QuadricsReg: Large-Scale Point Cloud Registration Using Semantic Quadric Primitives

Ji Wu, Huai Yu, *Member, IEEE*, Shu Han, Ximeng Cai, Mingfeng Wang, Wen Yang, *Senior Member, IEEE*, and Gui-Song Xia, *Senior Member, IEEE*

***Abstract*—Designing an effective and scalable scene primitive representation is fundamental for large-scale point cloud registration. Existing studies that predominantly rely on dense point clouds or single-type geometric primitives struggle to scale to scenes characterized by massive data volume, structural diversity, and wide viewpoint variations. To address these, this article introduces *QuadricsReg*, a novel point cloud registration framework based on quadric primitives for large-scale environments. We compactly model diverse scene structures within a unified semantic quadric formulation, achieving high compression while preserving geometric richness and discriminability. This representation enables efficient quadric matching initialization via intrinsic similarity and robust correspondence pruning by maximizing geometric consistency in a multilevel graph, ensuring reliable associations even under large viewpoint variations. Furthermore, we design a factor graph based on degeneracy-aware quadric residual to estimate the transformation, ensuring accurate alignment in heterogeneous scenes. We evaluate QuadricsReg on 5 public datasets, where its exceptional registration performance with low overhead demonstrates strong scalability for large-scale scenarios. With a compact representation of $\sim 29.5\mathrm{KB/scan}$ on KITTI, nearly 100% registration success rate is achieved for point cloud pairs within $10\mathrm{m}$. Real-world testing on the self-collected dataset further validates its robustness and generalization ability across different LiDAR sensors and robot platforms.**

***Index Terms*—Geometric primitives, localization and mapping, point cloud registration, quadric representation.**

Received 24 January 2026; accepted 8 April 2026. Date of publication 21 April 2026; date of current version 11 May 2026. This work was supported in part by the National Natural Science Foundation of China under Grant 62301370, Grant 62325111, and Grant U22B2011, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2025A1515010440, in part by the Shenzhen Science and Technology Program under Grant JCYJ20240813111202004, and in part by the Wuhan University-Huawei Geoinformatics Innovation Laboratory. This article was recommended for publication by Associate Editor T. A. Vidal-Calleja and Editor M. Walter upon evaluation of the reviewers' comments. *(Corresponding authors: Huai Yu; Gui-Song Xia.)*

Ji Wu, Shu Han, Ximeng Cai, and Mingfeng Wang are with the School of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: ji.wu@whu.edu.cn; hanshu@whu.edu.cn; ximengcai@whu.edu.cn; evanswang@whu.edu.cn).

Huai Yu is with the School of Electronic Information, Wuhan University, Wuhan 430072, China, and also with the Wuhan University Shenzhen Research Institute, Shenzhen 518057, China (e-mail: yuhuai@whu.edu.cn).

Wen Yang is with the School of Electronic Information, Wuhan University, Wuhan 430072, China (e-mail: yangwen@whu.edu.cn).

Gui-Song Xia is with the School of Artificial Intelligence, Wuhan University, Wuhan 430072, China (e-mail: guisong.xia@whu.edu.cn).

Data is available on-line at https://levenberg.github.io/QuadricsReg.

This article has supplementary downloadable material available at https://doi.org/10.1109/TRO.2026.3686263, provided by the authors.

Digital Object Identifier 10.1109/TRO.2026.3686263

I. Introduction

POINT cloud registration is a fundamental task in robotics and serves as the cornerstone of a wide range of applications, such as SLAM [1], large-scale mapping [2], and autonomous navigation [3]. By matching primitive representations that encode the structural information, global registration estimates the 6-DoF transformation between point clouds without prior information [4], [5], facilitating precise ego-pose estimation and coherent map construction in large-scale environments. The scene primitive representation plays a pivotal role in the performance of registration, as it directly influences the quality of subsequent feature matching and transformation estimation. However, as illustrated at the top of Fig. 1, the characteristics inherent in point clouds, such as massive amounts of data, structural diversity, and wide viewpoint variations, pose significant challenges to effective and scalable scene representations during long-term on-board resilient operation in large-scale scenarios.

Raw point clouds directly generated from LiDAR preserve fine-grained scene details but incur considerable storage and computational overhead in large-scale scenarios. Urban environments are generally characterized by structural regularity, which has motivated existing methods to abstract scenes using geometric primitives such as keypoints [6], lines [7], planes [8], [9], cylinders [10], and ellipsoids [5]. While reducing redundancy, these individual primitives lack the expressiveness to uniformly represent diverse structures, such as linear edges, planar buildings, and volumetric vehicles. This obscures the structural saliency of scenes, which is critical for reliable matching, and narrows the applicability of the residual metrics relied on by transformation solvers such as RANSAC [11], [12] and GNC [1], [4]. Moreover, simple geometric models exhibit limited discriminability in scenes with significant appearance variations, making it difficult to construct consistent matching features from partial observations under wide viewpoint changes. Therefore, although existing methods are effective in constrained scenarios, they struggle to handle structural diversity uniformly and are sensitive to viewpoint changes. These limitations highlight the necessity of designing a *compact*, *expressive*, and *discriminative* scene representation for efficient, accurate, and robust global registration, which can be scalable to large-scale complex environments.

Motivated by these, we find that quadrics provide a unified representation for heterogeneous structures (*e.g.*, ellipsoids for vehicles, cylinders for poles, and planes for walls). As a family of parameterized surfaces, quadrics are both compact and

1941-0468 © 2026 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

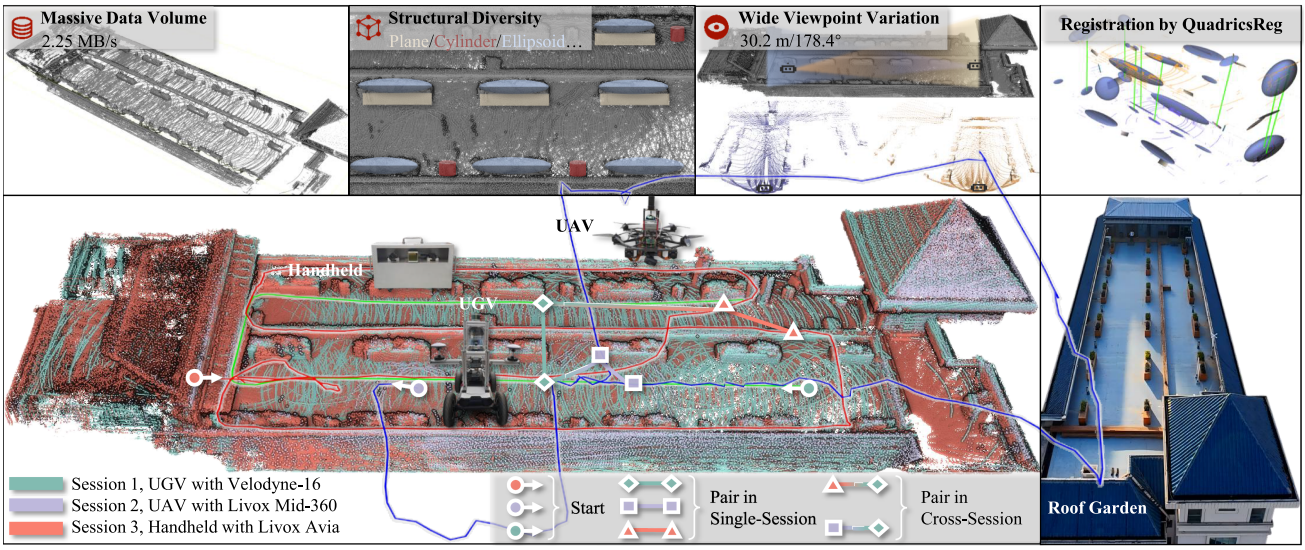


Fig. 1. Global point cloud registration using QuadricsReg. The raw point clouds are collected by an UGV, a UAV, and a handheld platform equipped with different LiDAR sensors in a roof garden. In overlapping regions, QuadricsReg effectively aligns point clouds and enables cross-platform mapping with different LiDAR sensors.

expressive compared to other geometric primitives, capable of modeling 17 commonly occurring shape types using only ten parameters [13], [14]. This representation enables efficient encoding of structural saliency and facilitates both the reconstruction of discriminative geometry and the accurate measurement of effective residuals, even from partial point clouds. Furthermore, their mathematical formulation, being tightly coupled with geometric attributes (type, pose, scale, and degeneracy), streamlines both efficient modeling of quadrics from the scene and intuitive geometric analysis. The semantic information, as an object-level feature, enhances the extraction of scene structures and strengthens data association by its inherent discriminability. The effective integration of quadrics with semantic cues is expected to improve the compactness, expressiveness, and discriminability of scene representation, thereby improving feature matching and transformation estimation for large-scale point cloud registration.

In this article, we propose a semantic quadrics-based global registration framework, namely *QuadricsReg*. The framework achieves a compact, expressive, and discriminative scene representation using semantic quadrics, upon which we develop a robust feature matching and a degeneracy-aware transformation estimation, significantly improving registration performance. Specifically, given a pair of point clouds, we first infer their semantic labels and cluster within semantic types to extract key elements. For each clustered element, we efficiently model a mathematical quadric by aligning geometric attributes from partial point clouds. On this basis, we efficiently initialize correspondences based on the similarity in type and scale of semantic quadrics. To robustly prune outlier correspondences and tolerate representational bias, a multilevel compatibility graph is constructed based on geometric consistency, with inlier matches identified through maximal clique search. Finally, these matches are used to initialize the 6-DoF transformation between paired point clouds. The transformation is further optimized in a factor-graph with a degeneracy-aware quadric residual, yielding an accurate global registration. Fig. 1 illustrates a global point cloud registration example using QuadricsReg, which enables efficient and robust heterogeneous LiDAR mapping for real-world robots.

The contributions of this work are listed as follows.

1) We propose *QuadricsReg*, a systematic global registration framework designed for efficient and robust operation in large-scale scenes.
2) We introduce a novel semantic quadric model that unifies common geometric primitives into a compact, expressive, and discriminative scene representation.
3) We present a robust quadric matching strategy leveraging intrinsic semantic-geometric consistency to resiliently establish correspondences under viewpoint variations.
4) We design a degeneracy-aware quadric residual within a factor-graph optimization to ensure accurate transformation estimation across diverse structural environments.
5) We demonstrate superior robustness and generalization across five public datasets and heterogeneous field tests using unmanned aerial vehicle (UAV), unmanned ground vehicle (UGV), and handheld platforms.

II. Related Works

This section reviews related works in global point cloud registration, with a focus on 3-D scene representation, 3-D correspondence estimation and outlier pruning, and correspondence-based 6-DoF transformation estimation.

A. 3-D Scene Representation

As a foundational front-end task, 3-D scene representation models environmental structures to underpin subsequent back-end registration. Low-level spatial representations derived via

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

points [15], [16], [17], meshes [18], [19], voxels [20], or surfels [21] offer dense spatial detail. However, in large-scale scenarios, the number of candidate frames for registration increases significantly as the map grows over time. This expansion results in substantial storage, computational, and communication overheads that limit long-term scalability.

To improve efficiency, registration methods often prioritize key elements (landmarks) over dense data. Recent studies [22], [23], [24], [25], [26] achieve scene compression by clustering semantic categories to extract key objects or structures. CAD models [27] demonstrate a versatile capability to abstract small-scale indoor scenes. While high-level primitives like lines [28], [29], planes [30], cuboids [31], and ellipsoids [5], [32] capture basic scene structures, their application in heterogeneous environments faces a critical tradeoff. Individual primitives lack the expressiveness to uniformly represent diverse structures, whereas hybrid models increase complexity due to separate extraction and fitting [33], [34], [35]. Moreover, geometric information alone is often insufficient to reliably extract salient primitives.

Quadrics offer a compact solution by representing various geometric shapes using only ten parameters [13], [14], [36]. By integrating semantic cues, we enhance primitive extraction to ensure a mathematically concise yet geometrically rich description of the environment. Our work extends these capabilities to large-scale scenes by designing a unified representation for multiple primitives, effectively balancing compactness with expressiveness to improve alignment robustness and efficiency.

B. 3-D Correspondence Establishment and Outlier Pruning

Global point cloud registration methods are typically classified into correspondence-based and correspondence-free approaches. This work focuses on the former, which establishes initial matches by comparing descriptors of key scene elements and performs robust pruning to remove outliers.

Traditionally, correspondences are established based on local keypoint descriptors. Handcrafted methods such as USC [37], SHOT [38], PFH [39], and FPFH [40] construct covariance-based or handcrafted histograms. Learning-based methods like FCGF [41] and D3Feat [42] predict keypoint saliency to improve matchability. However, significant viewpoint differences often lead to limited overlap, making point-level features inconsistent and unreliable. This limitation further motivates the transition to high-level representations. SegMatch [22] computes feature histograms at the segment level, while BoxGraph [43] establishes correspondences based on bounding box elongation. G3Reg [5] employs the Wasserstein distance between Gaussian models for similarity measurement. Nevertheless, in large-scale scenes, geometric descriptors or a single primitive type often lack the discriminability required for accurate matching under wide viewpoint variations.

Outlier pruning aims to eliminate incorrect matches while retaining at least three valid correspondences to prevent degeneracy in 6-DoF estimation [1]. RANSAC [11], [12] and its variants identify solutions through iterative sampling, whereas GNC [44] suppresses outliers via robust optimization. Graph-based methods [45], [46], [47], [48], [49] filter correspondences through geometric consistency checks and identify inliers via maximum clique detection. Check criteria within these graphs range from global length invariance [4] to Gaussian-modeled discrepancies [25]. Despite their robustness, these methods remain sensitive to consistency thresholds that are difficult to quantify due to uncertainty boundaries. Multilevel strategies [5], [50] address this by evaluating transformations across pyramid compatibility graphs.

By fusing quadrics with semantic information, our method improves feature consistency and matching reliability through joint geometric and semantic reasoning. Based on initial quadric matching, a multilevel compatibility graph is employed to prune outliers while accounting for matching uncertainty. This architecture ensures robust registration even under significant viewpoint variations.

C. Correspondence-Based 6-DoF Transformation Estimation

Estimating the transformation between point clouds is the ultimate objective of registration. Closed-form 6-DoF solutions exist when correspondences are known and noise is isotropic [51], [52], yet guaranteeing all-inlier correspondences remains difficult. Existing strategies for handling mismatches include RANSAC-based sampling [11], [12] and GNC-based robust optimization [1], [4], [44], [53]. These methods typically evaluate alignment quality using residuals between corresponding points to constrain the transformation in three directions. However, large-scale environments often contain regions lacking distinctive geometric features, such as those dominated by planar or cylindrical surfaces [14], [30], [54]. These structures exhibit geometric degeneracy in specific directions, leading to insufficient constraints and substantial alignment errors when using conventional point-to-point metrics [55]. Prior research has addressed this by proposing point-to-line [56] and point-to-plane [57] residuals, which offer directionally specific constraints for different configurations. Nevertheless, the limited expressiveness of existing scene representations hampers their ability to perceive structural diversity. While techniques like smoothness analysis [15] can infer geometry types, they introduce additional algorithmic complexity.

Based on our expressive quadric representation, we develop a degeneracy-aware metric and optimization strategy. This approach preserves geometric constraints while accommodating structural degeneracy, effectively addressing scene diversity and reducing registration errors in large-scale environments.

III. Problem Formulation

A. Problem Definition

Let $\mathcal{P}_x$ and $\mathcal{P}_y$ represent two partially overlapping 3-D LiDAR point clouds captured from different viewpoints. Global registration aims to estimate the optimal rigid transformation $\mathbf{T}^* \in SE(3)$, comprising a rotation $\mathbf{R}^* \in SO(3)$ and a translation $\mathbf{t}^* \in \mathbb{R}^{3\times 1}$, to align $\mathcal{P}_x$ with $\mathcal{P}_y$ without requiring an initial guess.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

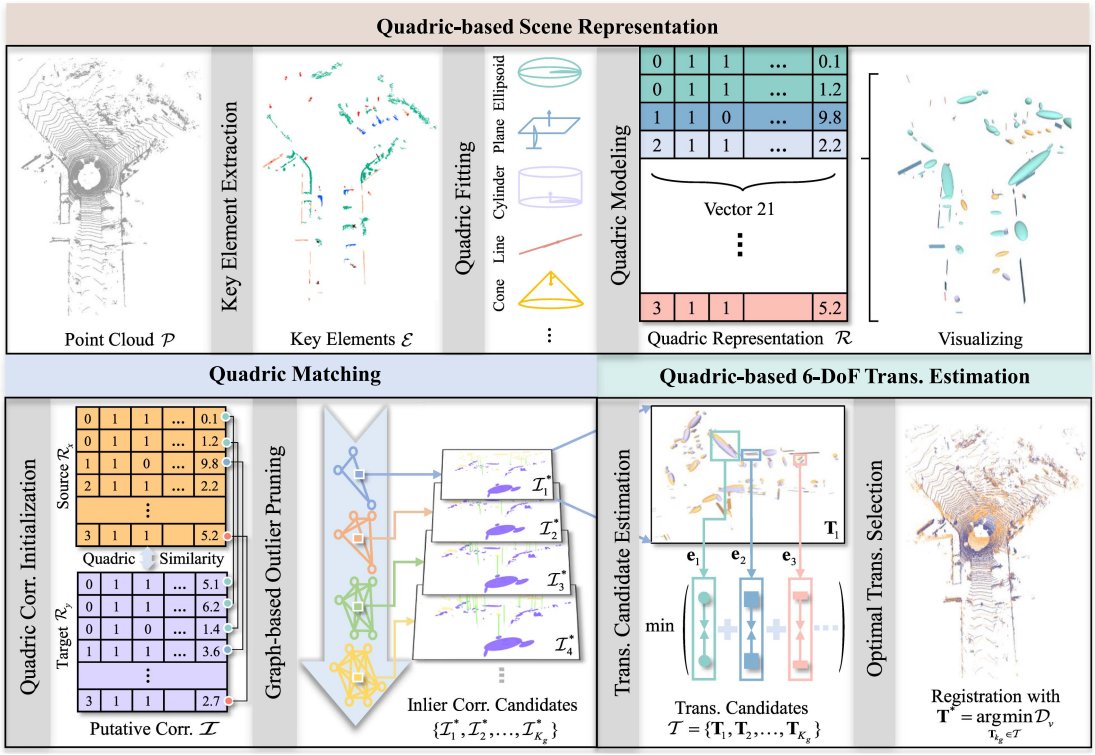


Fig. 2. Pipeline of QuadricsReg, consisting of quadric-based scene representation, matching, and 6-DoF transformation estimation.

The point clouds are first abstracted into sets of scene representations $\mathcal{R}_x = \{\mathbf{r}_x^i\}_{i=1}^{N_x}$ and $\mathcal{R}_y = \{\mathbf{r}_y^j\}_{j=1}^{N_y}$, encoding elements such as lines, planes, or objects. The registration problem is then formulated as finding the optimal $\mathbf{T}^*$ that minimizes the aggregate residual over a set of reliable correspondences $\mathcal{I}^* \subset \mathcal{R}_x \times \mathcal{R}_y$

$$\mathbf{T}^* = \underset{\mathbf{T} \in SE(3)}{\arg\min} \sum_{(\mathbf{r}_x, \mathbf{r}_y) \in \mathcal{I}^*} \rho\left(e\left(f_t\left(\mathbf{r}_x \mid \mathbf{T}\right), \mathbf{r}_y\right)\right) \tag{1}$$

where $\rho(\cdot)$ is the robust kernel function, $e(\cdot)$ is the error function, and $f_t(\cdot)$ denotes the rigid transformation.

B. System Overview

The pipeline of QuadricsReg, illustrated in Fig. 2, consists of three stages: quadric-based scene representation, quadric matching, and quadric-based transformation estimation. Given a pair of point clouds $\mathcal{P}_x$ and $\mathcal{P}_y$, we first extract salient primitives and fit them into quadrics, yielding concise representations $\mathcal{R}_x$ and $\mathcal{R}_y$. Initial correspondences $\mathcal{I}$ are established based on the similarity metric for quadrics. Multithreshold consistency checks are then applied to construct multilevel compatibility graphs for outlier pruning among the correspondences. The maximum clique from each graph level is identified to generate inlier correspondence candidates $\{\mathcal{I}_1^*, \mathcal{I}_2^*, \ldots, \mathcal{I}_{K_g}^*\}$. For each correspondence candidate, an initial transformation is computed using singular value decomposition (SVD). A degeneracy-aware quadric residual is subsequently designed to further refine the transformation, resulting in transformation candidates $\mathcal{T} = \{\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_{K_g}\}$. The one with the minimum residual between $\mathcal{R}_x$ and $\mathcal{R}_y$ after registration is selected as the optimal transformation $\mathbf{T}^*$.

IV. Methodology

A. Quadrics

This section introduces quadrics as a compact parametric surface representation capable of modeling diverse primitive types. We decompose their mathematical form to reveal the geometric characteristics of each component, describe how to infer attributes such as type, scale, and pose, and conclude with a discussion of their inherent degeneracies.

1) Quadrics Basics: Quadrics are a class of surfaces defined implicitly by a second-degree polynomial equation $\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x} = 0$, where $\mathbf{x} = [x, y, z, 1]^{\mathrm{T}}$ represents the homogeneous coordinate. The symmetric matrix $\mathbf{Q}$ is parameterized by the coefficient vector $\mathbf{q} = [A, B, C, D, E, F, G, H, I, J]$ as

$$\mathbf{Q} = \begin{bmatrix} A & D & E & G \\ D & B & F & H \\ E & F & C & I \\ G & H & I & J \end{bmatrix}. \tag{2}$$

Despite being defined by just ten parameters, this representation uniformly models 17 geometric primitives, such as points, lines, planes, spheres, and cylinders, covering the majority of structures in urban environments.

2) Quadric Decomposition: As illustrated in Fig. 3, any general quadric $\mathbf{Q}$ can be decomposed into a canonical quadric $\mathbf{C}$

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

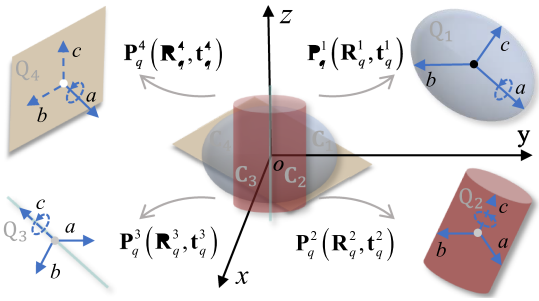


Fig. 3. Illustration of quadric derivation and degeneracy. Quadrics $\mathbf{Q}_1$ to $\mathbf{Q}_4$ (ellipsoid, cylinder, line, and plane) are transformed from canonical forms $\mathbf{C}$ via poses $\mathbf{P}_q$. In each quadric, solid arrows denote local principal axes; dashed lines indicate scale and translation degeneracies, while dashed circular arrows highlight rotational degeneracies. For $\mathbf{Q}_1$, rotation around axis a is degenerate due to symmetry along axes b and c.

TABLE I
CHARACTERISTICS OF TYPICAL QUADRICS; DIAG($\mathbf{C}$) DENOTES THE COEFFICIENTS OF THE CANONICAL QUADRIC; $\mathbf{I_C}$ FLAGS THE RANK AND TYPE OF THE QUADRIC, WHILE INDICATORS $\mathbf{I_{s,R,t}} \in \{0,1\}^3$ EXPLICITLY IDENTIFY DEGENERACY FOR SCALE, ORIENTATION, AND POSITION, RESPECTIVELY

Type	Diag($\mathbf{C}$)	$\mathbf{I_C}$	$\mathbf{I_s}$	$\mathbf{I_R}$	$\mathbf{I_t}$
Point	$[\lambda_a, \lambda_b, \lambda_c, 0]$	$[1,1,1,0]$	$[0,0,0]$	$[0,0,0]$	$[1,1,1]$
Line	$[\lambda_a, \lambda_b, 0, 0]$	$[1,1,0,0]$	$[0,0,0]$	$[0,0,1]$	$[1,1,0]$
Plane	$[\lambda_a, 0, 0, 0]$	$[1,0,0,0]$	$[0,0,0]$	$[1,0,0]$	$[1,0,0]$
Sphere	$[\lambda_a, \lambda_b, \lambda_c, -1]$	$[1,1,1,-1]$	$[1,1,1]$	$[0,0,0]$	$[1,1,1]$
Cylinder	$[\lambda_a, \lambda_b, 0, -1]$	$[1,1,0,-1]$	$[1,1,0]$	$[0,0,1]$	$[1,1,0]$

(aligned with coordinate axes) and a pose $\mathbf{P}_q$

$$\mathbf{Q} = \mathbf{P}_q^{-\mathrm{T}}\mathbf{C}\mathbf{P}_q^{-1} = \begin{bmatrix}\mathbf{R}_q & \mathbf{t}_q \\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix}^{-\mathrm{T}} \begin{bmatrix}\mathbf{\Lambda}_q & \mathbf{0} \\ \mathbf{0}^{\mathrm{T}} & c_{44}\end{bmatrix} \begin{bmatrix}\mathbf{R}_q & \mathbf{t}_q \\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix}^{-1} \tag{3}$$

where the form of Diag($\mathbf{C}$) determines the quadric type (see Table I) and the values of Diag($\mathbf{\Lambda}_q$) $= [\lambda_a, \lambda_b, \lambda_c]$ define the scale. Expanding (3) further reveals the explicit connection between geometric parameters and algebraic coefficients

$$\mathbf{Q} = \begin{bmatrix}\mathbf{R}_q\mathbf{\Lambda}_q\mathbf{R}_q^{\mathrm{T}} & -\mathbf{R}_q\mathbf{\Lambda}_q\mathbf{R}_q^{\mathrm{T}}\mathbf{t}_q \\ -\mathbf{t}_q^{\mathrm{T}}\mathbf{R}_q\mathbf{\Lambda}_q\mathbf{R}_q^{\mathrm{T}} & k\end{bmatrix} \triangleq \begin{bmatrix}\mathbf{Q}_{33} & \mathbf{l} \\ \mathbf{l}^{\mathrm{T}} & k\end{bmatrix}. \tag{4}$$

Based on this, we can infer the geometric attributes of the quadric from its coefficients. First, $\mathbf{Q}$ is normalized to resolve scale ambiguity [13]. The primitive type is identified by the spectral signature of $\mathbf{Q}_{33}$ and the value of c_{44} (via indicator $\mathbf{I_C} \in \{0, 1, -1\}^4$ in Table I). Subsequently, by diagonalizing $\mathbf{Q}_{33} = \hat{\mathbf{R}}_q\hat{\mathbf{\Lambda}}_q\hat{\mathbf{R}}_q^{\mathrm{T}}$, we derive the valid scale $\mathbf{s}_q$, orientation axes $\mathbf{R}_q$, and center $\mathbf{t}_q$

$$\begin{cases}\mathbf{s}_q = & \mathrm{diag}(\mathbf{I_s})\sqrt{|\mathrm{diag}(\hat{\mathbf{\Lambda}}_q)^{-1}|} \\ \mathbf{R}_q = & \pm\hat{\mathbf{R}}_q\,\mathrm{diag}(\mathbf{I_R}) \\ \mathbf{t}_q = & \mathrm{diag}(\mathbf{I_t})\tilde{\mathbf{t}}_q\end{cases} \tag{5}$$

where $\mathbf{I_{s,R,t}} \in \{0,1\}^3$ are binary indicators from Table I that explicitly flag degenerate dimensions (e.g., rotational symmetry or infinite extension). The algebraic center $\tilde{\mathbf{t}}_q$ is determined by solving the linear system derived from (4): $\mathbf{Q}_{33}\tilde{\mathbf{t}}_q + \mathbf{l} = \mathbf{0}$. The rank of $\mathbf{Q}_{33}$ determines the dimensionality of the solution space,

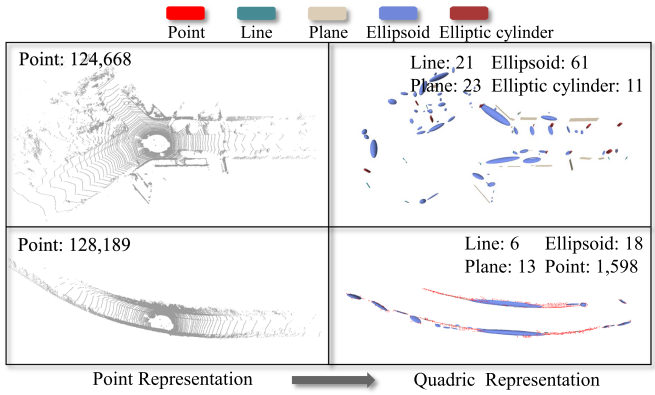


Fig. 4. Quadric-based representation for LiDAR scans from KITTI dataset. Street structures like the buildings and poles are modeled as planes and lines, while objects such as vehicles and trunks are treated as ellipsoids and elliptic cylinders. Vegetation is modeled with ellipsoids to enrich scene representation.

classifying quadrics into central (rank-3, e.g., points, spheres), linear-center (rank-2, e.g., lines, cylinders), and planar-center (rank-1, e.g., planes).

This decomposition establishes a reversible algebraic-geometric mapping, facilitating both the synthesis of $\mathbf{Q}$ for robust quadric modeling and the extraction of explicit attributes for downstream registration.

3) Quadric Degeneracy: Degeneracy refers to the invariance of a quadric shape under specific transformations, a property that can be analyzed from $\mathbf{Q}$. Specifically, a rank-deficient $\mathbf{Q}$ implies scale degeneracy along certain axes. Further analysis of $\mathbf{Q}_{33}$ reveals spatial degeneracies: zero eigenvalues in $\hat{\mathbf{\Lambda}}_q$ indicate that translation along the corresponding axis is unconstrained (e.g., infinite extension), while duplicate eigenvalues imply rotational symmetry, rendering rotation around the symmetry axis degenerate. Fig. 3 offers an intuitive visualization of these geometric behaviors.

A notable advantage of quadric representation is the characterization of degeneracy, which allows the optimization to decouple valid geometric constraints from spurious degrees of freedom. For instance, in planar scene registration, only deviations along the surface normal should be penalized by enforcing point-to-plane residual constraints.

B. Quadric-Based Scene Representation

This section details the abstraction of the dense point cloud $\mathcal{P}$ into a compact representation $\mathcal{R}$. By extracting and fitting key semantic or geometric elements, we model the scene using a 21-parameter quadric representation that explicitly encodes semantics, pose, scale, and degeneracy.

1) Quadric Extraction: Human scene perception relies on identifying geometrically or semantically salient elements. These elements are typically stable, distinctive, and widely present, as illustrated in Fig. 4. We follow this principle and extract a set of key structures and key objects, whose types provide reliable priors for subsequent quadric fitting.

Key structure: These elements represent the primary and stable geometric structures in a scene, mainly composed of

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

planar and linear elements. Following the geometric parsing procedures in TRAVEL [58] and G3Reg [5], we extract three types of structures using default parameter configurations.

1) *Ground:* TRAVEL is applied to estimate the ground surface, producing a dominant planar region denoted as $\mathcal{E}_g$ and its associated label $\mathcal{L}_g$.
2) *Plane:* The remaining cloud $\mathcal{P} \setminus \mathcal{E}_g$ is parsed following the plane-extraction pipeline of G3Reg. $\mathcal{P} \setminus \mathcal{E}_g$ is voxelized, and each voxel is analyzed using local covariance statistics to detect locally planar structures. Voxels exhibiting planar signatures are then merged into coherent planar regions through region growing, using constraints on normal consistency and point-to-plane proximity. The merged planar components constitute $\mathcal{E}_p$ and are assigned label $\mathcal{L}_p$.
3) *Line:* For the remaining points $\mathcal{P} \setminus (\mathcal{E}_g \cup \mathcal{E}_p)$, TRAVEL is applied to cluster them into set $\mathcal{E}_c$. Each cluster in $\mathcal{E}_c$ is subsequently parsed using a RANSAC-based 3-D line detector, and clusters consistent with a dominant linear segment are identified as $\mathcal{E}_l$ with label $\mathcal{L}_l$.

Key object: These elements correspond to discriminative objects that typically exhibit volumetric shapes. Semantic information is valuable for identifying such objects, but its reliability in LiDAR data is limited by the restricted generalization of segmentation models. We therefore perform object extraction in two complementary modes.

1) *Geometry-based:* When semantic predictions are unreliable, the unstructured clusters $\mathcal{E}_c \setminus \mathcal{E}_l$ that remain after structural extraction are treated as key objects $\mathcal{E}_o$ and assigned the corresponding labels $\mathcal{L}_o$.
2) *Semantic-guided:* When the scene permits reliable semantic prediction, we apply PVKD [59] to obtain pointwise labels $\mathcal{L}_s$. From $\mathcal{L}_s$, we select classes with stable and distinctive geometric patterns, including vehicles, trunks, and vegetation. For each selected class s_k, all points with label $\mathcal{L}_{s_k} \subseteq \mathcal{L}_s$ are grouped into clusters using TRAVEL. Each cluster is treated as an object element $\mathbf{e}_{s_k}^i$, forming $\mathcal{E}_{s_k} = \{\mathbf{e}_{s_k}^1, \ldots, \mathbf{e}_{s_k}^{K_o}\}$ with semantic label $\mathcal{L}_{s_k}$. Collecting all such objects yields $\mathcal{E}_o = \{\mathcal{E}_{s_1}, \ldots, \mathcal{E}_{s_{K_s}}\}$ and $\mathcal{L}_o = \{\mathcal{L}_{s_1}, \ldots, \mathcal{L}_{s_{K_s}}\}$.

Through the above process, the raw point cloud $\mathcal{P}$ is reduced to the element point clouds $\mathcal{E} = \{\mathbf{e}_k \mid \mathbf{e}_k \in \mathcal{E}_g, \mathcal{E}_p, \mathcal{E}_l, \mathcal{E}_o\}$ and labels $\mathcal{L}_e = \{l_k \mid l_k \in \mathcal{L}_g, \mathcal{L}_p, \mathcal{L}_l, \mathcal{L}_o\}$ are recorded to facilitate quadric fitting.

2) Quadric Fitting: In this section, each element point cloud $\mathbf{e} \in \mathcal{E}$ is fitted into a parametric quadric $\mathbf{Q}$. We aim to stably obtain a reliable quadric abstraction that remains consistent across viewpoints.

As discussed in Section IV-A, directly regressing the algebraic coefficients of $\mathbf{Q}$ is numerically unstable, where small perturbations can induce drastic geometric distortions. To address this, as illustrated in Fig. 5, we decompose the fitting process into two robust stages: 1) determining the quadric type and 2) estimating the geometric attributes (scale and pose). These components are subsequently composed to form the final quadric $\mathbf{Q}$ via (3).

Guided by this decomposition strategy, we first associate each element type with a specific quadric category using the mapping in Table II. This mapping acts as a midlevel geometric abstraction: rather than reconstructing object shapes in detail, it constrains each element to a plausible quadric type based on its dominant spatial extent. As summarized in Table I, this step directly determines the canonical form $\mathbf{I_C}$, effectively stabilizing the fitting process by eliminating topological ambiguities while preserving the geometric discriminability required for registration.

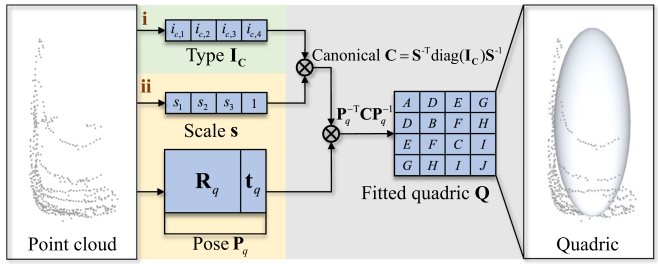


Fig. 5. Decomposition quadric fitting strategy for point clouds. The process is factorized into two phases: 1) determining the quadric type $\mathbf{I_C}$ and 2) estimating the geometric parameters (scale $\mathbf{s}$ and pose $\mathbf{P}_q$). These components are subsequently composed to stably fit the final quadric coefficient $\mathbf{Q}$.

TABLE II
MAPPING FROM ELEMENTS (GEOMETRIC AND SEMANTIC) TO QUADRICS; THIS REFLECTS COMMON ELEMENT–QUADRIC ASSOCIATIONS AND CAN BE ADAPTED OR EXTENDED FOR SPECIFIC APPLICATION SCENARIOS

	Element type	Quadric type	Justification
Geo.	Ground	Plane	Broad smooth surface
	Plane	Plane	–
	Line	Line	–
	Unstructured cluster	Ellipsoid	Generic volumetric shape
Sem.	Vehicle	Ellipsoid	Convex volumetric shape
	Trunk	Elliptic cylinder	Vertical column
	Vegetation	Ellipsoid	Irregular volumetric mass
	Augmented point	Point	–

With the type fixed, the remaining task is to estimate the geometric components. However, point cloud observations are typically imperfect (i.e., noisy and incomplete), which compromises cross-view consistency. To ensure a robust fit, we estimate the geometric components using two complementary methods in parallel, from which the optimal quadric is adaptively selected based on a point-to-surface residual metric:

Learning-based: We utilize QuadricsNet [13] to infer the components $\mathbf{s}_l$ and $\mathbf{P}_l$ from the element $\mathbf{e}$, which are then combined to form the quadric $\mathbf{Q}_l$. During training, fragmented and noisy point clouds are used as inputs, while the quadric coefficients serve as supervision. This enables the network to recover reliable shapes even in the presence of noisy or incomplete observations caused by viewpoint changes.

Statistics-based: To complement the learning model and ensure valid geometry under distribution shifts, we additionally compute a statistics-based estimate.

1) *Centralization:* $\mathbf{e}_c = \mathbf{e} - \mathbf{t}_s$, where $\mathbf{t}_s = \frac{1}{|\mathbf{e}|}\sum_{\mathbf{x}\in\mathbf{e}} \mathbf{x}$ is the center of the quadric.
2) *Eigenvalue decomposition:* We leverage the covariance matrix to capture the spatial distribution of points:

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

Structure 1: Quadric Representation of Element

```
1: l ∈ ℕ_{1×1}; % Semantic label or quadric type
2: q ∈ ℝ_{1×10}; % Quadric parameters
3: s_f ∈ ℝ^+_{1×3}; % Full scale
4: η_f ∈ ℝ_{1×4}; % Full rotation in quaternion form
5: t_f ∈ ℝ_{1×3}; % Full center
```

$\frac{1}{|\mathbf{e}|}\mathbf{e}_c^{\mathrm{T}}\mathbf{e}_c = \mathbf{R}_s\mathbf{\Lambda}_s\mathbf{R}_s^{\mathrm{T}}$. Here, $\mathbf{R}_s$ aligns with the principal geometric axes, and $\boldsymbol{\sigma}_s = \sqrt{\mathrm{diag}(\mathbf{\Lambda}_s)}$ quantifies the spread along these axes.

3) *Quadric composition:* The spatial scale is approximated as $\mathbf{s}_s = k_s\boldsymbol{\sigma}_s$, where $k_s \approx 1.6$ scales the standard deviations to cover the majority of points. Denoting $\mathbf{S}_s = \mathrm{diag}([\mathbf{s}_s, 1])$ as the scale matrix, the canonical matrix $\mathbf{C}_s$ and pose matrix $\mathbf{P}_s$ are computed as

$$\mathbf{C}_s = \mathbf{S}_s^{-\mathrm{T}}\mathrm{diag}(\mathbf{I_C})\mathbf{S}_s^{-1}, \quad \mathbf{P}_s = \begin{bmatrix}\mathbf{R}_s & \mathbf{t}_s \\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix} \tag{6}$$

where $\mathbf{I_C}$ indicates the form of $\mathbf{C}_s$ determined by the quadric type l_q. Then, the quadric parameters can be formed as $\mathbf{Q} = \mathbf{P}_s^{-\mathrm{T}}\mathbf{C}_s\mathbf{P}_s^{-1}$.

We evaluate the agreement between the fitted quadric $\mathbf{Q}$ and the points $\mathbf{e}$ using the point-to-surface residual [60]

$$f_{p2s}(\mathbf{e}, \mathbf{Q}) = \frac{1}{|\mathbf{e}|}\sum_{\mathbf{x}\in\check{\mathbf{e}}}\frac{\|\mathbf{x}^{\top}\check{\mathbf{Q}}\mathbf{x}\|}{\|\nabla_{\mathbf{x}}(\mathbf{x}^{\top}\check{\mathbf{Q}}\mathbf{x})\|} \tag{7}$$

where the superscript $\vee$ indicates normalization to unit space, and $\nabla_{\mathbf{x}}$ denotes the gradient of the quadric implicit function at $\mathbf{x}$. The final quadric is selected as

$$\mathbf{Q} = \underset{\mathbf{Q}'\in\{\mathbf{Q}_l, \mathbf{Q}_s\}}{\arg\min} f_{p2s}(\mathbf{e}, \mathbf{Q}'). \tag{8}$$

This selection mechanism ensures that each element receives a stable and reliable quadric fitting. The two fitting processes serve complementary roles: The learning-based method offers robustness to noisy and partial observations, while the statistics-based estimator guarantees a valid geometric fit under out-of-distribution cases. The latter also provides a nondegenerate scale $\mathbf{s}_s$ and pose $\mathbf{P}_s$, which are further utilized in the quadric modeling stage.

3) Quadric Modeling: In this section, we model the point cloud $\mathcal{P}$ with a concise quadric representation $\mathcal{R} = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}$ based on the extraction and fitting results. We aim to create a representation that efficiently supports global registration while minimally describing the scene.

Each quadric representation $\mathbf{r} \in \mathcal{R}$ is detailed in Struct. 1, comprising only 21 parameters. Within this representation, the semantic label l serves to constrain the search space during registration. The algebraic coefficients $\mathbf{q}$ yield geometric attributes $(\hat{\mathbf{s}}_q, \hat{\mathbf{R}}_q, \hat{\mathbf{t}}_q)$ robust to viewpoint changes and simultaneously infer indicators $\mathbf{I}_{\mathbf{s},\mathbf{R},\mathbf{t}}$ to flag degenerate dimensions. While these degenerate dimensions are decoupled during transformation optimization, they provide valuable cues for correspondence establishment and transformation initialization. Therefore, the statistical scale $\mathbf{s}_s$ and pose $\mathbf{P}_s(\mathbf{R}_s, \mathbf{t}_s)$ are employed as a complement to recover the full attributes

$$\begin{cases}\mathbf{s}_f = & \mathrm{diag}(\mathbf{I_s})\hat{\mathbf{s}}_q + \mathrm{diag}(\mathbf{1} - \mathbf{I_s})\mathbf{s}_s \\ \mathbf{R}_f = & \hat{\mathbf{R}}_q\,\mathrm{diag}(\mathbf{I_R}) + \mathbf{R}_s\,\mathrm{diag}(\mathbf{1} - \mathbf{I_R}) \\ \mathbf{t}_f = & \mathrm{diag}(\mathbf{I}'_{\mathbf{t}})\hat{\mathbf{t}}_q + \mathrm{diag}(\mathbf{1} - \mathbf{I}'_{\mathbf{t}})\mathbf{t}_s\end{cases} \tag{9}$$

where $\mathbf{1} - \mathbf{I}$ denotes the negation of $\mathbf{I}$. $\mathbf{I}'_{\mathbf{t}}$ is set to $\mathbf{0}$ if $\mathbf{I_t}$ indicates degeneracy (i.e., not all-ones). The rotation $\mathbf{R}_f$ is simplified to a quaternion form in Struct. 1. To further ensure a compact representation, we retain only the top-K_e quadrics for each semantic type in $\mathcal{R}$. We prioritize larger quadrics as they are typically more distinctive, measuring size as the product of the first n components of $\mathbf{s}_f$ (where $n \in \{1, 2, 3\}$ corresponds to lines, planes, and volumes, respectively).

Global point cloud registration necessitates sufficient reliable correspondences and geometric constraints for stable $SE(3)$ estimation. Specifically, transformation initialization using position-based solvers (e.g., SVD) mathematically mandates at least three noncollinear matching pairs. However, as illustrated in Fig. 4, extended scenes (e.g., elongated roads or corridors) are often dominated by sparse and degenerate elements. This sparsity limits the number of available matches, while geometric degeneracy fails to provide sufficient constraints for transformation estimation. Furthermore, partial observations of these large elements induce viewpoint-dependent fitting instability, which hinders reliable data association. Consequently, representing each element as a single quadric in such scenarios is inadequate.

To address this, when $|\mathcal{R}| < \delta_a$, we apply voxel downsampling (voxel size v_a) to the K_a largest elements of each semantic class, instantiating voxel centers as augmented points. Following Table II, these are treated as pointtype quadrics that retain only their centers, with all other geometric attributes marked as degenerate. This allows them to be seamlessly appended to $\mathcal{R}$ as $\mathbf{r}_a$, providing auxiliary geometric constraints during quadric matching in sparse scenes.

The final scene is modeled by a compact representation $\mathcal{R} \in \mathbb{R}^{N\times 21}$. The element count is strictly bounded by $N \leq M_l \times K_e + N^a$, covering M_l semantic types and N^a augmented points. This abstraction achieves massive data compression: a typical KITTI scan ($\approx$ 120k points) is reduced to merely $N \approx 100$ quadrics, equivalent to the memory footprint of roughly 700 points. Thus, $\mathcal{R}$ provides a highly concise yet geometrically expressive description, encoding semantics, pose, scale, and degeneracy in a lightweight format.

C. Quadric Matching

Given source and target representations $\mathcal{R}_x$ and $\mathcal{R}_y$, we aim to establish correspondences $\mathcal{I}^*$ between these quadrics. As illustrated in Fig. 6, our strategy proceeds in two phases: we first establish putative correspondences based on geometric similarity within each semantic type, and subsequently extract reliable inliers using multilevel graph-based pruning.

1) Quadric Correspondence Initialization: To obtain an adequate but tractable set of putative correspondences, we match quadrics only within the same semantic class and further filter them using a geometric similarity measure.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

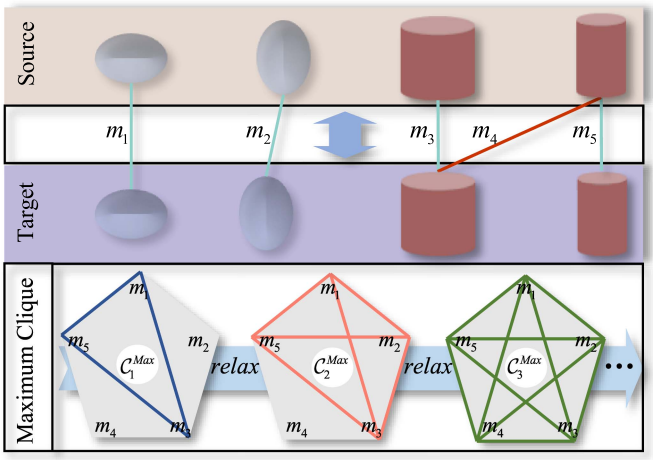


Fig. 6. Quadric matching using multilevel compatibility graphs. Putative correspondences $m_k \in \mathcal{I}$ link source and target quadrics within shared semantics, comprising valid inliers (cyan, $m_{1,2,3,5}$) and an outlier (red, m_4). The grey pentagon arranges these matches as vertices of a compatibility graph, where edges are defined by pairwise consistency checks. Progressively relaxing the consistency threshold δ_m yields varying maximum cliques (colored polygons): The strict threshold-derived clique $\mathcal{C}_1^{Max}$ successfully rejects outlier m_4 resulting in sparse correspondences, whereas the loose $\mathcal{C}_3^{Max}$ retains more matches but includes the outlier.

For each semantic label shared by the source $\mathcal{R}_x$ and target $\mathcal{R}_y$, we compare quadrics based on their full scales $\mathbf{s}_f$, using

$$f_{qsq}(\mathbf{r}_x, \mathbf{r}_y) = -\left\|\mathrm{diag}(\mathbf{I_s})(\mathbf{s}_{f,x} - \mathbf{s}_{f,y})\right\|_2 \tag{10}$$

where the subscripts x/y denote the scene domains, and $\mathbf{I_s}$ masks degenerate dimensions. For each quadric, we retain its top-K_s most similar partners across scenes, producing an initial set of semantic-consistent correspondences. The union of all semantic classes forms the putative correspondence set $\mathcal{I}$. For augmented point quadrics, correspondences are established using FPFH similarity [40] with semantic filtering, and these matches are likewise inserted into $\mathcal{I}$.

The number of correspondences in $\mathcal{I}$ is strictly bounded by $K_e \times K_s \times |\mathcal{L}_\cap| + N_\cap^a$, where $N_\cap^a \leq \min(N_x^a, N_y^a)$ and $|\mathcal{L}_\cap|$ denotes the number of shared semantic classes. In contrast, the upper limit for an all-to-all strategy is $|\mathcal{R}_x| \times |\mathcal{R}_y| = (K_e \times |\mathcal{L}_x| + N_x^a) \times (K_e \times |\mathcal{L}_y| + N_y^a)$. This explicit comparison highlights that matching within semantic types via geometric similarity significantly reduces the search space while ensuring the inclusion of high-likelihood pairs.

2) Graph-Based Outlier Pruning: The putative correspondence set $\mathcal{I}$ inevitably contains outliers. We construct a compatibility graph over these correspondences and robustly identify inliers via maximum-clique reasoning.

To construct the graph, we evaluate pairwise geometric consistency based on translation and rotation invariant measurements [4]. Consider correspondence pairs $(\mathbf{r}_x^i, \mathbf{r}_y^i)$ and $(\mathbf{r}_x^j, \mathbf{r}_y^j)$, where subscripts indicate the domains, and superscripts (i, j) denote the pair indices. Using the full quadric center $\mathbf{t}_f$ as the position proxy, the pairs are deemed consistent if their relative distance variation falls within a tolerance δ_m

$$d_{i,j} = \left|\|\mathbf{t}_{f,x}^i - \mathbf{t}_{f,x}^j\|_2 - \|\mathbf{t}_{f,y}^i - \mathbf{t}_{f,y}^j\|_2\right| \leq \delta_m. \tag{11}$$

This constraint guides the construction of a compatibility graph where the vertices represent correspondences, and the edges connect mutually consistent pairs.

Ideally, rigid transformations preserve relative distances ($d_{i,j} = 0$). In practice, however, a tolerance δ_m is required to accommodate nonzero residuals caused by positional deviations. In keypoint matching, this deviation is dominated by sensor noise and remains approximately uniform across the scene, making a single global threshold feasible. In contrast, for element-level quadric matching, the deviation stems from the instability of center estimation and exhibits heterogeneous uncertainty. Compact primitives (e.g., vehicles modeled as ellipsoids) produce well-constrained centers, whereas extended primitives (e.g., walls modeled as planes) are more susceptible to drift along their unconstrained axes under partial observations. This discrepancy creates conflicts: a small δ_m rejects valid matches of extended elements, whereas a large δ_m fails to filter outliers among compact objects. Therefore, no single global threshold can accommodate all cases.

As illustrated in Fig. 6, we adopt a multilevel thresholding strategy. We construct a sequence of graphs $\{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_{K_g}\}$ using progressively relaxed thresholds $\delta_m^1 < \delta_m^2 < \ldots < \delta_m^{K_g}$, where superscripts denote hierarchy levels. At each level, we extract the maximum clique $\mathcal{C}_{k_g}^{Max}$ applying a graduated strategy to the parallel maximum clique algorithm [61]. Specifically, the clique size determined at level k_g is propagated as a strict lower bound to initialize the search at level $k_g + 1$. This hierarchical scheme reduces the search space by first preserving high-precision constraints captured at strict levels and then gradually recovering valid but noisy matches at relaxed levels. Finally, the union of resulting cliques $\{\mathcal{I}_1^*, \mathcal{I}_2^*, \ldots, \mathcal{I}_{K_g}^*\}$ constitutes a robust inlier set for subsequent transformation estimation.

D. Quadric-Based 6-DoF Transformation Estimation

This section details the estimation of the optimal transformation $\mathbf{T}^* \in SE(3)$ aligning $\mathcal{R}_x$ to $\mathcal{R}_y$. Specifically, we first initialize the pose via SVD on inlier centers and subsequently refine it using a degeneracy-aware factor graph. This formulation unifies diverse geometric constraints to robustly handle structural degeneracies, with the final solution selected by minimizing the global alignment residual.

1) Transformation Candidate Estimation: For each correspondence candidate set $\mathcal{I}_{k_g}^*$ in $\{\mathcal{I}_1^*, \mathcal{I}_2^*, \ldots, \mathcal{I}_{K_g}^*\}$, we aim to estimate the transformation $\hat{\mathbf{T}}_{k_g}$ between the paired point clouds only based on $\mathcal{R}_x$ and $\mathcal{R}_y$. We first estimate an initial transformation directly with SVD based on $\mathcal{I}_{k_g}^*$ and then refine the transformation by optimizing it with a factor graph.

Transformation initialization: According to the correspondence pairs $\{(\mathbf{r}_x, \mathbf{r}_y) \in \mathcal{R}_x \times \mathcal{R}_y \mid \text{associated in } \mathcal{I}_{k_g}^*\}$, we first estimate an initial transformation $\mathbf{T}_{k_g}^0$ by applying SVD to the full centers $\mathbf{t}_f$ of the corresponding quadrics. Starting from $\mathbf{T}_{k_g}^0$, we can construct a nonlinear factor graph to refine the transformation $\mathbf{T}_{k_g}(\mathbf{R}_{k_g}, \mathbf{t}_{k_g})$.

Ground-aligned filtering: Before adding factors, we exclude structural quadrics (planes and lines) $\mathbf{r}$ in $\mathcal{R}_x$ and $\mathcal{R}_y$ whose

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1: Quadric-based Trans. Estimation

Input: Quadric representations of source and target point clouds $\mathcal{R}_x$ and $\mathcal{R}_y$, inlier correspondence candidates set $\{\mathcal{I}_1^*, \mathcal{I}_2^*, \ldots, \mathcal{I}_{K_g}^*\}$
Output: Optimal transformation $\mathbf{T}^*(\mathbf{R}^*, \mathbf{t}^*)$
% Transformation Candidate Estimation
Initialize transformation candidates set: $\mathcal{T} = \emptyset$
for $\mathcal{I}_{k_g}^*$ in $\{\mathcal{I}_1^*, \mathcal{I}_2^*, \ldots, \mathcal{I}_{K_g}^*\}$ **do**
  $\mathbf{T}_{k_g}^0 = \mathtt{SVD}(\mathcal{R}_x, \mathcal{R}_y, \mathcal{I}_{k_g}^*)$
  Initialize factor graph: $\mathcal{F}_{k_g}.\text{initial}(\mathbf{T}_{k_g}^0)$
  for $(\mathbf{r}_x, \mathbf{r}_y)$ in $\mathcal{I}^*$ **do**
    if $f_{pp}(\mathbf{r}_x)$ or $f_{pp}(\mathbf{r}_y)$ **then**
      continue
    `Supplement` $(\mathbf{r}_x, \mathbf{r}_y)$
    Quadric residual: $\mathbf{e} = [\mathbf{e_R}, \mathbf{e_t}]^{\mathrm{T}}$
    Jacobian: $\mathbf{J} = \left[\frac{\partial \mathbf{e_R}}{\partial \mathbf{R}}, \frac{\partial \mathbf{e_R}}{\partial \mathbf{t}}; \frac{\partial \mathbf{e_t}}{\partial \mathbf{R}}, \frac{\partial \mathbf{e_t}}{\partial \mathbf{t}}\right]$
    $\mathcal{F}_{k_g}.\text{addFactor}(\mathbf{e}, \mathbf{J})$
  $\mathbf{T}_{k_g} = \mathtt{LMOptimizer}(\mathcal{F}_{k_g}).\text{pose}$
  $\mathcal{T} = \mathcal{T} \cup \{\mathbf{T}_{k_g}\}$
% Optimal Transformation Selection
Initialize residual set: $\mathcal{D}_v = \emptyset$
for $\mathbf{T}_{k_g}$ in $\mathcal{T}$ **do**
  $\mathcal{I}_{snn} = \mathtt{SNN}(f_t(\mathcal{R}_x|\mathbf{T}_{k_g}), \mathcal{R}_y)$
  $d_v = []$
  for $(\mathbf{r}_x, \mathbf{r}_y)$ in $\mathcal{I}_{snn}$ **do**
    if $f_{pp}(\mathbf{r}_x)$ or $f_{pp}(\mathbf{r}_y)$ **then**
      continue
    `Supplement` $(\mathbf{r}_x, \mathbf{r}_y)$
    $d_v.\text{append}(\rho(\|\mathbf{e_R}\|_2 + \|\mathbf{e_t}\|_2))$
  $\mathcal{D}_v = \mathcal{D}_v \cup \{\sum d_v/|d_v|\}$
Optimal transformation: $\mathbf{T}^* = \underset{\mathbf{T}_{k_g} \in \mathcal{T}}{\arg\min}\, \mathcal{D}_v$

dominant directions deviate from the ground reference

$$f_{pp}(\mathbf{r}) = (\theta > \delta_g) \wedge (|\theta - 90^\circ| > \delta_g) \wedge (|\theta - 180^\circ| > \delta_g) \tag{12}$$

where $\theta = \arccos(\hat{\mathbf{R}}_q[:, i] \cdot \hat{\mathbf{v}}_g)$ is the angle between the ground normal $\hat{\mathbf{v}}_g$ and the ith nondegenerate axis ($\mathbf{I_R}[i] = 1$), and δ_g is the angular tolerance. Here, $\hat{\mathbf{R}}_q$ denotes the quadric rotation and $\mathbf{I_R}$ the rotation-degeneracy indicator, both inferred from the parameters $\mathbf{q}$ in $\mathbf{r}$ according to (5). This constraint restricts the refinement stage to reliable rotational cues provided by ground-aligned structures, effectively pruning nonaligned elements that yield unstable orientation constraints.

Degeneracy-aware quadric residual: We refine the transformation via a factor graph. For a matched pair $(\mathbf{r}_x, \mathbf{r}_y)$ with estimate $\mathbf{T}(\mathbf{R}, \mathbf{t})$, we define the residual $\mathbf{e} = [\mathbf{e_R}, \mathbf{e_t}]^{\mathrm{T}}$ as

$$\begin{bmatrix} \mathbf{e_R} \\ \mathbf{e_t} \end{bmatrix} = \begin{bmatrix} \mathrm{diag}(\mathbf{I}_{\mathbf{R},y})(\mathbf{R}\hat{\mathbf{R}}_{q,x} \otimes \hat{\mathbf{R}}_{q,y}) \\ \mathrm{diag}(\mathbf{I}_{\mathbf{t},y})\hat{\mathbf{R}}_{q,y}^{\mathrm{T}}(\mathbf{R}\mathbf{t}_{f,x} + \mathbf{t} - \mathbf{t}_{f,y}) \end{bmatrix} \tag{13}$$

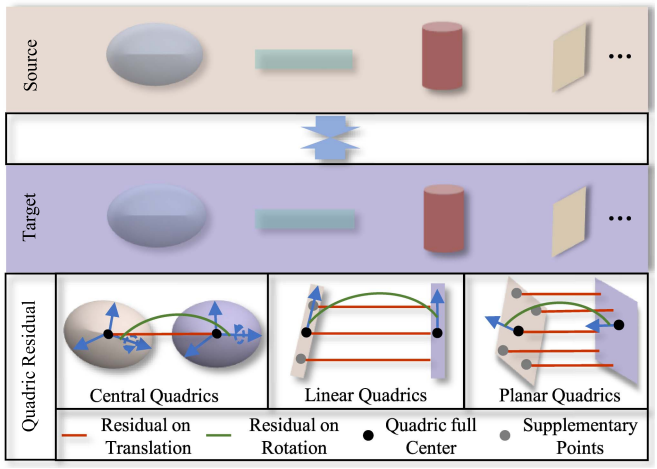


Fig. 7. Illustration of the unified quadric residual metric. The metric comprises translational (red lines) and rotational (green curves) components. Leveraging quadric degeneracy, translational residuals for central, linear, and planar quadrics are measured via point-to-point, point-to-line, and point-to-plane metrics, respectively. Supplementary points (grey dots) provide spatial leverage on noncentral shapes. Rotational residuals are evaluated only along nonsymmetric axes.

where $\otimes$ denotes the columnwise cross product. The diagonal matrices $\mathrm{diag}(\mathbf{I}_{\mathbf{R},y})$ and $\mathrm{diag}(\mathbf{I}_{\mathbf{t},y})$ act as geometric projection operators. Specifically, the rotational term penalizes the misalignment of principal axes solely along nondegenerate directions (e.g., ignoring rotation around a plane's normal), while the translational term projects the displacement error onto valid subspaces. As illustrated in Fig. 7, this yields a unified formulation that enforces principal axis alignment while naturally reducing to point-to-point residual for central quadrics, point-to-line residual for linear quadrics, and point-to-plane residual for planar quadrics.

Constraint supplementary: Equation (13) explicitly suppresses error components along degenerate axes, preventing the optimization from being influenced by geometrically invalid directions. However, for noncentral quadrics (e.g., planes, lines, and cylinders), applying this constraint solely at the centroid leads to conditioning issues: 1) the reduced rank of the translational constraint (e.g., 1-DoF for planes) provides insufficient gradients for efficient alignment; and 2) single-point constraints lack the spatial lever arm, yielding vanishing translational residuals when centers intersect despite significant boundary misalignment.

To address this, as shown in Fig. 7, we sample supplementary points for noncentral quadrics in $\mathbf{r}_x$ using a radius σ defined as the mean of its nondegenerate scales. Linear quadrics (e.g., lines and cylinders) are supplemented with endpoints $\mathbf{t}_f \pm \sigma\, \mathbf{v}_{dir}$, while planar quadrics (e.g., planes) utilize four peripheral vertices $\mathbf{t}_f \pm \sigma\mathbf{u} \pm \sigma\mathbf{v}$, where $\mathbf{v}_{dir}$ is the principal direction and $\mathbf{u}, \mathbf{v}$ form the orthonormal basis. These supplementary points are paired with the target $\mathbf{r}_y$ and evaluated using the same residual formulation (13). This strategy introduces a spatial lever arm that recovers the rank of the geometric constraints, ensuring that misalignment is robustly penalized across the structure's extent without violating the validity of degenerate directions.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

TABLE III
SUMMARY OF EXPERIMENTAL DATASETS

	Dataset	LiDAR-Beam	Points/Frame	Evaluation
Public	KITTI [62]	Velodyne-64	120 K	LC, ODO
	KITTI-360 [63]	Velodyne-64	120 K	LC, ODO
	Apollo [64]	Velodyne-64	120 K	LC, ODO
	Waymo [65]	Spinning-64*	177 K	ODO
	nuScenes [66]	Velodyne-32	34 K	ODO
Self.	Hetero-Reg	Velodyne-16	30 K	SLAM, MM, CS
		Livox *Mid-360*	20 K	
		Livox Avia	24 K	

∗: Spinning-state LiDAR (exact model undisclosed).

Factor graph optimization: For each candidate set $\mathcal{I}_{k_g}^*$, we construct a factor graph aggregating the residuals of all correspondences. The optimal transformation is estimated by minimizing the total squared error via the Levenberg-Marquardt algorithm. Analytical Jacobians are derived using standard Lie algebra perturbation models. Iterating this process for all multilevel cliques yields a set of transformation candidates $\mathcal{T} = \{\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_{K_g}\}$.

2) Optimal Transformation Selection: We seek the optimal $\mathbf{T}^* \in \mathcal{T}$ that aligns $\mathcal{R}_x$ and $\mathcal{R}_y$. Equation (1) is rewritten as

$$\mathbf{T}^* = \arg\min_{\mathbf{T}_{k_g} \in \mathcal{T}} \frac{1}{|\mathcal{I}_{snn}|} \sum_{(\mathbf{r}_x, \mathbf{r}_y) \in \mathcal{I}_{snn}} \rho\left(\|\mathbf{e}_\mathbf{R}\|_2 + \|\mathbf{e}_\mathbf{t}\|_2\right) \quad (14)$$

where $\mathbf{e}_\mathbf{R}$ and $\mathbf{e}_\mathbf{t}$ are the residual components (13), and $\mathcal{I}_{snn}$ denotes the one-to-one correspondence established by performing a nearest neighbor search for each semantic type (SNN) on full centers from $f_t(\mathcal{R}_x|\mathbf{T}_{k_g})$ to $\mathcal{R}_y$. Specifically, $f_t(\mathcal{R}_x|\mathbf{T}_{k_g}) = \{\mathbf{R}_{k_g}\mathbf{t}_{f,x} + \mathbf{t}_{k_g} | \mathbf{t}_{f,x} \in \mathbf{r}_x, \mathbf{r}_x \in \mathcal{R}_x\}$ represents the set of transformed quadric centers of $\mathcal{R}_x$ using $\mathbf{T}_{k_g}$. We also filter out misaligned structures and employ supplementary constraints for noncentral quadrics to enhance robustness.

V. EXPERIMENTS AND ANALYSIS

A. Dataset

To evaluate the performance of global point cloud registration methods, we conduct experiments using partially overlapping point clouds from both loop closure (LC) and odometry (ODO) scenarios. The LC scenario provides a broader distribution of transformations, whereas the ODO scenario serves as a standard benchmark for assessing registration robustness across continuous trajectories [1], [5], [25], [41], [67], [68].

As summarized in Table III, our evaluation utilizes five public benchmarks and one self-collected dataset, spanning six distinct LiDAR sensor types (visualized in Fig. 8). The KITTI [62], KITTI-360 [63], and Apollo [64] datasets serve as large-scale SLAM benchmarks, providing abundant loop closure candidates. While Waymo [65] and nuScenes [66] typically consist of short-duration sequences (∼20 s), their vast volume allows for a comprehensive evaluation across diverse environments. Specifically, the 32-beam configuration of nuScenes is leveraged to assess generalization across varying sensor resolutions. Furthermore, to validate real-world applicability, we established the Hetero-Reg dataset using three heterogeneous robotic platforms equipped with different LiDARs. We apply QuadricsReg to this dataset to evaluate performance in loop closure (SLAM), multisession mapping (MM), and challenging scenarios (CSs). The dataset comprises six sequences, and further details are presented in Section V-H.

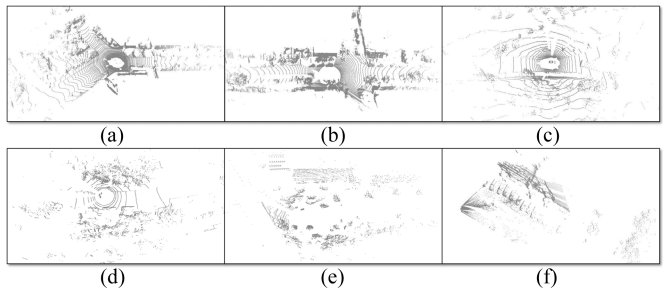


Fig. 8. Point clouds dataset with differences in LiDAR sensors, spatial arrangement, density, and FOV. We select LiDAR sensors of spinning-state, semisolid-state, and solid-state types. (a) Velodyne-64. (b) Spinning-64. (c) Velodyne-32. (d) Velodyne-16. (e) Livox Mid-360. (f) Livox Avia.

1) Dataset From Loop Closure: For the loop closure scenario, the set of loop pairs is defined as follows:

$$\left\{(i,j) \mid d_{lc}^{\min} \le \|\bar{\mathbf{t}}_i - \bar{\mathbf{t}}_j\|_2 \le d_{lc}^{\max}, |i-j| \ge t_{lc}, \forall i,j \in \mathbb{N}_p\right\} \quad (15)$$

where $\mathbb{N}_p$ represents the set of frame indices for point clouds within the same sequence, and $\bar{\mathbf{t}}$ denotes the ground truth (GT) position of the point cloud. $d_{lc}^{\min}$ and $d_{lc}^{\max}$ denote the minimum and maximum allowable distances between two frames, respectively, while t_{lc} is a threshold ensuring sufficient temporal separation between selected frames to avoid consecutive selections. Since registration difficulty increases with translation distance (due to reduced overlap), we stratify the dataset into three difficulty levels: Easy (0–10 m), Medium (10–20 m), and Hard (20–30 m).

We construct KITTI-LC using sequences 00, 02, 05, 06, and 08, adopting refined ground truth poses from SemanticKITTI [69]. Similarly, we select the five most loop-rich sequences from KITTI-360 (00, 02, 04, 06, 09) and Apollo (21, 22, 23, 25, 26) to generate KITTI-360-LC and Apollo-LC, respectively. To ensure balanced evaluation, we apply uniform spatial sampling to retain approximately 1000 pairs per difficulty level across all datasets.

2) Dataset From Odometry: For the odometry scenario, the set of registration pairs is defined as follows:

$$\left\{(i,j) \mid j = \arg\min_k \left(\left|\|\bar{\mathbf{t}}_i - \bar{\mathbf{t}}_k\|_2 - d_{odo}\right|\right), \forall i,j \in \mathbb{N}_p\right\} \quad (16)$$

where d_{odo} represents the distance between two point clouds. We vary the distance from 1 to 10 m, generating 10 difficulty levels of registration datasets to analyze the effect of viewpoint shift on the performance of registration methods.

For odometry testing, we spatially sample approximately 500 pairs per level across diverse scenes to validate generalization. KITTI-ODO utilizes the standard test sequences 08-10 [41]. Apollo-ODO is constructed from sequences 20–23 and 25–26. For Waymo-ODO and nuScenes-ODO, we sample from the

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
SUMMARY OF EVALUATION PROTOCOL

	Experiment		Dataset	Sequence	Semantics
Public	Corr. Estab.		KITTI-LC	00, 02, 05, 06, 08	✓
	Global Reg.	Loop	KITTI-LC	00, 02, 05, 06, 08	✓
			KITTI-360-LC	00, 02, 04, 06, 09	✓
			Apollo-LC	21, 22, 23, 25, 26	✗
		Odometry	KITTI-ODO	08, 09, 10	✓
			Apollo-ODO	20–23, 25–26	✗
			Waymo-ODO	00-99	✓
			nuScenes-ODO	00-99	✓
	Trans. Est.		KITTI-LC	00, 02, 05, 06, 08	✓
	Scene Rep.		KITTI-LC	08	✓
Self.	SLAM		Hetero-Reg	1, 2, 3, 4	✗
	Multi. Mapping		Hetero-Reg	1, 2, 3, 4	✗
	Chall. Scen.		Hetero-Reg	5, 6	✗

first 100 sequences to obtain environmental diversity. Note that KITTI-360 is excluded from odometry benchmarks due to its similarity with KITTI for the LiDAR sensors and capturing scenarios. The resulting collection comprises KITTI-ODO, Apollo-ODO, Waymo-ODO, and nuScenes-ODO.

B. Experimental Settings and Metrics

1) Implementation Details: Unless otherwise stated, we adopt a unified set of default parameters across all experiments. Specifically, the ground extraction and clustering for quadric modeling are conducted using the default parameters provided by TRAVEL [58] and G3Reg [5]. For datasets where semantics can be reliably predicted, PVKD [59] is employed to predict semantic labels. The threshold δ_p for adopting the fitting quadrics of QuadricsNet is set to 0.5, following the network configuration in [13]. In the final quadric representation $\mathcal{R}$, $K_e = 50$ quadrics are retained for each semantic type. If the total number of quadrics in $\mathcal{R}$ is less than $\delta_a = 60$, the top $K_a = 10$ largest quadrics from each semantic type are downsampled using a voxel size of $v_a = 0.5$ to serve as augmented points. In the correspondence establishment, up to $K_s = 20$ putative correspondences are established for each quadric, and a four-level compatibility graph is constructed using thresholds $[0.2, 0.4, 0.6, 0.8]$. During transformation estimation, irregular structures are removed based on a threshold of $\delta_g = 5^\circ$. We adopt the dynamic covariance scaling [70] as the robust kernel function ρ in optimal transformation selection.

All experiments are conducted on a workstation with an Intel Xeon Platinum 8336 C CPU running at 2.30 GHz, 128 GB RAM, and an Nvidia RTX4090 GPU.

2) Evaluation Protocols: Table IV provides a summary of the datasets, sequences, and semantic strategies allocated to each evaluation task. Overall, our experiments encompass two primary dimensions: standardized evaluation on public datasets and practical validation on self-collected heterogeneous datasets. On public benchmarks, we evaluate fundamental performance across correspondence establishment, global registration (covering both loop closure and odometry), transformation estimation, and scene representation. For the self-collected Hetero-Reg dataset, evaluations extend to loop closure within a SLAM system, multisession mapping, and registration in challenging scenarios with geometric sparsity and degeneracy. Regarding semantic utilization, we adopt an adaptive strategy: semantic-guided quadric extraction is enabled for datasets supporting robust predictions (KITTI, KITTI-360, Waymo, nuScenes), while the system reverts to purely geometric clustering for Apollo and Hetero-Reg to mitigate instabilities arising from missing annotations and significant domain discrepancies.

3) Evaluation Metrics: For evaluation of correspondence establishment, the number of correspondences (Num.), inlier ratio (IR), and correspondence recall (CR) are utilized. Given a pair of point clouds with an established correspondence set $\mathcal{I}$, Num. denotes the size of this set. IR is defined as

$$\frac{1}{\text{Num.}} \sum_{(i,j)\in\mathcal{I}} 1\left(\left\|\bar{\mathbf{R}}\mathbf{t}_x + \bar{\mathbf{t}} - \mathbf{t}_y\right\|_2 \leq 0.5\,\text{m}\right) \tag{17}$$

where $\mathbb{1}(\cdot)$ is the indicator function, and $(\bar{\mathbf{R}}, \bar{\mathbf{t}})$ represents the GT transformation. The GT transformations of all datasets are refined using the iterative closest point (ICP) algorithm. The threshold of 0.5 m defines the residual tolerance within which correspondence is considered an inlier. If the total number of inliers reaches or exceeds 3, the point cloud pair is considered successfully matched. Based on this criterion, CR can be computed for the dataset.

For transformation estimation, we use the relative translation error (RTE) and relative rotation error (RRE) as metrics:

$$\text{RTE} = \|\bar{\mathbf{t}} - \mathbf{t}\|_2, \text{RRE} = \arccos\left[\frac{\text{tr}\left(\mathbf{R}^{\mathrm{T}}\bar{\mathbf{R}}\right) - 1}{2}\right]. \tag{18}$$

Furthermore, we adopt the registration success rate (SR) to evaluate global registration. Since global registration typically serves as initialization for fine registration (e.g., ICP), the estimate must fall within a convergence basin. We consider a registration successful if $RTE \leq 2\,\text{m}$ and $RRE \leq 5^\circ$.

Finally, efficiency is assessed via storage overhead (memory usage per frame for scene representation) and computational runtime (average registration time per point cloud pair).

C. Evaluation of Correspondence Establishment

In this section, we evaluate the capability of establishing putative correspondences based on various representations. We compare QuadricsReg against representative baselines from two categories. 1) Low-level: Handcrafted FPFH [40] and learning-based FCGF [41]. 2) High-level: Semantic-based Segregator [25] and geometry-based G3Reg [5]. For QuadricsReg, we utilize a fixed default configuration ($K_s = 20$) for the primary comparison. Additionally, since a larger upper limit K_s yields a more conservative approximation of all-to-all matching, we also report results for $K_s = 15$ and 25 to evaluate the effectiveness of our matching strategy based on quadric similarity. Table V lists the results on the KITTI-LC dataset.

In general, methods based on low-level representations achieve relatively higher IR due to their dense features and the use of a one-to-one matching strategy, resulting in fewer mismatches. In contrast, high-level representations tend to yield lower IR because of feature sparsity, which necessitates a one-to-many matching strategy to maintain a sufficient number of inliers. Nevertheless, QuadricsReg achieves higher CR and SR

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

TABLE V
EVALUATION OF CORRESPONDENCE ESTABLISHMENT ON KITTI-LC DATASET

		Level	Easy				Medium				Hard			
		Metrics	Num.	IR ↑	CR ↑	SR ↑	Num.	IR ↑	CR ↑	SR ↑	Num.	IR ↑	CR ↑	SR ↑
KITTI-LC	Low.	FPFH [40]	707.20	9.15	**100.00**	**99.38**	633.12	2.55	95.60	82.37	595.63	0.89	66.20	45.17
		FCGF [41]	2483.55	**17.70**	80.89	73.74	1824.31	**6.74**	60.06	54.12	1568.15	2.30	43.55	39.09
	High.	Segregator [25]	351.98	14.52	98.64	91.61	235.56	6.18	86.13	80.71	187.19	**2.92**	55.84	54.11
		G3Reg [5]	1353.92	1.52	99.20	97.53	1326.64	0.76	94.80	90.60	1313.10	0.40	76.40	69.20
		QuadricsReg	1456.42	2.05	**100.00**	98.77	1427.26	1.05	98.00	93.22	1395.62	0.60	85.60	80.00
		QuadricsReg-15*	1107.18	2.50	**100.00**	98.77	1082.75	1.26	97.40	92.91	1053.49	0.71	83.80	78.20
		QuadricsReg-25*	1787.60	1.75	99.88	98.64	1769.24	0.87	**98.33**	**93.64**	1741.44	0.50	**88.43**	**81.42**

(Num.: Number of correspondences; IR: Inlier ratio (%); CR: Correspondence recall (%); SR: Registration success rate, (%)). Bold indicates the best, and underlined denotes the second-best.
∗: QuadricsReg-K_s, where K_s denotes the upper limit for the initial correspondences of each quadric.

across all levels despite a lower IR, demonstrating the effectiveness of its correspondence outlier pruning based on the multilevel compatibility graph. Additionally, QuadricsReg and Segregator yield higher IR than G3Reg, highlighting the benefit of semantic cues. As the viewpoint distance increases from easy to hard levels, changes in the surrounding points make it challenging for low-level features to establish reliable matches, resulting in reduced CR and lower SR. High-level representations, leveraging element-level information, are robust to viewpoint changes, maintaining higher CR and SR at medium and hard levels. Notably, QuadricsReg performs best on the hard-level due to the expressive power of quadrics and the effectiveness of their similarity measurement.

As shown in the lower section of Table V, increasing K_s leads to a notable increase in Num. but results in a decrease in IR. Meanwhile, CR and SR exhibit minimal fluctuations. This indicates that our quadric-based matching effectively establishes reliable correspondences based on similarity scores, without requiring a conservative strategy (i.e., a large K_s) to ensure high performance.

D. Evaluation of Global Registration

1) Loop Closure Test: In this section, we assess QuadricsReg in the loop closure task against three categories of global registration methods. 1) Learning methods based on low-level representations: Descriptor-based FCGF [41] and end-to-end pipelines including DGR [68], PointDSC [67], and LCDNet [71]. DGR and PointDSC primarily focus on outlier pruning after matching, while LCDNet specializes in joint loop detection and registration. 2) Handcrafted methods based on low-level representations: Robust estimators including RANSAC, TEASER++ [4], Quatro [72], and 3-DMAC [46]. These solvers operate as back-ends to filter outliers and estimate poses based on initial correspondences generated by FPFH. 3) Methods based on high-level representations: Object-based Segregator [25] and ellipsoid-based G3Reg [5]. Quantitative results are presented in Table VI.

Learning-based methods generally exhibit inferior performance compared to handcrafted and high-level representation-based approaches. This is primarily because the learning-based methods are trained on registration datasets involving short distances or continuous viewpoint variations. In contrast, loop closure scenarios feature greater diversity in translational and rotational changes.

At the easy-level, point-based handcrafted methods achieve near-perfect success rates around 100%. As viewpoint distance increases, although the registration success rate decreases, those based on high-level representations generally exhibit a smaller decline in performance compared to point-based learning and handcrafted methods. This is because the integration of element-level semantic and geometric information by high-level representations mitigates the challenges of matching under significant viewpoint differences. Consequently, these high-level representation-based methods establish more stable correspondences under challenging conditions, thereby ensuring robust registration. Among high-level methods, both G3Reg and QuadricsReg outperform Segregator, demonstrating the benefits of element-level geometric matching and multilevel compatibility filtering. QuadricsReg further exploits semantic and geometric information to segment and represent the scene, leveraging this representation during matching and transformation estimation to achieve superior overall performance. We observe that it consistently achieves a near-100% registration success rate for point cloud pairs 10 m apart. It also maintains strong performance under large viewpoint differences in hard-level scenarios. As shown in Fig. 9(a)–(d), it models different semantic elements using distinct geometric primitives, which not only reduces the matching search space but also provides reliable geometric similarity measures for later matching. In object-sparse environments, such as the KITTI-LC sequence 02 [see Fig. 9(b)], QuadricsReg leverages the augmented points [red points in Fig. 9(b)] to effectively compensate for the reduced number of matches caused by the sparsity of key elements. In the KITTI-LC sequence 08 and KITTI-360-LC, most loops are reversed, where the opposite direction poses challenges to estimating the geometric properties of objects. QuadricsReg nonetheless maintains its robustness. Regarding the Apollo-LC dataset, where reliable semantic predictions are unavailable, our framework adaptively utilizes geometric clustering categories as surrogate labels. By effectively leveraging these geometric features, QuadricsReg yields commendable performance, thereby demonstrating the flexibility of our quadric representation.

2) Odometry Test With Random Rotations: In this section, we evaluate the point cloud registration performance in the odometry test. Raw odometry scenarios typically exhibit a narrow pose distribution dominated by forward translation and minimal rotation. As visualized in Fig. 10, the ground truth rotations in the raw dataset are heavily concentrated near zero ($< 5^\circ$). Consequently, this setting may primarily evaluate local tracking capabilities

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

TABLE VI
GLOBAL REGISTRATION TEST FOR LOOP CLOSURE ON KITTI-LC; KITTI-360-LC AND APOLLO-LC DATASETS

		Sequence	00			02			05			06			08		
KITTI-LC		Level	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard
	Learning	FCGF [41]	89.50	67.50	48.00	62.50	49.00	32.50	82.95	69.85	51.00	93.50	49.50	37.00	2.24	0.61	0.00
		DGR [68]	83.00	61.50	40.50	62.50	50.50	33.00	78.75	66.83	47.00	96.90	49.50	25.50	2.99	1.23	0.00
		PointDSC [67]	79.50	61.00	45.00	53.48	46.46	37.50	77.27	62.81	49.00	89.92	49.00	36.14	2.24	0.00	0.00
		LCDNet [71]	95.00	-	-	68.60	-	-	95.45	-	-	**100.00**	-	-	88.81	-	-
	Handcraft	RANSAC	99.50	74.50	27.50	98.83	70.20	32.00	98.86	78.89	35.50	**100.00**	96.50	80.50	**100.00**	93.82	50.81
		TEASER++ [4]	99.50	79.50	32.00	97.67	71.21	27.00	99.43	86.93	42.00	**100.00**	98.50	87.50	**100.00**	96.29	59.45
		Quatro [72]	98.50	91.00	51.00	**100.00**	**74.74**	39.00	**100.00**	92.96	50.00	**100.00**	98.50	85.00	99.25	95.06	63.78
		3DMAC [46]	99.50	66.00	15.50	95.34	51.51	15.50	98.86	64.32	20.00	**100.00**	89.00	58.00	**100.00**	79.01	32.43
	High.	Segregator [25]	99.00	91.00	48.00	62.21	48.48	26.00	99.43	74.37	44.00	**100.00**	99.50	87.50	**100.00**	93.80	69.73
		G3Reg [5]	**100.00**	95.00	56.00	88.37	67.17	50.00	**100.00**	96.48	73.50	**100.00**	**100.00**	**97.50**	**100.00**	98.14	78.91
		QuadricsReg	99.50	**99.50**	**80.00**	96.51	71.21	**53.50**	**100.00**	**98.49**	**87.50**	**100.00**	**100.00**	96.00	**100.00**	**98.77**	**84.32**
		Sequence	00			02			04			06			09		
KITTI-360-LC		Level	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard
	Learning	FCGF [41]	38.50	29.00	25.50	21.00	18.00	16.50	14.00	8.00	10.00	41.00	35.00	24.50	56.00	47.50	31.00
		DGR [68]	39.00	26.50	22.00	20.50	16.00	14.00	16.00	9.00	6.50	40.00	34.00	20.50	56.00	42.00	23.00
		PointDSC [67]	37.20	26.50	21.00	18.50	16.50	16.00	14.50	9.50	8.00	38.50	35.00	27.00	54.00	42.50	28.50
		LCDNet [71]	92.00	-	-	89.00	-	-	94.50	-	-	96.00	-	-	94.00	-	-
	Handcraft	RANSAC	99.50	89.50	49.00	**99.00**	**87.00**	55.00	97.50	77.00	32.50	97.00	71.00	34.50	98.00	78.00	35.50
		TEASER++ [4]	**100.00**	91.50	56.00	98.50	83.50	54.50	97.00	75.00	36.50	96.50	70.00	35.00	99.50	83.00	34.00
		Quatro [72]	**100.00**	93.50	66.00	97.50	81.00	59.50	99.00	86.50	49.00	97.00	76.00	41.00	99.50	89.50	50.00
		3DMAC [46]	99.50	72.00	33.50	96.00	72.50	33.00	98.00	61.50	21.00	93.00	51.50	19.50	98.50	58.00	15.50
	High.	Segregator [25]	97.00	86.50	62.00	94.00	67.50	41.50	91.00	64.00	34.00	87.00	52.00	20.00	97.50	74.50	30.50
		G3Reg [5]	**100.00**	94.00	73.50	98.50	84.00	64.00	**100.00**	88.00	49.50	**99.50**	79.50	43.00	**100.00**	91.50	59.50
		QuadricsReg	**100.00**	**97.50**	**82.00**	97.00	85.50	**68.50**	**100.00**	**91.00**	**68.00**	**99.50**	**84.50**	**51.00**	**100.00**	**96.50**	**75.00**
		Sequence	21			22			23			25			26		
Apollo-LC		Level	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard
	Learning	FCGF [41]	85.50	74.50	61.50	**100.00**	4.50	4.00	92.50	90.00	77.50	93.75	4.00	5.50	66.50	65.00	50.50
		DGR [68]	82.50	71.50	41.00	93.10	7.50	4.00	89.00	84.00	45.00	93.75	5.50	4.00	65.00	63.50	44.00
		PointDSC [67]	78.50	63.50	36.50	**100.00**	4.00	4.00	89.00	77.50	49.50	75.00	4.00	5.00	67.00	65.00	45.00
		LCDNet [71]	79.00	-	-	89.66	-	-	77.00	-	-	93.75	-	-	92.00	-	-
	Handcraft	RANSAC	99.50	94.50	60.50	**100.00**	96.50	73.00	99.50	92.00	60.00	93.75	88.50	75.50	**100.00**	99.00	76.50
		TEASER++ [4]	**100.00**	93.00	70.50	**100.00**	96.00	81.50	**100.00**	97.00	72.00	**100.00**	**92.00**	85.00	**100.00**	98.50	80.50
		Quatro [72]	**100.00**	92.50	70.00	**100.00**	92.50	80.00	99.00	96.50	76.00	**100.00**	89.50	87.00	99.50	**100.00**	84.00
		3DMAC [46]	98.50	82.50	32.50	96.55	83.00	50.50	99.50	81.50	38.50	93.75	76.50	60.50	**100.00**	93.50	55.00
	High.	G3Reg [5]	**100.00**	**100.00**	97.50	**100.00**	**99.50**	96.50	**100.00**	**100.00**	96.50	**100.00**	90.50	86.50	**100.00**	**100.00**	**97.00**
		QuadricsReg	**100.00**	**100.00**	**98.00**	**100.00**	99.00	**98.00**	**100.00**	99.00	**98.50**	**100.00**	88.00	**87.50**	**100.00**	**100.00**	**97.00**

(Registration success rate (%)); Approximately 1000 loop pairs of point clouds are evaluated in total per difficulty level; Bold indicates the best, and underlined denotes the second-best; The "-" indicates that the method's designed scope is not applicable to the corresponding difficulty level.

rather than global registration performance. To address this bias, we perturb the point clouds by introducing random yaw rotations in the range of $[-45°, 45°]$. This perturbation creates a balanced rotation distribution while maintaining the original translation diversity (1–10 m). We compare QuadricsReg against the same methods detailed in Section V-D1, which are categorized into learning-based, handcrafted, and high-level methods.

As shown in Fig. 11, learning-based methods (e.g., DGR, PointDSC, LCDNet) suffer performance degradation as viewpoint distance increases, except for FCGF, which retains robustness via its RANSAC backend. Handcrafted methods (e.g., RANSAC, TEASER++, Quatro, 3-DMAC) perform well on high-resolution datasets but falter on nuScenes, where 32-beam LiDAR sparsity hinders FPFH matching. High-level methods (e.g., G3Reg, Segregator) exhibit instability across diverse scenes. Notably, QuadricsReg consistently outperforms all methods across all datasets. Its success rate remains stable as viewpoint distance increases, demonstrating the effectiveness of QuadricsReg in the odometry test.

Furthermore, we present the qualitative registration performance of QuadricsReg in the odometry test in Fig. 9(e) and (f). The introduction of random rotations further impacts the geometric attributes of objects in the scene, such as their orientation. The odometry test dataset spans a diverse range of scenes, including challenging cases such as sparse-object environments and unstructured settings, which pose substantial challenges to the global registration. Despite these difficulties, QuadricsReg exhibits robust performance, owing to its ability to accurately estimate the geometric properties of underlying surfaces from point clouds and to effectively represent them using quadrics. Moreover, experiments conducted on the nuScenes-ODO dataset, collected using a 32-beam LiDAR, further validate QuadricsReg's robustness to variations in spatial arrangement and point cloud density due to different sensor types, as also illustrated in Fig. 9(f).

E. Evaluation of Transformation Estimation

In this section, we assess the transformation estimation performance by comparing QuadricsReg against the same baselines detailed in Section V-D1, which are categorized into learning-based, handcrafted, and high-level methods. To demonstrate the effectiveness of our quadric-based transformation optimization, we additionally report the results of QuadricsReg without transformation optimization. The RRE and RTE results on the KITTI-LC dataset are shown in Fig. 12. Generally, transformation errors increase from easy to hard viewpoint distances for all methods. Learning-based methods exhibit the largest error variances in both RRE and RTE. Methods based on high-level representations generally obtain lower rotation and translation errors compared to those utilizing lower-level representations. Notably, QuadricsReg achieves competitive performance even before

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

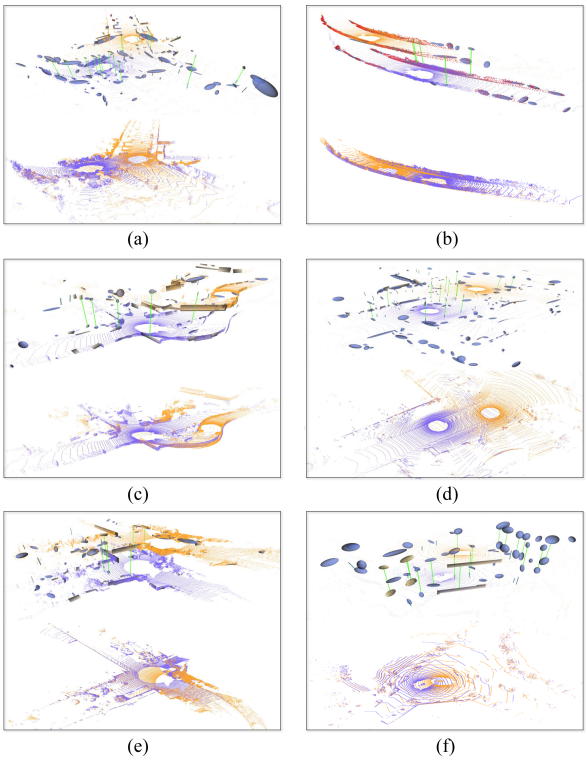


Fig. 9. Global point cloud registration results across diverse datasets. (a) (a) KITTI: 20.48 m/55.63°. (b) KITTI: 23.27 m/9.67°. (c) KITTI-360: 25.59 m/112.18°. (d) Apollo: 28.26 m/177.81°. (e) Waymo: 6.15 m/22.84°. (f) nuScenes: 7.76 m/66.58°.

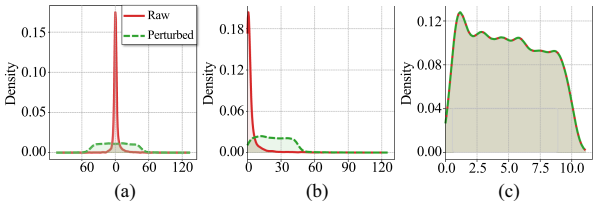


Fig. 10. Distributions of ground truth relative transformations in the KITTI-ODO dataset. (a)–(b) Raw data (red) is heavily concentrated on small rotational variations, with $\sim 85\%$ of rotations smaller than $5°$; in contrast, applying random yaw perturbations in $[-45°, 45°]$ (green) results in a more uniform distribution. (c) Translation magnitude uniformly covers the 1–10 m range. (a) Yaw (deg). (b) Rotation (deg). (c) Translation (m).

transformation optimization, primarily due to the effectiveness of its quadric-based representation and matching, which result in accurate correspondences and precise initial estimation. After incorporating transformation optimization, the errors are further reduced, surpassing all other methods. This outcome not only demonstrates the effectiveness of the optimization process but also proves the robustness of quadrics to geometric degeneracy. This is attributed to our residual computation strategy, which evaluates residual strictly on nondegenerate attributes, thereby enhancing the validity of both the optimization objective and the selection criteria.

F. Evaluation of Scene Representation

In this section, we evaluate the spatial and computational efficiency of scene representations, specifically contrasting

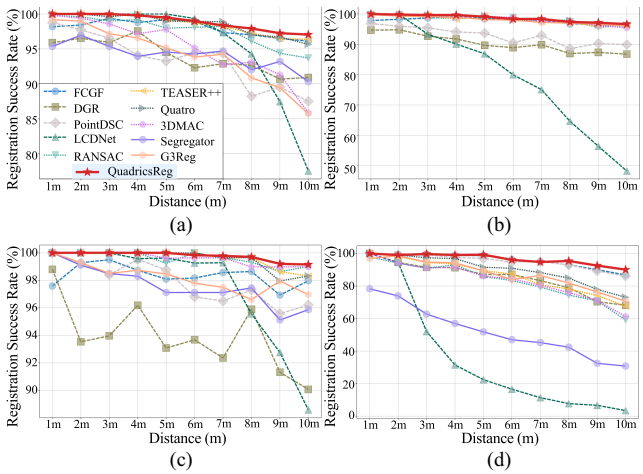


Fig. 11. Evaluation of the global registration success rate in LiDAR odometry test on KITTI-ODO, Apollo-ODO, Waymo-ODO, and nuScenes-ODO datasets. (a) KITTI-ODO. (b) Apollo-SouthBay-ODO. (c) Waymo-ODO. (d) nuScenes -ODO.

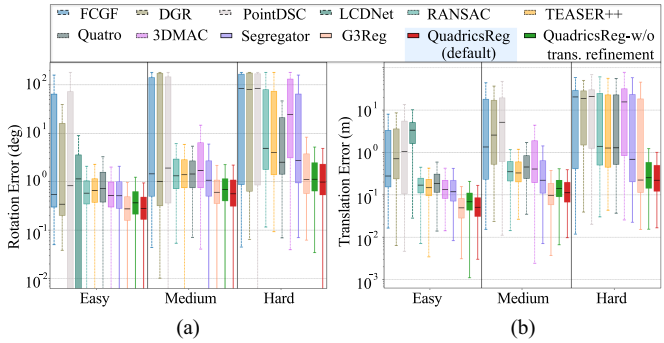


Fig. 12. Evaluation of transformation estimation on KITTI-LC dataset. (a) Relative rotation error (RRE). (b) Relative translation error (RTE).

our high-level quadric primitives against standard point representations. To establish a fair efficiency benchmark, we select RANSAC and TEASER++ as representative baselines. As widely recognized standards for point-based global registration, we utilize their off-the-shelf implementations, the RANSAC module from the Open3-D library [73] and the official TEASER++ release. All methods are evaluated on an identical computing setup to ensure a consistent experimental environment. We evaluate these baselines under voxel downsampling rates of 0.3, 0.5, and 0.7 to analyze the impact of point density. For QuadricsReg, in addition to the default, we introduce a fast variant with reduced quadric density ($K_e = 30$) and fewer compatibility graph levels (thresholds $[0.2, 0.8]$), targeting computation-constrained platforms or applications requiring faster response.

The comparison results on the hard-level of the KITTI-LC sequence 08 are presented in Fig. 13. In terms of storage, the quadric-based representation significantly reduces overhead compared to point-based representations. At a voxel downsampling rate of 0.5, each scan ($\sim$ 29.5 KB) requires less than one-thirtieth of the space of point clouds, achieved by efficiently encoding key structures using vectorized quadrics. In terms of runtime, the default QuadricsReg performs comparably to the baselines (between RANSAC-0.7 and TEASER++-0.7),

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

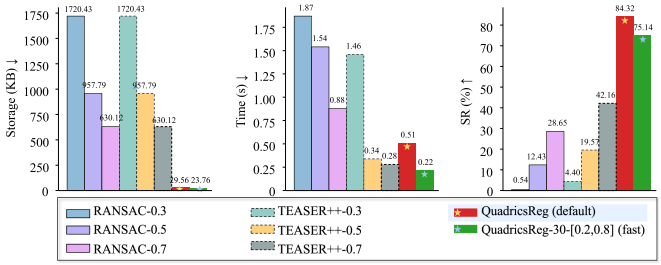


Fig. 13. Efficiency evaluation of scene representation on the hard-level of the KITTI-LC sequence 08. Storage is measured by uniformly storing the representation data in text format.

TABLE VII
PERFORMANCE EVALUATION OF DIFFERENT CONFIGURATIONS ON THE HARD-LEVEL OF THE KITTI-LC SEQUENCE 08

	Runtime			Performance					
	Rep.	Mat.	Est.	Num.	IR ↑	CR ↑	SR ↑	RTE ↓	RRE ↓
QuadricsReg	0.33	0.17	0.33	1439.63	0.60	90.81	84.32	0.27	1.01
QuadricsReg (fast)	0.33	0.08	0.13	841.59	0.79	81.62	75.14	0.29	1.11

Runtime metrics: rep., mat., and est. Denote the runtime (s) for scene representation, matching, and transformation estimation; Performance metrics: Num.: Number of correspondences; IR: Inlier ratio (%); CR: Correspondence recall (%); SR: Registration success rate (%); RTE: Relative translation error (m); RRE: Relative rotation error (deg).

while the fast variant outperforms TEASER++-0.7. We provide a detailed breakdown of runtime and registration metrics in Table VII. Although scene representation is computationally demanding, it typically executes independently during the odometry thread on selected keyframes. Consequently, the backend (matching and estimation), achieves ∼2 Hz (default) and 5 Hz (fast), both sufficient for loop closure.

Table VII further elucidates the tradeoff between speed and accuracy. Specifically, the strict quadric selection acts as a denoising filter that prioritizes prominent structures. This strategy significantly boosts IR by removing unstable primitives, yet inevitably lowers CR since valid objects beyond the fixed capacity are excluded. Coupled with the coarser outlier rejection resulting from reduced pruning levels, this leads to a moderate decrease in SR on hard sequences. Nevertheless, these results demonstrate that even under challenging hard-level conditions (20–30 m), the fast variant maintains a satisfactory success rate despite the performance decline, thereby offering a viable high-speed alternative.

G. Ablation Study

This section presents the ablation study focusing on scene representation, correspondence establishment, and transformation estimation to demonstrate the contributions of key modules in QuadricsReg and validate the rationality of parameter settings. Results on the KITTI-LC dataset are shown in Table VIII, with the last row representing the default settings.

1) Impact of Scene Representation: Ablation results on scene representation are shown in rows 1–10. Comparing the geometry-only version (row 1) with the default (row 24) reveals that incorporating semantic information yields higher success rates across all difficulty levels by refining primitive

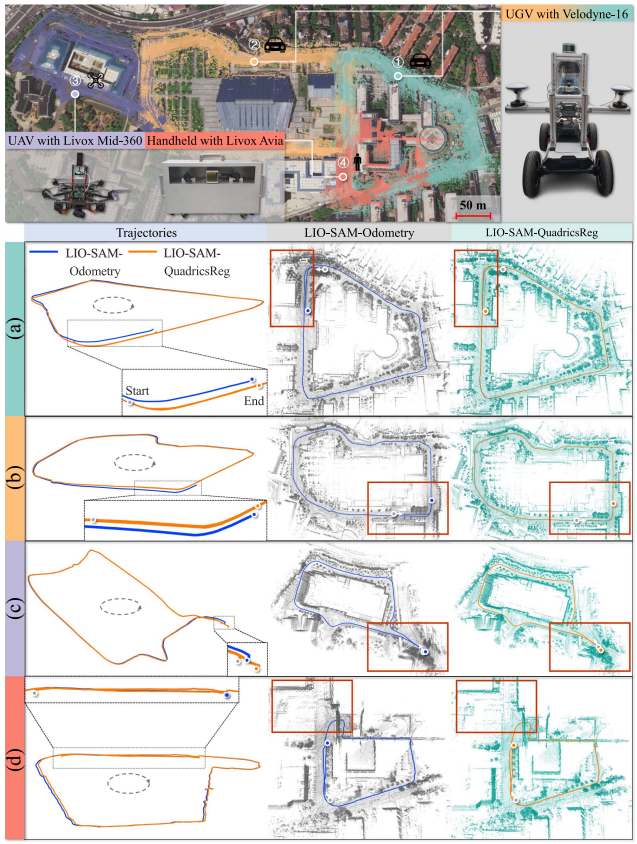


Fig. 14. Localization and mapping results on the self-collected Hetero-Reg dataset with QuadricsReg as the loop closure module. (a) Session 1: 795 m. (b) Session 2: 962 m. (c) Session 3: 601 m. (d) Session 4: 459 m.

extraction and reducing mismatches. Notably, cross-referencing the KITTI-LC results in Table VI indicates that our geometry-only baseline (row 1) outperforms all competing methods, ranking second only to our semantic-guided version. Regarding the fitting strategy, removing either the learning-based (row 2) or statistics-based module (row 3) degrades performance. This confirms their complementarity: the former provides robustness to noise and partial data, while the latter ensures geometric validity in out-of-distribution cases. We also evaluate quadric types. Omitting central quadrics (row 4) has the most significant impact, as they provide essential 3-D constraints. Linear-center (row 5) and planar-center (row 6) quadrics are also vital. The absence of any type leads to inadequate scene representation and poor registration, especially in hard-level cases. Notably, point augmentation (row 7) is crucial for sparse or repetitive environments (*e.g.*, KITTI-02) where quadric abstraction alone is insufficient. Finally, rows 8–10 show that increasing the semantic quadric limit (K_e) improves completeness but may increase overhead. $K_e = 50$ is selected as the optimal balance.

2) Impact of Correspondence Establishment: Ablation results for matching are shown in rows 11–17. Increasing the matching candidate bound (K_s) generally improves potential correspondences (Table V). However, comparing $K_s = 15, 25$ (rows 11–12) with the default ($K_s = 20$, row 24) shows that larger bounds yield diminishing returns while increasing the computational cost of clique searching. Determining a single

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

TABLE VIII
ABLATION STUDY ON KITTI-LC DATASET (REGISTRATION SUCCESS RATE (%)); APPROXIMATELY 1000 PAIRS OF POINT CLOUDS ARE EVALUATED IN TOTAL PER DIFFICULTY LEVEL

	Sequence	00			02			05			06			08		
	Level	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard	Easy	Medium	Hard
Scene Representation	(1) W/O semantics-aided	**100.00**	96.50	59.00	95.93	**73.73**	51.50	**100.00**	95.48	71.50	**100.00**	99.50	**98.50**	**100.00**	98.77	80.00
	(2) W/O learning-based fit	99.50	97.00	75.50	94.19	68.69	49.50	**100.00**	95.48	84.00	**100.00**	**100.00**	93.50	**100.00**	98.77	80.00
	(3) W/O statistics-based fit	99.50	92.00	60.50	86.05	55.05	32.00	99.43	91.46	68.00	**100.00**	99.00	85.50	98.51	96.91	74.05
	(4) W/O central quadrics	98.50	80.00	40.50	61.05	50.51	38.00	89.20	57.79	28.00	**100.00**	99.00	85.00	97.01	83.33	58.38
	(5) W/O linear-center quadrics	99.50	95.50	55.00	92.44	56.57	30.50	**100.00**	95.48	76.00	**100.00**	**100.00**	96.00	**100.00**	95.06	67.57
	(6) W/O planar-center quadrics	99.50	98.00	68.50	94.77	61.11	40.00	**100.00**	95.98	79.50	**100.00**	**100.00**	88.50	**100.00**	97.53	76.22
	(7) W/O augmented points	99.50	98.00	71.00	86.63	66.67	50.50	**100.00**	**98.49**	84.50	**100.00**	**100.00**	95.00	**100.00**	98.77	83.24
	(8) $K_e = 30$	**100.00**	97.50	70.50	94.77	71.21	52.50	**100.00**	97.49	78.50	**100.00**	**100.00**	88.00	**100.00**	98.15	77.84
	(9) $K_e = 40$	99.50	98.00	72.50	94.77	71.21	52.50	**100.00**	97.49	80.00	**100.00**	**100.00**	93.50	**100.00**	98.77	80.00
	(10) $K_e = 60$	99.50	98.00	72.00	94.77	70.71	52.50	**100.00**	97.99	84.00	**100.00**	**100.00**	95.00	**100.00**	98.77	**86.49**
Corr. Estab.	(11) $K_s = 15$	99.50	98.00	71.50	94.77	70.20	52.50	**100.00**	98.49	84.00	**100.00**	**100.00**	95.00	**100.00**	98.77	82.70
	(12) $K_s = 25$	**100.00**	99.00	**84.00**	93.60	72.21	53.00	**100.00**	96.98	**88.00**	**100.00**	**100.00**	98.00	**100.00**	**100.00**	**86.49**
	(13) $\delta_m : [0.2]$	99.50	91.00	47.00	87.79	60.10	34.00	**100.00**	91.46	56.50	**100.00**	99.00	82.00	99.25	94.44	65.41
	(14) $\delta_m : [0.8]$	99.00	97.50	61.50	94.19	70.71	49.50	**100.00**	95.48	77.50	**100.00**	**100.00**	88.50	**100.00**	98.77	78.91
	(15) $\delta_m : [0.2, 0.8]$	99.50	97.00	70.00	94.77	68.18	52.50	**100.00**	97.99	80.50	**100.00**	**100.00**	94.50	**100.00**	98.77	81.08
	(16) $\delta_m : [0.2, 0.6, 0.8]$	99.50	97.50	71.50	94.77	69.70	53.00	**100.00**	**98.49**	83.50	**100.00**	**100.00**	93.50	**100.00**	98.77	82.70
	(17) $\delta_m : [0.2, 0.4, 0.6, 0.8, 1.0]$	99.50	98.00	76.00	94.77	71.21	53.00	**100.00**	97.99	84.50	**100.00**	**100.00**	95.00	**100.00**	98.77	84.86
Trans. Est.	(18) W/O ground-aligned filtering	99.50	**99.50**	79.00	**96.51**	68.18	51.50	**100.00**	97.99	85.00	**100.00**	**100.00**	95.00	**100.00**	98.77	82.16
	(19) $\mathbf{e}_{p2p}$	86.50	81.50	70.00	92.44	67.67	47.00	86.93	89.44	81.00	93.02	86.50	86.50	**100.00**	96.91	84.32
	(20) W/O $\mathbf{e_R}$	99.50	98.00	71.50	95.35	70.71	52.50	**100.00**	**98.49**	84.00	**100.00**	**100.00**	95.00	**100.00**	98.77	83.24
	(21) W/O $\mathbf{e_t}$	99.50	98.00	73.00	94.19	70.71	53.00	**100.00**	97.99	84.00	**100.00**	**100.00**	95.00	**100.00**	98.77	82.16
	(22) W/O supplementary points	99.50	98.00	72.00	94.19	69.19	53.00	**100.00**	97.49	84.50	**100.00**	**100.00**	95.00	**100.00**	98.77	82.70
	(23) W/O trans. refinement	99.50	98.00	72.00	93.60	69.70	52.50	**100.00**	97.99	85.00	**100.00**	**100.00**	94.50	**100.00**	98.77	82.16
	(24) Default	99.50	**99.50**	80.00	**96.51**	71.21	**53.50**	**100.00**	**98.49**	87.50	**100.00**	**100.00**	96.00	**100.00**	98.77	84.32

Bold indicates the best and underlined denotes the second-best.

consistency threshold is difficult due to varying noise. Single-level graphs, whether strict (row 13) or loose (row 14), fail to robustly identify inliers. In contrast, our multilevel strategy (rows 15–17) tolerates observation errors by integrating constraints at different scales. However, excessively loose thresholds (row 17, up to 1.0) expand the search space and may introduce erroneous edges. Thus, a four-level hierarchy capped at 0.8 is adopted for efficiency and accuracy.

3) Impact of Transformation Estimation: The results in rows 18–23 evaluate the transformation estimation components. Ground-aligned filtering (row 18) is beneficial, as pruning non-aligned elements ensures that refinement relies on reliable rotational cues. Replacing our degeneracy-aware quadric metric with a standard point-to-point residual ($\mathbf{e}_{p2p}$, row 19) leads to significant performance drops. This confirms that point metrics struggle with degenerate primitives (e.g., sliding along planes), whereas our metric provides unified measurement across points, lines, and planes. Rows 20–21 demonstrate that both rotation ($\mathbf{e_R}$) and translation ($\mathbf{e_t}$) terms are critical for stability. Furthermore, supplementary points (row 22) help mitigate the limited constraints provided by degenerate quadrics during optimization. Finally, omitting transformation refinement (row 23) reduces accuracy, particularly in hard-level cases. As shown in Fig. 12, the coarse-to-fine process consistently minimizes errors across all levels.

H. Real-World Applications

To demonstrate the real-world applicability of QuadricsReg, we establish the Hetero-Reg dataset by collecting multiple mapping sessions on a campus using three heterogeneous platforms: a UGV equipped with a Velodyne-16 (and RTK for ground truth), a UAV with a Livox Mid-360, and a handheld Livox Avia. The dataset consists of two parts. First, we collect four large-scale sessions (1–4, covering ∼2.8 km) to evaluate single-session SLAM loop closure and multisession mapping capabilities, as illustrated in Fig. 14. Second, to assess robustness in challenging scenarios, we collect two specialized sequences (5–6, covering ∼170 m each) using the UAV in unstructured forest and narrow corridor environments, respectively (see Fig. 18).

1) Loop Closure in SLAM: We integrate QuadricsReg as a loop closure module into LIO-SAM [16] to evaluate its effectiveness for localization and mapping. LIO-SAM utilizes a radius search strategy to identify revisited locations and generate loop candidates. Upon loop detection, QuadricsReg estimates the transformation between point clouds. The loop closure is then incorporated as constraints into a pose graph for further optimization. However, inaccuracies in pose estimation can introduce noise into the pose graph, resulting in suboptimal optimization. As shown in Fig. 14, the results indicate that incorporating QuadricsReg for loop closure allows the trajectory to be correctly aligned from start to end, effectively eliminating map inconsistencies in revisited areas. In particular, the significant z-axis drift in the LIO-SAM odometry for sessions 1 and 2 is successfully corrected by applying QuadricsReg in the pose graph. In sessions 3 and 4, the shorter distances result in minimal drift along the odometry. However, map inconsistencies still exist in revisited areas, which are also eliminated after using the pose graph optimization.

To quantitatively evaluate the loop closure performance of QuadricsReg in real-world SLAM application, we generate loop candidates for sessions 1 and 2 using RTK-provided positioning information with a search radius of 5 m. We select two typical competitors, i.e., RANSAC and TEASER++. The results are shown in Fig. 15. QuadricsReg achieves the highest success rate for both sessions, demonstrating its robust performance even with relatively sparse point clouds.

Furthermore, we test the generalization ability of QuadricsReg on different LiDAR sensors and the results are presented in

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

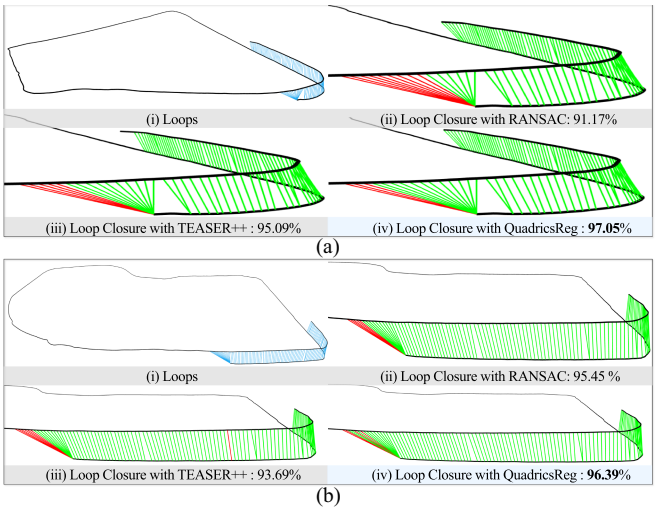


Fig. 15. Evaluation of loop closure on session 1 and session 2 of Hetero-Reg dataset. Black lines show trajectories, elevated for clarity. (a) Loop closure for Session 1. (b) Loop closure for Session 2. Subfigure (i) displays the full trajectory with blue lines marking GT loop frames. The remaining subfigures zoom in on loop regions, where green lines highlight successful closures by global registration and red lines denote failures.

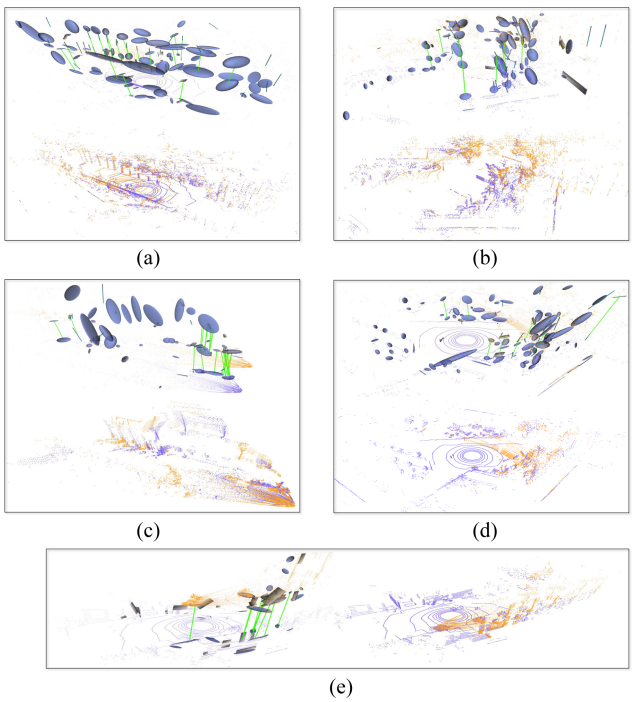


Fig. 16. Real-world global registration results on point clouds scanned by different sensors. (a) Velodyne-16: 12.27 m/3.23°. (b) Livox Mid-360: 15.10 m/19.86°. (c) Livox Avia: 1.87 m/9.01°. (d) Livox Mid-360-Velodyne-16: 10.08 m/35.72°.

Fig. 16(a)–(c). QuadricsReg effectively represents point clouds from different LiDAR types, accounting for spatial arrangement, density, and field of view (FOV) differences, while establishing precise correspondences and achieving robust registration. Specifically, the narrow view of the Livox Avia (horizontal FOV of approximately 70°) poses challenges for object clustering, leading to minor inaccuracies in scene representation.

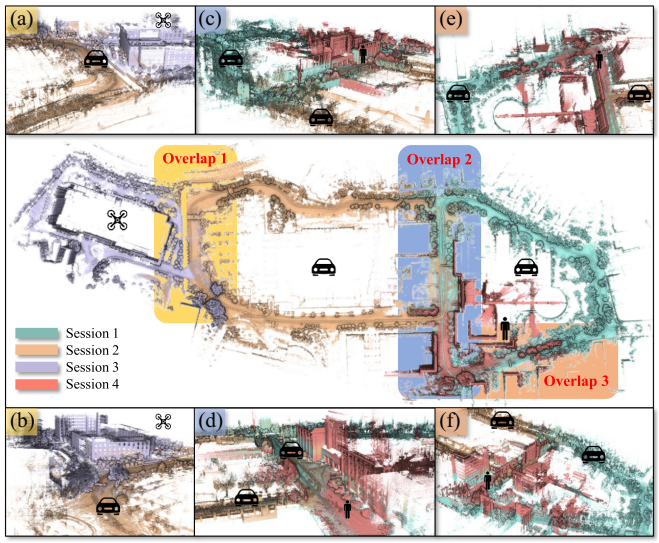


Fig. 17. Multisession mapping on Hetero-Reg dataset. (a)-(f) Provide detailed views of the map. QuadricsReg is utilized for global registration within the same session for the same LiDAR and across sessions for different LiDARs.

Nevertheless, QuadricsReg successfully achieves accurate point cloud registration, demonstrating the robustness of its backend matching and transformation estimation modules.

2) Multisession Mapping: We apply QuadricsReg to multi-session mapping, focusing on evaluating the global registration performance across heterogeneous robot platforms and different LiDAR sensors. Fig. 16(d)–(e) presents the results, indicating that representing the scene using quadrics mitigates the registration challenges posed by inconsistent point distributions from various LiDAR sensors. Based on these registration results, we conduct the multisession mapping experiment, as shown in Fig. 17. The local session maps are constructed using LIO-SAM, integrating QuadricsReg for loop closure. Fig. 17(a)–(f) illustrates the map merging results in three overlapping regions, demonstrating that maps from heterogeneous robotic platforms align effectively despite discrepancies in their operational workspaces. Therefore, QuadricsReg can provide effective transformations for map merging tasks on heterogeneous robotic platforms equipped with different sensors, even with significant viewpoint differences and inconsistent point distributions. For merging multiple sessions, QuadricsReg is triggered when overlaps between robots' trajectories are detected. The estimated transformation using QuadricsReg is incorporated into the pose graph for joint optimization. Finally, we obtain the merged large-scale map of 4 sessions under different LiDAR sensors and heterogeneous robot platforms, demonstrating the capability of QuadricsReg on global registration of large-scale scenarios.

3) Global Registration in Challenging Scenarios: Finally, we evaluate the performance of QuadricsReg in two challenging scenes: an unstructured forest and a geometrically degenerated corridor. Following the setup in Section V-A2, we adopt an odometry test protocol with random rotations. We configure three difficulty levels with distances of 1, 2, and 3 m, and sample 140, 70, and 45 pairs per sequence for each level, respectively.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

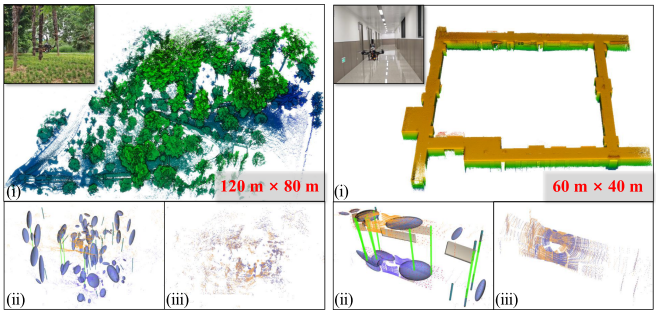


Fig. 18. Illustration of QuadricsReg in challenging scenarios. (a) Forest. (b) Corridor.

TABLE IX
GLOBAL REGISTRATION TEST IN CHALLENGING SCENES (REGISTRATION SUCCESS RATE (%))

	Sequence	5 (Forest)			6 (Corridor)		
	Level	1 m	2 m	3 m	1 m	2 m	3 m
Handcraft	RANSAC	90.27	73.23	61.36	55.10	32.86	11.36
	TEASER++ [4]	83.33	70.42	50.00	41.49	27.14	9.09
	Quatro [72]	90.97	83.09	65.90	**89.79**	**62.85**	**27.27**
	3DMAC [46]	84.72	56.33	45.45	45.57	25.71	13.64
High.	G3Reg [5]	95.23	90.14	79.54	26.53	11.43	2.27
	QuadricsReg	**98.61**	**92.96**	**82.22**	64.63	34.29	13.64

Bold indicates the best and underlined denotes the second-best.

The GT transformation for each sample pair is derived from SLAM odometry and refined with ICP. Qualitative results are shown in Fig. 18, and quantitative comparisons are provided in Table IX.

The results demonstrate that even in geometrically sparse environments such as forests, QuadricsReg achieves robust global registration by relying solely on geometric structures as trunks, canopies, and vegetation, represented as distinct quadric primitives. Its registration success rate consistently exceeds that of point-based methods and the Gaussian ellipsoid-based G3Reg. In the corridor scenario, performance deteriorates markedly due to highly degenerate geometric features. High-level representation-based methods are especially affected, as the absence of isolated objects or structures results in insufficient feature correspondences and frequent registration failures. QuadricsReg partially alleviates this issue through point augmentation and achieves better performance.

I. Discussion

1) Robustness Against Representational Bias: QuadricsReg achieves strong registration performance in large-scale scenes by compactly and uniformly modeling diverse structures. However, its geometric abstraction may exhibit bias in unstructured scenarios, which directly affects the quality of matching. As shown in Fig. 19, we compare the IR and SR of QuadricsReg on the structured KITTI-ODO dataset and the self-collected unstructured forest dataset under difficulty levels of 1–3 m intervals. In unstructured scenes, the IR drops significantly compared to

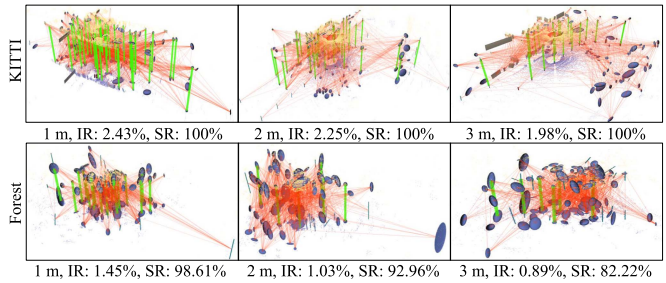


Fig. 19. Robust matching and registration under representational bias.

TABLE X
ROBUSTNESS ANALYSIS AGAINST SEMANTIC DETERIORATION (REGISTRATION SUCCESS RATE (%))

Deterioration Ratio (%)	KITTI-LC			KITTI-360-LC		
	Easy	Medium	Hard	Easy	Medium	Hard
0	99.20	93.59	80.26	99.30	91.00	68.90
10	98.60	92.81	79.56	99.00	89.67	64.33
30	98.02	92.73	75.85	98.58	88.33	61.25
50	97.44	92.12	73.12	98.17	85.58	55.75

structured scenes, primarily due to representational bias weakening the discriminability of similarity-driven matching. However, essential inliers are effectively established. Consequently, the SR remains stable owing to the multilevel graph pruning strategy, which successfully isolates these preserved inliers by maximizing distance-consistent subsets. Thus, our framework exhibits sufficient robustness to the inevitable modeling bias of unstructured point segments, ensuring that quadric-based representation maintains effectiveness even in complex environments.

2) Robustness Against Semantic Deterioration: To evaluate the resilience of our method to the quality of semantic segmentation, we conduct a stress test by intentionally deteriorating the predicted semantic labels. In our default configuration, semantic labels of key objects are predicted by PVKD. We represent challenging real-world scenarios where reliable or complete semantic information is often unavailable by randomly masking a specific percentage of the predicted labels as unlabeled. As shown in Table X, we report the registration success rates on KITTI-LC and KITTI-360-LC under various label-omission ratios from 0% to 50%. The results indicate that while semantic deterioration leads to performance drops, the impact is primarily observed in hard-level (20–30 m) cases. Notably, QuadricsReg exhibits minimal performance decline under a 10% omission ratio. Even under severe deterioration where 50% of labels are removed, our method maintains high success rates in easy (0–10 m) and medium (10–20 m) levels. These findings confirm that our framework is not critically dependent on high-fidelity segmentation. We effectively leverage key structures, including planes, lines, and unstructured clusters, that are extracted independently of predicted semantics and are sufficiently discriminative to support robust registration. Furthermore, the geometry-informed matching and graph-based pruning effectively compensate for deficient semantic cues. This

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

architecture ensures that the system remains stable even under suboptimal semantic segmentation quality.

3) Limitations and Future Work: As an element-level registration framework, the primary limitation of QuadricsReg lies in the robust extraction and matching of valid quadrics. Quadric extraction relying solely on LiDAR-based semantic and geometric cues may degrade in complex scenarios, especially when low overlaps lead to inconsistent quadric models, thereby compromising matching accuracy. Future work will explore fusing rich image texture to enhance extraction, while investigating a principled integration of semantics and geometry for adaptive quadric modeling, which is a critical step towards enhancing matching robustness. Additionally, in dynamic environments, although moving objects can be modeled as quadrics, the relative geometric consistency between them is disrupted, yielding noisy connections in the matching graph. Consequently, another avenue is to explicitly exclude dynamic entities by estimating their motion states, thereby maintaining registration stability in dynamic scenes.

VI. Conclusion

This article presented QuadricsReg, which leveraged semantic quadric primitives to compactly, expressively, and discriminatively represent 3-D scenes through a concise mathematical formulation. These strengths were seamlessly integrated into the design of matching and transformation estimation, contributing to an efficient and robust point cloud registration framework with strong scalability to large-scale environments. Extensive experiments on five public and one self-collected datasets demonstrated that QuadricsReg achieved outstanding performance across different LiDAR sensors, robot platforms, and scenes in a lightweight manner, and can be effectively applied to real-world SLAM systems. In addition, concerns regarding robustness to representational bias and semantic deterioration were discussed, and future improvements were suggested in the direction of robust quadric extraction, matching, and dynamic entity filtering. With its unified and lightweight nature, this framework showed potential in bridging modality gaps across heterogeneous 3-D sensors and improving the scalability of collaborative robot systems in practical deployments.

References

[1] H. Lim, B. Kim, D. Kim, E. Mason Lee, and H. Myung, "Quatro : Robust global registration exploiting ground segmentation for loop closing in LiDAR SLAM," *Int. J. Robot. Res.*, vol. 43, no. 5, pp. 685–715, 2024.

[2] T. Shiratori et al., "Efficient large-scale point cloud registration using loop closures," in *Proc. Int. Conf. 3D Vis.*, 2015, pp. 232–240.

[3] H. Yin et al., "A survey on global LiDAR localization: Challenges, advances and open problems," *Int. J. Comput. Vis.*, vol. 132, no. 8, pp. 3139–3171, 2024.

[4] H. Yang, J. Shi, and L. Carlone, "TEASER: Fast and certifiable point cloud registration," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 314–333, Apr. 2021.

[5] Z. Qiao, Z. Yu, B. Jiang, H. Yin, and S. Shen, "G3Reg: Pyramid graph-based global registration using Gaussian ellipsoid model," *IEEE Trans. Autom. Sci. Eng.*, vol. 25, pp. 3416–3432, 2024.

[6] Z. J. Yew and G. H. Lee, "RPM-Net: Robust point matching using learned features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11821–11830.

[7] M. Prokop, S. A. Shaikh, and K.-S. Kim, "Low overlapping point cloud registration using line features detection," *Remote Sens.*, vol. 12, no. 1, 2020, Art. no. 61.

[8] S. Chen, L. Nan, R. Xia, J. Zhao, and P. Wonka, "Plade: A plane-based descriptor for point cloud registration with small overlap," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2530–2540, Apr. 2020.

[9] P. Geneva, K. Eckenhoff, Y. Yang, and G. Huang, "LIPS: LiDAR-inertial 3D plane SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 123–130.

[10] L. Zhou, G. Huang, Y. Mao, J. Yu, S. Wang, and M. Kaess, "$\mathcal{PLC}$-LiSLAM: LiDAR SLAM with planes, lines, and cylinders," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 7163–7170, Mar. 2022.

[11] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[12] C. Papazov, S. Haddadin, S. Parusel, K. Krieger, and D. Burschka, "Rigid 3D geometry matching for grasping of known objects in cluttered scenes," *Int. J. Robot. Res.*, vol. 31, no. 4, pp. 538–553, 2012.

[13] J. Wu, H. Yu, W. Yang, and G.-S. Xia, "Quadricsnet: Learning concise representation for geometric primitives in point clouds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 4060–4066.

[14] W. Zhen, H. Yu, Y. Hu, and S. Scherer, "Unified representation of geometric primitives for graph-SLAM optimization using decomposed quadrics," in *Proc. Int. Conf. Robot. Autom.*, 2022, pp. 5636–5642.

[15] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," in *Proc. Robot.: Sci. Syst.*, 2014, pp. 1–9.

[16] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and R. Daniela, "LIO-SAM: Tightly-coupled LiDAR inertial odometry via smoothing and mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5135–5142.

[17] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO2: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.

[18] J. Ruan, B. Li, Y. Wang, and Y. Sun, "Slamesh: Real-time LiDAR simultaneous localization and meshing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2023, pp. 3546–3552.

[19] J. Lin et al., "ImMesh: An immediate LiDAR localization and meshing framework," *IEEE Trans. Robot.*, vol. 39, no. 6, pp. 4312–4331, Dec. 2023.

[20] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "Octomap: An efficient probabilistic 3D mapping framework based on octrees," *Auton. Robots*, vol. 34, pp. 189–206, 2013.

[21] X. Chen, A. Milioto, E. Palazzolo, P. Giguère, J. Behley, and C. Stachniss, "SuMa: Efficient LiDAR-based Semantic SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4530–4537.

[22] R. Dubé, D. Dugas, E. Stumm, J. Nieto, R. Siegwart, and C. Cadena, "Segmatch: Segment based place recognition in 3D point clouds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 5266–5272.

[23] X. Kong et al., "Semantic graph based place recognition for 3D point clouds," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 8216–8223.

[24] A. Cramariuc et al., "Semsegmap–3D segment-based semantic localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 1183–1190.

[25] P. Yin, S. Yuan, H. Cao, X. Ji, S. Zhang, and L. Xie, "Segregator: Global point cloud registration with semantic and geometric cues," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2023, pp. 2848–2854.

[26] M. B. Peterson, Y. X. Jia, Y. Tian, A. Thomas, and J. P. How, "Roman: Open-set object map alignment for robust view-invariant global localization," in *Proc. Robot.: Sci. Syst.*, 2025.

[27] A. Rosinol et al., "Kimera: From SLAM to spatial perception with 3D dynamic scene graphs," *Int. J. Robot. Res.*, vol. 40, no. 12-14, pp. 1510–1546, 2021.

[28] X. Zuo, X. Xie, Y. Liu, and G. Huang, "Robust visual SLAM with point and line features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 1775–1782.

[29] H. Yu, W. Zhen, W. Yang, J. Zhang, and S. Scherer, "Monocular camera localization in prior LiDAR maps with 2D-3D line correspondences," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 4588–4594.

[30] D. Chen et al., "Vip-slam: An efficient tightly-coupled RGB-D visual inertial planar slam," in *Proc. Int. Conf. Robot. Autom.*, 2022, pp. 5615–5621.

[31] S. Yang and S. Scherer, "CubeSLAM: Monocular 3-D object SLAM," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 925–938, Aug. 2019.

[32] L. Nicholson, M. Milford, and N. Sünderhauf, "QuadricSLAM: Dual quadrics from object detections as landmarks in object-oriented SLAM," *IEEE Robot. Autom. Lett.*, vol. 4, no. 1, pp. 1–8, Jan. 2019.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.

[33] R. Schnabel, R. Wahl, and R. Klein, "Efficient ransac for point-cloud shape detection," *Comput. Graph. Forum*, vol. 26, no. 2, 2007, pp. 214–226.
[34] L. Li, M. Sung, A. Dubrovina, L. Yi, and L. J. Guibas, "Supervised fitting of geometric primitives to 3D point clouds," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2652–2660.
[35] G. Sharma, D. Liu, S. Maji, E. Kalogerakis, S. Chaudhuri, and R. Mech, "Parsenet: A parametric surface fitting network for 3D point clouds," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 261–276.
[36] C. Xia et al., "Quadric representations for LiDAR odometry, mapping and localization," *IEEE Robot. Autom. Lett.*, vol. 8, no. 8, pp. 5023–5030, Aug. 2023.
[37] F. Tombari, S. Salti, and L. Di Stefano, "Unique shape context for 3D data description," in *Proc. ACM Workshop 3D Object Retrieval*, 2010, pp. 57–62.
[38] S. Salti, F. Tombari, and L. Di Stefano, "Shot: Unique signatures of histograms for surface and texture description," *Comput. Vis. Image Understanding*, vol. 125, pp. 251–264, 2014.
[39] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 3384–3391.
[40] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 3212–3217.
[41] C. Choy, J. Park, and V. Koltun, "Fully convolutional geometric features," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 8958–8966.
[42] 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. 6358–6366.
[43] G. Pramatarov, D. De Martini, M. Gadd, and P. Newman, "Boxgraph: Semantic place recognition and pose estimation from 3D LiDAR," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 7004–7011.
[44] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone, "Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1127–1134, Apr. 2020.
[45] P. C. Lusk, K. Fathian, and J. P. How, "Clipper: A graph-theoretic framework for robust data association," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 13828–13834.
[46] X. Zhang, J. Yang, S. Zhang, and Y. Zhang, "3D registration with maximal cliques," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 17745–17754.
[47] J. Yang et al., "MAC: Maximal cliques for 3D registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10645–10662, Dec. 2024.
[48] K. Fathian and T. Summers, "Clipper: A fast maximal clique algorithm for robust global registration," *IEEE Robot. Autom. Lett.*, vol. 9, no. 4, pp. 3562–3569, Apr. 2024.
[49] H. Lim et al., "Kiss-matcher: Fast and robust point cloud registration revisited," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2025, pp. 11104–11111.
[50] Z. Qiao, Z. Yu, H. Yin, and S. Shen, "Pyramid semantic graph-based global point cloud registration with low overlap," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 11202–11209.
[51] B. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Amer. A*, vol. 4, no. 4, pp. 629–642, 1987.
[52] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 698–700, Sep. 1987.
[53] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 766–782.
[54] S. W. Chen et al., "SLOAM: Semantic LiDAR odometry and mapping for forest inventory," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 612–619, Apr. 2020.
[55] P. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
[56] A. Censi, "An icp variant using a point-to-line metric," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 19–25.
[57] K.-L. Low, *Linear Least-Squares Optimization for Point-to-Plane ICP Surface Registration*," vol. 4. Chapel Hill, NC, USA: Univ. North Carolina, 2004, pp. 1–3.
[58] M. Oh et al., "Travel: Traversable ground and above-ground object segmentation using graph representation of 3D LiDAR scans," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 7255–7262, Jul. 2022.
[59] Y. Hou, X. Zhu, Y. Ma, C. C. Loy, and Y. Li, "Point-to-voxel knowledge distillation for LiDAR semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8479–8488.
[60] G. Taubin, "Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 11, pp. 1115–1138, Nov. 1991.
[61] R. A. Rossi, D. F. Gleich, and A. H. Gebremedhin, "Parallel maximum clique algorithms with applications to network analysis," *SIAM J. Sci. Comput.*, vol. 37, no. 5, pp. C589–C616, 2015.
[62] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
[63] Y. Liao, J. Xie, and A. Geiger, "KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3292–3310, Mar. 2023.
[64] W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, "L3-Net: Towards learning based LiDAR localization for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6389–6398.
[65] P. Sun et al., "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2446–2454.
[66] H. Caesar et al., "nuscenes: A multimodal dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11621–11631.
[67] X. Bai et al., "Pointdsc: Robust point cloud registration using deep spatial consistency," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15859–15869.
[68] C. Choy, W. Dong, and V. Koltun, "Deep global registration," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2514–2523.
[69] J. Behley et al., "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9297–9307.
[70] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 62–69.
[71] D. Cattaneo, M. Vaghi, and A. Valada, "LCDNet: Deep loop closure detection and point cloud registration for LiDAR SLAM," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2074–2093, Aug. 2022.
[72] H. Lim et al., "A single correspondence is enough: Robust global registration to avoid degeneracy in urban environments," in *Proc. Int. Conf. Robot. Autom.*, 2022, pp. 8010–8017.
[73] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," 2018, *arXiv:1801.09847*.



Ji Wu received the B.E. degree in automation from Jiangnan University, Wuxi, China, in 2021. He is currently working toward the Ph.D. degree in computer science with the School of Computer Science, Wuhan University, Wuhan, China.

His research interests include multirobot perception and embodied intelligence.



Huai Yu (Member, IEEE) received B.S. and Ph.D. degrees in communication and information systems from Wuhan University, Wuhan, China, in 2015 and 2020, respectively.

He was a Visiting Scholar from 2018 to 2020 and a Postdoctoral Fellow from 2020 to 2021 with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA. He is currently working as an Associate Professor with the School of Electronic Information, Wuhan University. His research interests include structure perception, 3-D reconstruction, and SLAM.

Dr. Yu received the Best Paper Award for the ISPRS Journal of Photogrammetry and Remote Sensing in 2021. He serves on the Editorial Board of *IEEE Robotics and Automation Letters*.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.



Shu Han is currently working toward the B.E. degree in software engineering with the School of Computer Science, Wuhan University, Wuhan, China.

His research interests include robot perception, generative models, and autonomous driving.



Ximeng Cai received the B.E. degree in artificial intelligence from the School of Computer Science, Wuhan University, Wuhan, China, in 2024, where he is currently working toward the M.S. degree in artificial intelligence.

His research interests include robot perception and embodied intelligence.



Mingfeng Wang received the B.E. and M.S. degrees in computer science from the Southwestern University of Finance and Economics, Chengdu, China, in 2020, and Wuhan University, Wuhan, China, in 2023, respectively. He is currently working toward the Ph.D. degree in software engineering with the School of Computer Science, Wuhan University.

His research interests include robot perception and embodied intelligence.



Wen Yang (Senior Member, IEEE) received the B.S. degree in electronic apparatus and surveying technology, the M.S. degree in computer application technology, and the Ph.D. degree in communication and information systems from Wuhan University, Wuhan, China, in 1998, 2001, and 2004, respectively.

From 2008 to 2009, he worked as a Visiting Scholar with the Apprentissage et Interfaces (AI) Team, Laboratoire Jean Kuntzmann, Grenoble, France. From 2010 to 2013, he worked as a Postdoctoral Researcher with the State Key Laboratory of Information Engineering, Surveying, Mapping and Remote Sensing, Wuhan University. Since then, he has been a Full Professor with the School of Electronic Information, Wuhan University. His research interests include object detection and recognition, multisensor information fusion, and remote sensing.



Gui-Song Xia (Senior Member, IEEE) received the Ph.D. degree in image processing and computer vision from CNRS LTCI, Télécom Paris, Paris, France, in 2011.

From 2011 to 2012, he was a Postdoctoral Researcher with the Centre de Recherche en Mathématiques de la Décision, CNRS, Paris-Dauphine University, Paris, for one and a half years. In 2018, he worked as a Visiting Scholar with DMA, École Normale Supérieure (ENS-Paris) for two months. He was also a guest Professor with Future Lab AI4EO, Technical University of Munich (TUM). He is currently working as a Full Professor with the School of Artificial Intelligence, Wuhan University, Wuhan, China. His current research interests include artificial intelligence, robotics, and remote sensing.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on September 11,2026 at 05:21:36 UTC from IEEE Xplore. Restrictions apply.