Ray-Path-Aware Virtual Point Removal on 2D Layer-Wise Nearest Point Map

Dae Hyeon Jeon, Kyungdon Joo†, and Jae-Young Sim†

Ulsan National Institute of Science and Technology (UNIST), Republic of Korea
{dhjeon, kyungdon, jysim}@unist.ac.kr

Abstract. When LiDAR scans a scene with reflective materials, it often produces erroneous virtual points, which resemble real structures and degrade downstream 3D perception. Conventional methods rely on multi-stage pipelines that combine 3D surface reconstruction with heuristic symmetry detection, making them highly sensitive to surface noise, reflection irregularities, and accumulated estimation errors across stages. To address this, we propose an end-to-end framework that reformulates virtual point removal as ray-aligned geometric reasoning in the 2D equirectangular projection domain, eliminating the dependency on explicit surface modeling. Specifically, we introduce a Layer-Wise Nearest Point Map, which organizes multi-return LiDAR points by layer correspondence rather than fixed distance intervals, preserving geometric continuity between real and reflected surfaces. Building on this representation, we propose a Ray-Path-Aware Attention mechanism that constrains feature aggregation along physically plausible reflection trajectories derived from surface normals and incident ray directions. This physically guided attention enables the network to learn reflection-induced symmetry between real and reflected regions without stage-wise surface estimation. Extensive experiments on real-world LiDAR datasets demonstrate that our method achieves more accurate and stable virtual point removal than prior symmetry-based approaches.

Keywords: Virtual Point Removal · 3D LiDAR Point Cloud

1 Introduction

Light Detection and Ranging (LiDAR) sensors are widely used in robotics and computer vision applications, including autonomous driving and indoor or outdoor mapping, due to their ability to capture high-resolution 3D point clouds of the environment. In particular, multi-echo LiDAR sensors enhance the accuracy of 3D reconstruction by recording multiple reflections from a single laser pulse, which is beneficial in complex environments, such as forests and urban scenes. However, when LiDAR sensors encounter reflective materials, such as glass, marble, or polished metal, they may produce inaccurate 3D points. If a reflective

† Corresponding author.

surface lies along the path of a LiDAR sensor ray, a portion of the ray may undergo specular reflection and hit an object in another direction. In this situation, since the sensor does not detect this change in path, it may capture erroneous points that mimic real structures. These points are referred to as *virtual points* in prior work [24].

Virtual points exhibit similar shapes and densities to actual points, making them difficult to distinguish. More fundamentally, reflection-induced points remain geometrically consistent with the surrounding structure while not corresponding to physically valid surfaces, introducing ambiguity in LiDAR perception. Because they preserve plausible geometric structure, simple geometric filtering or density-based criteria are often insufficient for reliable separation. This causes problems in downstream tasks such as object detection, registration, semantic segmentation, localization, and mapping, where virtual points may be misinterpreted as physical objects. Therefore, identifying and removing virtual points requires reasoning beyond local geometric similarity.

Several approaches have been proposed to remove virtual points by exploiting the geometric symmetry of specular reflections [3, 11, 19, 24, 25]. These methods typically estimate reflective surfaces and then detect virtual points based on the assumed symmetry across them. However, this reliance on explicitly modeled reflective surfaces makes them sensitive to surface noise, contamination, or slight deviations in reflection direction caused by surface roughness or incident angles. Errors in surface estimation can propagate to the symmetry detection step, leading to accumulated inaccuracies and unstable performance across scenes.

To overcome these limitations, we propose a novel end-to-end framework that removes virtual points directly in the 2D equirectangular projection (ERP) domain without relying on explicit surface reconstruction (see Fig. 1). Instead of first estimating reflective surfaces and then performing symmetry comparison in separate stages, our framework reasons about reflection symmetry within a unified representation. This reformulates virtual point removal as ray-aligned geometric reasoning in the 2D domain, rather than as surface-first symmetry modeling in 3D. Specifically, we introduce a Layer-Wise Nearest Point Map, a variant of the multi-layer range map that organizes points based on layer correspondence rather than fixed distance intervals. This layer-centric formulation allows both real and reflected points along the same LiDAR ray to be projected onto different layers while preserving geometric continuity and symmetry cues that are crucial for identifying virtual reflections.

Building on this representation, we design a Ray-Path-Aware Attention mechanism that constrains feature aggregation along physically plausible reflection paths derived from surface normals and ray directions. By guiding the attention along these physically consistent trajectories, the network can implicitly learn reflection-induced symmetry between real and virtual regions without explicitly modeling reflective surfaces. Finally, the network outputs a 2D label map indicating whether each pixel corresponds to a virtual or real point, which is then unprojected back into 3D space through the inverse Layer-Wise Nearest Point Mapping process to reconstruct a physically consistent point cloud. Experiments

on real-world LiDAR datasets demonstrate that our framework achieves more accurate and stable virtual point removal than prior symmetry-based approaches, while significantly reducing computational overhead. These results indicate that reasoning about reflection geometry in the ERP domain provides a reliable alternative to conventional multi-stage pipelines. The main contributions are:

- We propose an end-to-end virtual point removal framework that operates in the ERP domain while implicitly modeling the symmetry between virtual and real points.
- We introduce a Layer-Wise Nearest Point Map that replaces interval-based projection with layer-centric sampling, preserving geometric continuity between real and reflected surfaces.
- We design a Ray-Path-Aware Attention mechanism that guides feature aggregation along physically plausible reflection trajectories, enabling the network to learn reflection-induced symmetry.
- Our framework achieves stable and accurate removal of reflection artifacts in real-world LiDAR scans, demonstrating significant improvement over existing 3D-2D coupled pipelines.

2 Related Work

Virtual Point Removal Methods. Existing methods for virtual point removal can be categorized into multi-view methods and single-scan multi-stage pipelines. Multi-view methods exploit cross-view geometric consistency and assume that virtual points violate correspondences across different scan positions [4, 5]. Although cross-view redundancy provides additional constraints, these methods require accurate registration and synchronized acquisitions. Since virtual points often distort correspondences, the multi-view consistency assumption may break down in practice and is not applicable in single-scan scenarios.

Single-scan methods instead rely on reflective surface estimation followed by symmetry reasoning [3, 11, 24, 25]. These multi-stage pipelines typically involve reflective region detection, surface estimation, and symmetry comparison. Despite the adoption of learning-based modules [11], the overall framework remains fundamentally dependent on accurate reflective surface estimation. In realistic environments, reflective surfaces may be incomplete or noisy, causing errors that propagate across stages and destabilize virtual point classification.

In contrast, our framework eliminates explicit reflective surface estimation and reformulates virtual point removal as ray-aligned reasoning in the ERP domain. By operating directly on per-ray structured representations, our method avoids stage-wise error propagation and enables single-stage virtual point classification.

Range Map Representations for LiDAR Processing. LiDAR point clouds are irregular and sparsely sampled in Euclidean space, making dense learning and neighborhood reasoning difficult. A common solution is to project 3D points into a 2D range image using ERP, where each pixel corresponds to a fixed ray

direction and stores attributes such as range or intensity [13, 20, 26, 29]. This ray-aligned representation enables efficient processing with 2D convolutional networks and has been widely adopted for LiDAR perception tasks, including segmentation, detection, and odometry [13, 26, 29].

While conventional range maps store only the nearest return per ray, LiDAR sensors often capture multiple returns along a ray due to semi-transparent surfaces, vegetation, or multipath reflections. Prior works extend range images to multi-layer representations by discretizing radial distance into fixed intervals and assigning points to the corresponding depth bins [14,20]. However, interval-based layering can fragment continuous surfaces near bin boundaries and scatter real and reflected points across different layers, weakening geometric consistency and making correspondence learning difficult.

To address this limitation, our Layer-Wise Nearest Point Map adopts a layer-centric mapping strategy that preserves cross-layer geometric continuity and structurally aligns real and reflection-induced returns in the ERP domain.

Attention Mechanisms for Geometric Reasoning. Attention architectures such as ViT [2], Uformer [23], Swin Transformer [12], and Neighborhood Attention [7] have demonstrated strong performance in various vision tasks. However, these designs are primarily appearance-driven and do not explicitly model geometric or physical constraints. In virtual point removal, attention can facilitate correspondence reasoning between spatially separated yet geometrically related regions. Prior work [16] constrains attention using fixed spatial priors for RGB reflection removal, where symmetry is largely image-plane based. Such assumptions do not generalize to LiDAR data, where virtual points arise from three-dimensional ray-surface interactions. Consequently, conventional attention mechanisms lack the physically grounded priors required to model reflection-induced structures.

To address this limitation, we introduce Ray-Path-Aware Attention, which constrains feature aggregation along reflection trajectories estimated from surface normals and incident ray directions. By embedding reflection geometry directly into the attention process, our design enables stable and geometry-consistent virtual point classification without relying on handcrafted spatial assumptions.

3 Method

3.1 Overview

We propose a novel end-to-end framework for effectively removing 3D virtual points in the ERP domain (see Fig. 1). Our framework takes as input a 3D LiDAR point cloud containing both real and virtual points, and then embeds it into a multi-layer range map representation in the ERP domain. Unlike the conventional multi-layer range map that assigns each point to the nearest layer, we introduce a Layer-Wise Nearest Point Map, a new layer-centric mapping strategy that aligns points based on their layer correspondence (Sec. 3.2). This

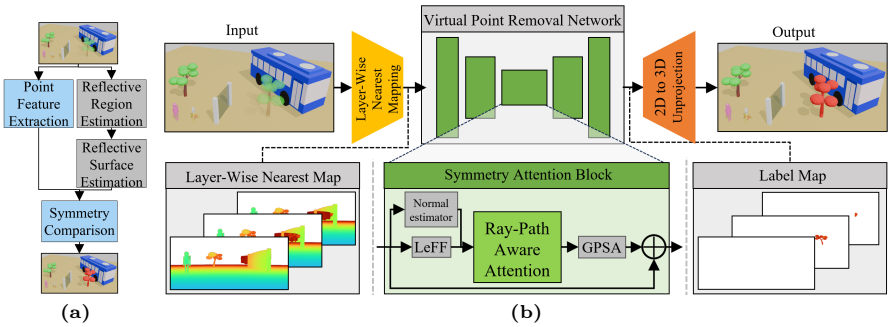

Fig. 1: Overview of the conventional pipeline and proposed framework. (a) Structure of conventional pipelines [3, 11, 24, 25]. (b) Our framework. Input 3D point cloud is mapped into the ERP domain as a Layer-Wise Nearest Point Map, processed by the Virtual Point Removal Network, and reconstructed into 3D via 2D-to-3D unprojection. Each block of the network is a Symmetry Attention Block that integrates Ray-Path-Aware Attention with the normal estimator, LeFF, and GPSA.

design allows real and virtual points along the same ray to co-exist within a layer, facilitating the capture of reflection-induced geometric symmetry.

Based on the constructed Layer-Wise Nearest Point Map, our framework adopts a U-Net style architecture equipped with a Symmetry Attention Block (Sec. 3.3). This block implicitly learns reflection-induced symmetry between real and virtual regions by constraining feature aggregation along physically valid ray paths. Finally, the network generates a label map in the ERP domain, indicating whether each pixel corresponds to a virtual point. The predicted map is then unprojected through the inverse of the Layer-Wise Nearest Point Mapping process to estimate the 3D point cloud with virtual points removed (Sec. 3.4).

3.2 Layer-Wise Nearest Point Map

Representing LiDAR point clouds as a 2D range map in the ERP domain provides a compact and structured form that preserves the ray-based sampling pattern of the sensor. This representation converts irregular 3D points into a regular 2D grid, enabling the use of standard convolutional networks and significantly improving computational efficiency. Thanks to these advantages, range maps have been widely adopted in LiDAR-based perception tasks such as semantic segmentation, object detection, odometry, and sensor calibration [13, 15, 21, 26, 29].

While conventional range maps store only the nearest return along each ray, LiDAR data containing virtual points includes multiple points at different ranges along the same ray. To represent such multi-return information within the range map domain, the single-range structure needs to be extended to a multi-layer range map. A multi-layer range map divides the range along each ray into uniformly spaced layers and assigns each 3D point to its nearest layer, effectively capturing multiple returns along the same viewing direction. This formulation

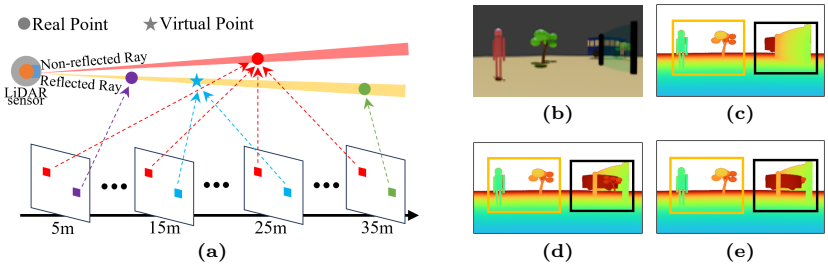

Fig. 2: Illustration of the Layer-Wise Nearest Point Map. (a) Overview of the mapping process. The top shows LiDAR returns from reflective and non-reflective rays, and the bottom shows layer assignment based on distance in the proposed map. (b) Reference RGB image. (c–e) Results for the first, middle, and last layers. Yellow boxes denote non-reflective regions; black boxes indicate reflective regions containing virtual points. Real and reflected points lie on the same layer in (c).

enables ray-wise structural reasoning, which is beneficial for analyzing occlusions and virtual points. However, due to its uniform interval partitioning, points belonging to the same physical surface can be separated across adjacent layers near the interval boundaries, resulting in geometric discontinuities and inconsistent feature extraction. Moreover, because virtual points lie behind real surfaces, they are typically mapped to more distant range layers than their real counterparts. This separation scatters symmetric pairs across layers, making it difficult for the network to capture their correspondence.

To address these issues, we propose a novel Layer-Wise Nearest Point Map that constructs a layer-centric representation for multi-layer range mapping. Unlike a naive multi-layer range map that assigns each 3D point to its nearest layer, the proposed representation assigns the nearest 3D point along each ray to its corresponding layer. Our proposed representation is based on an ERP, where each pixel (u, v) corresponds to an azimuth-elevation (θ, ϕ) pair on the unit sphere. Each pixel (u, v) is spatially aligned with the emitted LiDAR ray map $\mathcal{R}(u, v)$. Based on this ERP, we define a multi-layer map $\mathcal{M} = \{\mathbf{M}_i\}_{i=1}^{L}$, where each layer $\mathbf{M}_i \in \mathbb{R}^{H \times W}$ corresponds to a specific distance d_i along each ray direction, with H and W representing the height and width of the ERP map, respectively, and L denoting the total number of layers. All layers share an identical pixel grid, meaning that the same pixel location represents the same ray direction across layers. Under this setting, we perform a layer-centric mapping that assigns each 3D point to its corresponding layer.

To this end, given an input 3D point cloud $\mathcal{P} = \{\mathbf{p}_j\}_{j=1}^{N}$, where each point $\mathbf{p}_j = [p_x, p_y, p_z]^\top \in \mathbb{R}^3$, we first transform the input point cloud into the ERP domain by quantizing each point according to its azimuth θ and elevation ϕ:

$$(u, v) = \Pi_{\text{ERP}}(\mathbf{p}) = \left(\lfloor \frac{W}{2\pi}(\theta + \pi) \rfloor, \lfloor \frac{H}{\pi} \left(\frac{\pi}{2} - \phi \right) \rfloor \right), \tag{1}$$

where $\lfloor \cdot \rfloor$ denotes the floor function, $\theta = \arctan(p_y / p_x)$, $\phi = \arcsin(p_z / \|\mathbf{p}\|)$, and $\| \cdot \|$ represents the L2 norm. We group all points that fall along the same

pixel (*i.e.*, along the same ray $\mathcal{R}(u,v)$) into a multi-return set $\mathcal{S}(u,v) = \{\, \mathbf{p}_j \in \mathcal{P} \mid \Pi_{\text{ERP}}(\mathbf{p}_j)=(u,v)\}$ for ray-wise aggregation.

For each layer $\mathbf{M}_i$, we then find the point whose range is closest to d_i and encode its range value into the corresponding pixel of the layer map:

$$\mathbf{M}_i(u,v)=\|\hat{\mathbf{p}}_j\|, \quad \text{where} \quad \hat{\mathbf{p}}_j=\underset{\mathbf{p}_j\in\mathcal{S}(u,v)}{\arg\min} \,|\, \|\mathbf{p}_j\| - d_i \,|. \tag{2}$$

The proposed representation inherently possesses structural properties that make it effective in distinguishing virtual points. First, for rays without reflection, only a single physical surface exists along the ray path; thus, all layers sample the same real 3D point. Second, for rays with reflection, both real and virtual points are present, allowing each layer to sample a different 3D point along the ray (see Fig. 2). As a result of these two characteristics, some layers contain both real points and their corresponding virtual points simultaneously, as illustrated in Fig. 2(d).

In other words, real and virtual points that exist at different ranges can be sampled within the same pixel when the reference depth d_i lies between their actual distances, which facilitates symmetry learning in the subsequent attention mechanism. Furthermore, since each layer directly samples the nearest point along the ray, the proposed map naturally preserves dense and continuous surface information.

3.3 Ray Path-Aware Attention

Attention mechanisms are widely used in 2D domains, yet most existing variants rely on local geometry for spatial reasoning. Although several studies have proposed geometry-aware local attention (*e.g.*, [1,7–10,12,18,22,27]), they primarily encode low-level geometric cues such as distances or relative coordinates.

To address this, we propose a Ray Path-Aware Attention module that leverages a high-level geometric prior based on the physical symmetry between virtual and real points. Benefiting from the previously constructed Layer-Wise Nearest Point Map, we can easily identify the first point where a ray hits a surface, which corresponds to a potential reflection region and is associated with the first layer M_1 in our representation. Given the surface normal at this point and the incident ray direction derived from the ERP domain, we can then estimate the reflected ray direction according to the laws of geometric reflection. Furthermore, by utilizing the multi-return set $\mathcal{S}(u,v)$ captured along the same ray, we approximate the 3D trajectory of potential virtual points by uniformly sampling along the reflected ray within the observed return interval. Projecting this trajectory back onto the ERP domain yields physically plausible reflection paths in the 2D domain, which explicitly encode the geometric relationships between real and virtual regions. By guiding the attention region to follow these physically valid reflection paths, the network can implicitly learn reflection-aware correspondences between real and virtual structures without requiring explicit estimation of reflective surfaces. This transforms generic spatial attention into physics-constrained correspondence reasoning.

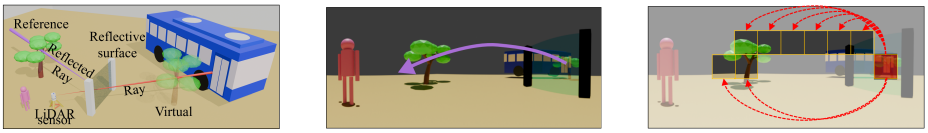

Fig. 3: Illustration of the proposed Ray-Path-Aware Attention. (a) Diagram showing how the paths of incident LiDAR rays change upon reflection from a surface. (b) Reflected ray trajectories. (c) Attention regions along the reflected ray paths.

Specifically, following the conventional procedure [28], we first compute the surface normal map $\mathcal{N}(u, v)$ from $\mathbf{M}_1$. Here, each pixel value in $\mathcal{N}(u, v)$ encodes the surface normal vector at the corresponding surface point. Next, we utilize both the ray map $\mathcal{R}(u, v)$ and the surface normal map $\mathcal{N}(u, v)$ to compute the reflection ray map. Let $\mathbf{r} = \mathcal{R}(u, v)$ and $\mathbf{n} = \mathcal{N}(u, v)$ be the incident ray direction and the surface normal at pixel (u, v), the reflected ray direction $\bar{\mathbf{r}}$ can be computed by the reflection law: $\bar{\mathbf{r}} = \mathbf{r} - 2(\mathbf{r} \cdot \mathbf{n})\mathbf{n}$. Then, to restrict the valid range traversed by the ray, we compute the reflection interval $[t_{\min}, t_{\max}]$ using the multi-return set $\mathcal{S}(u, v)$ as follows:

$$t_{\min} = \min_{\mathbf{p}_j \in \mathcal{S}(u,v)} \|\mathbf{p}_j\|, \quad t_{\max} = \max_{\mathbf{p}_j \in \mathcal{S}(u,v)} \|\mathbf{p}_j\|. \tag{3}$$

Within the valid interval $[t_{\min}, t_{\max}]$, the reflected ray corresponding to pixel (u, v) is represented by a set of sampled points $\mathcal{Q}(u, v) = \{\mathbf{q}(t) \mid t \in [t_{\min}, t_{\max}]\}$, where each point $\mathbf{q}(t)$ on the ray is computed as

$$\mathbf{q}(t) = \mathbf{o} + t\bar{\mathbf{r}}, \tag{4}$$

with $\mathbf{o}$ denoting the LiDAR origin and $\bar{\mathbf{r}}$ the reflected ray direction. When the reflected-ray point set $\mathcal{Q}(u, v)$, obtained by tracing the LiDAR ray corresponding to pixel (u, v) along the reflected direction $\bar{\mathbf{r}}$, is projected onto the ERP domain via Π_{ERP}, it forms a curved path connecting the real and virtual regions, as illustrated in Fig. 3. For each projected point along $\Pi_{\text{ERP}}(\mathcal{Q}(u, v))$, the attention region is computed only within its K nearest neighboring pixels along the projected reflection path, ensuring that feature aggregation is confined to physically valid reflection trajectories.

This physically guided attention improves virtual point removal by reducing false correspondences in non-reflective regions and reinforcing the geometric relationship between real and reflected surfaces.

Symmetry Attention Block. The Symmetry Attention Block integrates the proposed Ray-Path-Aware Attention with complementary modules to exploit the inherent reflection symmetry between real and virtual regions, enabling reasoning over mirrored geometric structures along physically valid ray paths. The Symmetry Attention Block consists of three core components: the Local Enhancement Feed-Forward (LeFF) [23], Global Position Self-Attention (GPSA) [23],

and Ray-Path-Aware Attention modules. LeFF enhances local geometric features within each layer, facilitating the recognition of surface-level symmetry, while GPSA enables coherent interaction of features across layers. Finally, the Ray-Path-Aware Attention constrains feature aggregation along physically plausible reflection trajectories derived from surface normals. Together, these modules form a unified feature space where reflection symmetry can be effectively compared and learned, allowing the network to associate real and virtual regions in a geometrically consistent manner without explicit surface reconstruction or heuristic reflection modeling.

3.4 2D to 3D Unprojection

The network outputs a label map $\mathcal{Y} = \{\hat{\mathbf{Y}}_i\}_{i=1}^{L}$, indicating whether each mapped 3D point on the Layer-Wise Nearest Point Map corresponds to a virtual or real surface point. Given the predicted virtual point label $\hat{\mathbf{Y}}_i$, the reconstruction process operates inversely to the Layer-Wise Nearest Point Mapping. For each 3D point $\mathbf{p}$, we first identify the nearest layer whose reference distance d_i is closest to the range of $\mathbf{p}$:

$$\hat{i} = \arg\min_i \mid \|\mathbf{p}\| - d_i \mid. \tag{5}$$

This ensures consistency with the layer assignment strategy defined in Eq. (2). Since the unprojection follows the same layer-centric assignment principle, no additional geometric post-processing is required.

Using Eq. (1), we identify the corresponding pixel (u, v) to which point $\mathbf{p}$ is quantized in the ERP domain. The predicted label $\hat{\mathbf{Y}}_{\hat{i}}(u, v)$ from the associated layer is then used to determine whether the point $\mathbf{p}$ is classified as a virtual reflection or a real surface point. Points classified as virtual are removed to obtain the final cleaned 3D point cloud.

3.5 Loss Function

The loss is computed in the ERP domain rather than in 3D space. This formulation provides a one-to-one correspondence between each pixel and a LiDAR ray, enabling dense and stable supervision while avoiding the irregular sampling and non-uniform density problems of raw 3D point clouds.

The virtual point removal network is trained using a Binary Cross-Entropy (BCE) loss computed from the predicted label $\hat{\mathbf{Y}}_i(u, v)$ and the ground-truth label $\mathbf{Y}_i(u, v)$. Since not every pixel in the ERP domain corresponds to a valid LiDAR return, a validity mask $\mathcal{V} \in \{0, 1\}^{H \times W \times L}$ is applied to exclude pixels with no mapped point:

$$\mathcal{L} = -\frac{1}{|\mathcal{V}|} \sum_{(u,v,i) \in \mathcal{V}} \big[\mathbf{y} \log \hat{\mathbf{y}} + (1 - \mathbf{y}) \log(1 - \hat{\mathbf{y}})\big], \tag{6}$$

where $\hat{\mathbf{y}}$ and $\mathbf{y}$ denote the scalar values of $\hat{\mathbf{Y}}_i(u, v)$ and $\mathbf{Y}_i(u, v)$ at the same pixel (u, v) and layer position i, respectively.

4 Experiments

4.1 Datasets and Evaluation Protocol

Synthetic Dataset for Scalable Supervision. Given the limited scale of real-world reflection datasets and the difficulty of reliably annotating reflection-induced virtual points, we construct a synthetic reflection dataset based on the Semantic3D benchmark [6]. From 30 outdoor scenes (terrestrial LiDAR scans), we generate 100 augmented variants per scene by inserting planar reflective surfaces with randomly sampled orientations. We then perform physically based ray tracing that models the LiDAR sensor geometry to simulate specular reflection paths. This procedure produces 3,000 synthetic point clouds with point-wise ground-truth annotations distinguishing real and virtual points. The resulting dataset provides controlled and diverse supervision for modeling reflection artifacts that are difficult to capture and annotate at scale in real-world acquisitions.

The synthetic dataset is used only for training supervision, while all reported evaluation results are obtained on real-world LiDAR scans. Detailed dataset generation procedures are provided in the supplementary material.

Real-World Datasets. We evaluate the proposed method on two real-world terrestrial LiDAR datasets containing naturally occurring reflection artifacts: the UNIST LS3DPC dataset [25] (11 scenes) and the 3DRN dataset [3] (12 scenes). Both datasets provide manually annotated point-wise labels for real and virtual points. The UNIST LS3DPC dataset was captured using a RIEGL VZ-400 terrestrial LiDAR scanner and provides raw point clouds with 3D coordinates, RGB color, intensity values, and ground-truth labels. The 3DRN dataset was acquired using a RIEGL VZ-200i LiDAR sensor. Unlike UNIST LS3DPC, the released data are density-sampled point clouds rather than raw scans. Each point therefore contains only its 3D coordinates and the corresponding ground-truth label, without RGB color or intensity attributes.

Synthetic-to-Real Training and Cross-Dataset Evaluation. Because the real-world datasets are relatively small, training high-capacity models solely on them can easily lead to overfitting. To mitigate this issue and improve generalization, we adopt a synthetic-to-real training strategy, where the model is first pre-trained on a large-scale synthetic dataset and then fine-tuned on real-world datasets. In the pre-training phase, we train the model on the synthetic dataset to learn generic reflection-aware geometric representations. Specifically, the 3,000 synthetic point clouds are split into 2,500 training samples and 500 validation samples, with the validation set used to monitor generalization during pre-training. The pre-trained model is then fine-tuned separately on each real-world dataset to adapt from synthetic data to real-world sensor characteristics. Since the real-world datasets are limited in size, we do not reserve an additional validation set during fine-tuning. Instead, each model is trained for a fixed number of epochs and evaluated on the held-out test split. Finally, to assess cross-dataset generalization, we evaluate the model fine-tuned on UNIST

Table 1: Quantitative comparison on two datasets.

Method	UNIST dataset					3DRN dataset				
	FT data	Prec.	Rec.	F1	Avg. Time	FT data	Prec.	Rec.	F1	
Yun and Sim	-	0.7206	0.7475	0.7151	2h 5m	-	0.5361	0.7192	0.6143	
Lee *et al.*	-	0.6089	0.7906	0.6848	2h 2m	-	-	-	-	
Fang *et al.*	-	0.7627	0.7534	0.7580	37m 43s	-	0.7436	0.8291	0.7840	
Proposed, synthetic only	-	0.7959	0.7238	0.7541	8.2s	-	0.8043	0.7711	0.7874	
Proposed, cross fine-tuning	3DRN	0.8194	0.7618	0.7895	8.2s	UNIST	0.8239	0.7894	0.8062	

LS3DPC on 3DRN, and conversely evaluate the model fine-tuned on 3DRN on UNIST LS3DPC.

Metrics. For quantitative evaluation, we follow the evaluation protocol used in prior virtual point removal works such as Yun and Sim [25] and Lee *et al.* [11], reporting Precision, Recall, and F1 score.

4.2 Implementation Details

The input LiDAR point cloud is first projected into a Layer-Wise Nearest Point Map with a resolution of 256×512 and $L = 32$ layers. The layer reference distances are defined with 5-meter intervals starting from 5 meters.

The virtual point removal network follows a U-Net-style encoder-decoder architecture composed of nine symmetry attention blocks with channel dimensions $[32, 64, 128, 256, 512, 256, 128, 64, 32]$. Strided convolutions are used for down-sampling in the encoder, while transposed convolutions are used for up-sampling in the decoder. The network is trained for 100 epochs on the synthetic dataset and fine-tuned for 20 epochs on the real-world datasets using the AdamW optimizer with an initial learning rate of 1×10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay of 0.2. Training uses a mini-batch size of 4 across four NVIDIA TITAN RTX GPUs. Random horizontal flipping is applied for data augmentation. The final output is passed through a logistic function to produce a probability map, and a threshold of 0.5 is used to classify virtual and real points.

4.3 Comparison

We compare the proposed method with three representative virtual point removal approaches that follow conventional multi-stage pipelines: Yun and Sim [25], Lee *et al.* [11], and Fang *et al.* [3]. These methods detect reflection-induced virtual points through a sequence of geometric processing stages, including reflective region detection, reflective surface estimation, and symmetry verification.

Quantitative Results. Tab. 1 reports the quantitative comparison on both real-world datasets. Our method achieves the highest Precision and F1 score across all evaluated settings, indicating more reliable identification of reflection-induced virtual points while preserving physically valid scene geometry.

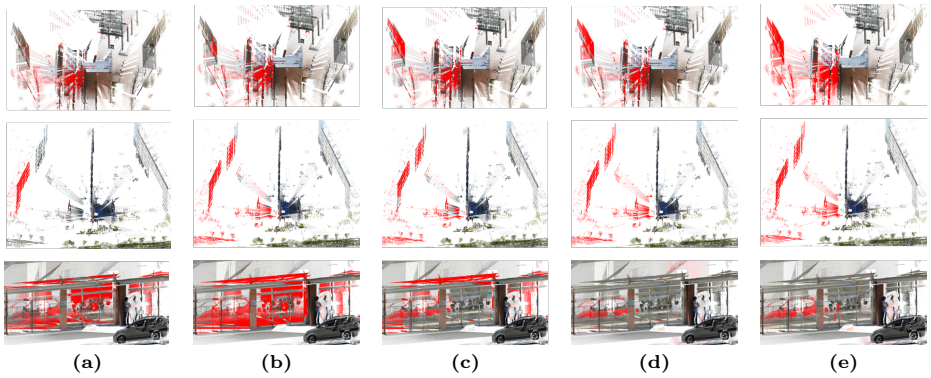

Fig. 4: Qualitative results on the UNIST dataset [25]. (a) Yun and Sim [25]. (b) Lee *et al.* [11]. (c) Fang *et al.* [3]. (d) Proposed method. (e) Ground truth.

The 3DRN dataset does not provide several sensor attributes required by existing methods, such as intensity measurements. Because the released data consist of density-sampled point clouds rather than raw LiDAR scans, some geometric processing steps used in prior pipelines cannot be reproduced under identical input conditions. Therefore, we report the results from Fang *et al.*, which also include the results of Yun and Sim. The results of Lee *et al.* are not included because the evaluation metrics reported in their work are not directly comparable under this evaluation setting.

Qualitative Results. Fig. 4 shows qualitative comparisons between the proposed method and existing symmetry-based approaches, where virtual points are highlighted in red. Conventional symmetry-based methods identify virtual points by enforcing local symmetric consistency around the estimated reflective surface, implicitly assuming that symmetric correspondences arise from mirror reflections. However, real-world scenes often contain naturally symmetric structures that are not caused by reflections. In such cases, reliance on local symmetry leads to ambiguous decisions and may result in erroneous removal of legitimate geometry. In contrast, the proposed method leverages broader scene-level contextual information beyond local symmetric correspondences. By modeling structural relationships at a wider scene scale, it more effectively distinguishes true geometric symmetry from reflection-induced artifacts, enabling stable preservation of valid structures while suppressing virtual reflections.

Runtime. Tab. 1 reports the runtime comparison between the proposed method and existing multi-stage virtual point removal pipelines. Conventional approaches rely on sequential geometric processing stages, leading to substantial computational overhead as the number of input points increases. In contrast, the proposed method performs end-to-end inference on a compact ERP representation, avoiding repeated 3D operations and significantly reducing runtime. The proposed method achieves over two orders of magnitude faster inference compared with

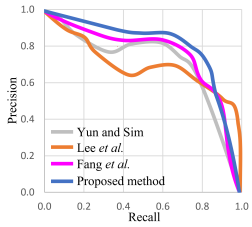

Fig. 5: PR curves. Methods are color-coded: Yun and Sim (Gray), Lee *et al.* (Orange), Fang *et al.* (Magenta), and the proposed method (Blue).

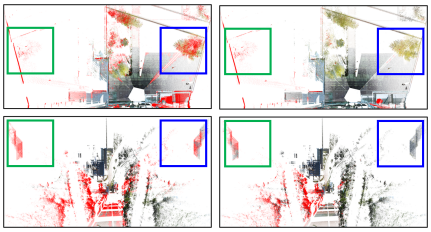

Fig. 6: Attention mechanism comparison. The green box contains virtual points, and the blue box contains the corresponding real points. (a) H-win+V-win. (b) Proposed Ray-Path Aware Attention.

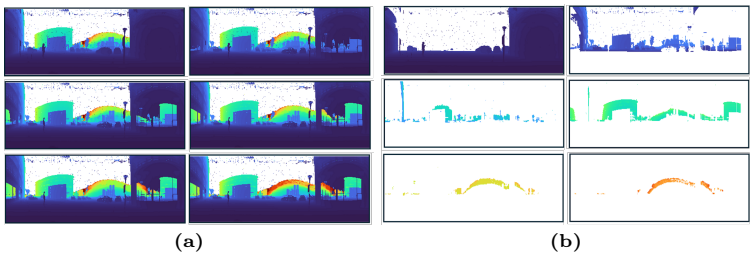

Fig. 7: Comparison of projection strategies. (a) Proposed Layer-Wise Nearest Point Map. (b) Naive interval-based projection.

existing pipelines. Importantly, this improvement in computational efficiency is achieved while maintaining superior detection performance.

Precision-Recall Curve. Fig. 5 shows the precision-recall (PR) curves on the UNIST dataset. In virtual point removal, false positives correspond to removing physically valid geometry, which can negatively affect downstream tasks such as mapping and registration. Therefore, maintaining high precision while preserving recall is particularly important. The proposed method achieves the best precision-recall trade-off, indicating more reliable discrimination between real and reflection-induced virtual points.

4.4 Ablation Study

3D to 2D Projection Strategy. To evaluate the impact of the proposed Layer-Wise Nearest Point Map, we replace it with a naive interval-based projection while keeping all other components unchanged. As shown in Tab. 2a, the naive projection yields the lowest performance, demonstrating that the projection strategy significantly affects the learned representations.

Fig. 8: Visualization of attention regions for virtual point removal. Green markers denote query pixels, blue overlays highlight the constrained attention regions, and brighter purple shades indicate higher attention weights.

The performance degradation mainly arises from surface fragmentation introduced by rigid range intervals. Points belonging to the same physical surface are often split across adjacent layers (Fig. 7), creating artificial discontinuities that do not reflect the true scene geometry. Such irregular sampling patterns are known to produce unstable responses in convolutional networks [14, 20]. In contrast, the proposed per-ray layer-wise sampling preserves cross-layer geometric coherence by selecting points relative to each layer's reference depth, resulting in more stable and discriminative features.

Attention Mechanisms. We compare the proposed Ray-Path Aware Attention with several alternative attention designs, including window-based methods (H-win and H-win+V-win) [17], and general-purpose vision transformers (ViT [2], UFormer [23]).

As reported in Tab. 2a, the proposed attention achieves the best F1 score. Window-based methods obtain relatively high recall but suffer from low precision due to excessive removal of real points. Although H-win+V-win slightly alleviates this issue, both rely on heuristic spatial matching and cannot reliably distinguish between symmetrically aligned real and virtual points (Fig. 6). ViT and UFormer exhibit the lowest performance. While they employ global self-attention, they do not explicitly encode the geometric structure of reflection symmetry. In contrast, our Ray-Path Aware Attention models geometric relationships along each ray path, enabling structured discrimination between real and virtual points and achieving superior precision–recall balance.

Visualization of Ray Path-Aware Attention. We visualize the attention regions and weights of the proposed attention mechanism (Fig. 8). The constrained attention region (blue) derived from the estimated ray path is overlaid with the network attention weights, where brighter colors indicate stronger responses. The attention consistently focuses on the symmetric counterparts of virtual points, indicating that the model effectively exploits reflection-aware correspondences when identifying virtual points.

Robustness to Normal Estimation. The proposed framework relies on surface normals to estimate reflection-aware ray paths. To evaluate its sensitivity to normal quality, we replace the default 3D PCA-based estimator with a 2D range-map gradient–based estimator and introduce synthetic uniform noise and random outlier perturbations to the normals. As shown in Tab. 2b, the performance degrades gradually as the noise level increases, while the F1 score

Table 2: Ablation and efficiency analysis of the proposed method. (a) Attention mechanisms ablation. (b) Robustness to normal estimation.

(a) Attention mechanisms

	Prec.	Rec.	F1
Naive projection	0.1715	0.2768	0.2043
ViT	0.6456	0.6613	0.6533
UFormer	0.5456	0.7427	0.6290
H-win	0.5331	**0.8654**	0.6597
H-win+V-win	0.6360	0.7293	0.6794
Proposed	**0.8194**	0.7618	**0.7895**

(b) Robustness to normals

Setting	Prec.	Rec.	F1
2D Depth	0.7997	0.7435	0.7705
3D PCA	0.8194	0.7618	0.7895
Uniform noise [0,3]	0.8013	0.7597	0.7799
Uniform noise [0,6]	0.8009	0.7417	0.7701
Uniform noise [0,9]	0.7800	0.7252	0.7516
Outliers [30–50] (10%)	0.7926	0.7334	0.7618
Outliers [30–50] (20%)	0.7561	0.7068	0.7306

remains relatively stable under moderate perturbations. Even under substantial perturbations, the performance degradation remains limited. These results indicate that the framework is robust to imperfect normal estimation and remains effective even when the normal inputs are moderately corrupted.

5 Conclusion

In this work, we have proposed an end-to-end framework for removing virtual points directly in the ERP domain. Central to our method is the Layer-Wise Nearest Point Map, which provides a continuous, layer-centric representation of multi-return LiDAR data and preserves geometric relationships between real and reflected surfaces in the 2D domain. Built on this, the proposed Ray-Path-Aware Attention constrains feature aggregation along physically plausible reflection trajectories derived from surface normals and incident ray directions, enabling stable comparison between real and virtual regions without explicit surface reconstruction. Experiments on real-world LiDAR scans demonstrate that our method achieves accurate and reliable virtual point removal, outperforming conventional 3D-2D coupled pipelines. These results highlight that reasoning along ray-aligned geometric constraints is crucial for resolving ambiguity between real and reflection-induced structures.

Acknowledgement

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korean government [Ministry of Science and ICT (MSIT)], including the Artificial Intelligence Graduate School Program [Ulsan National Institute of Science and Technology (UNIST)] under Grant RS-2020-II201336, the AI Star Fellowship Program (UNIST) under Grant RS-202525442824, the Leading Generative Artificial Intelligence (AI) Human Resources Development Program under Grant IITP-2025-RS-2024-00360227, and the Development of Egocentric Data Sensing and Spatial Immersive Experience Technology under Grant RS-2026-25507551.

References

1. Dong, X., Bao, J., Chen, D., Zhang, W., Yu, N., Yuan, L., Chen, D., Guo, B.: Cswin transformer: A general vision transformer backbone with cross-shaped windows. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12124–12134 (2022)
2. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
3. Fang, L., Li, T., Lin, Y., Zhou, S., Yao, W.: A coupled optical–radiometric modeling approach to removing reflection noise in tls data of urban areas. ISPRS Journal of Photogrammetry and Remote Sensing **220**, 217–231 (2025). https://doi.org/https://doi.org/10.1016/j.isprsjprs.2024.12.005, https://www.sciencedirect.com/science/article/pii/S0924271624004751
4. Gao, R., Li, M., Yang, S.J., Cho, K.: Reflective noise filtering of large-scale point cloud using transformer. Remote Sensing **14**(3), 577 (2022)
5. Gao, R., Park, J., Hu, X., Yang, S., Cho, K.: Reflective noise filtering of large-scale point cloud using multi-position lidar sensing data. Remote Sensing **13**(16), 3058 (2021)
6. Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M.: Semantic3d. net: A new large-scale point cloud classification benchmark. arXiv preprint arXiv:1704.03847 (2017)
7. Hassani, A., Walton, S., Li, J., Li, S., Shi, H.: Neighborhood attention transformer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6185–6194 (2023)
8. Hechen, Z., Huang, W., Yin, L., Xie, W., Zhao, Y.: Dilated-windows-based vision transformer with efficient-suppressive-self-attention for insect pests classification. Engineering Applications of Artificial Intelligence **127**, 107228 (2024)
9. Huang, Z., Wang, X., Huang, L., Huang, C., Wei, Y., Liu, W.: Ccnet: Criss-cross attention for semantic segmentation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 603–612 (2019)
10. Jiao, J., Tang, Y.M., Lin, K.Y., Gao, Y., Ma, A.J., Wang, Y., Zheng, W.S.: Dilateformer: Multi-scale dilated transformer for visual recognition. IEEE transactions on multimedia **25**, 8906–8919 (2023)
11. Lee, O., Joo, K., Sim, J.Y.: Learning-based reflection-aware virtual point removal for large-scale 3d point clouds. RA-L (2023)
12. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
13. Lu, F., Chen, G., Liu, Y., Zhang, L., Qu, S., Liu, S., Gu, R.: Hregnet: A hierarchical network for large-scale outdoor lidar point cloud registration. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16014–16023 (2021)
14. Ma, F., Karaman, S.: Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In: 2018 IEEE international conference on robotics and automation (ICRA). pp. 4796–4803. IEEE (2018)
15. Nicolai, A., Hollinger, G.A.: Denoising autoencoders for laser-based scan registration. IEEE Robotics and Automation Letters **3**(4), 4391–4398 (2018)

16. Park, J., Kim, H., Park, E., Sim, J.Y.: Fully-automatic reflection removal for 360-degree images. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1609–1617 (2024)
17. Park, J., Kim, H., Park, E., Sim, J.Y.: Fully-automatic reflection removal for 360-degree images. In: WACV. pp. 1609–1617 (2024)
18. Ramachandran, P., Parmar, N., Vaswani, A., Bello, I., Levskaya, A., Shlens, J.: Stand-alone self-attention in vision models. Advances in neural information processing systems **32** (2019)
19. Shao, W., Kakizaki, K., Araki, S., Mukai, T.: Reflections removal produced by multiple transparent and reflective glass objects in tls measurements. In: COMPSAC. pp. 370–379. IEEE (2023)
20. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: 2017 international conference on 3D Vision (3DV). pp. 11–20. IEEE (2017)
21. Vödisch, N., Unal, O., Li, K., Van Gool, L., Dai, D.: End-to-end optimization of lidar beam configuration for 3d object detection and localization. IEEE Robotics and Automation Letters **7**(2), 2242–2249 (2022)
22. Wang, H., Zhu, Y., Green, B., Adam, H., Yuille, A., Chen, L.C.: Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In: European conference on computer vision. pp. 108–126. Springer (2020)
23. Wang, Z., Cun, X., Bao, J., Zhou, W., Liu, J., Li, H.: Uformer: A general u-shaped transformer for image restoration. In: CVPR. pp. 17683–17693 (2022)
24. Yun, J.S., Sim, J.Y.: Reflection removal for large-scale 3d point clouds. In: CVPR. pp. 4597–4605 (2018)
25. Yun, J.S., Sim, J.Y.: Virtual point removal for large-scale 3d point clouds with multiple glass planes. PAMI **43**(2), 729–744 (2019)
26. Zeng, Y., Hu, Y., Liu, S., Ye, J., Han, Y., Li, X., Sun, N.: Rt3d: Real-time 3-d vehicle detection in lidar point cloud for autonomous driving. IEEE Robotics and Automation Letters **3**(4), 3434–3440 (2018)
27. Zhang, Q., Xu, Y., Zhang, J., Tao, D.: Vsa: Learning varied-size window attention in vision transformers. In: European conference on computer vision. pp. 466–483. Springer (2022)
28. Zhou, Q.Y., Park, J., Koltun, V.: Open3d: A modern library for 3d data processing. `http://www.open3d.org` (2018), accessed: 2025-07-26
29. Zhou, Y., Sun, P., Zhang, Y., Anguelov, D., Gao, J., Ouyang, T., Guo, J., Ngiam, J., Vasudevan, V.: End-to-end multi-view fusion for 3d object detection in lidar point clouds. In: Conference on Robot Learning. pp. 923–932. PMLR (2020)