RAPiD-Seg: Range-Aware Pointwise Distance Distribution Networks for 3D LiDAR Segmentation

Li Li[1], Hubert P. H. Shum[1], and Toby P. Breckon[1,2]

Department of {Computer Science[1] | Engineering[2]}, Durham University, UK
{li.li4, hubert.shum, toby.breckon}@durham.ac.uk

Abstract. 3D point clouds play a pivotal role in outdoor scene perception, especially in the context of autonomous driving. Recent advancements in 3D LiDAR segmentation often focus intensely on the spatial positioning and distribution of points for accurate segmentation. However, these methods, while robust in variable conditions, encounter challenges due to sole reliance on coordinates and point intensity, leading to poor isometric invariance and suboptimal segmentation. To tackle this challenge, our work introduces **R**ange-**A**ware **Poi**ntwise **D**istance Distribution (RAPiD) features and the associated RAPiD-Seg architecture. Our RAPiD features exhibit rigid transformation invariance and effectively adapt to variations in point density, with a design focus on capturing the localized geometry of neighboring structures. They utilize inherent LiDAR isotropic radiation and semantic categorization for enhanced local representation and computational efficiency, while incorporating a 4D distance metric that integrates geometric and surface material reflectivity for improved semantic segmentation. To effectively embed high-dimensional RAPiD features, we propose a double-nested autoencoder structure with a novel class-aware embedding objective to encode high-dimensional features into manageable voxel-wise embeddings. Additionally, we propose RAPiD-Seg which incorporates a channel-wise attention fusion and two effective RAPiD-Seg variants, further optimizing the embedding for enhanced performance and generalization. Our method outperforms contemporary LiDAR segmentation work in terms of mIoU on SemanticKITTI (**76.1**) and nuScenes (**83.6**) datasets.

Keywords: Semantic Segmentation · Point Cloud · Invariance Features

1 Introduction

LiDAR-based point clouds, characterized by pointwise 3D positions and LiDAR intensity/reflectivity [3, 11, 29, 49], play a pivotal role in outdoor scene understanding, particularly in perception systems for autonomous driving. Recent literature in the domain of 3D LiDAR segmentation has witnessed significant strides, with myriad approaches attempting pointwise 3D semantic scene labeling [48]. Leveraging color [7, 38, 39], range imagery [22, 35], and Birds Eye View (BEV) projections [35, 61], multi-modal methods integrate diverse data streams from LiDAR and other sensors to enhance feature representation and segmentation accuracy. Single-modal (LiDAR-only) methods [17, 31, 57, 65] effectively exploit spatial data interpretation and avoid multi-modal issues, *e.g.*, modality heterogeneity and limited sensor field of view intersection [5].
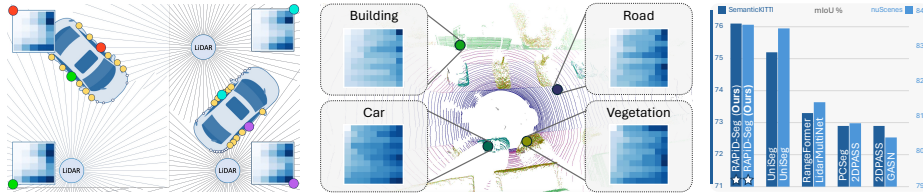
Fig. 1: Left: RAPiD exhibits excellent viewpoint invariance and geometric stability, visualizing comparable features around the vehicle door structure at varying ranges and viewpoints (feature matrix plots inset). **Middle:** RAPiD is distinctive in different semantic classes, as visualized by the matrices. Embedded RAPiD patterns corresponding to different points are visualized using a spectrum of colors, showcasing their capacity to represent different classes. **Right:** Our RAPiD-Seg achieves superior results over SoTA methods on nuScenes [3] and SemanticKITTI [2].

Despite these advancements, a common challenge inherent to these methods is their poor isometric invariance, characterized by heavy reliance on solely coordinates and intensity data, which often results in suboptimal segmentation outcomes [31,65]; this is primarily due to poor translational invariance and visibility (*e.g.*, occlusions), or sparse observations (*e.g.*, at long range) [26], affecting data spatial distribution.

In this work, we seek features that are (1) capable of capturing the localized geometric structure of neighboring points, (2) invariant to rotation and translation, and (3) applicable in noisy LiDAR outdoor environments. While numerous methods fulfill some of these requirements individually [21,33,41], they fall short of addressing them all [10]. Recognizing the need for higher-level features capable of capturing local geometry while potentially incorporating LiDAR point-specific attributes (*e.g.*, intensity and reflectivity), we instead turn our attention to pointwise Distance Distribution (PDD) features [53,54]. They are recognized for their remarkable efficacy in providing a robust and informative geometric representation of point clouds, excelling in both rotational and translational invariance while including intricate local geometry details.

However, directly employing PDD features is impractical due to their high-dimensional nature, and their significant memory and storage requirements for large-scale point clouds. PDD also overlooks local features, as including distant points subsequently dilutes the focus on immediate neighborhoods.

As an enabler, we propose the **R**ange-**A**ware **P**ointwise **D**istance Distribution (RAPiD-Seg) solution for LiDAR segmentation. As shown in Fig. 1, it utilizes invariance of RAPiD features to rigid transformations and point cloud sparsity variations, while concentrating on compact features within specific local neighborhoods. Specifically, our method leverages inherent LiDAR isotropic radiation and semantic categorization for enhanced local representations while reducing computational burdens. Moreover, our formulation computes a 4D distance, incorporating both 3D geometric and reflectivity differences to enhance semantic segmentation fidelity. To compress high-dimensional RAPiD features into tractable voxel-wise embeddings, we propose a novel embedding approach with our class-aware double-nested Au-

toEncoder (AE) module. We further incorporate a channel-wise attention fusion and a two-stage training strategy to optimize the AE independently before integrating into the full network, enhancing performance and generalization capability.

We conduct extensive experiments on SemanticKITTI [2] and nuScenes [3] datasets, upon which our approach surpasses the state-of-the-art (SoTA) segmentation performance.

Overall, as shown in Fig. 2, our contributions can be summarized as follows:

- A novel Range-Aware Pointwise Distance Distribution **feature (RAPiD)** for 3D LiDAR segmentation that ensures robustness to rigid transformations and viewpoints through isometry-invariant metrics within specific Regions of Interest, *i.e.*, intra-ring (R-RAPiD) and intra-class (C-RAPiD).

- A novel method for **embedding** RAPiD with class-aware double nested AE (RAPiD AE) to optimize the embedding of high-dimensional features, balancing efficiency and fidelity.

- A novel open-source network architecture **RAPiD-Seg** [1] and supporting training methodology, to enable modular LiDAR segmentation that achieves SoTA performance on SemanticKITTI (mIoU: **76.1**) and nuScenes (mIoU: **83.6**).
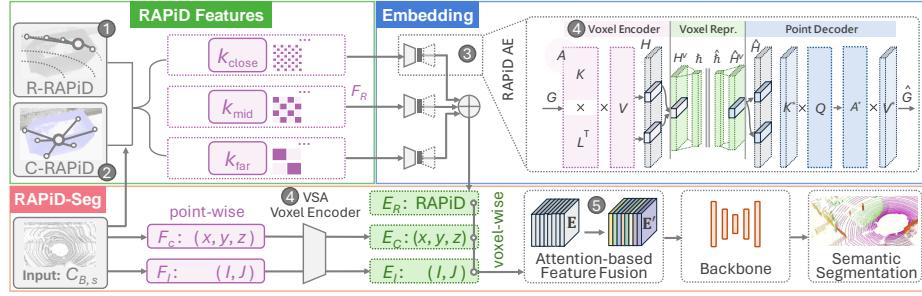
Fig. 2: Our proposed architecture for 3D segmentation framework leverages RAPiD features from the point cloud. We encode pointwise features into voxel-wise embeddings via the voxel encoder and multiple RAPiD AutoEncoders (RAPiD AE). After attention-based feature fusion, these fused embeddings go through the backbone network for segmentation results.

2   Related Work

LiDAR-Based Semantic Segmentation [7, 12, 17, 20, 22, 23, 27, 31, 35, 38, 39, 50, 51, 57–59, 61, 62, 65, 66] is fundamental for LiDAR-driven scene perception, aiming to label each point in a point cloud sequence. The majority of the approaches [17, 39, 50, 58, 65] solely rely on the point-based features of the point cloud, such as SPVCNN [50] which introduces a point-to-voxel branch, using combined point-voxel features for segmentation. Cylinder3D [65] proposes cylindrical partitioning with a UNet [9] backbone variant. LiM3D [31] utilizes coordinates combined with surface reflectivity attributes. Overall, such prior work relies solely on point-based features such as coordinates and intensity of the points, lacking an effective

---

[1] The code is publicly available at: https://github.com/l1997i/rapid_seg.

fusion mechanism, resulting in suboptimal performance [6, 28, 52]. They are also susceptible to changes in viewpoint, distance, and point sparsity [26] due to the lack of isometry across all inter-point distances.

Pointwise Distance Distribution (PDD) captures the local context of each point in a unit cell by enumerating distances to neighboring points in order. It is an isometry invariant proposed by Widdowson & Kurlin [53] to resolve the data ambiguity for periodic crystals, demonstrated through extensive pairwise comparisons across atomic 3D clouds from high-level periodic crystals of periodic structures [53–55]. Though the effectiveness of PDD in periodic crystals and atomic clouds has been proved by the aforementioned studies, to date, no work has applied PDD features to outdoor 3D point clouds. In outdoor settings, common invariant features [21, 32, 33, 41] and Azimuth-normalized method [4] often face issues due to the irregular and sparse nature of the data, which is compounded by increased noise and environmental complexities [32, 33]. Furthermore, the computational demands make them less suitable for the vast scale of outdoor settings [21, 41]. For instance, Melia *et al.* [41] introduce a rotation-invariant feature that struggles with generalization across diverse point cloud densities and scales due to its computational cost and susceptibility to outdoor noise. Recognizing these limitations, we identify an opportunity to leverage the PDD features in a new domain, where representing the local context of neighboring points in a transformation invariant and geometrically robust manner is crucial. Drawing from the advantages of the PDD design, we propose the RAPiD feature, tailored specifically for LiDAR-based point clouds, to capture the localized geometry of neighboring structures.

Point Cloud Representations and Embeddings are fundamental for LiDAR-driven tasks. 3D point cloud can be represented either as point-based [45, 46, 56, 64] or voxel-based [17, 36, 51, 65], each serving different purposes rooted in their unique characteristics. Voxel-based representation involves dividing the 3D space into grids of voxels, which provides a structured way of handling sparse and irregular LiDAR data [9, 13, 14, 65]. Previous work often utilizes Feedforward Neural Network (FNN) [17, 51, 65] to facilitate the conversion of pointwise features into structured voxel-wise features by projecting them onto voxel grids and generating dense embeddings. The primary drawback of FNN conversion is the potential loss of fine-grained details inherent in the original point cloud [15]. FNN may also struggle with the high dimensionality and sparsity of voxelized data, leading to computational inefficiency [47]. Our method addresses these issues by partitioning the point cloud into voxel grids and applying a novel class-ware double nested AE with self-attention mechanisms [15]. This approach not only retains more detailed information by focusing on local structures but also introduces high-dimensional feature compression, thereby improving computational efficiency.

3   Range-Aware Pointwise Distances (RAPiD)

In the context of LiDAR segmentation, a principal challenge lies in the lack of robustness against rigid transformations such as rotation and translation, as well as against variations in viewpoint, points sparsity, and occlusion [5, 26].

Traditional methods predominantly leverage 3D point coordinates to furnish spatial information [17,51,65]; however, they may be inadequate in the scenes with poor visibility (*e.g.*, occlusions) or sparse observations (*e.g.*, at long range) [26]. Such reliance on coordinates alone could lead to inaccuracies in recognizing object transformations or occlusions [26]. Data augmentation, *e.g.*, random geometric transformation can improve robustness under rigid transformations [51,65], but fail to guarantee comprehensive coverage of all potential transformations, resulting in vulnerability to previously unseen variations.

3.1 Overview

To achieve a transformation-invariant 3D data representation, we observe that distances within rigid bodies (*e.g.*, vehicles, roads, and buildings) remain constant under rigid transformations. In light of this, we focus on assimilating the principle of the isometric invariant into the LiDAR-driven point cloud perception. Specifically, we delve into the PDD [53] — an isometry invariant that quantifies distances between adjacent points. For outdoor point clouds, vanilla PDD features are computationally intensive and susceptible to noise and sparsity. Our Range-Aware Pointwise Distance Distribution (RAPiD), specifically designed for LiDAR data, instead calculates the PDD features for each point within specific Regions of Interest (RoI), which are typically associated with the intrinsic structure of LiDAR data.

Mathematical Formulation: Given a fixed number $k > 0$ representing the fixed number of point neighbors and a u-point cluster P_{RoI} comprising no fewer than k points based on RoI, the RAPiD is a $u \times k$ matrix, which retains both spatial distances and LiDAR reflectivity [31] disparities between points. RAPiD are adapted to LiDAR sparsity at different distances by using range-specific parameters k_{close}, k_{mid}, k_{far} for close, mid, and far ranges, respectively.

The k-point RAPiD in region P_{RoI} is defined as:

$$\text{RAPiD}\left(P_{\text{RoI}}; k\right) = \text{sort}\left(\left[\, \text{sort}\left(\left[\, \boldsymbol{\rho}_{j,1}, \ldots, \boldsymbol{\rho}_{j,k} \,\right]\right) \,\right]_{j=1}^{u}\right), \tag{1}$$

$\forall l \in \{1, \ldots, k\}, j \in \{1, \ldots, u\}, \; \boldsymbol{\rho}_{j,l}$ is given by:

$$\boldsymbol{\rho}_{j,l} = \left\|\left[\, \boldsymbol{p}_j - \boldsymbol{p}_{j,l}, \;\; g\left(r_j\right) - g\left(r_{j,l}\right) \,\right]\right\|_2, \tag{2}$$

where $\boldsymbol{p}_j$ and $\boldsymbol{p}_{j,l}$ denote the 3D coordinates of the j-th point and its l-th nearest neighbor within P_{RoI}, respectively; r_j and $r_{j,l}$ represent the reflectivity values of $\boldsymbol{p}_j$ and $\boldsymbol{p}_{j,l}$, correspondingly; $\|\cdot\|_2$ is the Euclidean norm. $g : \mathbb{R} \to [D_{\min}, D_{\max}]$ is the reflectivity mapping function that maps the numerical range of reflectivity onto a consistent scale with the range of Euclidean distances between points:

$$g(r) = \left(\frac{r - r_{\min}}{r_{\max} - r_{\min}}\right)\left(D_{\max} - D_{\min}\right) + D_{\min}, \tag{3}$$

$$D_{\min} = \min_{j,l} \|\boldsymbol{p}_j - \boldsymbol{p}_{j,l}\|_2, \quad D_{\max} = \max_{j,l} \|\boldsymbol{p}_j - \boldsymbol{p}_{j,l}\|_2. \tag{4}$$

where $[D_{\min}, D_{\max}]$ is the range of the Euclidean norms of coordinate differences for all considered point pairs.

The 4D distance in Eq. (2) integrates material reflectivity [31] into RAPiD features, enhancing feature representation and aiding in the discrimination of various materials and surfaces, which is crucial for accurate semantic segmentation.

For convenience to facilitate feature normalization and data alignment, we arrange RAPiD lexicographically by sorting Eq. (1), where sort($\cdot$) on the inner and outer brackets sorts the elements $\boldsymbol{\rho}_{j,l}$ within each row j, and the sorted rows based on their first differing elements, both in ascending order [53]. It treats RAPiD matrices with the same geometric structure but different orders as identical.

To improve the data robustness, we normalize $\boldsymbol{\rho}_{j,l}$ after excluding outliers exceeding a distance threshold δ, substituting these entries with the maximum of the normalized distribution to represent significant inter-point distances.

3.2   Intra-Ring and Intra-Class RAPiD

To enhance the RAPiD ability to capture local context, we propose R-RAPiD and C-RAPiD, based on the inherent structure of LiDAR data. The key benefit of C-RAPiD is its capacity to underscore the inherent traits of points within the same semantic class, achieved based on semantic labels before distance calculation. As a complementary, R-RAPiD is versatile, regardless of semantic label availability. **Intra-Ring RAPiD (R-RAPiD)** is a specialized variant of RAPiD, which confines the RoI to the ring encompassing the anchor points, thereby serving to economize on computational overhead while capitalizing on the inherent structural characteristics of LiDAR data (Fig. 2 ❶).

Given LiDAR sensors with fixed beam counts [2,3,29,49], $e.g.$, $B \in \{32, 64, 128\}$, beams radiate isotropically around the vehicle at set angles. We segment the m-point LiDAR point cloud $C_{B,s}$ by beams, where $m = s \times B$, with s representing measurements within a scan cycle and B the number of laser beams. Suppose $\boldsymbol{p}_i$ with Cartesian coordinate (x_i, y_i, z_i) is a point within $C_{B,s}$, its transformation to cylindrical coordinates (θ_i, ϕ_i) is formulated as:

$$\theta_i = \lfloor \arctan 2(y_i, x_i)/\Delta\theta \rfloor, \tag{5}$$

$$\phi_i = \left\lfloor \arcsin \left[z_i \ (x_i^2 + y_i^2 + z_i^2)^{-1/2} \right] /\Delta\phi \right\rfloor, \tag{6}$$

where $\Delta\theta$ and $\Delta\phi$ denote the mean angular resolutions horizontally and vertically between adjacent beams, and $\arctan 2(\cdot)$ is the 2-argument $\arctan(\cdot)$ function.

$C_{B,s}$ is then partitioned to B rings based on ϕ_b, $i.e.$, $C_{B,s} = \bigcup_{b=0}^{B-1} R_b$, where each ring $R_b = \{(\theta_{b,i}, \phi_{b,i}) \mid \phi_{b,i} = \phi_b, \ \forall i \leq m\}$; $\bigcup$ signifies the cumulative union. R-RAPiD is thus defined as the cumulative concatenate $\oplus$ of RAPiD in each ring R_b,

$$\text{R-RAPiD}(C_{B,s}; k) := \bigoplus\nolimits_{b=0}^{B-1} \text{RAPiD}(R_b; k). \tag{7}$$

Intra-Class RAPiD (C-RAPiD) concentrates on extracting point features within the confines of each semantic class (Fig. 2 ❷). It is complementary to R-RAPiD, preserving the embedding fidelity within individual semantic class.

Specifically, C-RAPiD concatenates RAPiD for each point $\boldsymbol{p}_j$ and its coresponding semantic label y_j within semantic class i, across N_c total classes:

$$\text{C-RAPiD}(\{C_{B,s}, \mathcal{Y}\}; k) := \bigoplus\nolimits_{i=0}^{N_c-1} \text{RAPiD}(S_i; k), \tag{8}$$

where $S_i = \{\boldsymbol{p}_j \mid y_j = i, \ \forall j \leq m\}$, $\mathcal{Y} = [y_j]_{j=1}^m$.

4 RAPiD Embedding

The direct utilization of high-dimensional RAPiD features imposes a substantial computational burden. Efficient processing of large data volumes requires methods that can condense dimensionality while preserving data integrity [63].

We explore data embedding methods to reduce computational costs while providing feature learning capacity. AE is efficient in embedding high-dimensional features utilizing the reconstruction loss [16, 43]. However, conventional AE, limited by their sensitivity to input order and reliance on fixed-size misinputs [24], struggle to align with the unordered and variable size inherent in point clouds [15].

We propose a double nested AE structure with a novel class-aware embedding objective, using the Voxel-based Set Attention (VSA) module [15] as a building block. This facilitates superior contextual awareness and adaptability to unordered and variably-sized point cloud data.

Nested RAPiD AE, illustrated in Fig. 2 ❸ and Fig. 3, is composed of two modules: the outer module is constituted by a VSA AE, primarily responsible for the point-to-voxel conversion. Within the inner module, which focuses on voxel-wise representation, we employ an additional AE specifically designed for dimensionality reduction. Specifically, considering an input of m-point RAPiD features $G \in \mathbb{R}^{m \times d}$, where each point encompasses d features. We compress pointwise G into a voxel-wise representation $H^v \in \mathbb{R}^{c \times l \times d}$ by an outer VSA Voxel Encoder (with c reduced voxel-wise dimension and l latent codes), then further reduces it to a compressed embedding $\hbar \in \mathbb{R}^{c \times l \times d'}$ with d' features via an inner Encoder. The inner and outer VSA Point Decoders then reconstruct the output feature set $\hat{G}$ akin to G.

Inner Voxel-Wise Representation is conducted via an inner AE, which takes H^v from outer VSA AE as the input, yielding a lower-dimensional embedding $\hbar$ (dimension reduced from d to d'). This process involves $u \times$ convolutional layers and a batch normalization layer [19]:

$$\hbar^{(i)} = \text{BatchNorm} \circ \text{Conv}(\hbar^{(i-1)}), \tag{9}$$

where $\hbar^{(0)} = H^v$, $\hbar^{(u-1)} = \hbar$. The Convolutional Feed-Forward Network (Con-vFFN) then promotes voxel-level information exchange, enhancing spatial feature interactivity. It maps reduced hidden features $\hbar$ to a 3D sparse tensor, indexed by voxel coordinates X^v, and employs dual depth-wise convolutions (DwConv) for spatial interactivity:

$$\hat{\hbar} = \text{DwConv}^{(2)} \circ \zeta \circ \text{DwConv}^{(1)} \left(\text{SpT}(\hbar, X^v) \right), \tag{10}$$

where ζ and $\text{SpT}(\cdot)$ represents the non-linear activation and sparse tensor construction. Subsequently, $\hat{\hbar} \in \mathbb{R}^{c \times l \times d'}$ is reconstructed into $\hat{H}^v \in \mathbb{R}^{c \times l \times d}$ through an inner Decoder, consisting of multiple DeConv Layers.

Outer VSA AE consists of a voxel encoder and point encoder. The voxel encoder project RAPiD features into key-value spaces, forming K and V, followed by a cross-attention mechanism with a latent query L, producing an attention
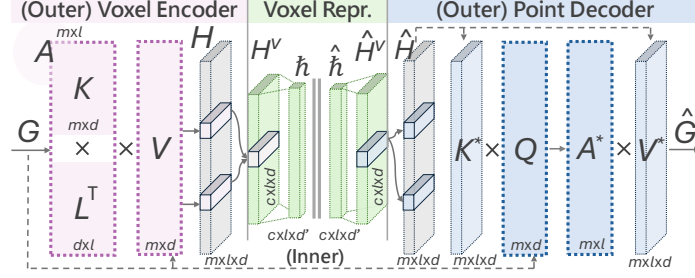
Fig. 3: Our RAPiD AE consists of an Encoder, Convolution Layer, and Decoder module, aiming to reproduce the input features and generate the compressed voxel-wise RAPiD representation $\hbar$.

matrix A. The voxel-wise representation H^v is the scatter sum [44] of pointwise H to aggregate the value vectors V:

$$H^v = \mathrm{Sum_{scatter}}(H, I^v), \quad H = \tilde{A}^\top V, \tag{11}$$

$$\tilde{A} = \mathrm{Softmax_{scatter}}(KL^\top, I^v), \quad (Q, V) = \mathrm{Proj}(G), \tag{12}$$

where $\mathrm{Proj}(\cdot)$ represents the linear projection, and I^v is the voxel indices. The Point Decoder reconstructs the output set from the enriched hidden features $\hat{H}^v$. We start by broadcasting $\hat{H}^v$ based on I^v, resulting in $\hat{H} \in \mathbb{R}^{m \times l \times d}$. The output $\hat{G}$ is analogous to the operational paradigm of Voxel Encoder:

$$\hat{G} = \tilde{A}^{*\top} V^*, \quad \tilde{A}^* = \mathrm{Softmax}(A^*), \tag{13}$$

$$A^* = \left[K_i^* Q_i^\top \right]_{i=1}^m, \quad (K^*, V^*) = \mathrm{Proj}(\hat{\mathcal{H}}). \tag{14}$$

Class-Aware Embedding Objective addresses the issue of non-uniqueness in embeddings produced by the AE, where various distinct inputs yield approximately the same embedding, leading to inaccurate representation [1, 25]. Our objective aims to facilitate the generation of AE embeddings that demonstrate robust semantic class discriminability.

Specifically, we introduce a novel class-aware contrastive loss $\mathcal{L}_{\mathrm{contr}}$ in Eq. (15), which aims to maximize the distance between embeddings of different semantic classes while minimizing the distance of the same class.

$$\mathcal{L}_{\mathrm{contr}} = \frac{1}{m} \sum_{i=1}^m \left[\frac{1}{|P(i)|} \sum_{p \in P(i)} \mathrm{ReLU}(\alpha_p - \mathrm{sim}(H_i, H_p)) \right.$$

$$\left. + \frac{1}{|N(i)|} \sum_{n \in N(i)} \mathrm{ReLU}(\mathrm{sim}(H_i, H_n) - \alpha_n) \right], \tag{15}$$

$$\mathcal{L}_{\mathrm{recon}} = \frac{1}{m \cdot d} \sum_{i=1}^m \sum_{j=1}^d (G_{i,j} - \hat{G}_{i,j})^2, \tag{16}$$

where i represents the point index; $P(i)$ and $N(i)$ are the indices of the nearest point in the same (Positive) and different class (Negative) relative to point i, respectively;

$\text{sim}(\cdot)$ is a similarity measure; α_p and α_n is the positive and negative margin controlling separation between classes, respectively. The pointwise representations H_i, H_p, and H_n are broadcasted from $\hbar$ based on voxel indices I^v.

We further combine this with the MSE [40] reconstruction loss in Eq. (16). The overall loss function, with λ balancing the fidelity of reconstruction with the distinctiveness of the embeddings, is formulated as: $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda \mathcal{L}_{\text{contr}}$.

5   RAPiD-Seg for 3D LiDAR Segmentation

We present RAPiD-Seg, a 3D LiDAR segmentation network leveraging RAPiD features. It incorporates multiple complementary features via the channel-wise fusion mechanism [18]. RAPiD-Seg takes point cloud as input and performs single-modal (LiDAR-only) 3D semantic segmentation. Specifically, the input point cloud has three types of point-wise features: (a) **coordinates-based features** $F_C = \{p_C \mid p_C = (x, y, z)\}$; (b) **intensity-based features** $F_I = \{p_I \mid p_I = (I, J)\}$, where I and J are intensity and reflectivity; (c) **RAPiD features** $F_R = \{p_R \mid p_R = \text{RAPiD}(P_{\text{RoI}}; k)\}$. The RoI varies depending on whether it is R- or C-RAPiD.

As shown in Fig. 2, $F_C \oplus F_I$ and F_R are fed to a VSA voxel encoder ❹ and RAPiD AE ❸ for voxelization, resulting in voxel-wise representation E_C, E_I and E_R. Subsequently, we incorporate the complementary voxel-wise features via the channel-wise fusion mechanism (Sec. 5.1). The backbone net takes in them for LiDAR segmentation prediction. Additionally, we introduce two effective RAPiD-Seg variants to expedite the convergence of the extensive segmentation network towards an optimal solution.

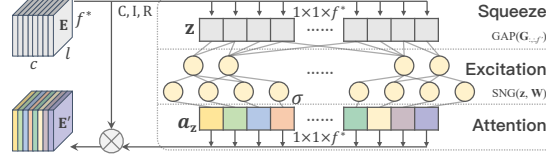### 5.1   RAPiD Channel-Wise Fusion with Attention

To combine various LiDAR point attributes, a highly efficient method for feature fusion is still a research topic [31, 51]. Current methods, which typically concatenate different LiDAR attributes (*e.g.*, intensity [39, 42, 60], reflectivity [31], PLS [51]) face issues with increased dimensionality and complexity, and the need for balancing weights to avoid biased training towards dominant attributes.

We thus propose a novel RAPiD Fusion with Attention module (FuAtten), pioneering application of channel-wise attention mechanism [18] in LiDAR point attribute fusion. With FuAtten, we effectively fuse E_C, E_I and E_R, emphasizing informative features adaptively while suppressing less relevant ones across different channels.

As shown in Fig. 2 ❺ and Fig. 4, we initially concatenate E_C, E_I and E_R along the voxel dimension, resulting in $\mathbf{E}$. In the squeeze operation, Global Average Pooling (GAP) [34] condenses each channel f' of the tensor into a representative channel descriptor. These descriptors form a channel-level embedding $\mathbf{z} = [z_{f'}]_{f'=1}^{f^*}$:

$$\mathbf{z} = [\text{GAP}(\mathbf{E}_{:,:,f'})]_{f'=1}^{f^*} = \frac{1}{c \times l} \left[\sum_{c'=0}^{c-1} \sum_{l'=0}^{l-1} \mathbf{E}_{c',l',f'} \right]_{f'=1}^{f^*}, \qquad (17)$$

where f^* is the dimension of the concatenated feature. A channel-wise attention $\boldsymbol{a}_{\mathbf{z}} \in \mathbb{R}^{1 \times 1 \times f^*}$ is then computed through the excitation step: $\boldsymbol{a}_{\mathbf{z}} =$

Fig. 4: Our feature fusion module with channel-wise attention.

$\sigma\left(\mathbf{W}_2\,\delta(\mathbf{W}_1\mathbf{z})\right)$, where δ and σ represent the ReLU and Sigmoid activation. With each element in $\boldsymbol{a_z}$ reflects the attention allocated to the corresponding feature of the voxel, the channel-wise fused features $\mathbf{E}' = \boldsymbol{a_z} \cdot \mathbf{E}$.

5.2   RAPiD-Seg Architectures for 3D Segmentation

The complexity inherent in 3D LiDAR-driven networks typically exacerbates the end-to-end training process [17, 31, 51], since the extensive parameter space increases the propensity for overfitting, slow convergence, and the possibility of settling into local minima [31].
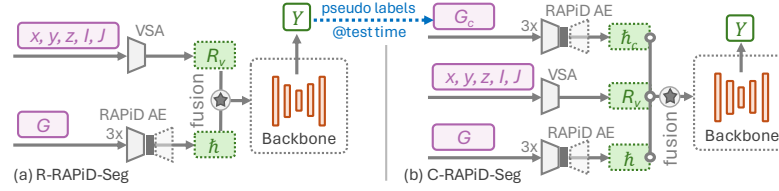
Fig. 5: Two variants of RAPiD-Seg. (a) **R-RAPiD-Seg** utilizes R-RAPiD features, and (b) **C-RAPiD-Seg** utilizes both R- and C-RAPiD features for better performance.

As shown in Fig. 5, we propose two novel and effective variants of RAPiD-Seg for quicker and better performance. R-RAPiD-Seg involves AE training from scratch, followed by C-RAPiD-Seg utilizing C-RAPiD features with the pre-trained AE and backbone for enhanced performance.

R-RAPiD-Seg. We first construct the lightweight R-RAPiD-Seg (Fig. 5 (a)) for fast 3D segmentation. R-RAPiD-Seg adopts an early fusion scheme utilizing LiDAR original points and R-RAPiD features. Specifically, 3D coordinate F_C and intensity-based features F_I are voxelized to voxel-wise representations $R_v = E_C \oplus E_I$ based on VSA voxelization [15]. We get the compressed d'-dimension voxel-wise R-RAPiD representations $\hbar$ (Sec. 4) from RAPiD AE. $\hbar$ is then fused into the voxel-wise representations $R_{v*} = \mathrm{FuAtten}(R_v, \hbar)$, where $\mathrm{FuAtten}(\cdot)$ is Fusion with Attention in Sec. 5.1. The fused voxel-wise features are taken into the backbone network for 3D segmentation.

C-RAPiD-Seg. To facilitate the embedding fidelity within individual semantic classes, we also design a class-aware framework with C-RAPiD features, *i.e.*, C-RAPiD-Seg (Fig. 5 (b)). Specifically, we fuse the voxel-wise C-RAPiD $\hbar_C$ into the representations $R_{v*} = \mathrm{FuAtten}(R_v, \hbar \oplus \hbar_C)$. C-RAPiD requires class labels to

compute the features regarding the semantic categories. Since the ground-truth labels are missing during the test time, we generate the reliable pseudo labels $\tilde{\mathcal{Y}}$ with a pretrained R-RAPiD-Seg based on confidence (Fig. 5 (b) blue dotted arrow). Subsequently, the pseudo voxel-wise C-RAPiD $\hbar_C$ are generated through $\tilde{\mathcal{Y}}$. We fuse them into the voxel-wise representations for 3D segmentation with the bonebone network.

6 Evaluation

Following the popular practice of LiDAR segmentation methods [17,38,65], we evaluate our proposed RAPiD-Seg network against SoTA 3D LiDAR segmentation approaches on the SemanticKITTI [2] and nuScenes [3] datasets.

6.1 Experimental Setup

Datasets: SemanticKITTI [2] comprises 22 point cloud sequences, with sequences 00-10, 08, and 11-21 for training, validation, and testing. 19 classes are chosen for training and evaluation by merging classes with similar motion statuses and discarding sparsely represented ones. Meanwhile, nuScenes [3] has 1000 driving scenes; 850 are for training and validation, with the remaining 150 for testing. 16 classes are used for LiDAR semantic segmentation, following the amalgamation of akin classes and the removal of rare ones.

Evaluation Protocol: Following the popular practice of [2,37,65], we adopt the Intersection-over-Union (IoU) of each class and mean IoU (mIoU) of all classes as the evaluation metric. The IoU of class i is defined as $\mathrm{IoU}_i = \mathrm{TP}_i/(\mathrm{TP}_i + \mathrm{FP}_i + \mathrm{FN}_i)$, where TP_i, FP_i and FN_i denote the true positive, false positive and false negative of class i, respectively.

Implementation Details: We construct the point-voxel backbone based on the Minkowski-UNet34 [8] (re-implemented by PCSeg [37] codebase), which is the open-access backbone with SoTA results to date. Before the AE training stage, we first generate pointwise RAPiD features (Sec. 3), which yield three RAPiD outputs for each frame, corresponding to different k values based on range. Notably, this stage does not impose additional burdens on the subsequent overall training. The number of maximum training epochs for AE and the whole network is set as 100 and the initial learning rate is set as 10^{-3} with SGD optimizer. We use 2 epochs to warm up the network and adopt the cosine learning rate schedule for the remaining epochs. All experiments are conducted on $4\times$ NVIDIA A100 GPUs ($1\times$ for inference).

6.2 Experimental Results

In Tab. 1 and Tab. 2, we showcase the performance of our RAPiD-Seg LiDAR segmentation method on SemanticKITTI and nuScenes *test* set in comparison with published leading contemporary SoTA approaches to demonstrate its superior efficacy (**top 10** results shown; full rankings in Supplementary Material).

Our method significantly outperforms others, especially for rigid object categories (*e.g.*, truck, o.veh, park, *etc.*), primarily due to our fusion of both localized 3D geometry and material reflectivity within RAPiD features, which

Table 1: Quantitative results of RAPiD-Seg and top-10 SoTA segmentation methods on SemanticKITTI [2] *test* set; **Best**/2nd best highlighted; ▲ for multi-modal methods.

Method	mIoU	car	bicy	moto	truc	o.veh	ped	b.list	m.list	road	park	walk	o.gro	build	fenc	veg	trun	terr	pole	sign
Cylinder3D [65]	68.9	97.1	67.6	63.8	50.8	58.5	73.7	69.2	48.0	92.2	65.0	77.0	32.3	90.7	66.5	85.6	72.5	69.8	62.4	66.2
AF2S3Net [7]	69.7	94.5	65.4	**86.8**	39.2	41.1	**80.7**	80.4	74.3	91.3	68.8	72.5	**53.5**	87.9	63.2	70.2	68.5	53.7	61.5	71.0
RPVNet [57]	70.3	97.6	68.4	68.7	44.2	61.1	75.9	74.4	73.4	**93.4**	70.3	**80.7**	33.3	93.5	72.1	86.5	75.1	71.7	64.8	61.4
SDSeg3D [27]	70.4	97.4	58.7	54.2	54.9	65.2	70.2	74.4	52.2	90.9	69.4	76.7	41.9	93.2	71.1	86.1	74.3	71.1	65.4	70.6
GASN [62]	70.7	96.9	65.8	58.0	59.3	61.0	80.4	**82.7**	46.3	89.8	66.2	74.6	30.1	92.3	69.6	87.3	73.0	72.5	66.1	**71.6**
PVKD [17]	71.2	97.0	67.9	69.3	53.5	60.2	75.1	73.5	50.5	91.8	70.9	77.5	41.0	92.4	69.4	86.5	73.8	71.9	64.9	65.8
2DPASS [59]	72.9	97.0	63.6	63.4	61.1	61.5	77.9	81.3	74.1	89.7	67.4	74.7	40.0	93.5	**72.9**	86.2	73.9	71.0	65.0	70.4
PCSeg [37]	72.9	97.5	51.2	67.6	58.6	68.6	78.3	80.9	**75.6**	92.5	71.5	78.3	36.9	93.1	71.4	85.4	73.6	69.9	66.1	68.7
RangeFormer [22]	73.3	96.7	69.4	73.7	59.9	66.2	78.1	75.9	58.1	92.4	73.0	78.8	42.4	92.3	70.1	86.6	73.3	72.8	66.4	66.6
▲UniSeg [38]	75.2	**97.9**	**71.9**	75.2	63.6	74.1	78.9	74.8	60.6	92.6	74.0	79.5	46.1	93.4	72.7	**87.5**	**76.3**	**73.1**	**68.3**	68.5
RAPiD-Seg (Ours)	76.1	97.7	71.1	76.2	**72.5**	**80.7**	79.9	79.1	59.8	91.8	**78.2**	78.6	46.0	**93.6**	72.1	86.9	74.6	72.3	65.9	68.5

Table 2: Quantitative results of RAPiD-Seg and top-10 SoTA segmentation methods on nuScenes [3] *test* set; **Best**/2nd best highlighted; ▲ for multi-modal methods.

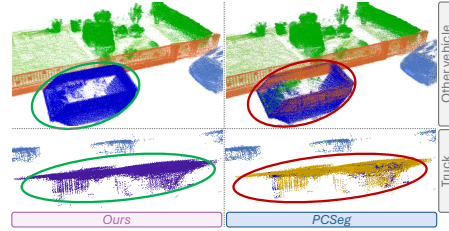| Method	mIoU	barr	bicy	bus	car	const	motor	ped	cone	trail	truck	driv	other	walk	terr	made	veg
▲PMF [66]	77.0	82.0	40.0	81.0	88.0	64.0	79.0	80.0	76.0	81.0	67.0	97.0	68.0	78.0	74.0	90.0	88.0
Cylinder3D [65]	77.2	82.8	29.8	84.3	89.4	63.0	79.3	77.2	73.4	84.6	69.1	97.7	70.2	80.3	75.5	90.4	87.6
AMVNet [35]	77.3	80.6	32.0	81.7	88.9	67.1	84.3	76.1	73.5	84.9	67.3	97.5	67.4	79.4	75.5	91.5	88.7
SPVCNN [50]	77.4	80.0	30.0	91.9	90.8	64.7	79.0	75.6	70.9	81.0	74.6	97.4	69.2	80.0	76.1	89.3	87.1
AF2S3Net [7]	78.3	78.9	52.2	89.9	84.2	77.4	74.3	77.3	72.0	83.9	73.8	97.1	66.5	77.5	74.0	87.7	86.8
▲2D3DNet [12]	80.0	83.0	59.4	88.0	85.1	63.7	84.4	82.0	76.0	84.8	71.9	96.9	67.4	79.8	76.0	92.1	89.2
GASN [62]	80.4	85.5	43.2	90.5	92.1	64.7	86.0	83.0	73.3	83.9	75.8	97.0	71.0	81.0	**77.7**	91.6	90.2
2DPASS [59]	80.8	81.7	55.3	92.0	91.8	73.3	86.5	78.5	72.5	84.7	75.5	97.6	69.1	79.9	75.5	90.2	88.0
▲LidarMultiNet [61]	81.4	80.4	48.4	94.3	90.0	71.5	87.2	**85.2**	**80.4**	86.9	74.8	**97.8**	67.3	80.7	76.5	92.1	89.6
▲UniSeg [38]	83.5	**85.9**	**71.2**	92.1	91.6	80.5	**88.0**	80.9	76.0	86.3	76.7	97.7	**71.8**	80.7	76.7	91.3	88.8
RAPiD-Seg (Ours)	83.6	84.8	64.3	**95.0**	**92.2**	84.6	87.9	81.8	76.8	**88.5**	79.0	**97.8**	66.6	**81.2**	76.7	**92.5**	88.4

enable segmentation based on material properties and local rigid structures. Remarkably, our single-modal methodology outperforms multi-modal approaches [22, 35,38,57,59,61], suggesting superior efficacy of our RAPiD features over alternative modalities like RGB and range images. Our inference time (105ms per frame) is comparable to other contemporary approaches [38,59,65]. Furthermore, we present supporting qualitative results in Fig. 6. Whereas the baseline method struggles with accurate vehicle type differentiation, ours achieves consistent segmentation (more visualization results in the supplementary materials).

RAPiD Features			Attention	mIoU	Δ
Geometric	Reflectivity	Embedding			
			✓	70.04	(Baseline)
✓				70.46	(+0.42)
✓	✓			71.21	(+1.17)
✓	✓	✓		71.93	(+1.89)
	✓	✓	✓	72.15	(+2.11)
		✓	✓	71.80	(+1.76)
✓		✓	✓	72.32	(+2.28)
✓	✓		✓	72.78	(+2.74)
✓	✓	✓	✓	**73.02**	(+2.98)

Table 3: Component-wise ablation of RAPiD-Seg on the SemanticKITTI *validation* set: Geometric (Eq. (1)), Reflectivity (Eq. (3)), (RAPiD) Embedding (Sec. 4), (channel) Attention (Sec. 5.1).

Fig. 6: Comparing our results and PCSeg (baseline) under multi-scan visualization, showing improved segmentation results.

6.3   Ablation Studies

Effectiveness of Components: In Tab. 3, we ablate each component of RAPiD-Seg step by step and report the performance on the SemanticKITTI *validation* set. We start from a baseline model, achieving an mIoU of 70.04 on *validation* set.

Table 4: Effects of RAPiD and Reflectivity features compared to other configurations on SemanticKITTI *validation* set. PDD: the compressed PDD embeddings; RAPiD-R: only RAPiD, without reflectivity feature fusion; RAPiD+R: both RAPiD and reflectivity.

Conf.	mIoU	truck	o.veh	park	walk	o.gro	build	fence
BaseL [37]	70.0	59.8	70.3	69.2	76.9	36.2	93.7	69.6
PDD [53]	66.2	40.3	65.8	67.5	74.6	33.1	92.8	68.2
RAPiD-R	71.8	62.5	69.0	70.3	77.4	35.8	93.6	70.8
RAPiD+R	**73.0**	**70.4**	**78.5**	**75.8**	**78.9**	**44.2**	**94.2**	**73.1**

Incorporating the RAPiD features leads to a notable increase in mIoU (+1.17), underscoring the efficacy of RAPiD in enhancing segmentation performance. Building upon the RAPiD framework, the integration of reflectivity further elevates our mIoU to 71.93, with an overall improvement of +1.89 from the baseline. This significant gain illustrates the critical role of reflectivity in capturing different object materials. Utilizing RAPiD AE to get the voxel-wise representation, we observe an additional improvement, taking the mIoU to 72.15 (+2.11 compared to the baseline), which demonstrates its capability in processing feature representations. We subsequently performed a module-by-module reduction to analyze the model performance. Specifically, without RAPiD-related modules (Geometric/Reflectivity/Embedding), mIoU drops from 73.02 to 70.46 - attributable to the lack of RAPiD features, which are robust to transformation and noise. Our ablation studies show that while disabling individual components leads to some improvement over the baseline, the peak performance is only attained when all components function together. This highlights the indispensable contribution of each component in maximizing segmentation accuracy.

Effectiveness of RAPiD and Reflectivity Features: In Tab. 4, we validate the efficacy of the proposed RAPiD features. The direct application of PDD results in a substantial mIoU decrease of 3.8, with most category IoUs falling below those of the baseline method. This indicates vanilla PDD features are not well-suited for LiDAR segmentation. While our proposed RAPiD (without reflectivity disparities in Eq. (2)) demonstrates an improvement of 1.4 in mIoU over the baseline. Moreover, the integration of reflectivity with RAPiD features significantly enhances performance, yielding 2.6 mIoU increase over the baseline. Notably, our approach exhibits superior performance in the segmentation of most rigid object categories compared to other configurations. Tab. 6 further shows the RAPiD outperforms PDD in varying k values at different ranges on SemanticKITTI *validation* set, with non-uniform k configurations yielding the most significant improvements in mIoU for both datasets, highlighting the effectiveness of the range-aware design of RAPiD.

Effectiveness of RAPiD AE and Architecture Variants: In Tab. 3, we replace our RAPiD AE with a commonly-used ConvFNN [17,51,65] to generate voxel-wise embeddings. Our AE demonstrated a 0.24 mIoU improvement over the FNN. In Tab. 5, we investigate the impact of the variant architectures, *i.e.*, R-RAPiD-Seg and C-RAPiD-Seg. Utilizing R-RAPiD features, R-RAPiD-Seg yields an mIoU of 72.3, which outperforms the baseline method by +2.3 mIoU.

Table 5: 3D segmentation results of different variants of RAPiD-Seg (ours) on SemanticKITTI *validation* set.

Method	mIoU %	car	ped	o.gro	pole
Baseline	70.0	97.2	78.1	35.4	63.5
R-RAPiD-Seg	72.3 (+2.3)	97.4	77.4	**45.0**	62.4
C-RAPiD-Seg	**73.0** (+3.0)	**97.7**	**79.3**	44.6	**66.4**

Table 6: Effects of k at various ranges.

		PDD [54]			RAPiD (ours)		
	k_{near} k_{mid} k_{far}		mIoU	k_{near} k_{mid} k_{far}			mIoU
SemK	7	7	7	64.74 (-7.1)	7	7	7	72.04 (+0.2)
	5	5	5	65.18 (-6.6)	5	5	5	72.28 (+0.5)
	10	7	5	66.23 (-5.6)	10	7	5	**73.02** (+1.2)
nuScene	6	6	6	72.19 (-6.5)	6	6	6	78.76 (+0.1)
	3	3	3	73.68 (-5.0)	3	3	3	79.43 (+0.8)
	8	6	3	72.24 (-6.4)	8	6	3	**79.91** (+1.3)

Concurrent fusing of both R- and C-RAPiD-Seg features in R-RAPiD-Seg improves performance by +0.7 mIoU, which shows a +3.0 overall mIoU enhancement.

Effectiveness of Feature Fusion with Channel-wise Attention: In Tab. 3, we assess the prevalent approach of direct feature concatenation. Our method, employing feature fusion with channel-wise attention, enhances mIoU by 0.87, thereby conclusively demonstrating its effectiveness.

Effectiveness of Backbone Networks: In Tab. 7, we evaluate the performance across multiple backbone networks to demonstrate the versatility and ease of integration of our modules. The results highlight the adaptability and effectiveness of our approach on both pointwise (P) and Voxel-wise (V) backbone architectures. Specifically, the point-based PTv2 backbone achieves an mIoU of 72.6, showing strong performance, particularly in the `car` and `other ground` categories. Among the voxel-based methods, Minkowski-UNet stands out with the highest overall mIoU of 73.0, excelling in the `pedestrian` and `pole` categories.

Table 7: Effects of using different backbones on SemanticKITTI *validation* set, where P and V for Point- and Voxel-wsie methods.

Repr.	Backbone	mIoU %	car	ped	o.gro	pole
P	PTv2 [56]	72.6 (+2.8)	**97.4**	77.4	**45.0**	62.4
V	Cylinder3D [65]	69.8	96.9	74.2	37.9	63.0
V	Minkowski-UNet [8]	**73.0** (+3.2)	97.2	**78.1**	44.2	**65.9**

7   Conclusion and Discussion

This paper presents a novel **R**ange-**A**ware **Po**intwise **D**istance Distribution (RAPiD) feature and the RAPiD-Seg network for LiDAR segmentation, adeptly overcoming the constraints of single-modal LiDAR methods. The rigid transformation invariance and enhanced focus on local details of RAPiD significantly boost segmentation accuracy. RAPiD-Seg integrates a two-stage training approach with reflectivity-guided 4D distance metrics and a class-aware nested AE, achieving SoTA results on the SemanticKITTI and nuScenes datasets. Notably, our single-modal method surpasses the performance of multi-modal methods, indicating the superior efficacy of RAPiD features even compared to other modalities, including RGB and range images.

Our RAPiD features hold significant potential for application in various tasks such as object detection [30], point cloud registration, semi-/weakly- supervised learning, and extend to promising applications in multi-modal research in the future.

References

1. Badger, B.L.: Depth and Representation in Vision Models (2023)
2. Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J.: SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. In: Int. Conf. Comput. Vis. pp. 9296–9306 (2019)
3. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: Nuscenes: A Multimodal Dataset for Autonomous Driving. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 11618–11628 (2020)
4. Chen, S., Wang, X., Cheng, T., Zhang, W., Zhang, Q., Huang, C., Liu, W.: Azi-Norm: Exploiting the Radial Symmetry of Point Cloud for Azimuth-Normalized 3D Perception. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 6387–6396 (2022)
5. Chen, X., Xu, S., Zou, X., Cao, T., Yeung, D.Y., Fang, L.: SVQNet: Sparse Voxel-Adjacent Query Network for 4D Spatio-Temporal LiDAR Semantic Segmentation. In: Int. Conf. Comput. Vis. pp. 8569–8578 (2023)
6. Chen, X., Shi, S., Zhu, B., Cheung, K.C., Xu, H., Li, H.: MPPNet: Multi-frame Feature Intertwining with Proxy Points for 3D Temporal Object Detection. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Eur. Conf. Comput. Vis. pp. 680–697. Springer Nature Switzerland (2022)
7. Cheng, R., Razani, R., Taghavi, E., Li, E., Liu, B.: (AF)2-S3Net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 12547–12556 (2021)
8. Choy, C., Gwak, J., Savarese, S.: 4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3070–3079 (2019)
9. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) Med. Image Comput. Comput. Assist. Interv. pp. 424–432. Springer International Publishing (2016)
10. Flitton, G., Breckon, T.P., Megherbi, N.: A Comparison of 3D Interest Point Descriptors with Application to Airport Baggage Object Detection in Complex CT Imagery. Pattern Recognition **46**(9), 2420–2436 (2013)
11. Geiger, A., Lenz, P., Urtasun, R.: Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3354–3361 (2012)
12. Genova, K., Yin, X., Kundu, A., Pantofaru, C., Cole, F., Sud, A., Brewington, B., Shucker, B., Funkhouser, T.: Learning 3D Semantic Segmentation with only 2D Image Supervision. In: Int. Conf. 3D Vis. pp. 361–372 (2021)
13. Graham, B., Engelcke, M., van der Maaten, L.: 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. In: IEEE Conf. Comput. Vis. Pattern Recog. (2018)
14. Han, L., Zheng, T., Xu, L., Fang, L.: OccuSeg: Occupancy-Aware 3D Instance Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2940–2949 (2020)
15. He, C., Li, R., Li, S., Zhang, L.: Voxel Set Transformer: A Set-to-Set Approach to 3D Object Detection from Point Clouds. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 8407–8417. IEEE (2022)
16. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked Autoencoders Are Scalable Vision Learners. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 16000–16009 (2022)

17. Hou, Y., Zhu, X., Ma, Y., Loy, C.C., Li, Y.: Point-to-Voxel Knowledge Distillation for LiDAR Semantic Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
18. Hu, J., Shen, L., Sun, G.: Squeeze-and-Excitation Networks. In: IEEE Conf. Comput. Vis. Pattern Recog. (2018)
19. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In: Int. Conf. Mach. Learn. pp. 448–456 (2015)
20. Jiang, L., Shi, S., Tian, Z., Lai, X., Liu, S., Fu, C.W., Jia, J.: Guided Point Contrastive Learning for Semi-supervised Point Cloud Semantic Segmentation. In: Int. Conf. Comput. Vis. (2021)
21. Jiang, M., Wu, Y., Zhao, T., Zhao, Z., Lu, C.: PointSIFT: A SIFT-like Network Module for 3D Point Cloud Semantic Segmentation (2018)
22. Kong, L., Liu, Y., Chen, R., Ma, Y., Zhu, X., Li, Y., Hou, Y., Qiao, Y., Liu, Z.: Rethinking Range View Representation for LiDAR Segmentation. In: Int. Conf. Comput. Vis. (2023)
23. Kong, L., Ren, J., Pan, L., Liu, Z.: LaserMix for Semi-Supervised LiDAR Semantic Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 21705–21715 (2023)
24. Kortvelesy, R., Morad, S., Prorok, A.: Permutation-Invariant Set Autoencoders with Fixed-Size Embeddings for Multi-Agent Learning. In: Int. Conf. Auton. Agents Multiagent Syst. pp. 1661–1669. International Foundation for Autonomous Agents and Multiagent Systems (2023)
25. Kumar, A., Poole, B.: On Implicit Regularization in β-VAEs. In: Int. Conf. Mach. Learn. pp. 5480–5490 (2020)
26. Li, E., Casas, S., Urtasun, R.: MemorySeg: Online LiDAR Semantic Segmentation with a Latent Memory. In: Int. Conf. Comput. Vis. pp. 745–754 (2023)
27. Li, J., Dai, H., Ding, Y.: Self-Distillation for Robust LiDAR Semantic Segmentation in Autonomous Driving. In: Eur. Conf. Comput. Vis. pp. 659–676 (2022)
28. Li, J., Dai, H., Han, H., Ding, Y.: MSeg3D: Multi-Modal 3D Semantic Segmentation for Autonomous Driving. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 21694–21704 (2023)
29. Li, L., Ismail, K.N., Shum, H.P.H., Breckon, T.P.: DurLAR: A High-Fidelity 128-Channel LiDAR Dataset with Panoramic Ambient and Reflectivity Imagery for Multi-Modal Autonomous Driving Applications. In: Int. Conf. 3D Vis. pp. 1227–1237 (2021)
30. Li, L., Qiao, T., Shum, H.P.H., Breckon, T.P.: TraIL-Det: Transformation-Invariant Local Feature Networks for 3D LiDAR Object Detection with Unsupervised Pre-Training. In: Brit. Mach. Vis. Conf. (2024)
31. Li, L., Shum, H.P.H., Breckon, T.P.: Less is More: Reducing Task and Model Complexity for Semi-Supervised 3D Point Cloud Semantic Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
32. Li, X., Li, R., Chen, G., Fu, C.W., Cohen-Or, D., Heng, P.A.: A Rotation-Invariant Framework for Deep Point Cloud Analysis. IEEE Trans. Vis. Comput. Graph. **28**(12), 4503–4514 (2022)
33. Liang, J., Lai, R., Wong, T.W., Zhao, H.: Geometric Understanding of Point Clouds Using Laplace-Beltrami Operator. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 214–221 (2012)
34. Lin, M., Chen, Q., Yan, S.: Network In Network. In: Int. Conf. Learn. Represent. (2014)
35. Liong, V.E., Nguyen, T.N.T., Widjaja, S., Sharma, D., Chong, Z.J.: AMVNet: Assertion-based Multi-View Fusion Network for LiDAR Semantic Segmentation (2020)

36. Liu, J., Yu, Z., Breckon, T.P., Shum, H.P.H.: U3DS3: Unsupervised 3D Semantic Scene Segmentation. In: IEEE Winter Conf. Appl. Comput. Vis. pp. 3759–3768 (2024)
37. Liu, Y., Bai, Y., Chen, R., Hou, Y., Shi, B., Li, Y., Kong, L.: PCSeg: An Open Source Point Cloud Segmentation Codebase. `https://github.com/PJLab-ADG/PCSeg` (2023)
38. Liu, Y., Chen, R., Li, X., Kong, L., Yang, Y., Xia, Z., Bai, Y., Zhu, X., Ma, Y., Li, Y., Qiao, Y., Hou, Y.: UniSeg: A Unified Multi-Modal LiDAR Segmentation Network and the OpenPCSeg Codebase. In: Int. Conf. Comput. Vis. pp. 21662–21673 (2023)
39. Liu, Y., Kong, L., Cen, J., Chen, R., Zhang, W., Pan, L., Chen, K., Liu, Z.: Segment Any Point Cloud Sequences by Distilling Vision Foundation Models. In: Adv. Neural Inform. Process. Syst. (2023)
40. Marmolin, H.: Subjective MSE measures. IEEE Trans. Syst. Man Cybern. **16**(3), 486–489 (1986)
41. Melia, O., Jonas, E., Willett, R.: Rotation-Invariant Random Features Provide a Strong Baseline for Machine Learning on 3D Point Clouds. Trans. Mach. Learn. Res. (2023)
42. Nunes, L., Wiesmann, L., Marcuzzi, R., Chen, X., Behley, J., Stachniss, C.: Temporal Consistent 3D LiDAR Representation Learning for Semantic Perception in Autonomous Driving. In: IEEE Conf. Comput. Vis. Pattern Recog. (2023)
43. Preechakul, K., Chatthee, N., Wizadwongsa, S., Suwajanakorn, S.: Diffusion Autoencoders: Toward a Meaningful and Decodable Representation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 10619–10629 (2022)
44. PyTorch: Scatter: PyTorch Scatter Documentation. `https://pytorch-scatter.readthedocs.io/en/latest/functions/scatter.html` (2023)
45. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 652–660 (2017)
46. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In: Adv. Neural Inform. Process. Syst. vol. 30 (2017)
47. Shao, Y., Tan, A., Yan, T., Sun, Z., Zhang, Y., Liu, J.: MS23D: A 3D Object Detection Method Using Multi-Scale Semantic Feature Points to Construct 3D Feature Layer (2023)
48. Sun, J., Xu, X., Kong, L., Liu, Y., Li, L., Zhu, C., Zhang, J., Xiao, Z., Chen, R., Wang, T., et al.: An Empirical Study of Training State-of-the-Art LiDAR Segmentation Models. arXiv:2405.14870 (2024)
49. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2446–2454 (2020)
50. Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S.: Searching Efficient 3D Architectures with Sparse Point-Voxel Convolution. In: Eur. Conf. Comput. Vis. (2020)
51. Unal, O., Dai, D., Van Gool, L.: Scribble-Supervised LiDAR Semantic Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2022)
52. Wang, S., Zhu, J., Zhang, R.: Meta-RangeSeg: LiDAR Sequence Semantic Segmentation Using Multiple Feature Aggregation. IEEE Robot. Autom. Lett. **7**(4), 9739–9746 (2022)

53. Widdowson, D., Kurlin, V.: Resolving the Data Ambiguity for Periodic Crystals. In: Adv. Neural Inform. Process. Syst. vol. 35, pp. 24625–24638 (2022)
54. Widdowson, D., Kurlin, V.: Recognizing Rigid Patterns of Unlabeled Point Clouds by Complete and Continuous Isometry Invariants With No False Negatives and No False Positives. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1275–1284 (2023)
55. Widdowson, D., Mosca, M.M., Pulido, A., Cooper, A.I., Kurlin, V.: Average Minimum Distances of Periodic Point Sets – Foundational Invariants for Mapping Periodic Crystals. MATCH Commun. Math. Comput. Chem. **87**(3), 529–559 (2022)
56. Wu, X., Lao, Y., Jiang, L., Liu, X., Zhao, H.: Point Transformer V2: Grouped Vector Attention and Partition-Based Pooling. In: Adv. Neural Inform. Process. Syst. (2022)
57. Xu, J., Zhang, R., Dou, J., Zhu, Y., Sun, J., Pu, S.: RPVNet: A Deep and Efficient Range-Point-Voxel Fusion Network for LiDAR Point Cloud Segmentation. In: Int. Conf. Comput. Vis. pp. 16024–16033 (2021)
58. Yan, X., Gao, J., Li, J., Zhang, R., Li, Z., Huang, R., Cui, S.: Sparse Single Sweep LiDAR Point Cloud Segmentation via Learning Contextual Shape Priors from Scene Completion. In: Conf. AAAI Artif. Intell. vol. 35, pp. 3101–3109 (2021)
59. Yan, X., Gao, J., Zheng, C., Zheng, C., Zhang, R., Cui, S., Li, Z.: 2DPASS: 2D Priors Assisted Semantic Segmentation on LiDAR Point Clouds. In: Eur. Conf. Comput. Vis. (2022)
60. Ye, D., Chen, W., Zhou, Z., Xie, Y., Wang, Y., Wang, P., Foroosh, H.: LidarMultiNet: Unifying LiDAR Semantic Segmentation, 3D Object Detection, and Panoptic Segmentation in a Single Multi-task Network. In: IEEE Conf. Comput. Vis. Pattern Recog. Worksh. (2022)
61. Ye, D., Zhou, Z., Chen, W., Xie, Y., Wang, Y., Wang, P., Foroosh, H.: LidarMultiNet: Towards a Unified Multi-Task Network for Lidar Perception. In: Conf. AAAI Artif. Intell. vol. 37, pp. 3231–3240 (2023)
62. Ye, M., Wan, R., Xu, S., Cao, T., Chen, Q.: Efficient Point Cloud Segmentation with Geometry-Aware Sparse Networks. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Eur. Conf. Comput. Vis. pp. 196–212. Springer (2022)
63. Zhang, C., Duc, K.D., Condon, A.: Visualizing DNA Reaction Trajectories with Deep Graph Embedding Approaches. In: Adv. Neural Inform. Process. Syst. Worksh. arXiv (2022)
64. Zhao, H., Jiang, L., Jia, J., Torr, P., Koltun, V.: Point Transformer. In: Int. Conf. Comput. Vis. (2021)
65. Zhu, X., Zhou, H., Wang, T., Hong, F., Ma, Y., Li, W., Li, H., Lin, D.: Cylindrical and Asymmetrical 3D Convolution Networks for LiDAR Segmentation. In: IEEE Conf. Comput. Vis. Pattern Recog. (2021)
66. Zhuang, Z., Li, R., Jia, K., Wang, Q., Li, Y., Tan, M.: Perception-Aware Multi-Sensor Fusion for 3D LiDAR Semantic Segmentation. In: Int. Conf. Comput. Vis. pp. 16280–16290 (2021)