
$\beta(vegetation)$	0.5	Vegetation Weight
$\beta(others)$	1.5	Others' Weight
W_{stable}	5.0	Pole Weight
$W_{ambiguous}$	1.0	Others' Weight

3.3.2. Group-Wise Matching and Adaptive Weight Assignment

The proposed adaptive voxel map stores not only geometric attributes (centroid, covariance, principal directions) but also aggregated semantic statistics at the voxel level. During map construction, all points falling into voxel V_j contribute their semantic labels and segmentation confidences. A dominant category is then assigned to the voxel based on confidence-weighted voting, together with a voxel-level semantic reliability score. This voxel-level semantic fusion suppresses point-wise classification noise and provides a consistent representation for subsequent correspondence grouping.

Different semantic categories exhibit distinct levels of geometric stability and localization value. By keeping semantic statistics within each voxel, correspondences can be grouped by semantic category before geometric matching, enabling category-specific priors to be embedded into the optimization.

For each scan point p_k^W in the global frame, we first locate its corresponding voxel V_j and retrieve the precomputed geometric structure. Geometric residuals are formulated following standard point-to-plane and point-to-line definitions. For planar features, the residual is

$$d_{p_k} = |(p_k^W - \mu_j)^\top \mathbf{n}_j|, \quad (14)$$

where μ_j is the voxel centroid, and $\mathbf{n}_j$ is the normal vector associated with the smallest eigenvalue λ_3 of the covariance. For edge features, the point-to-line distance is computed using the principal direction vector $\mathbf{v}_j$ (corresponding to λ_1):

$$d_{E_k} = |(p_k^W - \mu_j) \times \mathbf{v}_j|. \quad (15)$$

Importantly, semantic and geometric attributes are decoupled: semantic class determines the group to which the correspondence belongs, while the geometric feature type is determined purely from PCA of the voxel geometry. This separation prevents semantic labels from overriding actual geometric evidence and ensures robust feature extraction even under occasional segmentation errors.

Semantic constraints are then applied during association. A correspondence is considered valid only if the semantic label of p_k^W matches the dominant semantic category of voxel V_j , preventing spurious associations such as rail-to-ground or pole-to-vegetation matches. For voxels with low semantic confidence, we apply a relaxed criterion to avoid over-filtering.

Finally, all correspondences are grouped by semantic category, and category-dependent weights α_c are introduced as semantic residual weight within the optimization

cost function, allowing structurally reliable categories to exert stronger influence during pose estimation while suppressing unstable or ambiguous classes. In particular, poles and pillars are assigned a higher weight ($\alpha = 5.0$, while other categories use $\alpha = 1.0$ by default), as specified in Table 1, due to their high structural stability and strong observability, especially for mitigating longitudinal degeneracy in railway scenes. Let α_c denote the weight assigned to semantic class c . The aggregated residual for LiDAR odometry is expressed as:

$$r_{\mathcal{L}k} = \sum_{c \in \mathcal{C}} \alpha_c \left(\sum_{i=1}^{N_{\epsilon}^c} (d_{E_i}^c)^2 + \sum_{j=1}^{N_{\rho}^c} (d_{\rho_j}^c)^2 \right), \quad (16)$$

where N_{ϵ}^c and N_{ρ}^c denote the number of edge and planar correspondences within category c .

This voxel-level semantic grouping effectively amplifies the influence of structurally stable and geometrically reliable features, while suppressing the contribution of noisy or dynamic objects. It unifies segmentation trust and semantic residual weight into a single optimization framework. As a result, the proposed semantic-aware weighting enhances registration robustness, particularly in the geometrically degenerate and visually ambiguous environments commonly encountered in high-speed railway scenarios.

3.4. GNSS Factor

Since the GNSS can provide absolute positioning information in open environments, it is commonly employed to introduce absolute constraints into SLAM systems during long-term operation. In the domain of satellite-based positioning, real-time kinematic (RTK) technology, enabled by virtual reference stations, offers highly accurate and stable localization.

To integrate GNSS measurements into the optimization framework, we define a GNSS residual that constrains the absolute position of the vehicle. Let $\mathbf{z}_k^{\text{GNSS}} \in \mathbb{R}^3$ denote the GNSS observation in the world frame $\mathbb{W}$, and let $\mathbf{p}_k$ represent the estimated position of the vehicle at time step k . The residual of the GNSS factor is expressed as

$$\mathbf{r}_{\mathcal{G}_k} = \mathbf{p}_k - \mathbf{z}_k^{\text{GNSS}} \quad (17)$$

This residual formulation enforces global consistency of the trajectory by anchoring the SLAM estimates to the absolute GNSS measurements while adaptively leveraging the reliability of RTK solutions.

In situations where GNSS measurements are unavailable or deemed unreliable, such as during temporary outages or severe interference, GNSS factors are not incorporated into the estimator, and the system continues to operate in a tightly coupled LiDAR-IMU SLAM mode, which helps maintain local trajectory continuity. When GNSS measurements become available again and pass basic consistency checks, the corresponding GNSS factors can be incorporated to assist with global alignment, while the overall estimation remains stable due to the semantic-stable railway structures and LiDAR-IMU constraints.

Finally, all factors are jointly optimized in a nonlinear least-squares framework:

$$\min_{\mathcal{X}} \left(\sum_k \|\mathbf{r}_{\mathcal{I}_k}\|^2 + \sum_k \|\mathbf{r}_{\mathcal{L}_k}\|^2 + \sum_k \|\mathbf{r}_{\mathcal{G}_k}\|^2 \right) \quad (18)$$

where $\mathcal{X}$ represents the state variables of all keyframes. This formulation tightly couples IMU, LiDAR and GNSS information within a unified optimization framework.

4. Experiments

4.1. Experimental Setup

To validate the proposed framework, we established a customized data acquisition platform, as shown in Figure 4, equipped with a hybrid LiDAR setup (integrating Livox

solid-state and mechanical spinning units) and an Xsens MTi-680G GNSS/INS (Xsens Technologies B.V., Enschede, The Netherlands) module. Rigorous microsecond-level hardware synchronization among all sensors is achieved via a u-blox EVK-M8T timing kit using Pulse Per Second (PPS) triggering. The algorithms are executed on a high-performance onboard computer (Intel i7-14700KF, NVIDIA GTX 4080S, 32 GB RAM) running Ubuntu and ROS, ensuring real-time processing capabilities. Two representative railway datasets were collected for evaluation, whose basic statistics are summarized in Table 2. Table 2 reports the duration and travel distance of each dataset, which characterize the evaluation scale, motion length, and trajectory complexity of the experiments. We conducted experiments on representative railway lines, utilizing an industrial-grade M39 GNSS/INS system to record reference data; the ground truth trajectories were subsequently generated through post-processing kinematic (PPK) analysis combined with RTK corrections. This post-processing procedure provides a continuous and globally consistent reference trajectory and is independent of real-time GNSS availability during online operation.

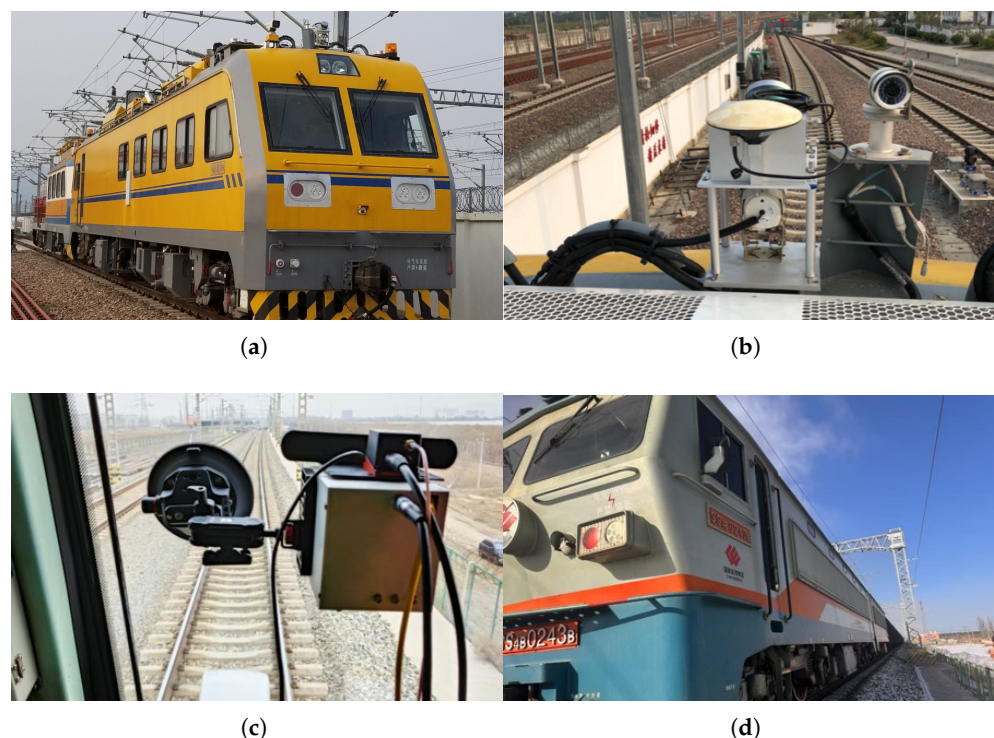


Figure 4. This figure illustrates the specialized railway vehicles and the sensor mounting configurations used to collect the two representative datasets. (a) The Hefei Railway Data Acquisition Vehicle. (b,c) The exterior sensor mounting configuration. (d) The Inner Mongolia Railway Data Acquisition Vehicle.

Table 2. Details of the two selected datasets.

Name	Duration (sec)	Distance (km)
Hefei	683	7.39
Inner Mongolia	434	2.38

For the confidence-aware filtering weight $\beta(\cdot)$ in Equation (12), we assign a higher weight to points belonging to vegetation compared to other objects to ensure they are effectively suppressed during the filtering process. Conversely, during the LiDAR odometry phase, we assign a higher weight to poles/pillars than to other objects in the category-weighted constraint (Equation (14)) to emphasize their structural significance in the registration process and mitigate longitudinal drift. Regarding semantic segmentation, point-level

predictions are generated by a SparseUNet model [30]. This model was first pre-trained on a proprietary railway point cloud semantic segmentation dataset, which comprises 20 distinct railway scenes. The segmentation network outputs predictions for seven categories: vegetation, poles/pillars, wires, support structures, rails, ground, fences, and others. During model training, we employed a cross-validation strategy where the data was split into a ratio of Training Set:Testing Set:Validation Set = 6:2:2. This allocation ensures that the testing sequences remain untouched during the training phase, thereby enhancing the consistency and relevance of the experimental results with real-world application scenarios. The main hyperparameters used in SERail-SLAM are summarized in Table 1. This table reports the key configuration settings of the proposed system and is provided to ensure reproducibility of the experimental results.

4.2. Filtering Performance Evaluation

To validate the robustness of the proposed filtering module against environmental noise, we conducted a comparative analysis using the Inner Mongolia railway dataset. As depicted in Figure 5a, this scenario is characterized by dense vegetation (bushes and shrubs) flanking the tracks. These objects are inherently unstructured and susceptible to airflow disturbances caused by high-speed trains, making them unreliable landmarks for precise registration. Figure 5 presents a qualitative comparison of different filtering strategies. First, Figure 5c shows the output of conventional Voxel Grid Filtering. Since this method downsamples data based solely on spatial distribution without semantic awareness, it retains a significant volume of vegetation noise (green points). In the subsequent optimization phase, these unstable points can generate erroneous point-to-plane residuals, leading to drift in the estimated trajectory. Second, Figure 5d illustrates the result of a Hard-Threshold Semantic Filter (direct masking based on labels). While this approach removes the majority of the vegetation, it exhibits limitations at object boundaries. Due to the limited receptive field and resolution of segmentation networks, points at the transition zones between vegetation and static structures are often misclassified. As highlighted by the red boxes, noise points that are incorrectly labeled as “static” are preserved. These “outliers” are particularly dangerous because they are treated as valid constraints in the registration. Finally, Figure 5e demonstrates the efficacy of our semantic-aware filtering. Instead of relying on a binary decision, our method leverages the raw probability output from the segmentation network. Points located at fuzzy boundaries typically possess lower classification confidence. By mapping this confidence to the filtering weight $\beta(\cdot)$, our system adaptively suppresses these ambiguous features. As a result, the proposed method yields the cleanest point cloud, effectively stripping away clutter while preserving the integrity of stable structures like rails and OCS poles. This high-quality pre-processing is crucial for minimizing false correspondences in the subsequent registration steps.

4.3. Mapping Performance Evaluation

As illustrated in Figure 6, the proposed semantic-enhanced adaptive voxel map provides a substantial improvement over conventional fixed-resolution voxel grids. The fixed-grid representation (Figure 6a,c) inevitably introduces coarse discretization, causing thin and elongated railway structures—such as overhead wires—to be partially truncated or ambiguously represented unless an excessively small voxel size is selected. In contrast, our adaptive strategy (Figure 6b,d) dynamically refines the voxel resolution according to local geometric complexity and semantic consistency. The close-up view demonstrates that the voxel subdivision process preserves the topology of the pole–wire junction and allocates finer voxels only where necessary. Moreover, the pre-computed eigenvectors

stored within each voxel encode the principal geometric directions, enabling stronger and more discriminative constraints during subsequent registration.

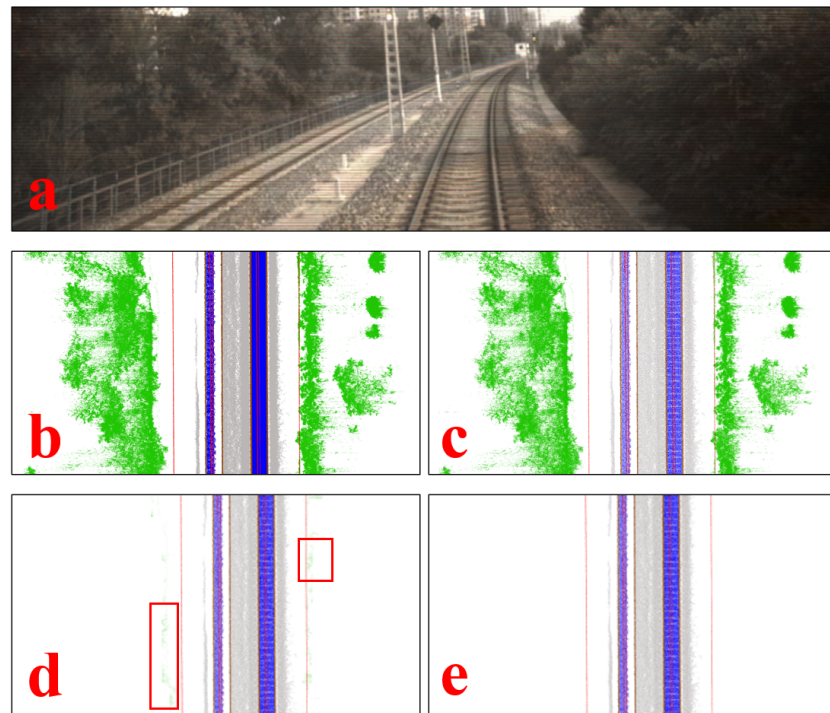


Figure 5. This figure illustrates the efficacy of the proposed semantic-aware filtering methods compared to conventional techniques. The green colored points consistently represent the point cloud categorized as vegetation (noise). (a) The camera image provides visual context of the surveyed railway corridor, characterized by dense vegetation along the trackside. (b) The raw, un-downsampled point cloud. (c) The point cloud after conventional Voxel Grid Filtering. (d) The result of simple binary filtering based on semantic labels. The red boxes indicate unfiltered noise point clouds. (e) The result of the proposed Confidence-Aware Weighted Semantic Filtering.

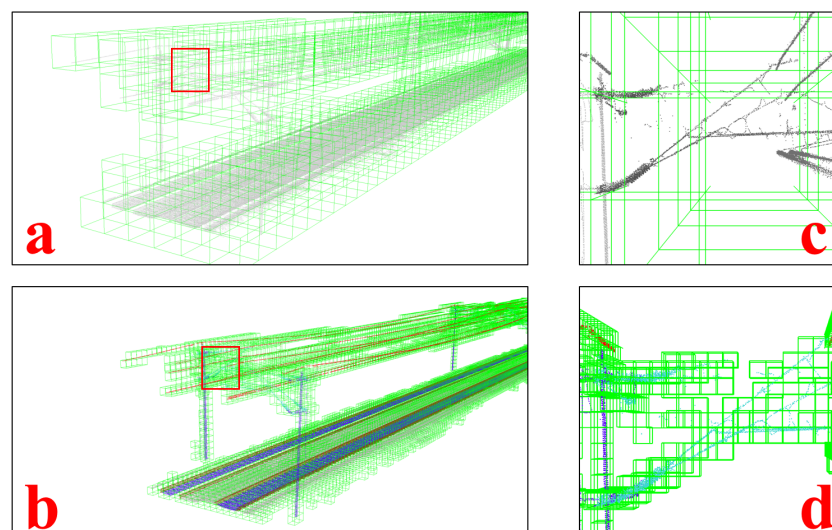


Figure 6. This figure illustrates the benefits of the proposed semantic-enhanced adaptive voxel map against a conventional fixed-resolution map, focusing on the fidelity of geometric feature representation. (a) Global view of a local map constructed using a Traditional Fixed-Resolution Voxel Map. (c) A close-up view of the area indicated by the red box in (a). (b) Global view of the proposed Semantic-Enhanced Adaptive Voxel Map. (d) A close-up view of the corresponding area marked in (b).

The large-scale mapping results generated by SERail-SLAM are shown in Figure 7. The reconstructed 3D corridor exhibits high continuity and structural completeness across diverse railway environments. The inset views further highlight the system’s ability to recover complex infrastructure details. The top-left inset shows the faithful reconstruction of a building structure; the top-right inset captures the rail–ballast geometry with clear separation of functional elements; and the bottom inset reveals the delicate and sparsely distributed overhead contact system. Consequently, the generated map achieves high structural fidelity and compactness, forming a reliable foundation for downstream applications such as railway inspection, asset management, and high-definition map production.

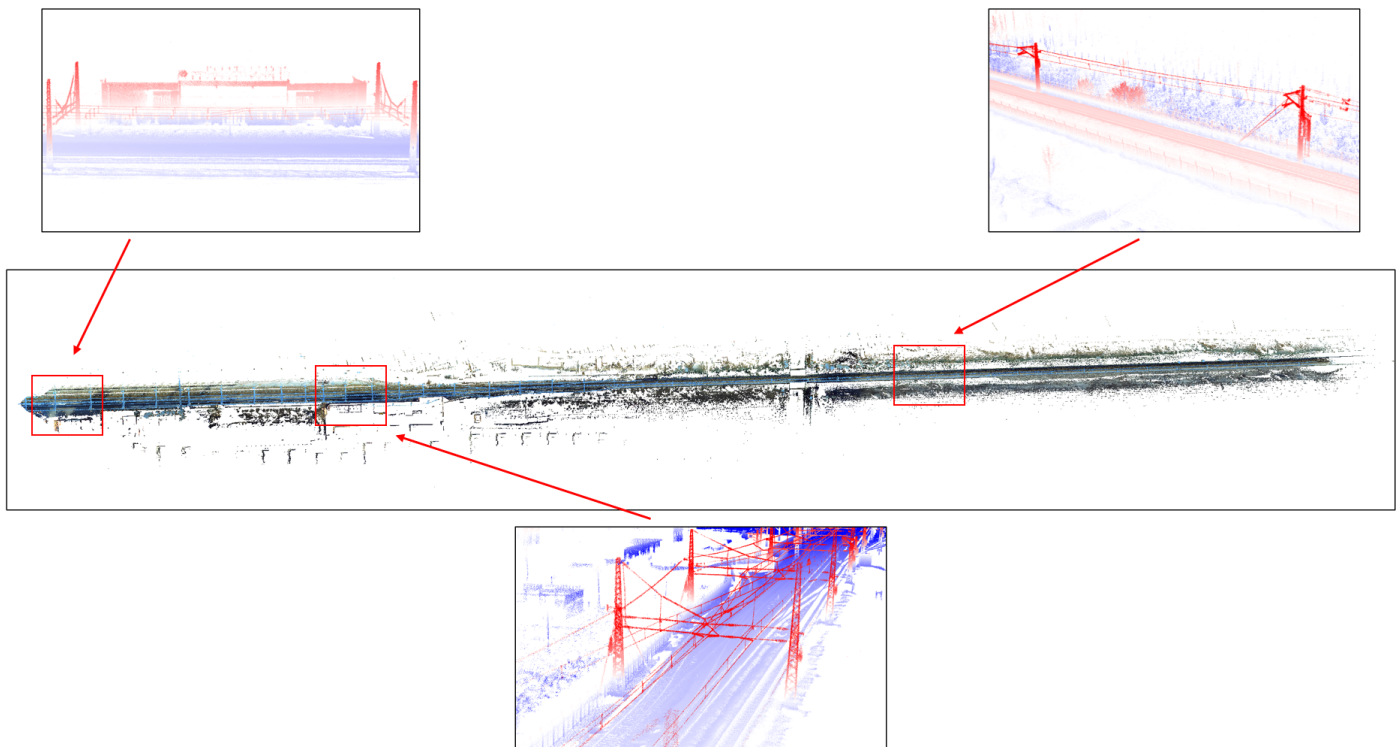


Figure 7. This figure provides a comprehensive visualization of the 3D point cloud map constructed by the proposed SERail-SLAM framework along the test railway corridor.

4.4. Positioning Performance Evaluation

To further evaluate the robustness and accuracy of the proposed framework, Figure 8 presents a trajectory comparison on a real railway line in Hefei, where several segments fall into RTK-denied environments. For fairness, FAST-LIO2 is equipped with the same RTK-based backend used in our system. The two zoom-in regions highlight the behavior of each system during temporary RTK outages of approximately 200 m. Due to the extended loss of RTK correction, both LIO-SAM and FAST-LIO2+RTK exhibit noticeable drift. In contrast, the proposed method maintains trajectory consistency even when RTK is unavailable, owing to the integration of surrounding semantic structures as long-term constraints. The right-most plots illustrate the per-axis drift in a full no-RTK scenario. Since FAST-LIO2 relies solely on LiDAR odometry without global correction, substantial drift—particularly along the Z-axis—accumulates over long-distance motion. In comparison, our approach preserves significantly higher stability, demonstrating its strong capability for GNSS-denied railway navigation.

Furthermore, to satisfy the stringent requirements of railway navigation for real-time state monitoring, we evaluate the system’s sampling rate and velocity estimation performance. The proposed algorithm provides full state estimates at a sampling rate of

10 Hz, synchronized with the LiDAR scan frequency. Quantitative analysis against the post-processed ground truth shows that the RMSE of the estimated speed is approximately 0.16 m/s. This high-fidelity velocity estimation, combined with the 10 Hz update rate, ensures the reliability of the system for critical tasks such as train braking distance prediction.

A quantitative comparison of positioning accuracy is provided in Table 3, which reports the RMSE and maximum positioning error for FAST-LIO2, FAST-LIO2+RTK, LIO-SAM, and the proposed method. The numerical results in Table 3 consistently support the qualitative observations in Figure 8, confirming the superior accuracy and robustness of our approach.

Table 3. SLAM performance comparison on the dataset (RMSE (m)/MAX (m)).

Method	RMSE (m)	Max (m)
FAST-LIO2	-	216
FAST-LIO2+rtk	6.09	8.21
LIO-SAM	4.13	7.34
Ours	1.88	2.17

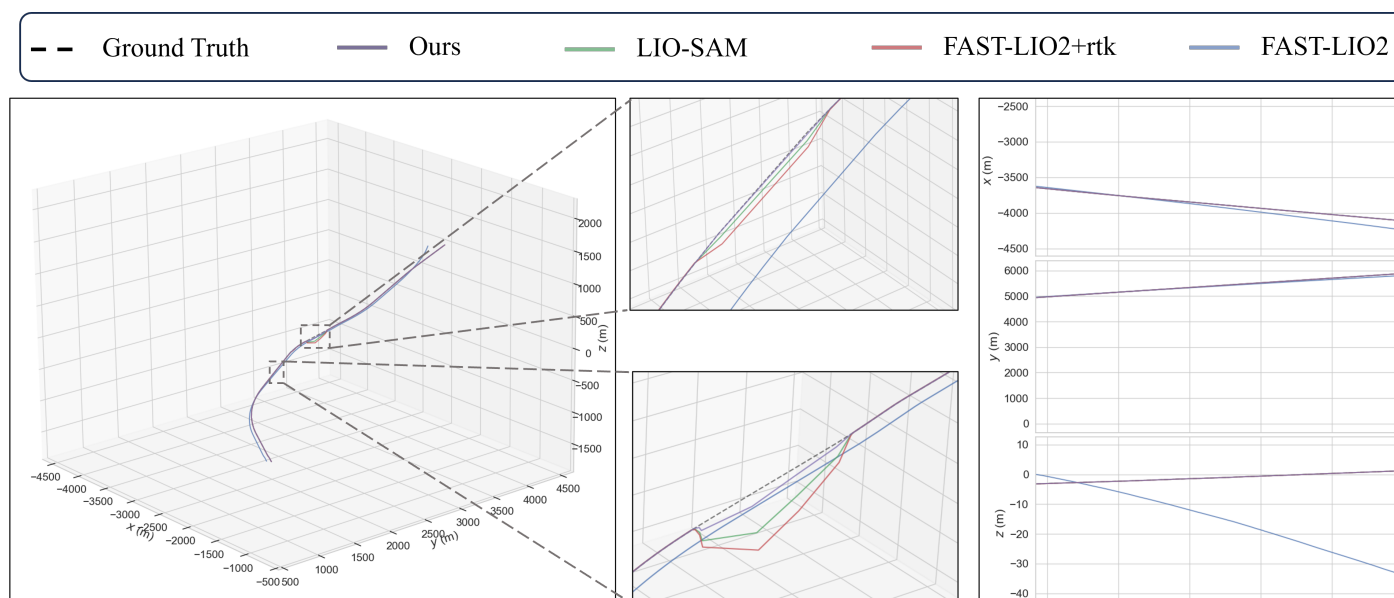


Figure 8. This figure shows the full 3D trajectories of the proposed method, Fast-LIO2, LIO-SAM, FAST-LIO2+rtk and the groundtruth reference. The two zoom-in subfigures highlight sections where RTK signals temporarily failed. The right-most column presents per-axis errors in a full no-RTK setting, where Fast-LIO2 suffers from significant drift, especially along the Z-axis.

5. Conclusions

In this paper, we propose SERail-SLAM, a Semantic-Enhanced LiDAR SLAM framework specifically tailored for high-precision mapping of railway environments. The core contribution of this work lies in demonstrating that the strategic integration of environmental semantics can fundamentally improve localization robustness in structurally repetitive and GNSS-challenged scenarios commonly encountered along rail corridors.

By explicitly modeling the structural stability and reliability of different semantic categories, semantic information is incorporated throughout the SLAM pipeline. This enables the development of a confidence-aware filtering mechanism to suppress dynamic and ambiguous clutter, as well as an adaptive voxel map structure that preserves the geometric fidelity of critical railway infrastructure such as rails and catenary-related components.

While the experimental validation in this work focuses on railway environments dominated by relatively flat terrain, it is worth noting that more complex topographic conditions—such as long tunnels, deep valleys, and mountainous corridors—may introduce additional challenges due to reduced geometric diversity and prolonged structurally constrained motion. Addressing such conditions will be an important aspect of future research. Looking ahead, future research will focus on two key directions: developing lightweight, structure-preserving semantic maps for efficient storage and leveraging semantic information to enable robust visual and LiDAR-based semantic relocalization capabilities.

Author Contributions: Conceptualization, W.S. and S.Z. (Shiqi Zheng); Methodology, S.Z. (Shiqi Zheng) and X.D.; Software, S.Z. (Shiqi Zheng), X.D., X.W., Y.W. and Z.W.; Validation, Y.W.; Formal analysis, Y.W.; Investigation, Z.W.; Resources, W.S. and Y.L.; Data curation, X.W. and W.L.; Writing—original draft, S.Z. (Shiqi Zheng); Writing—review & editing, S.Z. (Shiqi Zheng), X.W. and S.Z. (Shujie Zhou); Visualization, S.Z. (Shujie Zhou) and W.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by National Key Research and Development Program of China grant number 2023YFB3907301; in part by The National Natural Science Foundation of China grant number U23A20280; in part by Major Scientific and Technological Projects in Hubei Province grant number 2025BAA007 and in part by Major Program (JD) of Hubei Province grant number 2023BAA026.

Data Availability Statement: The datasets presented in this article are not readily available because the railway data used in this study contains confidential operational information. Requests to access the datasets should be directed to the corresponding author and are subject to institutional approval.

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

References

1. Li, K.; Li, M.; Hanebeck, U.D. Towards High-Performance Solid-State-LiDAR-Inertial Odometry and Mapping. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5167–5174. [\[CrossRef\]](#)
2. Pan, Y.; Xiao, P.; He, Y.; Shao, Z.; Li, Z. MULLS: Versatile LiDAR SLAM via Multi-Metric Linear Least Square. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 11633–11640. [\[CrossRef\]](#)
3. Jiao, J.; Ye, H.; Zhu, Y.; Liu, M. Robust Odometry and Mapping for Multi-LiDAR Systems with Online Extrinsic Calibration. *IEEE Trans. Robot.* **2022**, *38*, 351–371. [\[CrossRef\]](#)
4. Camurri, M.; Ramezani, M.; Nobili, S.; Fallon, M. Pronto: A Multi-Sensor State Estimator for Legged Robots in Real-World Scenarios. *Front. Robot. AI* **2020**, *7*, 68. [\[CrossRef\]](#)
5. Wang, Y.; Song, W.; Zhang, Y.; Huang, F.; Tu, Z.; Li, R.; Zhang, S.; Lou, Y. Four years of multimodal odometry and mapping on the rail vehicles. *J. Field Robot.* **2023**, *41*, 227–257. [\[CrossRef\]](#)
6. Li, L.; Kong, X.; Zhao, X.; Li, W.; Wen, F.; Zhang, H.; Liu, Y. SA-LOAM: Semantic-Aided LiDAR SLAM with Loop Closure. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 7627–7634. [\[CrossRef\]](#)
7. Li, L.; Kong, X.; Zhao, X.; Huang, T.; Li, W.; Wen, F.; Zhang, H.; Liu, Y. SSC: Semantic Scan Context for Large-Scale Place Recognition. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 2092–2099. [\[CrossRef\]](#)
8. Chen, X.; Milioto, A.; Palazzolo, E.; Giguère, P.; Behley, J.; Stachniss, C. SuMa++: Efficient LiDAR-based Semantic SLAM. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 4530–4537. [\[CrossRef\]](#)
9. Biagi, M.; Carnevali, L.; Paolieri, M.; Vicario, E. Performability evaluation of the ERTMS/ETCS—Level 3. *Transp. Res. Part C Emerg. Technol.* **2017**, *82*, 314–336. [\[CrossRef\]](#)
10. Deng, Z.; Song, H.; Huang, H.; Li, Y.; Dong, H. Multi-sensor based Train Localization and Data Fusion in Autonomous Train Control System. In Proceedings of the 2020 Chinese Automation Congress (CAC), Shanghai, China, 6–8 November 2020; pp. 5702–5707. [\[CrossRef\]](#)

11. Beugin, J.; Marais, J. Simulation-Based Evaluation of Dependability and Safety Properties of Satellite Technologies for Railway Localization. *Transp. Res. Part C Emerg. Technol.* **2012**, *22*, 42–57. [\[CrossRef\]](#)
12. Lu, D.; Schnieder, E. Performance Evaluation of GNSS for Train Localization. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 1054–1059. [\[CrossRef\]](#)
13. Reimer, C.; Müller, F.J.; Hinüber, E.L.V. INS/GNSS/odometer data fusion in railway applications. In Proceedings of the 2016 DGON Inertial Sensors and Systems (ISS), Karlsruhe, Germany, 20–21 September 2016; pp. 1–14. [\[CrossRef\]](#)
14. Sánchez-Rodríguez, A.; Soilán, M.; Cabaleiro, M.; Arias, P. Automated Inspection of Railway Tunnels' Power Line Using LiDAR Point Clouds. *Remote Sens.* **2019**, *11*, 2567. [\[CrossRef\]](#)
15. Guo, F.; Qian, Y.; Rizos, D.C.; Suo, Z.; Chen, X. Automatic Rail Surface Defects Inspection Based on Mask R-CNN. *Transp. Res. Rec.* **2021**, 2675, 655–668.
16. Wang, Y.; Song, W.; Lou, Y.; Huang, F.; Tu, Z.; Zhang, S. Simultaneous Localization of Rail Vehicles and Mapping of Environment with Multiple LiDARs. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8186–8193. [\[CrossRef\]](#)
17. Wang, Y.; Lou, Y.; Song, W.; Tu, Z.; Wang, Y.; Zhang, S. Simultaneous Localization of Rail Vehicles and Mapping of Surroundings with LiDAR-Inertial-GNSS Integration. *IEEE Sens. J.* **2022**, *22*, 14501–14512. [\[CrossRef\]](#)
18. Dai, X.; Lou, Y.; Weiwei, S.; Tang, W.; Wang, Y. LiDAR-Inertial Integration for Rail Vehicle Localization and Mapping in Tunnels. *IEEE Sens. J.* **2023**, *23*, 17426–17438. [\[CrossRef\]](#)
19. Zhang, J.; Singh, S. LOAM: Lidar Odometry and Mapping in Real-time. *Robot. Sci. Syst.* **2014**, *2*, 1–9.
20. Shan, T.; Englot, B. LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 4758–4765. [\[CrossRef\]](#)
21. Lin, J.; Zhang, F. Loam livox: A fast, robust, high-precision LiDAR odometry and mapping package for LiDARs of small FoV. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 3126–3131. [\[CrossRef\]](#)
22. Shan, T.; Englot, B.; Meyers, D.; Wang, W.; Ratti, C.; Rus, D. LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 5135–5142. [\[CrossRef\]](#)
23. Wang, H.; Wang, C.; Chen, C.L.; Xie, L. F-LOAM: Fast LiDAR Odometry and Mapping. In Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Prague, Czech Republic, 27 September–1 October 2021; pp. 4390–4396. [\[CrossRef\]](#)
24. Bai, C.; Xiao, T.; Chen, Y.; Wang, H.; Zhang, F.; Gao, X. Faster-LIO: Lightweight Tightly Coupled Lidar-Inertial Odometry Using Parallel Sparse Incremental Voxels. *IEEE Robot. Autom. Lett.* **2022**, *7*, 4861–4868. [\[CrossRef\]](#)
25. Xu, W.; Zhang, F. FAST-LIO: A Fast, Robust LiDAR-Inertial Odometry Package by Tightly-Coupled Iterated Kalman Filter. *IEEE Robot. Autom. Lett.* **2021**, *6*, 3317–3324. [\[CrossRef\]](#)
26. Xu, W.; Cai, Y.; He, D.; Lin, J.; Zhang, F. FAST-LIO2: Fast Direct LiDAR-Inertial Odometry. *IEEE Trans. Robot.* **2022**, *38*, 2053–2073. [\[CrossRef\]](#)
27. Yuan, C.; Xu, W.; Liu, X.; Hong, X.; Zhang, F. Efficient and Probabilistic Adaptive Voxel Mapping for Accurate Online LiDAR Odometry. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8518–8525. [\[CrossRef\]](#)
28. Wu, C.; You, Y.; Yuan, Y.; Kong, X.; Zhang, Y.; Li, Q.; Zhao, K. VoxelMap++: Mergeable Voxel Mapping Method for Online LiDAR(-Inertial) Odometry. *IEEE Robot. Autom. Lett.* **2024**, *9*, 427–434. [\[CrossRef\]](#)
29. Li, Y.; Trinh, H.; Haas, N.; Otto, C.; Pankanti, S. Rail Component Detection, Optimization, and Assessment for Automatic Rail Track Inspection. *IEEE Trans. Intell. Transp. Syst.* **2014**, *15*, 760–770. [\[CrossRef\]](#)
30. Graham, B.; Engelcke, M.; Maaten, L.v.d. 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9224–9232. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.