Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.058757

ARTICLE

LiDAR-Visual SLAM with Integrated Semantic and Texture Information for Enhanced Ecological Monitoring Vehicle Localization

Yiqing Lu[1], **Liutao Zhao**[2,*] **and Qiankun Zhao**[3]

[1]Foreign Environmental Cooperation Center, Ministry of Ecology and Environment of PRC, Beijing, 100035, China

[2]Beijing Computing Center Co., Ltd., Beijing Academy of Science and Technology, Beijing, 100083, China

[3]Beijing Guyu Interactive Artificial Intelligence Application Co., Ltd., Beijing, 100024, China

*Corresponding Author: Liutao Zhao. Email: zhaolt@bcc.ac.cn

Received: 20 September 2024 Accepted: 18 November 2024 Published: 03 January 2025

ABSTRACT

Ecological monitoring vehicles are equipped with a range of sensors and monitoring devices designed to gather data on ecological and environmental factors. These vehicles are crucial in various fields, including environmental science research, ecological and environmental monitoring projects, disaster response, and emergency management. A key method employed in these vehicles for achieving high-precision positioning is LiDAR (lightlaser detection and ranging)-Visual Simultaneous Localization and Mapping (SLAM). However, maintaining high-precision localization in complex scenarios, such as degraded environments or when dynamic objects are present, remains a significant challenge. To address this issue, we integrate both semantic and texture information from LiDAR and cameras to enhance the robustness and efficiency of data registration. Specifically, semantic information simplifies the modeling of scene elements, reducing the reliance on dense point clouds, which can be less efficient. Meanwhile, visual texture information complements LiDAR-Visual localization by providing additional contextual details. By incorporating semantic and texture details from paired images and point clouds, we significantly improve the quality of data association, thereby increasing the success rate of localization. This approach not only enhances the operational capabilities of ecological monitoring vehicles in complex environments but also contributes to improving the overall efficiency and effectiveness of ecological monitoring and environmental protection efforts.

KEYWORDS

LiDAR-Visual; simultaneous localization and mapping; integrated semantic; texture information

1 Introduction

In today's rapidly evolving society, accelerated urbanization coupled with increasingly severe environmental issues has significantly amplified the importance of ecological monitoring vehicles. These vehicles have become indispensable tools in modern urban management and environmental protection [1,2]. Equipped with cutting-edge sensors and sophisticated data processing systems, they are capable of real-time data collection and analysis in both complex urban landscapes and dynamic natural environments. Looking forward, these vehicles are expected to play a crucial role in providing accurate and comprehensive services for species surveys and environmental monitoring.

Their onboard monitoring systems are multifaceted, including meteorological sensors, air quality monitors, thermometers, noise sensors, high-resolution cameras, and LIDAR sensors. This extensive array of sensors allows for a comprehensive, multidimensional approach to environmental data collection. Such capabilities are vital, offering essential scientific and technical support for a range of applications including urban planning, tracking pollution sources, and responding effectively to sudden environmental events [3].

Despite notable advancements in technology, ecological monitoring vehicles continue to confront several challenges in practical deployment, with the enhancement of positioning accuracy being a particularly critical issue [4]. Achieving precise real-time positioning in dynamically changing ecological environments remains a complex and challenging task, largely dependent on the efficacy of Simultaneous Localization and Mapping (SLAM) technology [5]. Although significant progress has been made, existing technologies have proven effective in addressing specific environmental changes, such as puddles and snowdrifts [6–8], the issue of positioning module failures induced by environmental dynamics persists.

This study aims to address these challenges by integrating LiDAR inertial odometry (LIO) with a global LiDAR localization module that is based on map matching techniques. These measurements are synthesized within a pose graph optimization framework, creating a robust localization system capable of handling temporary ecological and environmental changes or inaccuracies in the map, all while ensuring consistent global localization. This innovative approach provides the technical foundation necessary for effective monitoring and surveying of field species. The specific contributions of this paper are outlined as follows:

(1) Development of an integrated framework for the localization of ecological monitoring vehicles that dynamically merges global matching and local odometry information. This integration enhances the system's resilience against failures caused by fluctuating indoor environments and other dynamic conditions.

(2) Creation of a closely integrated LiDAR-Inertial Odometry system that leverages both occupancy data and LiDAR intensity information. This system is designed to deliver precise real-time state estimations, improving the accuracy and reliability of ecological monitoring.

(3) Establishment of a reliable ecological monitoring vehicle localization system that has been rigorously tested in various environments, including busy indoor streets and diverse outdoor settings. This testing demonstrates the system's robustness and adaptability to dynamically changing conditions, further validating its effectiveness for real-world applications.

2 Related Work

2.1 Long-Term Localization

Constructing a continuous 7-day localization system poses significant challenges. Wolcott et al. [6] addressed the issue of varying environmental conditions such as repavement and snow by utilizing a robust LiDAR system coupled with multi-resolution mapping techniques. This approach ensures the system remains reliable under diverse and dynamic conditions. Wan et al. [8] made strides in navigation by incorporating altitude cues, which assist in improving the accuracy of localization in varying terrains. Levinson et al. [9] tackled these challenges by normalizing LiDAR scans, which effectively reduces variations in reflectance that can otherwise lead to inconsistencies in the data. Meanwhile, Aldibaja et al. [7] focused on enhancing system robustness in adverse weather conditions, including rain and snow, by applying Principal Component Analysis (PCA) and edge profiles to better handle the noise and distortions associated with such environments. Our research aims to provide a more generalized solution by integrating odometry with global matching cues, offering a versatile approach to continuous localization. In contrast, other studies [10–14] have relied predominantly on vision sensors, which can be susceptible to significant performance degradation due to changes in appearance and lighting conditions.

2.2 LiDAR Inertial Odometry

Several studies have made notable contributions to LiDAR odometry and SLAM [15–17]. In particular, inertial data has been used to support motion estimation [18,19] and distortion correction [20–22], enhancing the precision of constraints [23,24] and contributing to the development of integrated odometry solutions [25,26]. Building on Hess's work [16], which highlighted the benefits of LiDAR-inertial odometry due to its alignment with map representation, our approach integrates inertial sensors to achieve improved performance and robustness. Drawing inspiration from previous research, we incorporate these inertial sensors to further enhance the accuracy and reliability of our system.

2.3 Localization Fusion Methods

Fusion methods are crucial in combining estimates from various sensors to improve overall performance. Loosely-coupled fusion techniques leverage complementary sensors such as Global Navigation Satellite System (GNSS) [27], cameras [28], odometers [29], and Inertial Measurement Units (IMUs) to achieve more accurate and reliable localization results. Some methodologies [8,30,31] implement Kalman filters to perform sensor fusion, which helps in combining data from different sources effectively. Our approach aligns with [31] by employing a graph-based framework, which offers computational efficiency and robust performance [32,33]. In a related study, the authors [34] demonstrated the effectiveness of a tightly-integrated system combining GNSS, LiDAR, and inertial sensors, showcasing the benefits of a holistic approach to sensor fusion for enhanced localization.

3 Method

This section provides a detailed overview of the proposed architecture for the Visual Boosted Registration Framework, as illustrated in Fig. 1. Our system is designed to accurately register the source frame with the target frame, emphasizing crucial aspects such as data association and the estimation of relative pose transformations. Each frame within our system is composed of multi-camera images with minimal overlap, accompanied by a 64-line laser point cloud. The images are 1920∗1080 pixel surround images after de-distortion, and the laser point cloud is the point cloud after removing motion distortion.

We assume that the camera intrinsics and sensor extrinsics are precisely calibrated, and that time synchronization with IMU compensation is accurately accounted for, ensuring the precision of pose transformations. Under these assumptions, we consider these parameters to be sufficient for our calculations and subsequent processing. The system ultimately outputs the pose transformation that defines the spatial relationship between the two frames after analyzing their input data. The framework comprises four key modules: Data Association, Scale-free Transform Estimate, Recover Scale, and Robust Optimization. Fig. 1 provides a comprehensive depiction of the workflow involved in these modules. Initially, the Data Association module is responsible for extracting and associating image keypoints and semantic objects from the laser point cloud. This process involves matching these keypoints and semantic objects between the source and target frames to generate a set of corresponding keypoint and semantic matching pairs. These matching pairs are then processed through the Scale-free Transform Estimate module, which computes the scale-free pose transformation between the frames. This transformation is performed without considering the absolute scale, focusing instead on the relative spatial arrangement of the keypoints and semantic objects. The subsequent Recover Scale module is employed to address the scale of the transformation, adjusting the computed pose to accurately reflect the true scale of the frames. Finally, the Robust Optimization module refines the pose transformation by incorporating robust optimization techniques to improve accuracy and handle any discrepancies or errors in the initial data.

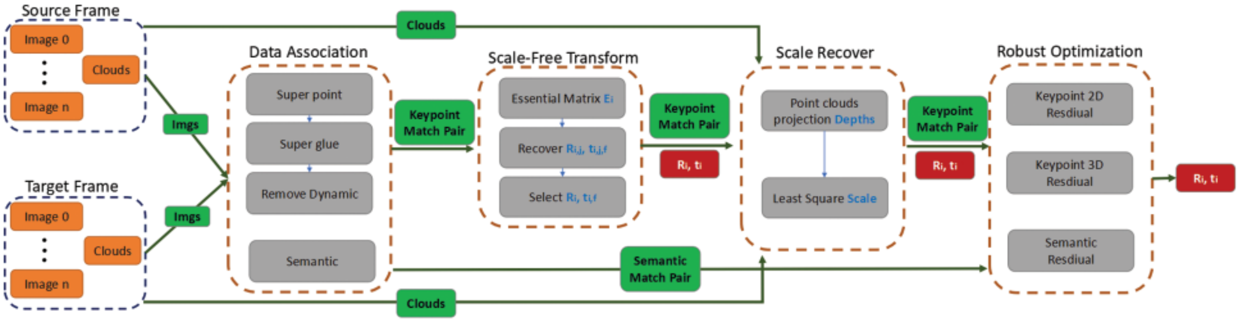

Figure 1: The proposed architecture

Overall, this workflow ensures that the registration process is both precise and resilient, leveraging the strengths of each module to produce an accurate pose transformation between the source and target frames. The pose transformation is recorded as (R_{st}, $t_{st,f}$) belong to the Special Orthogonal Group SO(3). Following the Scale Recover stage, the laser point cloud projection is used to estimate the depth of multiple image keypoints, and the scale S_{ts} of the pose transformation (R_{st}, $t_{st,f}$) is recovered to obtain the real scale pose transformation, which we recorded as (R_{st}, S_{ts}, t_{st}). Finally, the real scale pose transformation result is used as the initial value for robust optimization. We construct the optimization residuals based on keypoint and semantic matching pairs to obtain the final optimized pose transformation between the source and target frames. The following sections will provide a detailed introduction to these four system modules in sequence.

3.1 Data Association

Data association is a critical component for successful registration. A common approach involves iteratively finding nearest point correspondences within a predefined distance threshold, which requires an initial estimate of the relative pose between the datasets. However, associating data based solely on geometry poses significant challenges. The correspondence may be effective only for certain 3D geometric features or can lead to degeneracies in specific dimensions, as noted in Reference [35]. Texture information plays a crucial role in achieving stable correspondences, especially in environments with degenerate features, such as tunnels, highways, and complex urban scenes, where geometric features alone may not provide reliable matching.

(1) 2D Keypoints

Our method employs the SuperPoint [36] network to extract 2D keypoints from all images. To mitigate the impact of incorrect matches, we utilize the SuperGlue [37] network for improved matching accuracy across all pairs of images requiring keypoint matching. The presence of numerous moving objects in real-world road scenes can significantly affect the accuracy of keypoint-based calculations. To address this issue, we incorporate the SegNet [38] network for semantic segmentation to differentiate between dynamic and static objects. For each keypoint pair generated by SuperGlue, if either keypoint is found on a dynamic object, we discard the pair. Only the keypoint pairs that pass this semantic filtering are retained as final correspondences for further feature point-related calculations.

Once 2D-2D data association is established, we can recover the relative pose by decomposing the essential matrix. By associating LiDAR points with image features, we can estimate the scale using the depth information of each keypoint. These processes, including the detailed steps for pose recovery and scale estimation, are elaborated upon in the subsequent sections.

(2) Semantic Objects

In addition to determining the initial relative pose, another critical aspect of data association for registration is the quality of the correspondence ratio. In practice, texture-based data association often focuses on ground features, while distinctive landmarks in the point cloud can be utilized to construct correspondences that help constrain the horizontal dimension. We establish correspondences for the same landmark, such as road signs or pillars, ensuring both semantic and geometric consistency.

We parameterize semantic objects as primitives, such as lines or planes, and align them using corresponding distance errors. To achieve accurate object correspondences, we employ a Random Sample Consensus-based (RANSAC-based) approach to identify the closest (within a specified threshold distance) and unique landmarks across frame pairs. Typically, poles are particularly distinctive for lane lines. After determining an initial relative translation, this translation is used to search for additional correspondences by examining all semantic objects within the frame pairs.

3.2 Scale-Free Transform

Since 2D keypoints in images do not provide accurate depth information, directly applying methods such as PnP (Perspective-n-Point) or ICP (Iterative Closest Point) can result in significant errors. To address this issue, our approach first estimates the rotation and scale-free translation matrices using the Essential Matrix. Subsequently, we use the projected depth values to estimate the translation scale.

In order to get the scale-free pose transformation, we use the 2D Keypoint matching results to estimate the Essential matrix (E) between two images. After that, we inversely solve the rotation R and translation t (normalized results) of the pose transformation. For multi-camera scenes, we compute E_i, R_i, and t_i for each image matching pair. In order to use the multi-camera model to obtain a more accurate pose transformation, for each estimated R_i, and t_i, we use the external parameters between multiple cameras to get $R_{i,j}$, and $t_{i,j}$ (normalized results) of several other matching pairs. Then we calculate the fundamental matrix (F) according to Eq. (1), where K is the camera intrinsic.

$$F = K_1^{-T} t^{\wedge} R K_2^{-1} \tag{1}$$

where

$$t^{\wedge} = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$$

After that, the Fundamental Matrix is used to estimate the average distance between the two feature points and the respective baselines, and the feature points within the threshold are set as inlier points (P_1, P_2 represents the undistorted pixel position of thefeature point), as shown in Eq. (2). The result with the largest number of inlier points in multiple R_i and t_i is used as the estimated scale-free pose transformation, which are recorded as R_i, t_i (normalized results).

$$R_i, t_i = ArgMin(Err_{mean})$$
$$Err_{mean} = \frac{1}{2}\left(\frac{\left|P_2^T F P_1\right|}{\left\|P_2^T F\,|0{:}\,2|\right\|_2} + \frac{\left|P_2^T F P_1\right|}{\|FP_1\,|0{:}\,2|\|_2}\right) \tag{2}$$

3.3 Scale Recover

Since the previously restored pose transformations R_i and t_i are scale-free, they cannot directly participate in pose optimization. Therefore, we propose to use the true depth of Keypoints from multiple cameras to recover scale.

The first is to use the laser point cloud to estimate the depth of the 2D Keypoints in the images. We project the laser point cloud onto the image according to the extrinsics between the cameras and the Laser sensor and the instrinsics of the cameras according to Eq. (2). R_{Cl} and t_{Cl} represent the pose transformation from the Camera to the Laser sensor.

$$\pi(P_{ix}) = -\frac{1}{point.z} * K * (R_{\mathrm{Cl}} * Pt + t_{\mathrm{Cl}}) \tag{3}$$

By projection the point cloud to image plane, we can associate the 2D keypoint to the 3D plane patch, and obtain a good depth estimation for matched 2D keypoints. After obtaining the depth estimation results of multiple keypoints, we start to recover the scale later. According to Section 3.2, the scale-free pose transformations E_i, R_i, t_i between the corresponding multiple cameras in the source submap and the target submap are obtained. Then we set the actual scale result as s_i. We jointly use the Keypoints in multiple camera to estimate s_0. We assuming that R^s_{ik} and t^s_{ik} represent the transformation between the i-th camera and the k-th camera in the source frames and R^t_{ik} and t^t_{ik} represent the transformation between the i-th camera and the k-th camera in the target frames. According to the relationship between the camera extrinsic parameters, we can get:

$$\begin{bmatrix} R_{ik} & t_{ik} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R^s_{ki} & t^s_{ki} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_i & \widehat{s_i t_i} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R^t_{ik} & t^t_{ik} \\ 0 & 1 \end{bmatrix} \tag{4}$$

where R^s_{ki}, t^s_{ki} represents the extrinsic reference between the K-th image and the i-th image in MI_s, and R^t_{ik}, t^t_{ik} represents the extrinsic reference between the i-th image and the k-th image in MI_t.

Thus, for the j-th matching feature point pair u^s_{kj} and u^t_{kj} in the k-th image pair in MI_s and MI_t, we can get:

$$\lambda^s_{k,j} u^s_{kj} = R_{ik} \lambda^t_{k,j} u^t_{kj} + t_{ik} \tag{5}$$

Among them, λ^s_{kj} represents the actual depth of u^s_{kj}, and λ^t_{kj} represents the actual depth of u^t_{kj}. According to the previous method of generating Mappoints, some Mappoints correspond to image feature points, and these image feature points will have known depth values λ^s_{kj} or $\lambda^t_{k,j}$.

If λ^t_{kj} is known:

$$\lambda^s_{kj} u^s_{kj} = R_{ik} \lambda^t_{k,j} u^t_{kj} + t^s_{ki} + R^s_{ki} \mathrm{R}_i t^t_{ik} + s_i R^s_{ki} \hat{t}_i \tag{6}$$

$$s_i = \frac{u^s_{kj} \times (R_{ik} \lambda^t_{kj} u^t_{kj} + t^s_{ki} + R^s_{ki} \mathrm{R}_i t^t_{ik})}{u^s_{kj} \times R^s_{ki} \hat{t}_i} \tag{7}$$

The same reason, if λ_{kj}^{t} is known:

$$s_i = \frac{R_{ik}u_{kj}^{t} \times (\lambda_{kj}^{s}u_{kj}^{s} - t_{ki}^{s} - R_{ki}^{s}\mathrm{R}_i t_{ik}^{t})}{\lambda_{kj}^{s}u_{kj}^{s} \times R_{ki}^{s}\hat{t}_i} \tag{8}$$

Since the scale value s_i is a scalar, the result is very sensitive to noise, and our method of restoring the scale only requires more than one image feature point with accurate depth value to calculate the result. Therefore, we only use the image features corresponding to the Mappoint with higher confidence to participate in the scale restoration:

$$\begin{cases} \max_{i\in \mathtt{C}} \|\pi(pt) - u_i\|_2 < \theta_0 \\ \frac{1}{N_c}\sum_{i\in \mathtt{C}} \left\|\hat{\lambda}_i - \lambda_i\right\| < \theta_1 \end{cases} \tag{9}$$

pt represents the coordinates of the current Mappoint, $\mathtt{C}$ represents the set of image feature points corresponding to the 3D feature point, u_i represents the pixel position of the feature point, λ_i represents the depth of the feature point restored by the point cloud, $\hat{\lambda}_i$ represents the estimated depth of the feature point, and N_c represents the modulus of the set $\mathtt{C}$. This formula indicates that only when the reprojection error of the image feature corresponding to the current Mappoint and the depth error of the point cloud restoration are small enough, the Mappoint will be considered to have high confidence and participate in the calculation of scale recovery, so that a more accurate scale result can be obtained. In our experiment, we set reprojection error threshold θ_0 as 5 pixel and depth error threshold θ_1 as 0.25 m.

For each $s_i(i = 1, 2...m)$, we can get several formulas similar to the above s_i according to the image features corresponding to the selected Mappoint, and get a one-dimensional overdetermined equation. We use the least squares solution to get the final restored scale s_i.

In this way, for the previously estimated $R_0, R_1 \ldots R_m$, and $\hat{t}_0, \hat{t}_1 \ldots \hat{t}_m$, the corresponding $s_0, s_1 \ldots s_m$ can be estimated. We get the results of m pose transformations. In order to obtain a more accurate pose transformation, we need to screen these $[R_i, s_i\hat{t}_i]$.

We filter by counting the number of reprojection errors less than the threshold. For each $\left[R_i, s_i\hat{t}_i\right]$, we calculate the error between the position of all 3D feature points pt_l^{s} in MI_s projected on the image of MI_t and the corresponding 2D feature point pixel position u_l^{t}, and count the number of points less than the threshold. Similarly, the number of points less than the threshold between the position of all 3D feature points pt_l^{t} in MI_t projected on the image of MI_s and the corresponding 2D feature point pixel position u_l^{s}:

$$\underset{\left[R_i, s_i\hat{t}_i\right]}{\mathrm{Argmax}}[\underset{l\in MI_s}{\mathcal{N}}(\left\|\pi(pt_l^{s}) - u_l^{t}\right\|_2 < \theta_2) + \underset{l\in MI_t}{\mathcal{N}}(\left\|\pi(pt_l^{t}) - u_l^{s}\right\|_2 < \theta_2)] \tag{10}$$

The set with the largest sum of the two quantities $\left[R_i, s_i\hat{t}_i\right]$ is considered to be the restored pose transformation, and the pose transformation result $[R_{\mathcal{L}}, t_{\mathcal{L}}]$ of the point cloud between the Source Frame and the Target Frame can be obtained according to the external parameters in the Frame, where $[R_{\mathcal{L}i}^{s}, t_{\mathcal{L}i}^{s}]$ represents the external parameters from camera i to the point cloud in the Source Frame, and $[R_{i\mathcal{L}}^{t}, t_{i\mathcal{L}}^{t}]$ represents the external parameters from the point cloud to camera i in the Target Frame:

$$\begin{bmatrix} R_{\mathcal{L}} & t_{\mathcal{L}} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{\mathcal{L}i}^{s} & t_{\mathcal{L}i}^{s} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_i & s_i\hat{t}_i \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_{i\mathcal{L}}^{t} & t_{i\mathcal{L}}^{t} \\ 0 & 1 \end{bmatrix} \tag{11}$$

3.4 Robust Optimization

In the robust optimization stage, the residuals term of our method is mainly divided into two parts: keypoint residuals and semantic residuals.

For keypoint residuals, we mainly use the pixel error as the residual term to optimize the pose transformation. In the matching results of multiple camera Keypoints obtained above, only some of the Keypoints have depth values. For the convenience of recording, for the multiple camera images of the source frame, the set of Keypoints without depth is recorded as δ_s, and the set of Keypoints without depth in the images of the target frame is recorded as δ_t. In contrast, the set of Keypoints with depth in the images of source frame and target frame are recorded as γ_s and γ_t, respectively.

In order to obtain a more accurate pose transformation estimation, we designed residual items for both feature points to participate in the final optimization. As a result, the error term mainly includes two items, namely 2D error r_{2d} for Keypoints without depth and 3D error r_{3d} for Keypoints with depth.

For 2D Keypoints in δ_s or δ_t without depth, the estimated R and t are using to infer the fundamental matrix using Eq. (1) and calculate the distance from the Keypoint to the baseline as the residual. Among the formula, P_{ij} represents the 2D pixel position of the j-th Keypoint of the i-th image in the source frame, and $P^{'}_{ij}$ represents 2D pixel location of the matching Keypoint in the target frame. And F_i represents the fundamental matrix recovered by R_i, $s_i t_i$.

$$e_{i,j} = \left| P_{ij}^T F_i P^{'}_{ij} \right| \tag{12}$$

$$\begin{cases} l^s_{i,j} = \left\| P_{ij}^T F_i [0:2] \right\|_2 \\ l^t_{i,j} = \left\| F_i P^{'}_{ij} [0:2] \right\|_2 \end{cases} \tag{13}$$

$$r_{2d} = \sum\nolimits_{i=1}^{n} \sum\nolimits_{j\in\delta_s} \left| \frac{e_{i,j}}{l^s_{i,j}} \right|^2 + \sum\nolimits_{i=1}^{n} \sum\nolimits_{j\in\delta_t} \left| \frac{e_{i,j}}{l^t_{i,j}} \right|^2 \tag{14}$$

$$r_{3d} = \sum\nolimits_{i=1}^{n} \sum\nolimits_{j\in\gamma_s} \left\| \pi(Pt_{ij}) \right\|_2^2 + \sum\nolimits_{i=1}^{n} \sum\nolimits_{j\in\gamma_t} \left\| \pi(Pt^{'}_{ij} - P_{ij}) \right\|_2^2 \tag{15}$$

For other 2D Keypoints in γ_s or γ_t with depth, we calculate the reprojection error of 3D Keypoint to 2D image pixels as residual. As shown in Eq. (13), Pt_{ij} represents the 3D coordinate of the j-th Keypoint of the i-th image in the source frame, and $Pt^{'}_{ij}$ represents 3D coordinate of the matching Keypoint in the target frame. $\pi()$ represents projecting the 3D point to the 2D pixel in the image plane.

Semantic objects are parameterized to primitives as line or plane and then align them with corresponding distance error. Among them, $\mathbf{n}_i$ represents the direction of the Lane, $\mathbf{p}_i$ represents the three-dimensional coordinates of the center point of the element. $\mathbf{m}_i$ represents the direction of the Lane, $\mathbf{q}_i$ represents the three-dimensional coordinates of the center point of the element.

$$\begin{cases} \mathbf{r}_P^C = (\bar{\mathbf{p}}_i - \mathbf{p}_j) \cdot \bar{\mathbf{n}}_j \\ \mathbf{r}_L^C = (\bar{\mathbf{q}}_i - \mathbf{q}_j) \times \bar{\mathbf{v}}_j \end{cases} \tag{16}$$

$$\begin{cases} \mathbf{r}_P^O = \bar{\mathbf{n}}_i \times \bar{\mathbf{n}}_j \\ \mathbf{r}_L^O = \bar{\mathbf{v}}_i \times \bar{\mathbf{v}}_j \end{cases} \tag{17}$$

The semantic object residual shown in Fig. 2 can written as

$$r_s = \sum\nolimits_P \left(\omega_0 \left(r_P^C\right)^2 + \omega_1 \left\| \mathbf{r}_P^O \right\|_2^2\right) + \sum\nolimits_L \left(\omega_0 \left\| \mathbf{r}_L^C \right\|_2^2 + \omega_1 \left\| \mathbf{r}_L^O \right\|_2^2\right) \tag{18}$$

and the symbol definition is shown in Table 1.

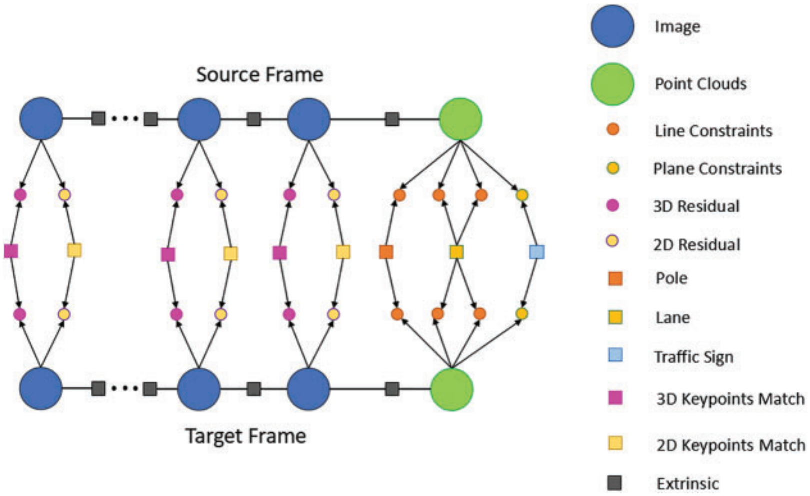

Figure 2: Graph structure of semantic objective

Table 1: The constraints use by each primitives

Primitive	Location		Orientation	
	Constraints	Covariance	Constraints	Covariance
Lane	$\mathbf{r}_L^C$	Σ_L	$\mathbf{r}_L^O$	Σ_P
Pole	$\mathbf{r}_P^C$	Σ_L	$\mathbf{r}_P^O$	Σ_P

After jointly optimizing all these residuals, the final high-precision positioning can be obtained, In actual experiments, we set ω_0 as 0.1, ω_1 as 3.0, and ω_2 as 1.5:

$$r = \omega_0 r_{2d} + \omega_1 r_{3d} + \omega_2 r_s \tag{19}$$

The overall framework is shown in Algorithm 1.

Algorithm 1: Visual registration

```
Input: Target Images $I_{t1}, \ldots, I_{t6}$, Cloud $C_t$. Source Images $I_{s1}, \ldots, I_{s6}$, Cloud $C_s$.
Output: Relative Transform $R, t$.
    $\{I_{ti}, I_{si}\}$ → SuperPoint → 2D keypoints $\{u_{ti}, u_{si}\}$
    Cloud Projection → keypoints depth $\{\lambda_{ti}, \lambda_{si}\}$
    Keypoint Match → Mappoints $\{Mp_i\}$
    Triangulate → supplement keypoints depth $\{\lambda_{ti}, \lambda_{si}\}$
    Calcualte:
        Eq. (2) to optimization no scale transform $R_i, t_i$
    For $Mp_i$ in Mappoints $\{Mp_i\}$:
        Eq. (8) to calculate scale $s_i$.
        Eq. (10) to count the number $N_i$ of mappoints that meet the threshold.
```

(Continued)

Algorithm 1 (continued)

```
    $s = \mathbf{Argmax}(\{N_i\})$,
    **Initial Pose:** $R_i, st_i$
    **Optimization:** Eq. (19) to optimization precise result $R, t$.
```

4 Experimental Results

4.1 Platforms and Datasets

Our system has been extensively tested in real-world scenarios primarily using two datasets, the Kitti dataset [39] and our internal hostital dataset.

The KITTI SLAM dataset, developed by the Karlsruhe Institute of Technology and the Toyota Technological Institute at Chicago, is a widely-used resource for benchmarking SLAM (Simultaneous Localization and Mapping) algorithms. It consists of data captured from a car equipped with multiple sensors, including high-resolution stereo cameras, a Velodyne 3D laser scanner, and GPS/IMU, providing rich information for 3D reconstruction and localization tasks. The dataset covers a variety of driving scenarios such as urban, rural, and highway environments, and includes ground truth poses obtained from GPS for accurate evaluation. With its diverse and challenging sequences, the KITTI SLAM dataset is essential for developing and testing autonomous driving and computer vision algorithms. It is publicly accessible and comes with extensive tools and benchmarks for performance evaluation.

Our Explainable-AI healthcare service robot platform is equipped with a Velodyne HDL-64E 360° LiDAR and a NovAtel PwrPak7D-E1 GNSS RTK receiver integrated with dual antennas and an Epson EG320N IMU. The ground truth poses used in the evaluation are generated using offline LiDAR SLAM methods typically formulated as a large-scale global least-square optimization problem, which are beyond the scope of this work.

4.2 Performance

We use the error between the predicted and true values as the primary criterion for determining the success of localization. Specifically, localization is deemed successful when the difference in the rotation angle is within 10 degrees, and the translation error remains within 50 cm. This criterion is essential for ensuring the accuracy and reliability of localization systems. As illustrated in Tables 2 and 3, our evaluation focuses primarily on the Root Mean Square Error (RMSE) of the relative translation error (RTE) and the relative rotation error (RRE), alongside the recall rate.

Table 2: Localition peformence in kitti dataset

Method	RTE (m)	RPE (°)	Recall (%)
ICP [40]	0.0624	0.2013	25.516
GICP [41]	0.0602	0.2381	45.883
DGR [42]	0.0986	0.2840	95.504
D3Feat [43]	0.0919	0.6096	99.946
PCAM [44]	0.1079	0.6014	99.296

(Continued)

Table 2 (continued)

Method	RTE (m)	RPE (°)	Recall (%)
Predator [45]	0.0938	0.6055	99.892
HregNet [46]	0.1277	0.6132	96.187
VLIS (Ours)	**0.1895**	**0.1944**	**99.187**

Table 3: Localition peformence in ecological park dataset

Method	RTE (m)	RPE (°)	Recall (%)
ICP [40]	0.2354	0.4949	3.677
GICP [41]	0.0902	0.1187	10.477
DGR [42]	0.1058	0.7790	65.928
D3Feat [43]	0.1112	0.8111	93.701
PCAM [44]	0.1238	0.7561	68.100
Predator [45]	0.1014	0.8195	94.433
HregNet [46]	0.0967	1.1623	95.598
VLIS (Ours)	**0.0796**	**0.0731**	**96.243**

In the context of the KITTI dataset, our proposed method demonstrates the highest level of precision among all algorithms assessed. This exceptional performance is reflected not only in the low RMSE values for both translation and rotation errors but also in a recall rate that stands competitively against leading methods in the field. Our approach achieves superior results when tested on our internal hostile dataset, showcasing a remarkable ability to handle challenging conditions. This includes excelling in both translation accuracy and angular error measurements, further validating the robustness and effectiveness of our method. The comprehensive evaluation across different datasets highlights the strength of our approach in real-world scenarios. The KITTI dataset, known for its diverse driving conditions, serves as a rigorous benchmark, and the exceptional performance of our method underscores its capability to provide reliable localization under varied and complex conditions. Additionally, the results on our internal hostile dataset emphasize our method's robustness in adverse situations, which is crucial for practical applications in autonomous driving and related fields.

In summary, the performance metrics detailed in Tables 2 and 3 underscore the effectiveness of our localization method. By achieving the highest precision in translation and rotation errors within the KITTI dataset and excelling on our internal hostile dataset, our approach demonstrates its superiority and reliability. This positions our method as a leading solution in the field, offering significant advancements in localization accuracy and robustness.

Our method has demonstrated improved performance on the local Ecological Park Datasets, which can be attributed primarily to the more advantageous external parameters provided by the Baidu dataset compared to the KITTI dataset. The Baidu dataset benefits from the inclusion of a greater number of cameras, specifically featuring a front view camera, a left view camera, and three additional cameras positioned at the rear and right rear. This increased number of cameras contributes to a wider field of view, which enhances the overall data richness and enables more accurate localization.

Furthermore, to facilitate a more nuanced comparison of success rates across different methods, we have plotted the recall rates of each method as the error threshold varies. This comparative analysis provides a clearer understanding of how different approaches perform under varying conditions. The results from the KITTI dataset and the Ecological Park Dataset are illustrated in Fig. 3, showcasing the effectiveness of our method in both scenarios.

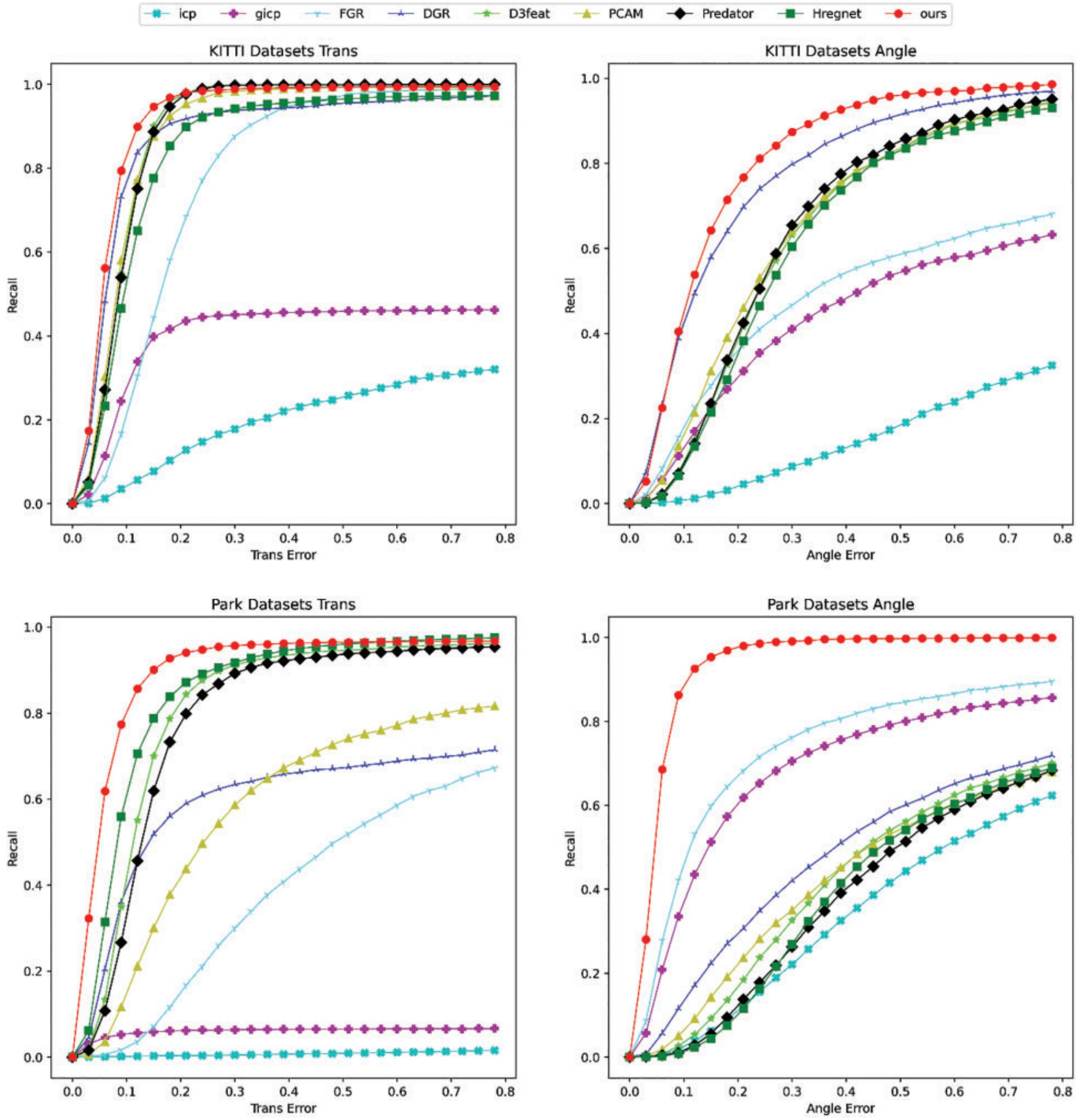

Figure 3: The recall rate of each method as the error threshold changes

The improved performance of the Ecological Park Datasets is indicative of the enhanced capabilities of our approach when supported by superior data quality and more comprehensive sensor setups. The wider field of view and the additional camera perspectives provided by the Baidu dataset play a crucial role in refining the accuracy of localization, demonstrating the advantage of using more detailed and varied data sources.

In summary, the comparison between the KITTI dataset and the Ecological Park Dataset highlights the benefits of using datasets with better external parameters and more extensive camera setups. By illustrating the recall rates as the error thresholds shift, Fig. 3 provides a valuable visual representation of our method's performance, further emphasizing its effectiveness and reliability across different datasets.

Compared with traditional registration methods, our method does not rely on initial values and can still obtain relatively good results when the difference is more than 10 m. Compared with deep learning methods, it has better generalization and does not require separate training. In addition, this registration algorithm is far better than other methods in the registration of weak structure and strong texture areas.

However, in the experiment, we also found some disadvantages of our method, such as the possibility of misregistration in repeated problem areas. Since a large number of mutual transformations between multiple sensors are used, the accuracy of external parameters is required to be quite high. These issues can be considered for optimization in subsequent work.

5 Conclusion

This paper introduces a robust LiDAR localization framework specifically designed for ecological monitoring vehicles to address localization issues in dynamic environments. The proposed approach employs a pose graph-based fusion framework that adaptively integrates both LiDAR semantic information and visual key points. Compared with traditional registration methods, the method we propose does not rely on the initial value and can still obtain better results when the difference is more than 10 m. Compared with deep learning methods, it has better generalization and does not require separate training. At the same time, the registration effect will be better for areas with weak structures and strong textures. The method we proposed proves that the solution of combining laser and vision for positioning can effectively improve the positioning accuracy and success rate, thereby further improving ecological monitoring vehicle's accuracy, and making it better for pollution monitoring and species survey and monitoring. In the future, we will consider researching more about the training or fine-tuning of the neural networks used (SuperPoint, SuperGlue, SegNet) to further improve the localization system.

Acknowledgement: This paper is supported by the project "Strengthening Coordinated Approaches to Reduce Invasive Alien Species (lAS) Threats to Globally Significant Agrobiodiversity and Agroecosystems in China". This paper has received funding from the Excellent Talent Training Funding Project in Dongcheng District, Beijing.

Funding Statement: This paper is supported by the project "GEF9874: Strengthening Coordinated Approaches to Reduce Invasive Alien Species (lAS) Threats to Globally Significant Agrobiodiversity and Agroecosystems in China". This paper has received funding from the Excellent Talent Training Funding Project in Dongcheng District, Beijing, with project number 2024-dchrcpyzz-9.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Yiqing Lu, Liutao Zhao; data collection: Yiqing Lu; analysis and interpretation of results: Yiqing Lu, Liutao Zhao, Qiankun Zhao; draft manuscript preparation: Liutao Zhao, Qiankun Zhao. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets generated during and/or analyzed during the current study are not publicly available due to personal privacy reasons but are available from the corresponding author on reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest toreport regarding the present study.

References

[1] Q. Tao, Z. Hu, G. Lai, J. Wan, and Q. Chen, "SMLAD: Simultaneous matching, localization, and detection for intelligent vehicle from LiDAR map with semantic likelihood model," *IEEE Trans. Vehicular Technol.*, vol. 73, no. 2, pp. 1857–1867, 2024. doi: 10.1109/TVT.2023.3321079.

[2] J. Song, Y. Chen, X. Liu, and N. Zheng, "Efficient LiDAR/inertial-based localization with prior map for autonomous robots," *Intell. Serv. Robot.*, vol. 17, no. 2, pp. 119–133, 2024. doi: 10.1007/s11370-023-00490-6.

[3] S. Hong, J. He, X. Zheng, and C. Zheng, "LIV-GaussMap: LiDAR-inertial-visual fusion for real-time 3D radiance field map rendering," *IEEE Robot. Autom. Lett.*, vol. 9, no. 11, pp. 9765–9772, 2024. doi: 10.1109/LRA.2024.3400149.

[4] C. Lee, S. Hur, D. Kim, Y. Yang, and D. Choi, "Insufficient environmental information indoor localization of mecanum mobile platform using wheel-visual-inertial odometry," *J. Mech. Sci. Technol.*, vol. 38, no. 9, pp. 5007–5015, 2024. doi: 10.1007/s12206-024-0836-z.

[5] H. Pham-Quang, H. Tran-Ngoc, T. Nguyen-Thanh, and V. Dinh-Quang, "Online robust sliding-windowed lidar slam in natural environments," in *2021 Int. Symp. Elect. Electr. Eng. (ISEE)*, Ho Chi Minh, Vietnam, 2021, pp. 172–177.

[6] R. W. Wolcott and R. M. Eustice, "Fast LiDAR localization using multiresolution gaussian mixture maps," in *IEEE Int. Conf. Robot. Automat.*, 2015, pp. 2814–2821. doi: 10.1109/ICRA.2015.7139582.

[7] M. Aldibaja, N. Suganuma, and K. Yoneda, "Robust intensity-based localization method for autonomous driving on snow-wet road surface," *IEEE Trans. Ind. Inform.*, vol. 13, no. 5, pp. 2369–2378, 2017. doi: 10.1109/TII.2017.2713836.

[8] G. Wan *et al.*, "Robust and precise vehicle localization based on multi-sensor fusion in diverse city scenes," in *Proc. IEEE Int. Conf. Robot. Automa. (ICRA)*, 2018, pp. 4670–4677.

[9] J. Levinson, M. Montemerlo, and S. Thrun, "Map-based precision vehicle localization in urban environments," in *Proc. Robot.: Sci. Syst.*, 2007.

[10] C. McManus, W. Churchill, W. Maddern, A. D. Stewart, and P. Newman, "Shady dealings: Robust, long-term visual localisation using illumination invariance," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2014, pp. 901–906.

[11] C. Linegar, W. Churchill, and P. Newman, "Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2015, pp. 90–97.

[12] C. Linegar, W. Churchill, and P. Newman, "Made to measure: Bespoke landmarks for 24-hour, all-weather localisation with a camera," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2016, pp. 787–794.

[13] M. Brki, I. Gilitschenski, E. Stumm, R. Siegwart, and J. Nieto, "Appearance-based landmark selection for efficient long-term visual localization," in *Proc. IEEE Int. Conf. Intell. Rob. Syst. (IROS)*, 2016, pp. 4137–4143.

[14] T. Sattler *et al.*, "Bench- marking 6DOF outdoor visual localization in changing conditions," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8601–8610.

[15] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot.: Sci. Syst. (RSS)*, 2014.

[16] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LiDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 1271– 1278.

[17] J. Zhang and S. Singh, "Low-drift and real-time LiDAR odometry and mapping," *Auton. Robots*, vol. 41, no. 2, pp. 401–416, 2017. doi: 10.1007/s10514-016-9548-2.

[18] C. Park, S. Kim, P. Moghadam, C. Fookes, and S. Sridharan, "Probabilistic surfel fusion for dense LiDAR mapping," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 2418–2426.

[19] J. Behley and C. Stachniss, "Efficient surfel-based SLAM using 3D laser range data in urban environments," in *Proc. Robot.: Sci. Syst. (RSS)*, 2018.

[20] D. Droeschel and S. Behnke, "Efficient continuous-time SLAM for 3D LiDAR-based online mapping," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1–9.

[21] C. Park, P. Moghadam, S. Kim, A. Elfes, C. Fookes and S. Sridharan, "Elastic LiDAR fusion: Dense map-centric continuous-time SLAM," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2018, pp. 1206–1213.

[22] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized LiDAR odometry and mapping on variable terrain," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst. (IROS)*, 2018, pp. 4758–4765.

[23] J. Deschaud, "IMLS-SLAM: Scan-to-model matching based on 3D data," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2018, pp. 2480–2485.

[24] Q. Li *et al.*, "Net: Deep real-time LiDAR odometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 8473–8482.

[25] J. Fan, X. Yang, R. Lu, Q. Li, and S. Wang, "Multi-modal scene matching location algorithm based on M2Det," *Comput. Mater. Contin.*, vol. 77, no. 1, pp. 1031–1052, 2023. doi: 10.32604/cmc.2023.039582.

[26] C. Qin, H. Ye, C. E. Pranata, J. Han, and M. Liu, "LINS: A LiDAR-Inerital state estimator for robust and fast navigation," 2019, *arXiv:1907.02233*.

[27] L. Chang, X. Niu, T. Liu, J. Tang, and C. Qian, "GNSS/INS/LiDAR- SLAM integrated navigation system based on graph optimization," *Remote Sens.*, vol. 11, no. 9, 2019, Art. no. 1009.

[28] R. Mur-Artal and J. D. Tards, "Orb-slam2: An open-source slam system formonocular, stereo, and rgb-d cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, 2017. doi: 10.1109/TRO.2017.2705103.

[29] H. Liu, M. Chen, G. Zhang, H. Bao, and Y. Bao, "ICE-BA: Incremental, consistent and efficient bundle adjustment for visual-inertial slam," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1974–1982.

[30] Y. Gao, S. Liu, M. Atia, and A. Noureldin, "INS/GPS/LiDAR integrated navigation system for urban and indoor environments using hybrid scan matching algorithm," *Sensors*, vol. 15, no. 9, pp. 23286–23302, 2015. doi: 10.3390/s150923286.

[31] H. Liu, Q. Ye, H. Wang, L. Chen, and J. Yang, "A precise and robust segmentation-based LiDAR localization system for automated urban driving," *Remote Sens.*, vol. 11, no. 11, 2019, Art. no. 1348. doi: 10.3390/rs11111348.

[32] H. Zhang *et al.*, "A LiDAR-INS-aided geometry-based cycle slip resolution for intelligent vehicle in urban environment with long-term satellite signal loss," *GPS Solut.*, vol. 28, 2024, Art. no. 61. doi: 10.1007/s10291-023-01597-0.

[33] H. Strasdat, J. Montiel, and A. J. Davison, "Real-time monocular SLAM: Why filter?" in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2010, pp. 2657–2664.

[34] T. S. Teoh, P. P. Em, and N. A. B. A. Aziz, "Vehicle localization based On IMU, OBD2, and GNSS sensor fusion using extended kalman filter," *Int. J. Technol.*, vol. 14, no. 6, pp. 1237–1246, 2023. doi: 10.14716/ijtech.v14i6.6649.

[35] W. Zhen, H. Yu, Y. Hu, and S. Scherer, "Unified representation of geometric primitives for graph-slam optimization using decomposed quadrics," 2021, *arXiv:2108.08957*.

[36] D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superpoint: Self- supervised interest point detection and description,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recogni. Workshops*, 2018, pp. 224–236.

[37] P. -E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, “Su-perglue: Learning feature matching with graph neural networks,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4938–4947.

[38] G. Zhou, F. Huang, W. Liu, Y. Zhang, H. Wei and X. Hou, “AWLC: Adaptive weighted loop closure for SLAM with multi-modal sensor fusion,” *J. Circuit Syst. Comp.*, vol. 33, no. 13, 2024, Art. no. 2450233. doi: 10.1142/S0218126624502335.

[39] L. Gao, X. Xia, Z. Zheng, and J. Ma, “GNSS/IMU/LiDAR fusion for vehicle localization in urban driving environments within a consensus framework,” *Mech. Syst. Signal. Process.*, vol. 205, 2023, Art. no. 110862. doi: 10.1016/j.ymssp.2023.110862.

[40] P. J. Besl and N. D. McKay, “Method for registration of 3-D shapes,” in *Sens. Fus. IV: Cont. Paradig. Data Struct.*, Boston, MA, USA, 1992, vol. 1611, pp. 586–606.

[41] A. Segal, D. Haehnel, and S. Thrun, “Generalized-icp,” in *Robot.: Sci. Syst.*, Seattle, WA, USA, 2009.

[42] C. Choy, W. Dong, and V. Koltun, “Deep global registration,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2514–2523.

[43] X. Bai, Z. Luo, L. Zhou, H. Fu, L. Quan and C. -L. Tai, “D3Feat: Joint learning of dense detection and description of 3D local features,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6359–6367.

[44] A. -Q. Cao, G. Puy, A. Boulch, and R. Marlet, “PCAM: Product of cross- attention matrices for rigid registration of point clouds,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 13229–13238.

[45] S. Huang, Z. Gojcic, M. Usvyatsov, A. Wieser, and K. Schindler, “PREDATOR: Registration of 3D point clouds with low overlap-supplementary material,” 2021, *arXiv:2011.13005*.

[46] F. Lu *et al.*, “HRegNet: A hierarchical network for large-scale outdoor lidar point cloud registration,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 16014–16023.