PointCT: Point Central Transformer Network for Weakly-supervised Point Cloud Semantic Segmentation

Anh-Thuan Tran[1] Hoanh-Su Le[3,4] Suk-Hwan Lee[2] Ki-Ryong Kwon[1]
[1]Department of Artificial Intelligence Convergence, Pukyong National University, South Korea
[2]Department of Computer Engineering, Dong-A University, South Korea
[3]Faculty of Information Systems, University of Economics and Law, Ho Chi Minh City, Vietnam
[4]Vietnam National University, Ho Chi Minh City, Vietnam
thuantran@pukyong.ac.kr, skylee@dau.ac.kr, krkwon@pknu.ac.kr

Abstract

Although point cloud segmentation has a principal role in 3D understanding, annotating fully large-scale scenes for this task can be costly and time-consuming. To resolve this issue, we propose Point Central Transformer (PointCT), a novel end-to-end trainable transformer network for weakly-supervised point cloud semantic segmentation. Divergent from prior approaches, our method addresses limited point annotation challenges exclusively based on 3D points through central-based attention. By employing two embedding processes, our attention mechanism integrates global features across neighborhoods, thereby effectively enhancing unlabeled point representations. Simultaneously, the interconnections between central points and their distinct neighborhoods are bidirectional cohered. Position encoding is further applied to enforce geometric features and improve overall performance. Notably, PointCT achieves outstanding performance under various labeled point settings without additional supervision. Extensive experiments on public datasets S3DIS, ScanNet-V2, and STPLS3D demonstrate the superiority of our proposed approach over other state-of-the-art methods.

1. Introduction

With the rapid growth of 3D techniques and their increasing scopes, point cloud segmentation has become an indispensable component for thoroughly understanding complex real-world scenes. It is also a critical factor in diverse practical applications such as autonomous driving, robotics, and smart cities to quickly capture the surrounding environment in 3D navigation and planning. Several studies have been developed to achieve efficient performance in large-scale point clouds, such as farthest point sampling to deal with vast points [20], adaptation of kernel points to local geometry [4, 25], enriched geometric features by integrating position encoding [10] or global contextual factors [7]. Additionally, transformer networks have gained attention due to their remarkable performance in 2D images [6]. Point Transformer [34] introduces self-attention mechanisms on raw point clouds to perform 3D understanding. The network is then improved by stratified sampling strategy [11] and group vector attention [27].

However, these works are built upon fully supervised point clouds, and in real-world scenes, such as cities, it is impractical and costly to annotate all points in kilometers of areas. As a result, weakly-supervised point cloud segmentation has become a more important and popular topic for near-future scenarios. Existing approaches has been introduced various techniques to overcome sparse annotations in large-scale point clouds, including augmentation [14], pseudo-labeling [17, 23, 26], pre-training [8, 29, 32], fine-tuning [18], multiple instance learning [31], Siamese network [13, 30], and contrastive learning [16, 33]. Although these methods achieve encouraging performance on multiple datasets, several limitations remain to be resolved. Firstly, existing approaches involve multiple stages of pre-training and fine-tuning [8, 32], which can be challenging to train and deploy in practical applications compared to the end-to-end training scheme. Secondly, the exploration of relationships between central points and their neighbors, in conjunction with the global characteristics of these 3D points, are inadequately explored [9, 30], resulting in an ineffective utilization of the limited valuable annotations.

Motivated by these challenges, we propose an end-to-end transformer network for weakly-supervised point cloud segmentation. In line with previous studies [9, 34], our network explores point representations through their corresponding neighborhoods, which are constructed by kNN. While we extract features from a query point, we define it as ”central point” with surrounding points referred to as

”neighboring points”. This approach implies that a single point in 3D space can be served as central point within its neighborhood, while concurrently role of a neighboring point within neighborhoods of other points. Fundamentally, our approach addresses sparse annotations by leveraging the relationships between central points and their neighborhoods to improve point representations through central-based attention. By utilizing two embedding processes, we extract global features across various neighborhoods and integrate them with central points to enhance unlabeled point features. Consequently, our method ensures that valuable global features, along with relevant central point characteristics, are efficiently shared across each point in neighborhoods. This approach provides flexibility and enriches unlabeled point representations. Furthermore, position encoding also has an important role in 3D large-scale understanding to provide essential geometric information. In weakly-supervised settings, we utilize this module with additional features to describe point positions more comprehensibly.

Overall, the proposed method, Point Central Transformer, presents a novel and straightforward approach that leverages central-based attention mechanisms and transformer architecture to tackle the challenge of sparse annotations. Our method has shown outstanding performance, demonstrating its capability to process a limited number of labeled points and outperforming other state-of-the-art methods without additional supervision or complex techniques, resulting in a highly efficient and effective solution. In general, our main contributions can be described as below:

- We propose a novel end-to-end trainable transformer network with central-based attention to overcome sparse annotations in point cloud segmentation.
- We introduce position encoding module in point cloud weak supervision, concentrating on different geometric components to extract point representations and improve model spatial reasoning.
- Our experimental results on benchmark datasets demonstrate the outstanding performance of the proposed method compared to state-of-the-art studies for both indoor and real-world point clouds.

2. Related Work

2.1. Large-scale point cloud segmentation

To deal with large-scale point clouds, various point-based methods have been developed. Specifically, PointNet++ [20], one of the prominent studies, introduces an encoder-decoder architecture and farthest point sampling to extract features through neighborhoods instead of the entire point cloud in PointNet [19]. PointNeXt [21] further optimizes this work with different training strategies. On the other side, RandLA-Net [10] concentrates on geometric positions to address irregular structures. Spatial features are incorporated into the network to reinforce point positions during training. KPConv [25] resolves this issue by applying kernel weights to points with local geometry, which provide more flexibility than grid convolutions.

Recently, transformer architecture has been extended to point clouds due to its remarkable performance in diverse fields by the capacity to learn on massive data points. Point Transformer [34] is one of the pioneering studies to propose a self-attention network on raw points and spatial feature supplements as position encoding. Stratified Transformer [11] further improves this approach with a stratified sampling strategy and context-relative position encoding. Subsequently, Point Transformer V2 [27] is introduced to leverage the encoder-decoder process through group vector attention. Although these methods have reached impressive performance on both indoor and outdoor point clouds, they rely on vast labeled data for training, a resource-intensive and impractical undertaking in real-world applications. Conversely, this study aims to extract semantics from a limited number of annotations by employing an attention mechanism solely based on 3D points, thereby providing a straightforward and cost-effective solution.

2.2. Weakly-supervised point cloud segmentation

Inspired by weak supervision of 2D images, several works are proposed to perform point cloud segmentation under sub-cloud level [26] or limited labeled point settings. $\prod$ Model [12] and Mean Teacher [24] use ensemble models for semi-supervised learning to tackle sparse annotations, while SSPC [3] explores super-point graphs for weak supervision settings. Other approaches utilize contrastive learning and pseudo-labeling to overcome sparse annotations. For example, 1T1C [17] introduces pseudo-labeling with contrastive loss. PSD [33] proposes perturbed branches to ensure predictive consistency with context-aware features. HybridCR [14] integrates pseudo labels and consistency regularization strategy with local and global guidance contrastive learning. Ren et al. [22] introduce an unsupervised 3D generation algorithm for pseudo-labeling. DAT [28] utilizes consistency constraints under local and regional adaptive transformations. In outdoor point clouds, Coarse3D [15] uses contrastive learning with entropy-driven sampling. LESS [16] leverages contrastive prototype learning and pre-segmentation to minimize manual labeling. Shi et al. [23] performs pseudo-labeling through temporal matching and graph propagation.

Furthermore, siamese networks are applied in weakly-supervised point cloud segmentation. GaIA [13] embeds relative entropy through Siamese network. Xu and Lee [30] exploit Siamese self-supervision with color smoothness constraints. Besides, pre-trained models are involved in this context. Hou et al. [8] propose a 3D pre-trained network to transfer to complex tasks with data-efficient learning. PointConstrast [29] proposes unsupervised pre-training in point clouds. Zhang et al. [32] utilize prior knowledge from self-supervised pretext tasks to overcome sparse annotations. Using hand-crafted features and a pre-trained model, Mei et al. [18] require a small fraction of input points for fine-tuning. On the other side, MIL-derived Transformer [31] integrates transformer backbone with three other losses in multi-instance learning to explore pair-wise cloud-level supervision. SQN [9] utilizes existing feature extraction for point embeddings, and these encoded features are upsampled by a semantic query network to explore native structures from neighboring points.

Different from previous approaches, we propose an end-to-end trainable weakly-supervised network purely based on 3D points to overcome limited point annotation setting. Specifically, we introduce a central-based attention mechanism that capitalizes on interconnections between central points and neighboring points, utilizing global features across multiple neighborhoods. Although SQN also integrates spatial features within its decoder stages, it computes central point features by aggregating neighborhood individually, without considering interactions with other neighborhoods. Consequently, this approach encounters limitations under weak supervision, especially in scenarios where both central points and neighboring points are unlabeled.

3. Proposed method

To address previous limitations, we propose a novel yet straightforward end-to-end trainable network to perform point cloud semantic segmentation using a small fraction of labeled points. The approach is applied directly on 3D raw points and can be operated without any pre-training, active learning, or pseudo-labeling and surpasses other studies through extensive experiments on benchmark datasets.

3.1. Preliminary

When the attention layer receives a point cloud D from previous downsampling stages, as illustrated in Figure 1(c), we first assume $P_d = \{p_i\}_{i=1}^N$ and $F_d = \{f_i\}_{i=1}^N$, where $p_i \in \mathbb{R}^3$ is the point position of the i-th point and $f_i \in \mathbb{R}^C$ is its corresponding features with N is the total number of points. The objective is to obtain point output features $F'_d = \{f'_i\}_{i=1}^N$ and then predicted labels $\hat{Y} = \{\hat{y}_i\}_{i=1}^N$. Instead of processing entire point cloud scene, several works [10, 20, 34] focus on extracting features locally based on neighboring points through k-Nearest Neighbor. Specifically, each query point, denoted as p_i, is explored through its K neighboring points $P_{ij} = \{p_{ij} : i \in N, j \in K\}$ and corresponding features $F_{ij} = \{f_{ij} : i \in N, j \in K\}$. During feature extraction of a given query point p_i, we identify it as "central point", while surrounding points p_{ij} in its neighborhood as "neighboring points".

We denote M is the number of labeled points in limited point annotation settings, $D^l = (P_M^l, F_M^l)$ and $D^u = (P_{N-M}^u, F_{N-M}^u)$ are the point coordinates and features of labeled and unlabeled points. Within D^l, point labels Y^l remain unchanged, whereas for D^u, point labels Y^u are assigned as unlabeled type. The network is expected to perform fully annotated segmentation F'_d from input point clouds $D^l \cup D^u$. During training, while feature extraction extends to all N points, only M points with labels Y^l are engaged in loss computation for back-propagation. To establish consistency and comparability, our approach aligns with the selection strategy deployed in previous works [9, 14, 30, 33] to choose M labeled points. In testing phase, our network is evaluated on fully annotated scenes to obtain predicted labels $\hat{Y}$ on total N points. This approach ensures a fair comparison with previous weakly-supervised studies [9, 14, 33] and fully-supervised methods [20, 34].

3.2. Central-based attention

This section describes our proposed central-based attention to address the challenges associated with sparse point annotations by integrating two embedding processes.

In weakly-supervised point clouds, the main challenge revolves around effectively optimize unlabeled point features to perform fully annotated point cloud segmentation while using only limited number of labeled points for training. The current transformer-based networks [11, 27, 34] employ local attention, which compute weights separately on each neighborhood. Although this attention mechanism demonstrates effectiveness in fully supervised large-scale point clouds, where labeled points tend to be densely interconnected and proximate to each other, it reveals several limitations when operating under weak supervision. In local attention, weights assigned to neighboring points within a given neighborhood are computed using features solely from that particular scope without considering relationships between other neighborhoods. The corresponding central point is integrated by weight summating of these neighboring points. However, in sparse annotations, training loss for back-propagation is computed using only limited labeled points. Consequently, this attention mechanism is only optimized when central points are labeled and unlabeled neighboring points could be updated accordingly. Conversely, this approach underestimates the value of labeled points if they serve as neighboring points while the central points remain unlabeled. The situation is exacerbated when both central points and neighboring points are unlabeled, which we define as "unlabeled neighborhoods".

The above shortcomings lead us to propose central-based attention, which effectively handles sparse annotations through two embedding processes. In the initial embedding, we resolve the problem related to unlabeled neigh-

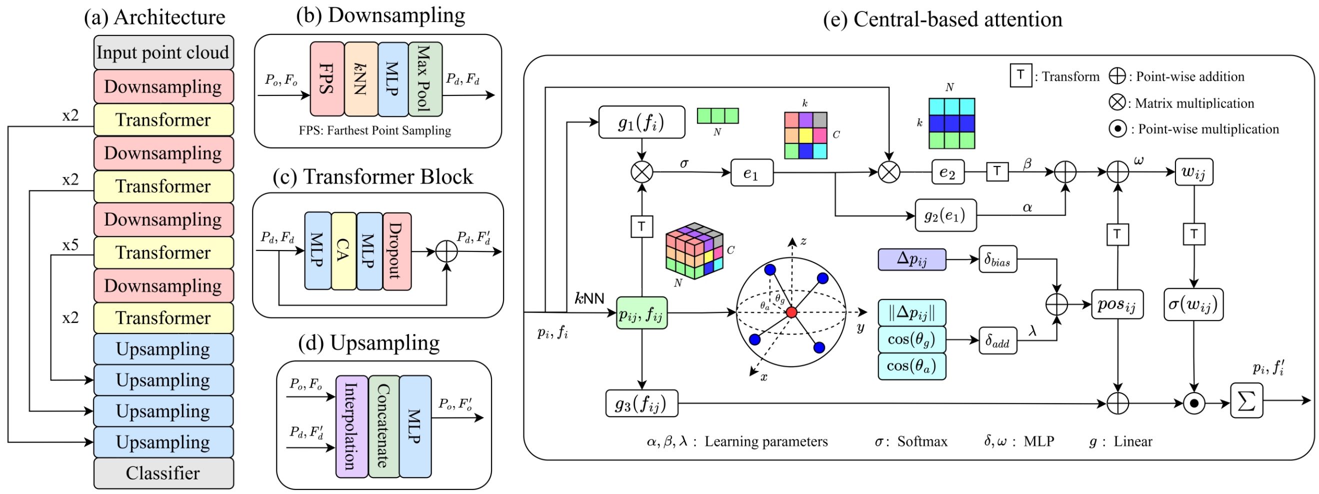

Figure 1. The overall PointCT network (a) receives input point clouds containing point coordinates and RGB colors. The features are then encoded through downsampling (b) with corresponding transformer block depth (c) and decoded by upsampling (d) with classifier layer to obtain predicted labels. Within each transformer block, a central-based attention layer (CA) (e) is integrated between MLP layers, facilitated by a residual connection. The attention weights are computed from global features in neighboring points e_1 and central point sharing e_2. The spatial features are then embedded through position encoding module pos_{ij}.

boring points by considering relationships with other neighborhoods. While previous studies [14, 26] obtain global features from only central points, we adopt a novel approach by extracting them in terms of neighboring points and then sharing across neighborhoods. In essence, we leverage existing labeled neighboring points to improve unlabeled features in multiple neighborhoods, regardless of whether central points are labeled. As a result, neighboring points, even in unlabeled neighborhoods, are also enhanced through valuable global features from other areas that might contain labeled points. Moreover, the benefits extend to unlabeled central points, which are further optimized through their integration from these corresponding neighborhoods.

Nonetheless, global feature extraction encounters obstacles involved in neighborhood construction. In 3D space, a single point can be served as a neighboring point in multiple neighborhoods with different order positions. Consequently, if global features in terms of neighboring points are simply aggregated from all these neighborhoods, the result would be a blend of all points, devoid of any meaningful semantic context. To overcome this issue, we distinguish points that attend in various neighborhoods by embedding their features with corresponding central weights. To be specific, central weights are derived from central point features, initially in dimension $N \times C$, and then transformed into dimension $N \times 1$. The global features in terms of neighboring points, denoted as e_1, are computed by aggregating these features based on their order positions within respective neighborhoods.

Particularly, the proposed attention mechanism with two embedding phases is illustrated in Figure 1(e). For a central point p_i, we compute central weights by linear layer g_1 with only one dimension. Then, the neighboring points $P_{ij} = \{p_{ij} : i \in N, j \in K\} \in \mathbb{R}^{N \times k \times 3}$, $F_{ij} = \{f_{ij} : i \in N, j \in K\} \in \mathbb{R}^{N \times k \times C}$ are explored using kNN to obtain K number of points. In the first embedding, global features are extracted by integrating central weights and neighboring point features. These features are further passed through softmax function σ to be served as neighboring point weights in second embedding process.

$$e_1 = \sigma(f_{ij}^T \times g_1(f_i)) \in \mathbb{R}^{k \times C \times 1} \tag{1}$$

$$e_2 = f_i \times e_1 \in \mathbb{R}^{k \times N \times 1} \tag{2}$$

Although e_1 improves unlabeled neighboring point representations in weakly-supervised point clouds through valuable global features, it tends to lack crucial local characteristics specific to separate neighborhoods and diminishes central point influence. As a result, its effectiveness in computing neighboring point weights might be suboptimal compared to previous studies [9, 34]. To tackle this limitation, we introduce the second embedding process aimed at sharing central point features to respective neighborhoods through the aforementioned global features e_1. This approach treats these global features as weights for neighboring points and subsequently performs matrix multiplication in combination with central points. Therefore, the second embedding, denoted as e_2, ensures that central point features are effectively propagated to each point within their corresponding neighborhoods, guided by appropriate global features.

To construct attention weights for neighboring points, which cover global features and valuable central weights, we amalgamate these embedding processes with position encoding as spatial supplements. Specifically, global features e_1 are transformed through g_2 linear layer and integrated with central sharing mechanism e_2 using learnable parameters α, β. This integration enhances local characteristics through central point sharing while also preserving global features. Moreover, position encoding pos_{ij} is incorporated into attention weights to supplement appropriate geometric features based on point positions. The enriched attention weights are then sent through softmax function σ and multiplied with the value feature map. Ultimately, we aggregate all neighboring points by weighted summation to avoid irregular point ordering problems and obtain output features.

$$w_{ij} = \omega(\alpha \odot g_2(e_1) + \beta \odot e_2^T + pos_{ij}^T) \in \mathbb{R}^{k \times C \times N} \quad (3)$$

$$f_i' = \sum_{j=1}^{K} \sigma(w_{ij}^T) \odot (g_3(f_{ij}) + pos_{ij}) \in \mathbb{R}^{N \times C} \quad (4)$$

3.3. Position encoding

Point clouds are characterized by an uneven distribution of points, leading to complex relationships and intricate connections. In transformer architecture and attention modules, previous works [10, 11, 27, 34] have utilized position encoding to capture spatial information and resolve features lost in high-level transformations. However, sparse annotations with limited labeled points have exacerbated this issue, highlighting the need for more comprehensible methods to explore rich geometric features from various perspectives. To address this demand, our proposed approach introduces geometric point information, which is merged into spatial features, as shown in Figure 1(e). We accomplish this effect by concatenating additional geometric features, such as altitude angle θ_a, azimuth angle θ_g, Euclidean distance $\|\Delta p_{ij}\|$, through MLP layer δ_{add} and integrate into position encoding δ_{bias} with appropriate learnable parameters λ.

The module enables us to capture more comprehensive geometric information and thereby cohere point representations, particularly in weakly-supervised settings with limited labeled points. In addition, position encoding is essential in balancing neighboring features after the two central embedding processes, ensuring that the model effectively leverages the inner relationships between points to obtain optimal results.

$$\Delta p_{ij} = p_{ij} - p_i = (\Delta x_{ij}, \Delta y_{ij}, \Delta z_{ij}) \quad (5)$$

$$\cos(\theta_g) = \frac{\sqrt{(\Delta x_{ij})^2 + (\Delta y_{ij})^2}}{\|\Delta p_{ij}\|} \quad (6)$$

$$\cos(\theta_a) = \frac{\|\Delta y_{ij}\|}{\sqrt{(\Delta x_{ij})^2 + (\Delta y_{ij})^2}} \quad (7)$$

$$pos_{ij} = \delta_{bias}(\Delta p_{ij}) + \lambda \odot \delta_{add}(\|\Delta p_{ij}\|, \cos(\theta_g), \cos(\theta_a)) \quad (8)$$

3.4. Network architecture

The overall encoder-decoder architecture is illustrated in Figure 1(a). Our point-based network utilizes both XYZ coordinates and RGB colors as input. Within the encoder branch, multiple Transformer blocks are integrated into each downsampling stage to extract essential features. Then, these encoded features are sent into the decoder branch, where upsampling blocks are used to obtain dense output labels for semantic segmentation.

Downsampling. The downsampling layers are illustrated in Figure 1(b). First, we use farthest point sampling [20] to select and group point indices through kNN from the original point cloud P_o and its corresponding features F_o. In our experiments, we define the downsampling scale of 4, reducing the number of points by four in each downsampling layer. The selected points are then grouped, and corresponding features are aggregated using max pooling.

Transformer Block. The transformer block mentioned in Figure 1(c) takes point coordinates P_d and corresponding features F_d from previous downsampling stages as inputs to compute the output features F_d'. In each transformer block, central-based attention is attached between MLP layers with a dropout module and a residual connection. The transformer block leverages point features with geometric factors, producing updated features for all points as its output.

Upsampling. As seen from Figure 1(d), the above points P_d and F_d' from transformer blocks, along with corresponding original ones P_o and F_o are concatenated by interpolation techniques [20] to get upsampling features F_o' with original points P_o. The resulting features F_o' are then passed through MLP layers to obtain the output features for semantic segmentation.

4. Experiment

4.1. Experiment setting

Datasets. We evaluate the proposed network performance on three large-scale point cloud datasets: S3DIS [1], ScanNet-V2 [5], and STPLS3D [2]. **S3DIS** is one of popular indoor point cloud datasets for semantic segmentation. It contains 271 rooms in six areas with 13 classes. We utilize six attributes of each points as inputs, including XYZ coordinates and RGB colors. **ScanNet-V2** is another indoor large-scale point cloud dataset, including 1,613 point cloud scenes with 20 classes. It provides point clouds with RGB attributes and well-annotated points. **STPLS** presents a real-world, large-scale, and synthetic aerial photogrammetry point cloud dataset. It was collected using a crosshatch-type flight pattern and covers more than 16 km^2 landscapes in four areas with six classes.

Implementation details. We implement Point Central Transformer in PyTorch. The input points are sent to grid sampling with different grid sizes before training. Following [9,34], S3DIS and ScanNet-V2 apply a grid size of 4cm. STPLS3D utilizes a grid size of 30cm based on default settings [2]. For a fair comparison, we follow previous studies [9, 14, 30, 33] to use Area (1,2,3,4,6) for training with different weak label settings (10%,1%,0.1%) while reserving Area-5 solely for testing. Moreover, we also expand the comparison on S3DIS 6-fold cross-validation, where each area is treated as the test set once and all others are utilized for training. Regarding the Scannet-V2 dataset, our network is trained on training set and evaluates results against the benchmarks set by the online test set [5]. For STPLS3D, we test network performance by conducting experiments on WMSC point cloud while training on other areas based on original work [2].

We train for 100 epochs with AdamW optimizer using weight decay 0.1 and cross-entropy loss. The learning rate is first set to 0.0005, dropped by 10x at epochs 60 and 80. In the encoder branch, transformer block depth is set to 2 – 2 – 5 – 2. The initial feature dimension is 32 and will double after each downsampling layer. Transformer blocks are connected to upsampling layers in decoder branch via skip connections. MLP layers are composed of linear layers, batch normalization, and ReLU activation functions. Following previous studies [9, 34], we construct neighboring points using k nearest neighbor with K number set to 16 and downsampling scale set to 4.

Evaluation metrics. We evaluate network performance on all points of test set. The mean Intersection-over-Union (mIoU) is used as the standard metric in our experiments.

4.2. Experimental result

The experimental results are shown in Table 1-4 with S3DIS, ScanNet-V2, and STPLS3D datasets. Underline and **Bold** represent the best results under fully-supervised and weakly-supervised settings, respectively.

Qualitative results on S3DIS and Scannet-V2. The comparison between our proposed model and other state-of-the-art methods with different point annotations has been conducted on S3DIS Area-5, and the results are summarized in Table 1. Obviously, PointCT consistently achieves the highest results across various point annotation levels. Compared to RandLA-Net [10], Zhang et.al [32], SQN [9], and PointTransformer [34], we outperform in mIoU by 5.9%, 3.6%, 2.9%, and 1.6% at the setting of 10% point annotations. Moreover, PointCT surpasses aforementioned studies with a lower number of labeled points under 1% setting by 7.8%, 5.8%, 4.0%, and 1.8%, respectively. Under 0.1% point annotation level, we obtain remarkable performance that exceeds RandLA-Net [10], SQN [9], Point-Transformer [34] by 15.4%, 6.9%, 2.0% in mIoU. We have also extended our experimentation to fully supervised point clouds on Area-5 and achieved competitive performance in comparison to current fully-supervised studies. Interestingly, our observations reveal that the model configured with the 0.1% settings achieves optimal performance, even outperforming models utilizing more labeled points. Further elaboration on this phenomenon can be found in Section 4.3.

For a comprehensive comparison with [14, 32, 33], we have also performed experiments on S3DIS using 6-fold cross-validation with 0.1% point annotations, as reported in Table 2. Specifically, PointCT surpasses other weakly-supervised studies [14, 32, 33] that use 1% labeled points, exceeding their mIoU performance by 5.3%, 3.2%, and 2.0%, respectively. Furthermore, we outperform SQN [9] by 7.5% in mIoU using a similar point annotation level.

In ScanNet-V2, as highlighted in Table 3, PointCT achieves exceptional performance across both annotation settings and outperforms other weakly-supervised methods. In comparison to Zhang et al. [32], PSD [33], and HybridCR [14], our approach under 1% point annotations exceeds their respective mIoU by 13.2%, 9.6%, and 7.5%, respectively. Furthermore, PointCT surpasses SQN [9] by 6.2% under 0.1% point annotations.

The visualization results are shown in Figure 2. It is observed that our proposed network achieves remarkable performance in indoor scenes compared to Point Transformer [34]. As illustrated in Figure 2, the proposed network demonstrates the ability to capture global context under limited annotations. Surprisingly, PointCT further optimizes this task by eliminating human errors in the last sample. In this instance, it accurately identifies chairs based on crucial patterns, demonstrating its proficiency despite deviations from the ground truth.

Qualitative results on STPLS3D. We evaluate our network using original benchmarks from real-world point cloud dataset STPLS3D [2], which results are detailed in

Table 1. Semantic segmentation results on S3DIS Area-5. Underline presents the best results under fully-supervised settings, and **Bold** shows the best results under weakly-supervised settings.

Settings	Method	mIoU	ceil.	floor	wall	beam	col.	wind.	door	chair	table	book.	sofa	board	clut.
100%	PointNet++ [20]	50.0	90.8	96.5	74.1	0.0	5.8	43.6	25.4	69.2	76.9	21.5	55.6	49.3	41.9
	HybridCR [14]	65.8	93.6	98.1	82.3	0.0	24.4	59.5	66.9	79.6	87.9	67.1	73.0	66.8	55.7
	RandLA-Net [10]	64.6	92.4	96.8	80.8	0.0	18.6	57.2	54.1	79.8	87.9	70.2	74.5	66.2	59.3
	SQN [33]	63.7	92.8	96.9	81.8	0.0	25.9	50.5	65.9	79.5	85.3	55.7	72.5	65.8	55.9
	PointTrans [34]	70.4	94.0	98.5	86.3	0.0	38.0	63.4	74.3	82.4	89.1	80.2	74.3	76.0	59.3
	PointCT	67.9	94.0	98.3	85.5	0.0	26.1	61.0	73.9	81.1	88.3	65.1	73.6	76.3	59.1
10%	RandLA-Net [10]	61.7	91.7	97.8	79.4	0.0	28.4	50.8	45.5	81.3	85.2	57.1	70.3	63.8	51.8
	Zhang et.al [32]	64.0	-	-	-	-	-	-	-	-	-	-	-	-	-
	SQN [9]	64.7	93.0	97.5	81.5	0.0	28.0	55.8	68.7	80.1	87.7	55.2	72.3	63.9	57.0
	PointTrans [34]	66.0	**93.7**	**98.3**	83.7	0.0	35.0	48.1	70.9	**81.9**	**88.3**	**60.3**	**73.2**	67.3	**57.2**
	PointCT	**67.6**	92.3	**98.3**	**84.6**	0.0	**35.8**	**62.6**	**79.7**	80.5	86.6	57.9	72.3	**73.5**	54.5
1%	HybridCR [14]	65.3	92.5	93.9	82.6	0.0	24.2	**64.4**	63.2	81.7	78.3	74.4	69.0	68.2	56.5
	RandLA-Net [10]	59.8	92.3	97.5	77.0	**0.1**	15.9	48.7	38.0	78.0	83.2	62.4	68.4	64.9	50.6
	Zhang et.al [32]	61.8	91.5	96.9	80.6	0.0	18.2	58.1	47.2	75.8	85.7	65.2	68.9	65.0	50.2
	SQN [9]	63.6	92.0	96.4	81.3	0.0	21.4	53.7	**73.2**	77.8	86.0	56.7	69.9	66.6	52.5
	PointTrans [34]	65.8	94.2	98.2	83.0	0.0	**44.2**	50.4	68.8	**83.0**	88.1	47.4	**75.2**	64.3	59.0
	PointCT	**67.6**	**94.7**	**98.5**	**85.3**	0.0	24.7	59.4	71.6	79.9	**88.6**	**69.5**	73.2	**73.8**	**59.4**
0.1%	RandLA-Net [10]	52.9	89.9	95.9	75.3	0.0	7.5	52.4	26.5	74.5	62.2	60.2	49.1	49.3	45.1
	SQN [9]	61.4	91.7	95.6	78.7	0.0	24.2	55.9	63.1	70.5	83.1	60.7	67.8	56.1	50.6
	PointTrans [34]	66.3	92.6	97.7	83.5	0.0	**35.4**	56.9	69.6	78.9	84.8	69.3	66.2	**74.0**	53.0
	PointCT	**68.3**	**92.7**	**98.3**	**85.1**	0.0	31.2	**60.7**	**73.0**	**79.9**	**89.2**	**82.3**	**71.6**	70.1	**54.2**

Table 2. Results on S3DIS 6-fold.

Setting	Method	mIoU
100%	PointNet++ [20]	54.5
	RandLA-Net [10]	70.0
	PointTrans [34]	73.5
1%	Zhang et al. [32]	65.9
	PSD [33]	68.0
	HybridCR [14]	69.2
0.1%	SQN [9]	63.7
	PointCT	**71.2**

Table 3. Results (mIoU) on ScanNet-V2.

Setting	Method	Val	Test
100%	PointNet++ [20]	53.5	55.7
	RandLA-Net [10]	-	64.5
1%	Zhang et al. [32]	-	51.1
	PSD [33]	-	54.7
	HybridCR [14]	56.9	56.8
	PointCT	**65.6**	**64.3**
0.1%	SQN [9]	58.4	56.9
	PointCT	**63.7**	**63.1**

Table 4. Results on STPLS3D.

Setting	Method	mIoU
100%	KPConv [25]	53.7
	RandLA-Net [10]	50.5
	SCF-Net [7]	50.7
	MinkowskiNet [4]	51.3
	PointTrans [34]	47.6
0.1%	PointCT	49.2
0.01%	PointCT	**53.2**

Table 4. PointCT, operating under two labeled point settings, consistently outperform PointTransformer [34] with full supervision by 1.6% and 5.6% in mIoU, respectively. Interestingly, PointCT, even with 0.01% labeled points, surpasses most of fully-supervised methods, except KPConv. One of the main reason behind this outstanding performance is the ability to eliminate noises in limited point annotations.

As illustrated in Figure 3, input point clouds contain confusing objects (yellow boxes), which are small buildings with similar shapes and colors to cars. Through leveraging fewer labeled points, PointCT efficiently differentiates these objects and improves performance in complex real-world scenes under 0.01% point annotations. Moreover, employing our method in weakly-supervised point clouds enhances identification of smaller objects, such as ”light pole” by a large margin, thereby leading to impressive mIoU increments using solely 3D points in our attention mechanisms. For further details, please refer to the per-class performance in the Supplementary Material, Section 1.

4.3. Ablation study

Central-based attention and position encoding. The module effectiveness is evaluated on S3DIS Area-5 under 0.1% labeled point setting by considering various factors and integrations, including central-based attention (CA), position encoding in attention weight (pos att), and value features (pos value). In model I, we remove transformer blocks in network architecture to obtain baseline model. Model II utilize central-based attention without position encoding by removing pos_{ij} in equations (3) and (4). Model III and IV compare performance of position encoding sep-

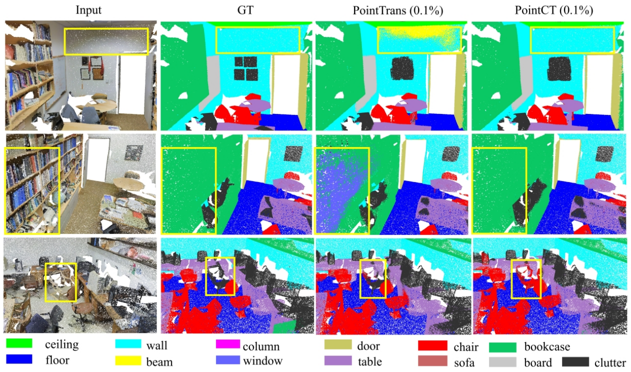

Figure 2. Visualization on indoor S3DIS Area-5 compared to Point Transformer under 0.1% setting with ground truth (GT).

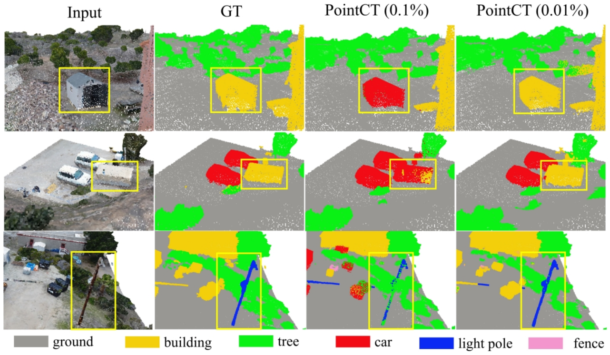

Figure 3. Visualization on real-world STPLS3D under two different labeled point annotations with ground truth (GT).

arately in attention weight with equation (3) and value features with equation (4). Finally, the proposed central-based attention was obtained by combining position encoding from these two modules.

As seen from Table 5, our central-based attention improves baseline model by 1.5% in mIoU. Additionally, the position encoding in value features has fewer impacts, only enhancing by 0.4% compared to 1.8% in attention weights. However, despite spatial features being integrated into attention weights to express point positions, the feature transformation process can diminish their impacts. Therefore, our network requires additional position reinforcement within the value features to achieve a more comprehensive improvement. The central-based attention with these two position encoding modules achieved the best result, improving the baseline by 6.5%.

Table 5. Ablation study on different modules.

ID	CA	*pos* att	*pos* value	mIoU
I				61.8
II	✓			63.3
III	✓		✓	63.5
IV	✓	✓		65.1
V	✓	✓	✓	**68.3**

Number of labeled points. The experimental results with different annotation levels (100%, 10%, 1%, 0.1%) are summarized in Table 1. While the overall mIoU exhibits fluctuations across various labeled point settings, substantial improvements are observed in several categories that contain noisy data. Specifically, for S3DIS-Area 5, the ”bookcase” and ”table” categories with complex backgrounds demonstrate interesting improvements under 0.1% point annotations compared to other settings. In contrast, categories such as ”sofa,” ”clutter,” and ”board” experience marginal fluctuations as the number of labeled points decreases. These findings highlight the effectiveness of our proposed approach with limited annotations in achieving a more balanced performance across diverse categories by mitigating these noises.

Similar patterns are observed in real-world point cloud STPLS3D, enhancing ”light pole” performance against other methods. This situation also extends to Scannet-V2 with ”cab,” ”shower,” ”bath,” and ”sofa” improvements. A comprehensive breakdown of per-class performance in Scannet-V2 and STPLS3D is available in Supplementary Section 1.

5. Conclusion

In conclusion, Point Central Transformer demonstrates the effectiveness of transformer networks with central-based attention for weakly-supervised point cloud segmentation. Our approach achieves impressive performance purely based on 3D points without additional supervision. Through central-based attention, we effectively handle intricate relationships between central points and neighborhoods using two embedding processes with appropriate global features. Geometric features are then improved using position encoding module. Extensive experiments validate PointCT's ability to capture global context and mitigate noises in weakly-supervised point clouds.

Acknowledgement This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2023-2020-0-01797) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation) and the MSIT (Ministry of Science and ICT), Korea, under the ICT Consilience Creative program (IITP-2023-2016-0-00318) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1534–1543, 2016. 5

[2] Meida Chen, Qingyong Hu, Hugues Thomas, Andrew Feng, Yu Hou, Kyle McCullough, and Lucio Soibelman. Stpls3d: A large-scale synthetic and real aerial photogrammetry 3d point cloud dataset. In *British Machine Vision Conference*, 2022. 6

[3] Mingmei Cheng, Le Hui, Jin Xie, and Jian Yang. Sspc-net: Semi-supervised semantic 3d point cloud segmentation network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 1140–1147, 2021. 2

[4] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3075–3084, 2019. 1, 7

[5] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 6

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 1

[7] Siqi Fan, Qiulei Dong, Fenghua Zhu, Yisheng Lv, Peijun Ye, and Fei-Yue Wang. Scf-net: Learning spatial contextual features for large-scale point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14504–14513, 2021. 1, 7

[8] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15587–15597, 2021. 1, 3

[9] Qingyong Hu, Bo Yang, Guangchi Fang, Yulan Guo, Aleš Leonardis, Niki Trigoni, and Andrew Markham. Sqn: Weakly-supervised semantic segmentation of large-scale 3d point clouds. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVII*, pages 600–619. Springer, 2022. 1, 3, 4, 6, 7

[10] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11108–11117, 2020. 1, 2, 3, 5, 6, 7

[11] Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3d point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8500–8509, 2022. 1, 2, 3, 5

[12] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. 2

[13] Min Seok Lee, Seok Woo Yang, and Sung Won Han. Gaia: Graphical information gain based attention network for weakly supervised point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 582–591, 2023. 1, 2

[14] Mengtian Li, Yuan Xie, Yunhang Shen, Bo Ke, Ruizhi Qiao, Bo Ren, Shaohui Lin, and Lizhuang Ma. Hybridcr: Weakly-supervised 3d point cloud semantic segmentation via hybrid contrastive regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14930–14939, 2022. 1, 2, 3, 4, 6, 7

[15] Rong Li, Anh-Quan Cao, and Raoul de Charette. Class-prototypes for contrastive learning in weakly-supervised 3d point cloud segmentation. In *British Machine Vision Conference*, 2022. 2

[16] Minghua Liu, Yin Zhou, Charles R Qi, Boqing Gong, Hao Su, and Dragomir Anguelov. Less: Label-efficient semantic segmentation for lidar point clouds. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIX*, pages 70–89. Springer, 2022. 1, 2

[17] Zhengzhe Liu, Xiaojuan Qi, and Chi-Wing Fu. One thing one click: A self-training approach for weakly supervised 3d semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1726–1736, 2021. 1, 2

[18] Jilin Mei and Huijing Zhao. Incorporating human domain knowledge in 3-d lidar-based semantic segmentation. *IEEE Transactions on Intelligent Vehicles*, 5(2):178–187, 2019. 1, 3

[19] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2

[20] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 1, 2, 3, 5, 7

[21] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in Neural Information Processing Systems*, 35:23192–23204, 2022. 2

[22] Zhongzheng Ren, Ishan Misra, Alexander G Schwing, and Rohit Girdhar. 3d spatial recognition without spatially labeled 3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13204–13213, 2021. 2

[23] Hanyu Shi, Jiacheng Wei, Ruibo Li, Fayao Liu, and Guosheng Lin. Weakly supervised segmentation on outdoor 4d point clouds with temporal matching and spatial graph

propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11840–11849, 2022. 1, 2

[24] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017. 2

[25] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6411–6420, 2019. 1, 2, 7

[26] Jiacheng Wei, Guosheng Lin, Kim-Hui Yap, Tzu-Yi Hung, and Lihua Xie. Multi-path region mining for weakly supervised 3d semantic segmentation on point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4384–4393, 2020. 1, 2, 4

[27] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point transformer v2: Grouped vector attention and partition-based pooling. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 3, 5

[28] Zhonghua Wu, Yicheng Wu, Guosheng Lin, Jianfei Cai, and Chen Qian. Dual adaptive transformations for weakly supervised point cloud segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI*, pages 78–96. Springer, 2022. 2

[29] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. Pointcontrast: Unsupervised pretraining for 3d point cloud understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 574–591. Springer, 2020. 1, 3

[30] Xun Xu and Gim Hee Lee. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13706–13715, 2020. 1, 2, 3, 6

[31] Cheng-Kun Yang, Ji-Jia Wu, Kai-Syun Chen, Yung-Yu Chuang, and Yen-Yu Lin. An mil-derived transformer for weakly supervised point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11830–11839, 2022. 1, 3

[32] Yachao Zhang, Zonghao Li, Yuan Xie, Yanyun Qu, Cuihua Li, and Tao Mei. Weakly supervised semantic segmentation for large-scale point cloud. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3421–3429, 2021. 1, 3, 6, 7

[33] Yachao Zhang, Yanyun Qu, Yuan Xie, Zonghao Li, Shanshan Zheng, and Cuihua Li. Perturbed self-distillation: Weakly supervised large-scale point cloud semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15520–15528, 2021. 1, 2, 3, 6, 7

[34] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 1, 2, 3, 4, 5, 6, 7