Check for updates

Received: 6 January 2025 | Revised: 13 May 2025 | Accepted: 8 August 2025

DOI: 10.4218/etrij.2025-0011

ETRI Journal WILEY

ORIGINAL ARTICLE

Special Issue: Featured Article

ELiOT: End-to-end LiDAR odometry with transformers harnessing real-world, simulated, and digital twin

Daegyu Lee[1] | Hyunwoo Nam[2] | Insung Jang[1] | David Hyunchul Shim[3]

[1]Air Mobility Research Division, Electronics and Telecommunications Research Institute, Daejeon, Rep. of Korea

[2]Hyundai Motors, Hwaseong, Rep. of Korea

[3]Department of Electrical Engineering, Korea Advanced Institute of Science and Technologies, Daejeon, Rep. of Korea

Correspondence
Daegyu Lee, Department of Advanced Defense Technology and Industries, Jeonbuk National University, Jeonju, Rep. of Korea.
Email: lee.dk@jbnu.ac.kr

Funding information
Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government (25ZR1210, DNA-Based National Intelligent Core Technology Development).

Abstract

The development of smart cities depends on intelligent systems that integrate data from diverse environments. In this work, we present **ELiOT**, an end-to-end LiDAR odometry framework with transformer architecture designed to utilize real-world data, simulations, and digital twins. ELiOT leverages high-fidelity simulators and digital twin environments to enable sim-to-real applications, training on the real-world KITTI odometry dataset while benefiting from simulated data for improved generalization. Our self-attention-based flow embedding network eliminates the need for traditional 3D-2D projections by implicitly modeling motion from sequential LiDAR scans. The framework incorporates a 3D transformer encoder-decoder to extract rich geometric and semantic features. By integrating digital twin environments and simulated data into the training process, ELiOT bridges the gap between simulation and real-world applications, offering robust and scalable solutions for urban navigation challenges. This work underscores the potential of combining real-world and virtual data to advance LiDAR odometry and highlights its role for the future smart cities.

KEYWORDS

AI-enabled robotics, robot learning, robot vision

1 | INTRODUCTION

The integration of digital twin technologies and high-fidelity simulation environments has become a key enabler in robotics and autonomous systems, providing scalable platforms for algorithm testing and validation under diverse virtual conditions. These advances align with broader trends in industrial metaverse research, where generative artificial intelligence (AI), cyber-physical systems, and internet-of-things (IoT) are increasingly used for simulation-based development, decision-making, and system management [1–3]. Among the core capabilities required in such virtual and physical autonomous systems, LiDAR Odometry (LO) plays a vital role in estimating ego-motion, especially when developing and validating localization algorithms within simulation environments. Accurate LO is essential for estimating the motion of a robot or vehicle using consecutive point cloud scans, particularly in urban environments where GNSS signals may be unreliable [4, 5]. Especially, point-clouds obtained from a LiDAR or a RGB-D camera can be utilized to estimate the robot motion between two consecutive scans. More precisely, LO algorithms are typically divided into

This is an Open Access article distributed under the term of Korea Open Government License (KOGL) Type 4: Source Indication + Commercial Use Prohibition + Change Prohibition (http://www.kogl.or.kr/info/licenseTypeEn.do).
1225-6463/$ © 2025 ETRI

geometry-based and learning-based methods. A foundational geometry-based method [6] introduced core scan-matching techniques, while [7] improved real-world applicability by efficiently extracting planar and edge features. [8, 9] further extended this by incorporating inertial measurements to enhance robustness. Early learning-based methods [10, 11] applied deep networks to LiDAR data, and [12] advanced this by leveraging LiDAR intensity information to improve localization accuracy.

The goal of the LO is to minimize the error of translation and rotation while accumulating the sequential robot motions. Typically, traditional LO methods are based on the point registration, which iteratively finds point-to-geometry correspondences, and minimizes the pose transformation error. Iterative Closest Point (ICP) [13], G-ICP [14], Normal Distributions Transform (NDT) [15] and its variants [16–18] are well-known examples utilizing the point registration algorithm. Rather than employing computationally expensive dense registration methods, feature-based LO utilizes efficient geometry constraint calculations for its implementation, as exemplified in previous works [7–9]. These registration methods have shown high performance on various benchmarks including the KITTI dataset [19]. Despite these advancements, geometry-based methods may not adequately account for memory considerations, whereas learning-based methods leverage data-driven approaches as demonstrated in [20].

In addition, learning-based approaches have a strength in handling the point-cloud acquired in deteriorated conditions because deep learning for feature extraction is more promising than handcrafted feature extractions by modeling robust and effective cross-domain generalizability as in [21]. Recent learning-based LiDAR odometry methods leverage projection-aware representations to improve efficiency and accuracy. [22] organizes point clouds into cylindrical grids and uses a coarse-to-fine cost volume with attention masks for robust pose estimation. [23] projects 3D points onto 2D surfaces and applies lightweight transformer modules. This form of projection representation has the benefit of allowing the use of image-based network pipelines and concepts from recent architectural studies. However, we note that emerging 3D data detection studies [24, 25] indicates that 2D represented methods cannot fully reflect the characteristics of 3D data and easy to loose features.

In this work, we propose an end-to-end LiDAR odometry framework using 3-D Transformer architectures, **ELiOT**, as illustrated in Figure 1. The main contributions of this paper are as follows:

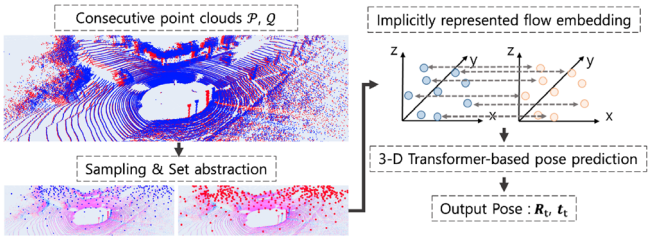


FIGURE 1 Propose end-to-end LO using 3-D transformer frameworks, **ELiOT**.

- We present an end-to-end approach that eliminates the need for conventional geometric concepts such as K-nearest neighbors (*KNN*) and clustering algorithms, enabling a seamless and efficient pipeline.
- We introduce a self-attention flow embedding network designed to implicitly represent motion within sequential LiDAR scenes, enhancing feature extraction and motion estimation.
- To the best of our knowledge, we are the first to propose a 3D transformer-based LiDAR Odometry framework that eliminates the need for 3D-2D projections.
- We leverage diverse datasets for training and validation, including real-world data from the KITTI odometry dataset, simulation data generated in the simulator, and data from a digital twin environment. This multi-environment approach demonstrates the framework's robustness and adaptability across domains.

The remainder of this paper is organized as follows. Section 2 introduces related 3D object detectors and deep LO studies. Section 3 describes our problem definition, network, and its modules for LO. Section 4 discusses implementation details. Experiments conducted on the KITTI odometry dataset, CARLA simulation data, and digital twin environments are detailed in Section 5. Finally, Section 6 concludes this study.

2 | RELATED WORK

2.1 | Feature extraction of 3-D detector

In recent years, 3-D detectors are studied to handle the sparse and unordered characteristic of points. Before directly processing point clouds in 3D, there has been a lot of research that has rasterized data into 2D by converting it to a BEV (Bird's Eye View) in order to utilize image-based recognition models for object recognition [26, 27]. This method of representing 3D data in 2D does not fully reflect the characteristics of 3D data and has difficulty considering the sparse data characteristics. As a

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

result, recent research has explored direct processing of sparse, unordered 3D point sets. [28] pioneered this direction by proposing a permutation-invariant network for point cloud classification and segmentation without voxelization. [29] extended this by capturing local geometric structures through hierarchical feature learning, enabling robust performance on complex 3D scenes.

However, because processing data arrays with tens of millions of points requires a large amount of computation, research has been proposed to efficiently detect objects by voxelizing the 3D space [30, 31]. However, this voxelization method can decrease the accuracy of recognizing small objects, leading to the emergence of research that combines the benefits of both point-wise and voxel-wise approaches through a hybrid method of point-voxel feature extraction [32, 33].

In addition, the Transformer architecture [34] has been effectively applied to image recognition tasks. [35] introduced an end-to-end object detection framework, while [36] extended Vision Transformers to detection with strong pretraining and competitive performance. Recently, there has been a study on the 3D data Transformer backbone [25], incorporating a self-attention algorithm to enhance the receptive field. The main focus of these studies is to enhance the performance of learning-based detection methods by leveraging efficient feature extraction and multi-head self-attention modules, improving the model's ability to capture important spatial features and relationships within 3D data, ultimately leading to more accurate and robust detection results [24, 25].

Our idea is to apply self-attention mechanisms to sequential tasks, specifically pose prediction using LiDAR data. This utilization aims to capture long-range dependencies and temporal relationships, improving the accuracy of pose prediction for applications like autonomous vehicles and robotics.

2.2 | Lidar odometry

LO algorithms are generally categorized into geometry-based and learning-based approaches. One of the most direct approaches to LiDAR odometry is point-cloud registration, commonly implemented using ICP and NDT variants. The original ICP method was introduced in [13], with [16] enhancing it through point-to-line distance minimization, and [14] generalizing it into the widely adopted G-ICP. For NDT methods, [15] proposed the foundational concept, which was extended in [37] for mobile robot mapping in real-world environments. Further refinements include improved convergence strategies [17] and computational efficiency techniques that preserve accuracy [18]. Owing to registration algorithm's large computing consumption, feature-based LO methods [7–9] implement handcrafted feature extraction and constraint for solution space by utilizing inertial measurement unit (IMU) values. Geometry-based methods are intuitive for robot pose prediction and have shown high accuracy in benchmarks [19].

Recently, CNN-based odometry methods have been proposed, and these learning-based studies have originated from research in visual odometry [38, 39]. To make use of this pipeline for visual odometry, research has also been conducted on rasterizing 3D points [40]. Furthermore, rasterized 3D points can be leveraged in multi-modal LiDAR-visual odometry by combining RGB images, depth maps, and rasterized imagery as input modalities [41]. In the wake of the PointNet proposals [28, 29], 3D CNN-based methods have been proposed for end-to-end registration [42–44]. In the wake of the PointNet proposals [28, 29], 3D CNN-based methods have enabled end-to-end LiDAR registration by learning keypoint features [44], achieving accuracy comparable to geometric methods with efficient outlier rejection [42], and estimating full 6-DoF poses using parallel networks for translation and rotation [43]. Deep point-cloud registration has been previously proposed as a solution to suppress the substantial noise that can pervade real-world data [45]. However, conventional geometric concepts such as the *KNN* algorithm are still utilized for motion embedding part [46]. In addition, A learning-based registration algorithm can be extended for domain adaptation work, as it has the advantage of being able to solve localization problems without a prior map by utilizing extracted feature information [47]. Moreover, recent research has expanded into unsupervised learning, enabling learning-based registration without manually or automatically labeled annotations—enhancing overall generalizability [48].

3 | PROPOSED METHOD

3.1 | Problem definition

Input of LO problem is unordered set of 3D points $\mathcal{P} = \{\mathbf{p}_1, ..., \mathbf{p}_n\}$ where n is the size of the query point-cloud. Let us define target set as $\mathcal{Q} = \{\mathbf{q}_1, ..., \mathbf{q}_m\}$ where m is the size of the target point-cloud. Moreover, Individual points in the point sets are $\mathbf{p}_i = \{\mathbf{x}_i, \mathbf{f}_i\}$ and $\mathbf{q}_i = \{\mathbf{y}_i, \mathbf{g}_i\}$ where $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{R}^3$ denote its a vector of position (x, y, z) and $\mathbf{f}_i, \mathbf{g}_i \in \mathbb{R}^c$ are optical feature such as reflection intensity, color or surfaces where c is the number of feature

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

dimension. The objective of **ELiOT** is to learn $\mathbf{T}$ from the sequential set $\{\mathcal{P}, \mathcal{Q}\}$. Let us define $\hat{\mathbf{x}}_t \in \mathrm{SE}(3)$ be the odometry at time t. Then, we can transform $\hat{\mathbf{x}}_t$ to $\hat{\mathbf{x}}_{t+1}$ using $\mathbf{T}_t$, which denotes estimated transformation matrix at time t as $\hat{\mathbf{x}}_{t+1} = \mathbf{T}_t \hat{\mathbf{x}}_t$. Therefore, if the initial pose $\hat{\mathbf{x}}_0$ is obtained from $\mathbf{T}_0 = \mathbf{I}$, we can formulate the odometry pose at time t as following:

$$\hat{\mathbf{x}}_t = \prod_{i=0}^{t} \mathbf{T}_i \mathbf{x}_0. \tag{1}$$

In this study, we train our model using supervised learning, with the aim of minimizing the error between the outcome predicted by (1) and the ground truth odometry. Furthermore, we utilize dual-quaternions to represent the translation and rotation components of $\mathbf{T}_t$ [45, 49, 50].

3.2 | Overall network architecture

Figure 2 presents an overview of the proposed network architecture. We utilize a PointNet++ backbone module [29] in order to extract geometric features via a learning-based approach. This module processes two consecutive point-cloud frames, $\mathcal{P}$ and $\mathcal{Q}$, generating keypoints and their corresponding set of abstracted features.

Subsequently, these keypoints and features are input into the implicitly represented flow embedding (*IRFE*) layers. This allows for the learning of global and local features for motion flows. Finally, Transformer-based encoder-decoder modules are employed to predict the dual-quaternion of the consecutive point-cloud frames.

3.3 | Feature extraction and set abstraction

The first module of our framework is designed to extract features, thereby generating a subsampled set of abstracted data. Because the size of the $\mathbf{P}$ is too large for a training model to process directly, sampling of point clouds is necessary. More precisely, we adopt the Furthest-Point-sampling (FPS) algorithm to extract a small number of n_{key} keypoints $\mathcal{K}_{\mathcal{P}} = \{\mathbf{p}_1, ..., \mathbf{p}_{n_{\text{key}}}\} \subset \mathbf{P}$. As previous deep LO [45, 46] extended multi-scale grouping set abstraction [29] for deep registration algorithm, we also generate a feature set $\mathcal{F}_{\mathcal{P}} = \{\mathbf{f}_1, ..., \mathbf{f}_{n_{\text{key}}}\}$ with n_r radii in multi-scale grouping, where $\mathcal{F}$ is fed into PointNet [28] with nonlinear function $h_{\text{sa}} : \mathbb{R}^{3+c} \to \mathbb{R}^{c'}$, realized as MLP_{sa}, and element-wise max pooling.

As a result, we generate an feature aggregation set for input $\mathcal{S}^{\mathcal{P}}$ which contains $\mathcal{K}_{\mathcal{P}}, \mathcal{F}_{\mathcal{P}}$ and identical mechanism for target set as follows:

$$\begin{aligned} \mathcal{S}^{\mathcal{P}}_{\text{sa},i} &= \{\mathbf{p}_i, \mathbf{f}_i | \forall \mathbf{p}_i \in \mathcal{K}_{\mathcal{P}}, \forall \mathbf{f}_i \in \mathcal{F}_{\mathcal{P}}\}, \\ \mathcal{S}^{\mathcal{Q}}_{\text{sa},i} &= \{\mathbf{q}_i, \mathbf{g}_i | \forall \mathbf{q}_i \in \mathcal{K}_{\mathcal{Q}}, \forall \mathbf{g}_i \in \mathcal{F}_{\mathcal{Q}}\}. \end{aligned} \tag{2}$$

Then, we pass through each $\mathcal{S}^{\mathcal{P}}_{\text{sa},i}$ and $\mathcal{S}^{\mathcal{Q}}_{\text{sa},i}$ to batch normalization of linear layers to generate feature aggregation. Input and target point cloud with dimensions $[n, 3+c]$ is pooled and represented with dimension $[n_{\text{key}}, 3+c']$ where c' is depth of MLP_{sa}.

3.4 | Implicitly represented flow embedding

Our reference studies [45, 46] used flow embedding modules to concatenate consecutive LiDAR frames.

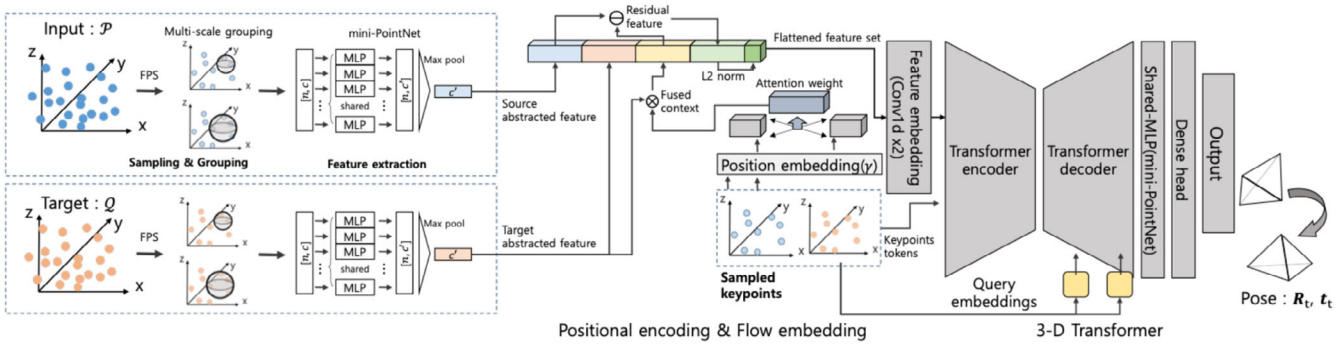


FIGURE 2 An overview of the proposed **ELiOT** network architecture. The pipeline begins with a PointNet++ backbone module that extracts geometric features from two consecutive point cloud frames, $\mathcal{P}$ and $\mathcal{Q}$. The extracted keypoints and features are passed to the Implicitly Represented Flow Embedding (IRFE) module, which computes attention-based soft correspondences between the frames using positional encodings. IRFE generates fused motion representations by aggregating contextual features, motion residuals, and their magnitudes. These motion-aware features are then projected through a Conv1d encoder to align with the transformer input dimension. Finally, a Transformer encoder-decoder predicts the relative pose between the frames in the form of a dual quaternion.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Specifically, they take the sequential feature aggregation set $\mathcal{S}^{\mathcal{P}}_{\text{sa},i}$ and $\mathcal{S}^{\mathcal{Q}}_{\text{sa},i}$ for flow embedding module. Using the (x,y,z) position vector of $\mathbf{p}_i \in \mathcal{S}^{\mathcal{P}}_{\text{sa},i}$ and $\mathbf{q}_i \in \mathcal{S}^{\mathcal{Q}}_{\text{sa},i}$, they implement the *k*-nearest neighbor (*KNN*) and clustering algorithms to index n_{group} of corresponding points between two sequential point clouds. Therefore, they can generate a flow embedding set $\mathcal{S}^{\text{fe}}$ by concatenate abstracted sets as follow:

$$\mathcal{S}^{\text{fe}}_i = \{\mathbf{q}_i - \mathbf{p}_i, \mathbf{f}_i, \mathbf{g}_i\}. \tag{3}$$

While effective in capturing short-range local motion, conventional grouping methods such as KNN are limited by their reliance on spatial proximity. These methods enforce rigid local neighborhoods, which can hinder the ability to represent motion in sparse, noisy, or structurally complex environments. In particular, they struggle to capture long-range or non-local correspondences, which are often critical in dynamic scenes with occlusions or large displacements.

To address these limitations, we propose the *Implicit Representation Feature Extractor* (IRFE) layer, designed to learn a spatially distributed motion representation. Unlike traditional methods, IRFE does not depend on explicit nearest-neighbor correspondences. Instead, it leverages cross-attention between globally encoded positions to extract contextual target features, which are then contrasted with local source features to form motion-aware residuals. This results in a rich representation that captures not only semantic similarity but also the direction and magnitude of motion. By integrating both local structure and global spatial context, IRFE enables the network to infer motion patterns in a fully implicit and data-driven manner, leading to more flexible and robust feature representations across varying environments. Drawing inspiration from NeRF [51], we employ positional encoding to transform a 3D pose into a higher dimensional space. This process stimulates our model to encapsulate more complex and high-frequency data about the surroundings. We define $\gamma(\mathbf{p})$ as the Fourier positional encoding:

$$\begin{aligned}\gamma(\mathbf{p}) = [\mathbf{p}, \quad & \sin(2^0\mathbf{p}), \cos(2^0\mathbf{p}), ..., \\ & \sin(2^{L-1}\mathbf{p}), \cos(2^{L-1}\mathbf{p})].\end{aligned} \tag{4}$$

where L is the dimension of Transformer's decoder.

Unlike conventional approaches such as KNN-based flow embedding, which explicitly select a fixed number of nearest neighbors based on spatial distance, our method avoids hard neighborhood assignments. Instead, we employ a cross-attention mechanism to softly infer correspondences between abstracted point sets across consecutive frames. This mechanism learns a dense, probabilistic relationship over all target points for each source point, allowing the network to attend to the most relevant regions regardless of spatial proximity. Given the source and target point sets $\mathcal{S}^{\mathcal{P}}_{\text{sa}} = \{\mathbf{p}_i, \mathbf{f}_i\}$ and $\mathcal{S}^{\mathcal{Q}}_{\text{sa}} = \{\mathbf{q}_j, \mathbf{g}_j\}$, where $\mathbf{f}_i$ and $\mathbf{g}_j$ are the abstracted feature vectors, we first compute positional encodings for each coordinate:

$$\gamma(\mathbf{p}_i), \gamma(\mathbf{q}_j) \in \mathbb{R}^{d_{pe}}, \text{for each } i,j. \tag{5}$$

To model relational similarity between point pairs $(\mathbf{p}_i, \mathbf{q}_j)$, we compute an affinity matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$ using dot-product attention between the positional encodings of source and target points:

$$\mathbf{A}_{ij} = \langle \gamma(\mathbf{p}_i), \gamma(\mathbf{q}_j) \rangle, \tag{6}$$

where n and m are the number of sampled points in the source and target clouds, respectively. We apply a row-wise softmax to normalize $\mathbf{A}$ into attention weights:

$$\mathbf{W}_{ij} = \frac{\exp(\mathbf{A}_{ij})}{\sum_{j'=1}^{m} \exp(\mathbf{A}_{ij'})}. \tag{7}$$

This results in a learned, differentiable weighting over all target points for each source point. Let the target feature matrix be defined as $\mathbf{G} = [\mathbf{g}_1, ..., \mathbf{g}_m]^\top$, where each $\mathbf{g}_i$ denotes an abstracted feature vector at target point $\mathbf{q}_i$. We then apply the attention weights $\mathbf{W} \in \mathbb{R}^{n\times m}$ to aggregate a fused context vector $\mathbf{c}_i$ for each source point:

$$\mathbf{c}_i = \sum_{j=1}^{M} \mathbf{W}_{ij} \cdot \mathbf{g}_j, \text{ for} i = 1, ..., n. \tag{8}$$

The resulting fused feature set $\mathcal{C} = \{\mathbf{c}_i\}_{i=1}^{n}$ encodes motion-aware context that reflects both the semantic similarity and spatial proximity in a soft, learnable manner.

To further enhance the motion representation capacity, we incorporate a motion residual term, defined as the difference between the contextually aggregated target feature and the original source feature: $\Delta\mathbf{r}_i = \mathbf{c}_i - \mathbf{f}_i$. This residual encodes the relative motion magnitude and direction for each point in a learnable manner. To explicitly inform the network about the strength of motion, we also compute the L2 norm of the residual, $\|\Delta\mathbf{r}_i\|_2$, which is appended as an additional scalar feature.

Consequently, the final IRFE representation for each point is reformulated as:

$$\mathcal{S}^{\text{irfe}}_i = \left\{\mathbf{f}_i, \mathbf{c}_i, \mathbf{g}_i, \Delta\mathbf{r}_i, \|\Delta\mathbf{r}_i\|_2\right\} \tag{9}$$

where $\mathcal{S}^{\text{irfe}}_i$ signifies the sequence of the flattened feature set with dimensions $[n_{\text{key}}, 4\times c' + 1]$.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

This refinement of IRFE complements the positional and contextual mechanisms by adding a direct encoding of motion change, which is critical for capturing dynamic behavior in 3D space.

3.5 | 3-D transformer-based pose prediction

Here, we propose a encoder-decoder transformer block for self-attentioning flow embedding set. We adopt and extend the framework of the 3D Transformer [25] to perform the LO task.

We feed $\mathcal{S}_i^{\text{irfe}}, \mathbf{p}_i$, and $\mathbf{q}_i$ into the Transformer layer. Motivated by the concept of learnable *classification tokens* [52, 53], our Transformer network follows a similar approach. Specifically, we enable the passage of sampled keypoint tokens $\mathbf{p}_i$ and $\mathbf{q}_i$ through the Transformer layers. As a result, the encoder effectively processes $\mathbf{p}_i$ to capture spatial features, while the decoder handles $\mathbf{q}_i$ to focus on sequential dependencies, preserving the geometric characteristics of consecutive LiDAR frames. Our implementation utilizes the encoder to handle $\mathbf{p}_i$ and generates queries for the decoder from $\mathbf{q}_i$. This design choice optimizes the model's ability to accurately predict pose using LiDAR data.

Finally, we employ an output multilayer perceptron using conventional Conv1d and Linear layers to predict the pose using a dual-quaternion representation.

3.6 | Training losses

The objective of LO is to minimize the error between (1) and ground truth odometry. Here, we can denote the predictive transformation as $\mathbf{T}$, and ground truth transformation as $\mathbf{T}^{\text{gt}}$, respectively. There is a intuitive loss function using geometric approach [54] such as Euler angles, SO(3) rotation matrices or quaternions.

In this study, as [45, 50] proposed, we adopt dual-quaternions $\boldsymbol{\sigma} = \mathbf{p}_{\text{r}} + \epsilon \mathbf{q}_{\text{d}}$ for representing rigid transformation where $\mathbf{p}_{\text{r}}, \mathbf{q}_{\text{d}}$ are real part, dual part, respectively. The details of dual-quaternions can be found here [49]. Therefore, we implement the dual loss as follows:

$$L_{\text{d}} = \mathbb{E}\left[\left\|\mathbf{q}_{\text{d}}^{\text{pred}} - \mathbf{q}_{\text{d}}^{\text{gt}}\right\|_2\right], \tag{10}$$

where $\mathbf{q}_d^{\text{pred}}, \mathbf{q}_d^{\text{gt}}$ are denoted as predicted transformation, ground-truth transformation represented by dual part in dual-quaternions form, respectively. Also, the real loss is given by

$$L_r = \mathbb{E}\left[\left\|\frac{\mathbf{p}_r^{\text{pred}}}{\left\|\mathbf{p}_r^{\text{pred}}\right\|} - \mathbf{p}_r^{\text{gt}}\right\|_2\right], \tag{11}$$

where $\mathbf{p}_r^{\text{pred}}, \mathbf{p}_r^{\text{gt}}$ are denoted as predicted rotation, ground-truth rotation represented by real part in dual-quaternions form, respectively.

4 | IMPLEMENTATION DETAILS

In our implementation, raw points are consecutively sampled using FPS methods, and aggregated into feature abstracted sets $\mathcal{S}_{\text{sa},i}^{\mathcal{P}}$ and $\mathcal{S}_{\text{sa},i}^{\mathcal{Q}}$.

For the flow embedding module, we deviate from conventional approaches such as KNN-based neighborhood search. Instead, we utilize IRFE, which operates entirely without hard correspondence selection. To relate abstracted point pairs between source and target sets, we use Fourier-based positional encodings $\gamma(\mathbf{p}_i)$ and $\gamma(\mathbf{q}_j)$ to compute a similarity matrix via dot-product attention. This attention is used to aggregate target features $\mathbf{g}_j$ into a fused context feature $\mathbf{c}_i$ for each source point.

To explicitly model motion, we incorporate a residual term $\Delta\mathbf{r}_i = \mathbf{c}_i - \mathbf{f}_i$, along with its L2 norm $\|\Delta\mathbf{r}_i\|_2$, which encodes motion strength. The final IRFE representation for each source point is constructed by concatenating the original source feature $\mathbf{f}_i$, the fused context feature $\mathbf{c}_i$, the original target feature $\mathbf{g}_i$, the motion residual $\Delta\mathbf{r}_i$, and its magnitude.

$$\mathcal{S}_i^{irfe} = \left\{\mathbf{f}_i, \mathbf{c}_i, \mathbf{g}_i, \Delta\mathbf{r}_i, \|\Delta\mathbf{r}_i\|_2\right\}.$$

Each of the four vector components has dimensionality c', and the norm term is a scalar. Thus, the resulting IRFE feature matrix has shape $[n_{\text{key}}, 4 \times c' + 1]$. This compact yet expressive representation is then passed to the transformer encoder as the motion-aware feature input, while the positional encodings are separately used for computing attention weights. Before feeding the IRFE representation into the transformer encoder, we apply a lightweight point-wise encoding module to compress the feature dimension to match the transformer's input requirements. Specifically, since the transformer expects input vectors of fixed dimension $d_{\text{model}} = 256$, we use a 1D convolutional block to compress the $4c' + 1$ dimensional IRFE features into 256-dimensional embeddings:

$$\texttt{Conv1d}(4c' + 1, 256) \rightarrow \texttt{ReLU} \rightarrow \texttt{Conv1d}(256, 256)$$

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

This encoder serves two purposes: (i) it ensures compatibility with the transformer architecture, and (ii) it enhances the semantic compactness of the fused motion representation through nonlinear feature transformation. As a result, the transformer receives well-normalized, high-level features that are more conducive to effective spatio-temporal attention modeling.

Lastly, our transformer block is composed of positional encoding, encoder, decoder, and query embedding modules. An overview of the network parameters is shown in Table 1.

During the learning process, random translations and rotation noise are applied to the input point sets for data augmentation. In addition, a key improvement in generalization and accuracy was achieved by increasing the batch size from 5 to 10. This adjustment significantly enhanced the model's robustness across varying sequences, leading to improved performance metrics.

TABLE 1 Hyperparameters for KITTI odometry dataset.

Module	Parameter for KITTI	Value
FE	n_{key}	1024
	MLP_{raw}	$[16, 16, 32] \times 2$
	SA radii	[0.5, 1.0]
FEPE	L	64
	PE method	Fourier
	PE normalization	True
	Gauss scale	1.0
Embedding	IRFE output dim.	$4 \times 64 + 1$
	Embedding type	Conv1d (2-layer)
	Embedding dim.	256
Transformer	Enc. layer	3
	Enc. dim.	256
	Enc. head number	4
	Enc. ffn dim.	16
	Enc. dropout	0.1
	Enc. activation	Relu
	Dec. layer	3
	Dec. dim.	256
	Dec. head number	2
	Dec. ffn dim.	16
	Dec. dropout	0.1
	Num. of queries	1024
Head	MLP_{pn}	[256, 512, 1024]
	MLP_{fc}	[1024, 512, 256, 8]

Abbreviations: FE, feature extraction; FEPE, flow embedding and positional encoding.

5 | EXPERIMENTS

5.1 | KITTI odometry

We validate the proposed end-to-end LO method with KITTI odometry dataset [19]. There are 22 LiDAR sequences and corresponding RGB/gray images in the KITTI odometry dataset. It provides ground-truth poses derived from IMU/GPS fusion algorithms for sequences 00-10. Also, ground-truth poses are not provided by the remaining sequences, which are for benchmark testing. There are different types of road environments in this dataset, as well as pedestrians, cyclists, and different types of vehicles. Data-collection vehicle drives from 0 km/h to 90 km/h in different areas. Therefore, we can evaluate whether the trained model can handle noisy real-world data with a large-scale point cloud.

5.2 | CARLA simulation

CARLA, an open-source simulator for autonomous driving research, provides a high-fidelity virtual environment with highly flexible sensor configurations. It includes a variety of assets that replicate small-scale urban environments. Built on Unreal Engine 5 (UE5), CARLA features dynamic elements such as non-player characters (NPCs) like pedestrians, as well as other city objects, including vehicles, bicycles, and more. In this research, we collected datasets from Towns 01, 02, 03, 05, and 10, utilizing vehicle pose data and LiDAR sensors. The LiDAR was a 64-channel sensor with specifications equivalent to those of the KITTI odometry benchmark, enabling effective bridging of the sim-to-real gap.

5.3 | Digital twin

We developed a digital twin environment for the Electronics and Telecommunications Research Institute (ETRI) in South Korea. This digital twin enables the collection of simulated LiDAR datasets along with corresponding vehicle pose data. Similar to CARLA, it is built on UE5, with its structural similarities illustrated in Figure 3. The LiDAR is a 64-channel sensor with specifications equivalent to those of the KITTI odometry benchmark, facilitating the bridging of the sim-to-real gap.

However, a notable limitation was encountered during the development of the digital twin environment: while buildings and terrain objects are accurately represented, plants and trees are highly simplified. This is due

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

to the significant challenges involved in replicating them with high fidelity, as well as the substantial computational resources required.

5.4 | Evaluation

All methods are tested on a machine equipped with an Intel Xeon(R) Gold 6244 CPU@3.60GHzx16, memory of 384.0GB, and a 10,752 CUDA cores with 48GB of graphics memory. The implementation is written in PyTorch.

FIGURE 3 Comparison between the real-world environment and its digital twin at ETRI. The top image shows the real-world satellite view of the ETRI campus, capturing the actual layout and infrastructure. The bottom image illustrates the corresponding simulation environment, created as a digital twin to replicate the real-world setup.

5.4.1 | KITTI odometry

As our training dataset consists of sequences 00-06, our method is compared against other competitive methods using sequences 07-10 from the KITTI odometry dataset. We compare our method with the classic method [14, 16]—that is, to demonstrate the classic methods, point-to-point ICP, point-to-plane ICP, and G-ICP are implemented based on the Open3D library. Additionally, we compare the proposed method against other CNN-based odometry methods [43, 45], and we discover that our method achieves competitive performance. We train a model with 00-06 sequences and evaluate with 07-10. The results were evaluated using the *KITTI Odometry Devkit*, which provides the average relative translation (t_{rmse}) and rotation errors (r_{rmse}) as part of its output metrics.

The evaluation results, which include a comparison with classic and learning-based methods, are presented in Table 2. We have visualized the generated trajectories on KITTI odometry sequences 07 through 10 as depicted in Figure 4. This visualization also includes each sequence's translation and rotation error segregated by speed and path length segments in Figure 5.

While our method underperforms classical frame-to-map methods like G-ICP in some sequences, it demonstrates better performance compared to learning-based frame-to-frame method [45] in terms of translation and

TABLE 2 Comparison with the state-of-the-arts on KITTI odometry dataset.

		7		8		9		10		Testing avg.	
Method	**Seq.**	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}	t_{rel}	r_{rel}
Classic	ICP-PO2PO	5.17	3.35	10.04	4.93	6.93	2.89	8.91	4.74	7.76	3.98
	ICP-PO2PL	1.55	1.42	4.42	2.14	3.95	1.71	6.13	2.60	4.01	1.97
	GICP [14]	**0.62**	**0.56**	**1.75**	**0.69**	**1.67**	**0.61**	**1.58**	**0.66**	**1.64**	**0.66**
	LOAM [6]	2.98	1.55	4.89	2.04	6.04	1.79	3.65	1.55	4.39	1.73
Learning-based	Nerf-LOAM [55]	1.00	-	-	-	1.63	-	2.08	-	1.69	-
	PIN-LO [56]	**0.34**	-	**0.80**	-	**0.54**	-	**0.50**	-	**0.55**	-
	PWCLO-Net [57]	0.60	0.44	1.26	0.55	0.79	0.35	1.69	0.62	1.09	0.49
	SelfVoxeLO [58]	3.09	1.81	3.16	1.14	3.01	1.14	3.48	1.11	3.19	1.30
	Efficient-LO [22]	0.46	**0.38**	1.14	**0.41**	0.78	**0.33**	0.80	**0.46**	0.80	**0.40**
	DEEPCLR [45]	4.82	3.47	6.37	2.27	3.87	1.63	8.10	2.77	5.79	2.54
	Proposed	2.05	1.50	2.88	1.39	2.53	1.52	4.36	2.02	2.91	1.52

Note: We list the classic methods and learning-based odometry methods. - indicates the number is not reported. The best results are highlighted in **bold**.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

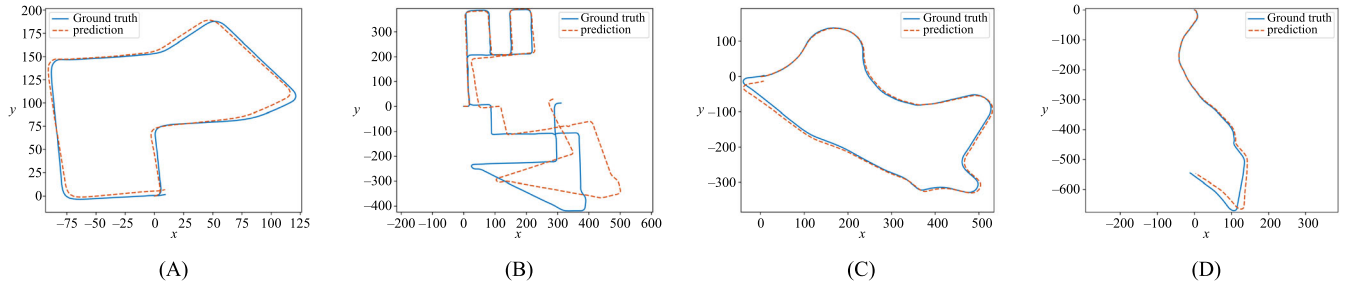


FIGURE 4 Qualitative comparison of predicted and ground truth trajectories on KITTI odometry sequences (A) 07, (B) 08, (C) 09, and (D) 10. These sequences were excluded from the training set and serve as unseen environments to evaluate the generalization capability of **ELiOT**: (A).

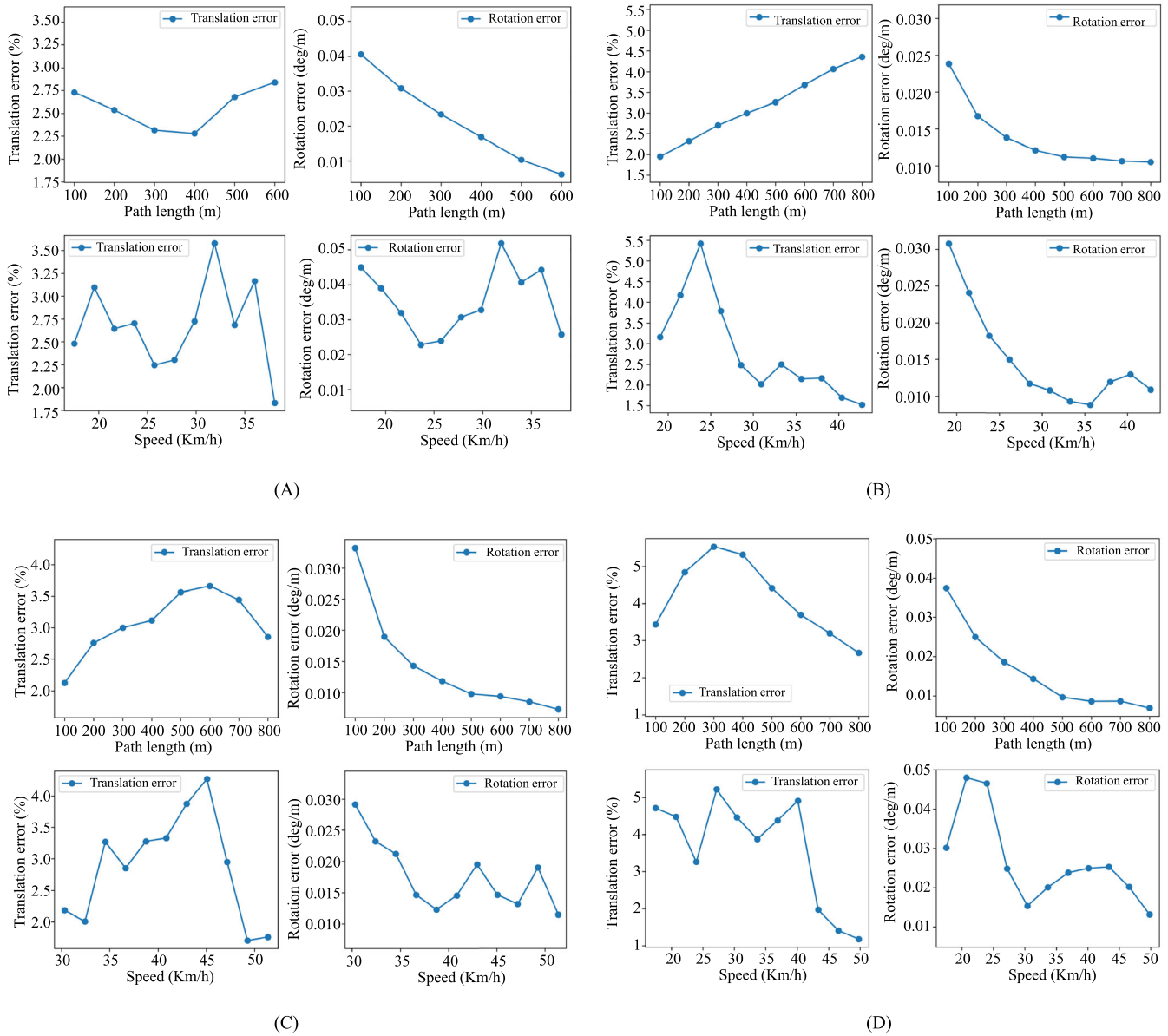


FIGURE 5 Illustration of translation and rotation errors on KITTI odometry sequences (A) 07, (B) 08, (C) 09, and (D) 10. Each subfigure demonstrates the respective errors for a particular sequence, showcasing the performance of our proposed method according to segment progress and up to speed.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

rotation accuracy. This suggests that our implicit motion representation improves robustness in dynamic or sparsely structured environments. Nevertheless, the lack of global map alignment in our approach limits its long-term consistency, which remains a direction for future improvement through map-based integration or optimization.

While comparing our method with traditional techniques and referencing learning-based studies, we acknowledge the potential for further enhancements. Specifically, we understand that our methodology requires more development to achieve the performance benchmarks set by 3D to 2D projection-based techniques and conventional approaches. As we continue our research, we aim to address these areas of improvement and strive for better results in future iterations.

One possible reason for the lack of generalization could be that the KITTI dataset used for training only covered limited sequences from 0 to 6. As a result, our model may not have fully generalized to other unseen sequences. To improve generalization, we plan to incorporate training on more diverse datasets and explore domain adaptation strategies for robust performance across unseen environments.

5.4.2 | CARLA simulation

As our training dataset consists of town 01, 02, 04 and validation dataset consists of town 03 and 10. We evaluate the performance of our method using three key metrics: *Absolute Position Error (APE)*, *Relative Position Error (RPE)*, and *Relative Rotational Error (RRE)*. These metrics quantify the accuracy of our learning-based approach to LiDAR odometry.

Table 3 presents the evaluation results for *APE*, *RPE*, and *RRE* across three different validation environments: Town 03, Town 10, and ETRI. Since the trajectory segments in the simulation dataset are shorter than those in the real-world KITTI dataset, we evaluate using a different approach.

First, we define *APE* as

$$\mathrm{APE}_i = \|\mathbf{p}_{\mathrm{est},i} - \mathbf{p}_{\mathrm{gt},i}\|_2 \tag{12}$$

where $\mathbf{p}_{\mathrm{est},i} = (x_{\mathrm{est},i}, y_{\mathrm{est},i}, z_{\mathrm{est},i})$ and $\mathbf{p}_{\mathrm{gt},i} = (x_{\mathrm{gt},i}, y_{\mathrm{gt},i}, z_{\mathrm{gt},i})$ represent the estimated and ground truth positions at frame i, respectively, in the 3D coordinate space.

The *RPE* quantifies the translational drift between consecutive frames. It is calculated as:

$$\mathrm{RPE}_i = \|\Delta\mathbf{p}_{\mathrm{est},i} - \Delta\mathbf{p}_{\mathrm{gt},i}\|_2 \tag{13}$$

where $\Delta\mathbf{p}_{\mathrm{est},i} = \mathbf{p}_{\mathrm{est},i+1} - \mathbf{p}_{\mathrm{est},i}$ and $\Delta\mathbf{p}_{\mathrm{gt},i} = \mathbf{p}_{\mathrm{gt},i+1} - \mathbf{p}_{\mathrm{gt},i}$ are the relative translations for the estimate and ground truth, respectively. The *RPE* remains small throughout the trajectory, demonstrating the accuracy of the local motion estimation. This indicates the effectiveness of the learning-based approach in capturing relative motion.

The *RRE* measures the angular difference between the estimated and ground truth relative rotations. It is defined using the geodesic distance:

$$\mathrm{RRE}_i = \arccos\left(\frac{\mathrm{trace}(\mathbf{R}_{\mathrm{gt},i}^T \mathbf{R}_{\mathrm{est},i}) - 1}{2}\right) \tag{14}$$

where $\mathbf{R}_{\mathrm{est},i}$ and $\mathbf{R}_{\mathrm{gt},i}$ are the relative rotation matrices for the estimate and ground truth at frame i, and $\mathrm{trace}(\cdot)$ denotes the sum of the diagonal elements of a matrix.

The *RRE* is also small across the trajectory, confirming the accuracy of rotational estimates. The pose estimation results in CARLA are depicted in Figure 6.

5.4.3 | Digital twin

We evaluate ELiOT to determine its suitability for direct application in the digital twin environment of ETRI as depicted in Figure 7. As shown in Table 3, the *RPE* and *RRE* demonstrate high accuracy, whereas the *APE* is relatively large due to accumulated rotational error. This

TABLE 3 Evaluation of *APE*, *RPE*, and *RRE* across different test environments.

	APE		*RPE*		*RRE*	
Town	**Mean**	**Std dev.**	**Mean**	**Std dev.**	**Mean**	**Std dev.**
03	19.95	12.90	0.11	0.10	0.22	0.42
10	19.88	9.72	0.12	0.09	0.26	0.33
ETRI	43.63	23.74	0.33	0.12	0.27	0.46

Note: The units for *APE* and *RPE* include the mean translational error in meters and its standard deviation (Std dev.), while *RRE* represents the mean rotational error in degrees and its standard deviation across the trajectories.
Abbreviations: *APE*, absolute position error; *RPE*, relative position error; *RRE*, relative rotational error.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

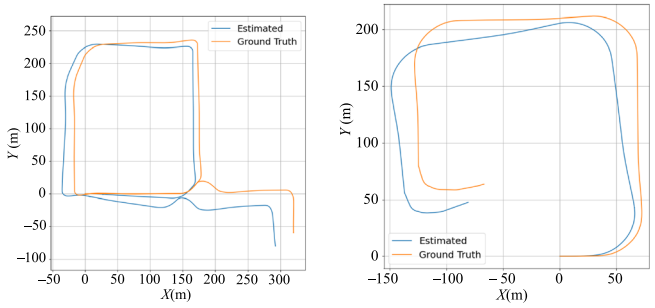


FIGURE 6 Pose estimation results of **ELiOT** compared to ground truth in CARLA simulation environments. Towns 03 (A) and 10 (B) were not seen during training, serving as test environments for evaluating generalization. The qualitative comparison shows that ELiOT is able to produce reasonable trajectory estimates even in structurally unseen scenes.

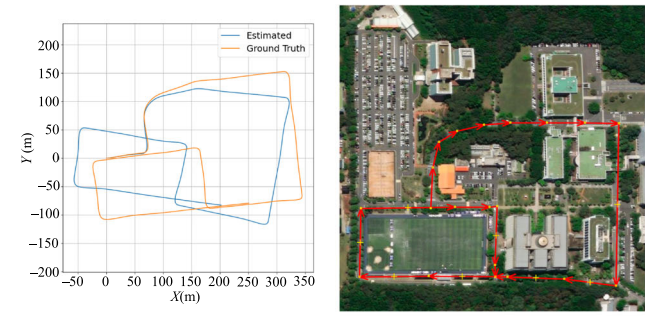
suggests that our learning-based LO can be effectively applied to sim-to-real-to-sim applications, provided that the dataset quality is curated, and a diverse dataset is collected from both the simulator and real-world environments.

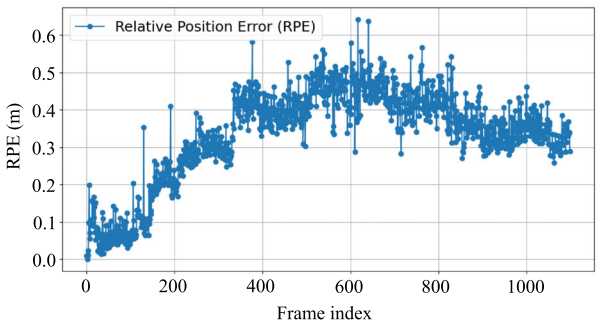
5.5 | Ablation study

We conducted an ablation study to compare our proposed IRFE against two baselines: a conventional KNN-based flow embedding and a hybrid fusion of IRFE with KNN features. As summarized in Table 4, IRFE consistently outperforms other methods across sequences 07–10, yielding the lowest translation error, rotation error, and ATE in nearly all cases.

For example, the proposed IRFE model achieved the lowest average translational error of 2.96%, compared to 5.58% for KNN-only and 4.31% for the fusion variant. Similarly, the average rotational error and RPE metrics were also more favorable in the IRFE setting. Notably, the hybrid model (*Proposed + KNN*) showed moderate improvement over the KNN baseline, but still underperformed compared to IRFE alone.

These results suggest that the soft, attention-based correspondence modeling in IRFE enables more accurate and generalizable motion representation. While KNN-based features provide geometric consistency, their explicit local correspondence may introduce redundancy or interfere with the contextual abstraction learned by IRFE. This likely explains why the fusion of both sources does not yield further improvements and, in some cases, slightly harms performance due to representational misalignment.





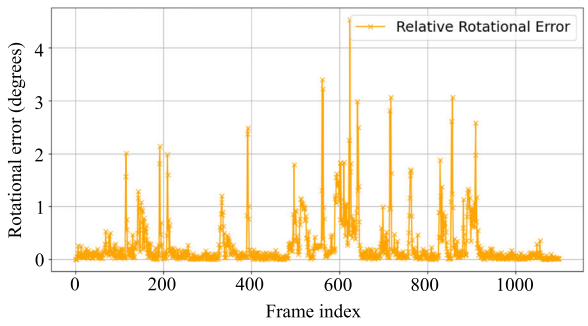


FIGURE 7 (A) The estimated trajectory (black) and ground truth trajectory (orange) are visualized in the digital twin environment of ETRI. (B) Satellite imagery of the ETRI environment overlaid with the trajectory, illustrating the real-world route taken in red. (C) The frame-wise *RPE* in meters, showing the local translational accuracy of the estimated trajectory compared to the ground truth. (D) The frame-wise *RRE* in degrees, showing the local rotational accuracy of the estimated trajectory compared to the ground truth.

Overall, the ablation validates the design philosophy of IRFE: By discarding rigid neighbor search and focusing on positional encoding, context attention, and motion residual modeling, it achieves robust motion understanding without reliance on hand-crafted local structures.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 4 Ablation study on flow embedding methods.

Method	Seq.	Trans. err. (%) ↓	Rot. err. (deg/100 m) ↓	ATE (m) ↓	RPE (m) ↓	RPE (deg) ↓
Proposed (IRFE)	07	**2.056**	**1.503**	**8.184**	**0.013**	0.155
	08	**2.881**	**1.399**	**59.788**	**0.024**	**0.182**
	09	**2.531**	**1.524**	**16.152**	**0.016**	**0.165**
	10	**4.363**	**2.023**	**17.648**	**0.016**	**0.214**
	Avg.	**2.96**	**1.61**	**25.94**	**0.017**	**0.179**
KNN only	07	4.822	2.696	18.432	0.027	0.159
	08	6.865	3.547	91.050	0.035	0.182
	09	5.875	2.746	72.606	0.028	0.164
	10	6.770	2.760	36.918	0.015	0.214
	Avg.	5.58	2.94	54.75	0.026	0.180
Proposed + KNN	07	3.425	1.978	16.323	0.013	**0.155**
	08	3.620	1.480	89.584	0.023	0.181
	09	3.127	1.638	36.115	0.015	0.167
	10	7.049	2.809	37.923	0.016	0.215
	Avg.	4.31	1.98	44.99	0.017	0.180

Note: IRFE shows consistently superior performance across sequences. The boldface values indicate the best performance and lowest error.

6 | CONCLUSION

In this work, we presented **ELiOT**, a novel architecture that seamlessly integrates point feature abstraction with a motion flow self-attentioning 3D transformer framework.

Through comprehensive evaluations, **ELiOT** demonstrates competitive performance in comparison to both classical and recent learning-based LiDAR odometry methods. In particular, the introduction of the IRFE enables more flexible and robust motion understanding, effectively reducing reliance on rigid geometric matching strategies.

This highlights ELiOT's potential as a generalizable and efficient alternative for LiDAR-based pose estimation, even in the absence of voxelization or handcrafted priors.

To address the generalization issue, we have harnessed the power of real-world simulation and digital twin environments by evaluating our trained models on datasets collected from CARLA and the ETRI digital twin. These datasets provide diverse and challenging scenarios for validating the robustness of our method.

Our results showed noticeable performance degradation when models trained exclusively on KITTI were deployed in these new environments, reinforcing the necessity of exposure to a wider range of geometric and semantic variability during training.

To mitigate this limitation, we plan to use digital twin technology not only for evaluation but also as a scalable means of generating high-fidelity training data with controlled diversity. By varying field-of-view (FoV), resolution, mounting positions, and scene configurations, we can simulate a broader distribution of real-world scenarios, which we believe will enhance the robustness and adaptability of ELiOT across domains.

Furthermore, we plan to expand our training dataset with more diverse sequences. Additionally, we aim to explore the integration of novel techniques and architectural enhancements to further improve **ELiOT**'s capabilities and push the boundaries of LiDAR odometry accuracy and efficiency.

ACKNOWLEDGMENTS

We sincerely thank Jinwoo Kim of Daejeon University for his generous contributions to our research through the provision of the CARLA and ETRI digital twin datasets. He was responsible for collecting the data in the CARLA and ETRI environments. His effort has greatly supported and enhanced our study.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

ORCID

Daegyu Lee https://orcid.org/0000-0002-9336-5759

REFERENCES

1. T. Kliestik, R. Dragomir, A. V. Băluță, I. Grecu, P. Durana, O. L. Karabolevski, P. Kral, R. Balica, P. Suler, O. V. Buşu, and M. Bugaj, *Enterprise generative artificial intelligence*

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

technologies, Internet of Things and blockchain-based fintech management, and digital twin industrial metaverse in the cognitive algorithmic economy, Oecon. Copern. **15** (2024), no. 4, 1183–1221.

2. G. Lazaroiu, T. Gedeon, E. Rogalska, K. Valaskova, M. Nagy, H. Musa, K. Zvarikova, M. Poliak, J. Horak, R. I. Cretoiu, and T. Krulicky, *Digital twin-based cyber-physical manufacturing systems, extended reality metaverse enterprise and production management algorithms, and internet of things financial and labor market technologies in generative artificial intelligence economics*, Oecon. Copern. **15** (2024), no. 3, 837–870.
3. G. Lazaroiu, T. Gedeon, K. Valaskova, J. Vrbka, P. Šuleř, K. Zvarikova, K. Kramarova, Z. Rowland, V. Stehel, L. Gajanova, and J. Horák, *Cognitive digital twin-based internet of robotic things, multi-sensory extended reality and simulation modeling technologies, and generative artificial intelligence and cyber-physical manufacturing systems in the immersive industrial metaverse*, Equilib. Quart. J. Econ. Economic Policy **19** (2024), no. 3, 719–748.
4. S. U. Kamat and K. Rasane, *A survey on autonomous navigation techniques*, (Second International Conference on Advances in Electronics, Computers and Communications, Bangalore, India), 2018, pp. 1–6.
5. R. Sun, Y. Yang, K.-W. Chiang, T.-T. Duong, K.-Y. Lin, and G.-J. Tsai, *Robust IMU/GPS/VO integration for vehicle navigation in GNSS degraded urban areas*, IEEE Sens. J. **20** (2020), no. 17, 10110–10122.
6. J. Zhang and S. Singh, *LOAM: lidar odometry and mapping in real-time*, in Robotics: Science and Systems, Vol. **2**. Berkeley, CA, 2014, pp. 1–9.
7. T. Shan and B. Englot, *LeGO-LOAM: lightweight and ground-optimized lidar odometry and mapping on variable terrain*, (IEEE/RSJ international conference on intelligent robots and systems, Madrid, Spain), 2018, pp. 4758–4765.
8. T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, *LIO-SAM: Tightly-coupled lidar inertial odometry via smoothing and mapping*, (IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, USA), 2020, pp. 5135–5142.
9. W. Xu and F. Zhang, *FAST-LIO: a fast, robust lidar-inertial odometry package by tightly-coupled iterated Kalman filter*, IEEE Robot. Autom. Lett. **6** (2021), no. 2, 3317–3324.
10. W. Lu, Y. Zhou, G. Wan, S. Hou, and S. Song, *L3-Net: Towards learning based lidar localization for autonomous driving*, (*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, USA) 2019, pp. 6389–6398.
11. H. Yin, L. Tang, X. Ding, Y. Wang, and R. Xiong, *LocNet: global localization in 3D point clouds for mobile vehicles*, (IEEE Intelligent Vehicles Symposium, Changshu, China), 2018, pp. 728–733.
12. I. A. Barsan, S. Wang, A. Pokrovsky, and R. Urtasun, *Learning to localize using a lidar intensity map*, arXiv preprint, 2020. DOI 10.48550/arXiv.2012.10902
13. P. J. Besl and N. D. McKay, *Method for registration of 3-D shapes*, Sensor Fusion IV: Control Paradigms and Data Structures, Vol. **1611**. SPIE, 1992, pp. 586–606.
14. A. Segal, D. Haehnel, and S. Thrun, *Generalized-ICP*, Robotics: Science And Systems, Vol. **2**. Seattle, WA, 2009, pp. 435.
15. P. Biber and W. Straßer, *The normal distributions transform: a new approach to laser scan matching*, (Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003), Las Vegas, NV, USA), 2003, pp. 2743–2748.
16. A. Censi, *An ICP variant using a point-to-line metric*, (IEEE International Conference on Robotics and Automation, Pasadena, CA, USA), 2008, pp. 19–25.
17. A. Das and S. L. Waslander, *Scan registration with multi-scale k-means normal distributions transform*, (IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal), 2012, pp. 2705–2710.
18. A. Das and S. L. Waslander, *Scan registration using segmented region growing NDT*, Int. J. Robot. Res. **33** (2014), no. 13, 1645–1663.
19. A. Geiger, P. Lenz, and R. Urtasun, *Are we ready for autonomous driving? The KITTI vision benchmark suite*, (IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA), 2012, pp. 3354–3361.
20. F. Xue, X. Wang, J. Wang, and H. Zha, *Deep visual odometry with adaptive memory*, IEEE Trans. Pattern Anal. Machine Intell. **44** (2020), no. 2, 940–954.
21. Y. Almalioglu, M. Turan, N. Trigoni, and A. Markham, *Deep learning-based robust positioning for all-weather autonomous driving*, Nature Mach. Intell. **4** (2022), no. 9, 749–760.
22. G. Wang, X. Wu, S. Jiang, Z. Liu, and H. Wang, *Efficient 3D deep lidar odometry*, IEEE Trans. Pattern Anal. Machine Intell. **45** (2022), no. 5, 5749–5765.
23. J. Liu, G. Wang, C. Jiang, Z. Liu, and H. Wang, *TransLO: a window-based masked point transformer framework for large-scale lidar odometry*, Proc. AAAI Conf. Artif. Intell. **37** (2023), 1683–1691.
24. J. Mao, Y. Xue, M. Niu, H. Bai, J. Feng, X. Liang, H. Xu, and C. Xu, *Voxel transformer for 3D object detection*, (Proceedings of the IEEE/CVF international conference on computer vision, Montreal, Canada), 2021, pp. 3164–3173.
25. I. Misra, R. Girdhar, and A. Joulin, *An end-to-end transformer model for 3D object detection*, (Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, Canada), 2021, pp. 2906–2917.
26. A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, *Pointpillars: fast encoders for object detection from point clouds*, (Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Long Beach, CA, USA) 2019, pp. 12697–12705.
27. B. Yang, W. Luo, and R. Urtasun, *PIXOR: real-time 3D object detection from point clouds*, (Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), Salt Lake City, UT, USA), 2018. DOI 10.1109/CVPR.2018.00798
28. C. R. Qi, H. Su, K. Mo, and L. J. Guibas, *Pointnet: deep learning on point sets for 3D classification and segmentation*, (Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA), 2017, pp. 652–660.
29. C. R. Qi, L. Yi, H. Su, and L. J. Guibas, *Pointnet++: deep hierarchical feature learning on point sets in a metric space*, Adv. Neural Inform. Process. Syst. **30** (2017), 5105–5114.
30. J. Deng, S. Shi, P. Li, W. Zhou, Y. Zhang, and H. Li, *Voxel R-CNN: towards high performance voxel-based 3D object detection*, Proc. AAAI Conf. Artif. Intell. **35** (2021), 1201–1209.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

31. Y. Zhou and O. Tuzel, *VoxelNet: end-to-end learning for point cloud based 3D object detection*, (Proceedings of the IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA), 2018, pp. 4490–4499.
32. S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, *PV-RCNN: point-voxel feature set abstraction for 3D object detection*, (Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA), 2020, pp. 10529–10538.
33. S. Shi, L. Jiang, J. Deng, Z. Wang, C. Guo, J. Shi, X. Wang, and H. Li, *PV-RCNN++: point-voxel feature set abstraction with local vector representation for 3d object detection*, Int. J. Comput. Vis. **131** (2023), 53–551.
34. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, *Attention is all you need*, Adv. Neural Inform. Process. Syst. **30** (2017).
35. N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, End-to-end object detection with transformers, European conference on computer vision. Springer, 2020, pp. 213–229.
36. J. Beal, E. Kim, E. Tzeng, D. H. Park, A. Zhai, and D. Kislyuk, *Toward transformer-based object detection*, arXiv preprint, 2020. DOI 10.48550/arXiv.2012.09958
37. E. Takeuchi and T. Tsubouchi, *A 3-D scan matching using improved 3-D normal distributions transform for mobile robotic mapping*,(IEEE/RSJ international conference on intelligent robots and systems, Beijing, China), 2006, pp. 3068–3073.
38. S. Wang, R. Clark, H. Wen, and N. Trigoni, *DeepVO: towards end-to-end visual odometry with deep recurrent convolutional neural networks*, (IEEE International Conference on Robotics and Automation, Singapore), 2017, pp. 2043–2050.
39. T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, *Unsupervised learning of depth and ego-motion from video*, (Proceedings of the IEEE conference on computer vision and pattern recognition, Honolulu, HI, USA), 2017, pp. 1851–1858.
40. Y. Cho, G. Kim, and A. Kim, *Unsupervised geometry-aware deep lidar odometry*, (IEEE International Conference on Robotics and Automation, Paris, France), 2020, pp. 2145–2152.
41. S. Han, M. Li, H. Tang, Y. Song, and G. Tong, *UVMO: deep unsupervised visual reconstruction-based multimodal-assisted odometry*, Pattern Recog. **153** (2024), 110573.
42. W. Lu, G. Wan, Y. Zhou, X. Fu, P. Yuan, and S. Song, *DeepVCP: an end-to-end deep neural network for point cloud registration*, (Proceedings of the IEEE/CVF international conference on computer vision, Seoul, Rep. of Korea), 2019. DOI 10.1109/ICCV.2019.00010
43. W. Wang, M. R. U. Saputra, P. Zhao, P. Gusmao, B. Yang, C. Chen, A. Markham, and N. Trigoni, *DeepPCO: end-to-end point cloud odometry through deep parallel neural network*, (IEEE/RSJ International Conference on Intelligent Robots and Systems, Macau, China), 2019, pp. 3248–3254.
44. Z. J. Yew and G. H. Lee, *3DFeat-Net: weakly supervised local 3D features for point cloud registration*, (Proceedings of the European conference on computer vision, Munich, Germany), 2018.
45. M. Horn, N. Engel, V. Belagiannis, M. Buchholz, and K. Dietmayer, *DeepCLR: correspondence-less architecture for deep end-to-end point cloud registration*, (IEEE 23rd International Conference on Intelligent Transportation Systems, Rhodes, Greece), 2020, pp. 1–7.
46. X. Liu, C. R. Qi, and L. J. Guibas, *FlowNet3D: learning scene flow in 3D point clouds*, (Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Long Beach, CA, USA), 2019. DOI 10.1109/CVPR.2019.00062
47. Y. Cho, G. Kim, S. Lee, and J.-H. Ryu, *OpenStreetMap-based lidar global localization in urban environment without a prior lidar map*, IEEE Robot. Autom. Lett. **7** (2022), no. 2, 4999–5006.
48. J. Liu, X. Lv, X. Gong, Y. Liang, and J. Hyyppä, *An unsupervised learning network for large-scale lidar point clouds registration*, IEEE Trans. Veh. Technol. **73** (2024), no. 11, 16187–16200.
49. B. Kenwright, *A beginners guide to dual-quaternions: what they are, how they work, and how to use them for 3D character hierarchies*, (International Conference in Central Europe on Computer Graphics and Visualization, Plzen, Czech Rep.), 2012.
50. N. Schneider, F. Piewak, C. Stiller, and U. Franke, *RegNet: multimodal sensor registration using deep neural networks*, (IEEE intelligent vehicles symposium, Angeles, CA, USA), 2017, pp. 1803–1810.
51. B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, *NeRF: representing scenes as neural radiance fields for view synthesis*, Commun. ACM **65** (2021), no. 1, 99–106.
52. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *BERT: pre-training of deep bidirectional transformers for language understanding*, (Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA), 2019, pp. 4171–4186.
53. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, and J. Uszkoreit, *An image is worth* 16×16 *words: transformers for image recognition at scale*, arXiv preprint, 2020. DOI 10.48550/arXiv.2010.11929
54. A. Kendall and R. Cipolla, *Geometric loss functions for camera pose regression with deep learning*, (Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA), 2017, pp. 5974–5983.
55. J. Deng, Q. Wu, X. Chen, S. Xia, Z. Sun, G. Liu, W. Yu, and L. Pei, *NeRF-LOAM: neural implicit representation for large-scale incremental lidar odometry and mapping*, (Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France), 2023, pp. 8218–8227.
56. Y. Pan, X. Zhong, L. Wiesmann, T. Posewsky, J. Behley, and C. Stachniss, *PIN-SLAM: lidar SLAM using a point-based implicit neural representation for achieving global map consistency*, IEEE Trans. Robot. **40** (2024), 4045–4064.
57. G. Wang, X. Wu, Z. Liu, and H. Wang, *PWCLO-Net: deep lidar odometry in 3D point clouds using hierarchical embedding mask optimization*, (Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Nashville, TN, USA),2021, pp. 15910–15919.
58. Y. Xu, Z. Huang, K.-Y. Lin, X. Zhu, J. Shi, H. Bao, G. Zhang, and H. Li, *SelfVoxeLO: self-supervised lidar odometry with voxel-based deep neural networks*, (Conference on Robot Learning. PMLR), 2021, pp. 115–125.

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

AUTHOR BIOGRAPHIES

Daegyu Lee received his B.S. degree in Automotive Engineering from Kookmin University, Seoul, South Korea, in 2018, and earned his M.S. and Ph.D. degrees in the Division of Future Vehicle and Electrical Engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2020 and 2024, respectively. He previously worked as a researcher at the Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea, and is currently an Assistant Professor in the Department of Advanced Defense Technology and Industries at Jeonbuk National University, Jeonju, South Korea, since October 2025. Daegyu's research interests lie in autonomous systems and localization, focusing on robust and learning-based approaches.

Hyunwoo Nam received his BS degree in Automobile and IT convergence, Kookmin University, Seoul, Republic of Korea, in 2022, and his M.S. degree in Robotics Program from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea, in 2024. He is currently a researcher at Hyundai Motors, Hwaseong, Republic of Korea.

Insung Jang received the BS and MS in computer engineering from Pusan National University, Busan, Republic of Korea, in 1999 and 2001, respectively. He has been with the Electronics and Telecommunications Research Institute (ETRI) since 2001 and currently serves as a principal researcher in Daejeon, Republic of Korea. His main research areas include platforms for Geo-Spatial, Geo-Sensor, LBS, Smart-City, Digital Twin, AAM, and Metaverse.

D. Hyunchul Shim received his BS and MS degrees in Mechanical Design and Production Engineering from Seoul National University, Seoul, Republic of Korea, in 1991 and 1993, respectively, and his PhD degree in Mechanical Engineering from the University of California at Berkeley, Berkeley, CA, USA, in 2000. He worked with the Hyundai Motor Company and Maxtor Corporation from 1993 to 1994 and from 2001 to 2005, respectively. In 2007, he joined the Department of Aerospace Engineering, KAIST, Daejeon, Republic of Korea, and is currently a tenured Professor with the Department of Electrical Engineering, and Adjunct Professor, Graduate School of AI, KAIST. His research interests include control systems, autonomous vehicles, and robotics.

How to cite this article: D. Lee, H. Nam, I. Jang, and D. H. Shim, *ELiOT: End-to-end LiDAR odometry with transformers harnessing real-world, simulated, and digital twin*, ETRI Journal **47** (2025), 815–829, DOI 10.4218/etrij.2025-0011

22337326, 2025, 5, Downloaded from https://onlinelibrary.wiley.com/doi/10.4218/etrij.2025-0011 by Electronics and Telecommunications, Wiley Online Library on [04/11/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License