# Weakly Supervised Learning Method for Semantic Segmentation of Large-Scale 3D Point Cloud Based on Transformers

Zhaoning Zhang[1,†], Tengfei Wang[2,†] , Xin Wang[2,*], Zongqian Zhan[2]

[1]Faculty of Geosciences and Environmental Engineering, SWJTU,Chengdu 611756, China-2020114062@my.swjtu.edu.cn
[2]School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China
tf.wang@whu.edu.cn, (xwang,zqzhan)@sgg.whu.edu.cn

**Keywords:** Weakly Supervised Learning, 3D Point cloud, Semantic Segmentation, Transformer.

**Abstract:**

Nowadays, semantic segmentation results of 3D point cloud have been widely applied in the fields of robotics, autonomous driving, and augmented reality etc. Thanks to the development of relevant deep learning models (such as PointNet), supervised training methods have become hotspot, in which two common limitations exists: inferior feature representation of 3D points and massive annotations. To improve 3D point feature, inspired by the idea of transformer, we employ a so-call LCP network that extracts better feature by investigating attentions between target 3D points and its corresponding local neighbors via local context propagation. Training transformer-based network needs amount of training samples, which itself is a labor-intensive, costly and error-prone work, therefore, this work proposes a weakly supervised framework, in particular, pseudo-labels are estimated based on the feature distances between unlabeled points and prototypes, which are calculated based on labeled data. The extensive experimental results show that, the proposed PL-LCP can yield considerable results (67.6% mIOU for indoor and 67.3% for outdoor) even if only using 1% real labels, and comparing to several state-of-the-art method using all labels, we achieve superior results in mIOU, OA for indoor (65.9%, 89.2%).

## 1. Introduction

Semantic segmentation is a key technique to assign a semantic label to each individual point in a point cloud. This technology is widely used in areas such as autonomous driving, augmented reality and 3D reconstruction. Traditional semantic segmentation methods such as Ransac (Jung, 2014), regional growth (Wang, 2015) and other methods are difficult to adapt to complex scenes. Emerging semantic segmentation methods (Zhao, 2021; Xu, 2020; Milioto, 2019) based on deep learning can process point clouds in multiple scenarios more accurately by learning the characteristics of supervised point cloud data. However, the large demand for supervised data and the difficulty of learning local features of point cloud are still unsolved problems.

The Transformer has achieved remarkable success in various fields such as natural language processing (NLP) (Vaswani, 2017; Wu, 2019; Devlin, 2018) and 2D image processing (Zhao, 2020; Ramachandran, 2019; Hu, 2019). In the domain of point cloud semantic segmentation, it plays a critical role in leveraging contextual features. The use of the Transformer (Zhao, 2021; Guo, 2021) has shown potential in capturing crucial features, thanks to its fundamental attention mechanism and the ability to capture long-range dependencies. This makes it a reasonable choice for handling unstructured and unordered point cloud data. However, the inherent limitation of the Transformer network is the lack of integration of local information, which has been described as a drawback (Liu, 2021). LCPFormer (Huang, 2023) introduces a simple and effective module called LCP (Local Context Propagation) to facilitate message passing between adjacent local regions. Specifically, it leverages information exchange between neighboring local regions to provide each local region with more informative and discriminative features. The goal of this method is to enhance the capability of Transformer in integrating local information by integrating relational information between adjacent local regions, thereby improving the performance of point cloud semantic segmentation networks. Furthermore, most existing large-scale point cloud datasets heavily rely on manually annotating each point, which is a labor-intensive, expensive, and error-prone task. Transformer architecture requires extensive datasets for training, which would incur significant costs. Weakly supervised training provides an effective solution to reduce annotation costs. Pseudo-labeling techniques (Lee, 2013) are a method of leveraging unlabeled data in weakly supervised training. Initially, the model (Zhang, 2021) is trained using a small amount of labeled data. Then, the trained model is used to predict the unlabeled data, and these predictions serve as pseudo-labels. Subsequently, the model is trained using a combination of labeled and unlabeled data, incorporating both the real labels and the pseudo-labels. The model's performance is evaluated on the test set. In addition, entropy regularization loss (Grandvalet, 2006; Shannon, 1948) and distribution alignment loss (Zhang, 2021; Saito, 2019) have been introduced in 3D segmentation tasks for weakly supervised learning. These techniques aim to utilize the information from all unlabeled points by mitigating the negative impact of pseudo-label noise and addressing distribution discrepancies (Li, 2023).

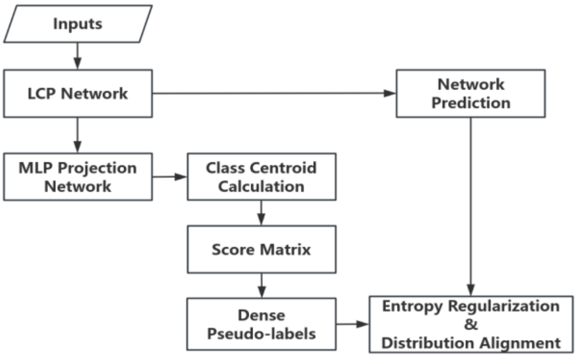

Figure 1. Flowchart of the proposed method

Overall, this paper combines weakly supervised learning with a Transformer framework with LCP to achieve superior results compared to other point cloud semantic segmentation methods such as RandLA-Net (Hu, 2020), even with limited annotations.

[†] These two authors make equal contributions.

[*] Corresponding author.

https://doi.org/10.5194/isprs-archives-XLVIII-1-2024-887-2024 | © Author(s) 2024. CC BY 4.0 License.

The methodology and workflow of our approach are illustrated in Figure 1. We begin by feeding the point cloud into an LCP network to predict the initial semantic information of the point cloud. Next, we employ a prototype pseudo-label generation strategy based on momentum (Xu, 2020) to generate pseudo-labels for unlabeled points. These pseudo-labels, along with the predicted results, are optimized using a loss function. Our main contributions are threefold:

1. Propose a novel PL-LCP framework that combines pseudo-labeling with Transformer, achieving good performance even with limited training samples.
2. Entropy regularization loss and distribution alignment loss are incorporated into pseudo-label generation, allowing for better utilization of information from all unlabeled points.
3. We employ varying degrees of labeling on the original data: 1%, 10%, and full, to investigate the performance of this framework with limited labelled 3D points.

## 2. Related Work

### 2.1 Semantic segmentation for large-scale point cloud

Point cloud semantic segmentation aims to assign semantic labels to 3D points. Early research in the field of point cloud segmentation witnessed numerous traditional segmentation methods, which achieved certain effectiveness. However, they were mostly constrained by specific scenarios and prior knowledge, rendering them less applicable across diverse contexts and often time-consuming. With the rapid advancement of deep learning in recent years, the focus of research on point cloud semantic segmentation has shifted towards methods primarily based on deep learning. Compared to the earlier traditional methods, these approaches have seen significant improvements in segmentation accuracy. State-of-art deep learning methods on point cloud can be categorized as projection-based, voxel-based, and point-based methods which are outlined here.

**2.1.1 Projection-based methods:** Projection-based methods benefit from mature 2D convolutional neural networks. These methods project 3D point clouds onto 2D images, then utilize existing 2D image segmentation methods to run customized image projection for identifying class labels. After labeling each point, the 2D images are projected back to their respective point clouds. In this process, the quality of point cloud projection determines the final segmentation outcome. Tatarchenko (2018) introduces tangent convolutions for dense point cloud segmentation. The method first projects the local surface geometry around each point onto a virtual tangent plane. Then, it directly performs tangent convolution operations on the surface geometry. This approach exhibits excellent scalability, capable of handling large-scale point clouds with millions of points. Zuo et al. (2021) proposes transforming three-dimensional point clouds into dense bird's-eye view projections and designs an attention-based fusion network for multi-modal learning of the projected images. The segmentation task is simplified due to the reduction of class imbalance and the feasibility of utilizing various 2D segmentation methods. Milioto (2019) proposes a real-time semantic segmentation method for LiDAR point clouds based on RangeNet++. Initially, it converts the semantic labels of 2D depth images to 3D point clouds, further utilizing KNN post-processing steps to mitigate issues related to discretization errors and blurry inference outputs. The performance of projection-based segmentation methods is sensitive to viewpoint selection and occlusion. Additionally, due to the inevitable information loss introduced by the projection step, these methods do not fully exploit the potential geometric and structural information.

**2.1.2 Voxel-based methods:** Voxel-based methods involve decomposing the entire point cloud into 3D regular cubic elements called voxels or volumetric elements, and applying 3D Convolutional Neural Networks (CNNs) on each voxel. Voxelization is the process of discretizing continuous three-dimensional space into a finite number of cubic units (referred to as voxels). In voxel-based semantic segmentation, the point cloud is partitioned into a uniform cubic grid, where each cubic unit represents a voxel in space. These voxels typically have the same size, allowing the point cloud to be represented and processed in a regular manner, such as with Convolutional Neural Networks (CNNs) or their variants, performing classification tasks on each voxel. Huang er al. (2016) divides the point cloud into a set of occupancy voxels, then feeds these intermediate data into a fully 3D CNN for voxel-wise segmentation. Finally, all points within a voxel are assigned the same semantic label as that voxel. In general, voxelization naturally preserves the neighborhood structure of three-dimensional point clouds. Its regular data format also allows for the direct application of standard 3D convolutions. However, voxelization inevitably introduces discretization artifacts and information loss, with high resolutions leading to high memory and computational costs, while low resolutions result in loss of detail. In practical applications, it's challenging to choose the appropriate grid resolution.

**2.1.3 Point-based methods:** Point-based methods directly work on the unstructured and irregular point clouds. These methods directly interact with points, taking individual points as input and outputting a labeled point or labeling the entire point cloud. PointNet (Qi, 2017) is a pioneering work and breakthrough that opened deep learning for direct work with points without rendering them to voxels or 2D images. PointNet used the max-pooling function with each layer in the network by learning an optimization function and aggregating the optimized values to a global descriptor. The final fully connected layers of the network aggregate these learnt optimal values into the global descriptor for the entire shape or are used to predict per point labels. Hu er al. (2020) proposes an efficient lightweight network, RandLA-Net, for large-scale point cloud segmentation. This network utilizes random point sampling and achieves very high efficiency in terms of memory and computation. Furthermore, it introduces the Local Feature Aggregation module to capture and retain geometric features. Pointvector (Deng, 2023) proposes a Vectororiented Point Set Abstraction that can aggregate neighboring features through higher-dimensional vectors. To facilitate network optimization, it constructs a transformation from scalar to vector using independent angles based on 3D vector rotations.

### 2.2 Transformers in 3D Point Clouds

Self-attention networks have revolutionized natural language processing and are making impressive strides in image analysis tasks such as image classification and object detection. The transformer models are especially suited for point cloud

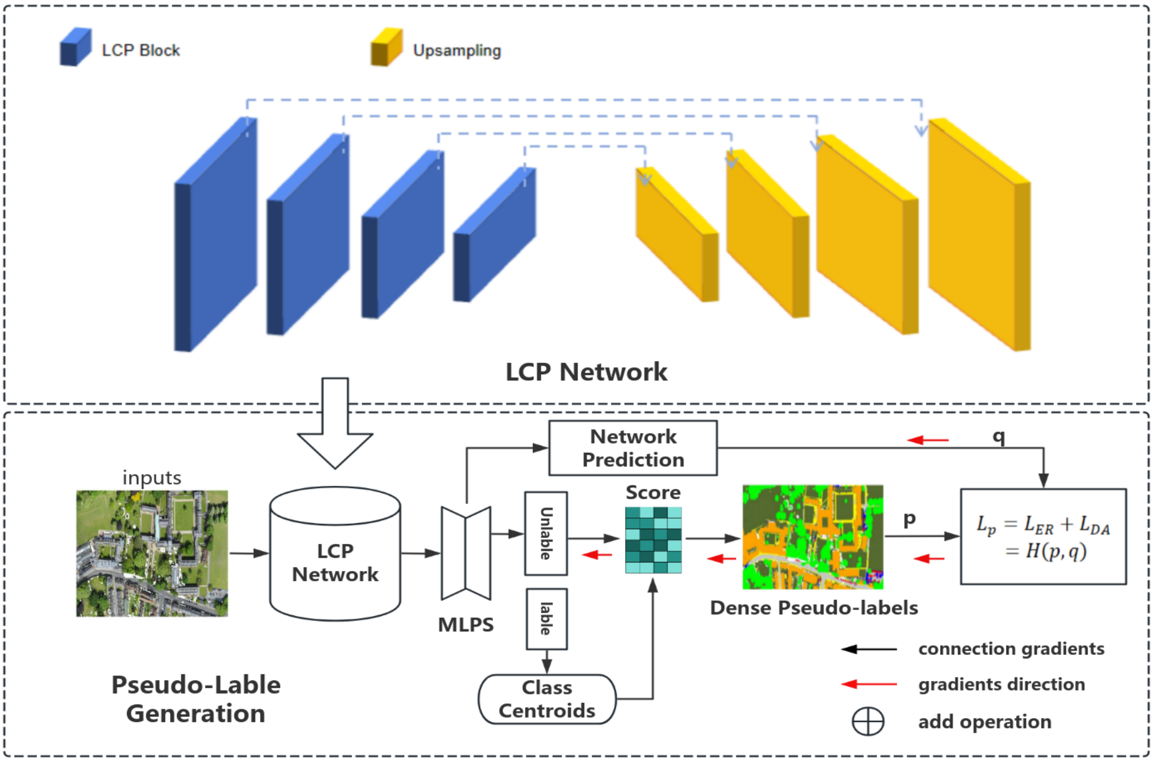

Figure 2. Overall architecture of our 3D vector rotations

processing because the self-attention operator, which is fundamental to transformer architectures, is in essence a set operator: it is invariant to permutation and cardinality of the input elements. The application of self-attention to 3D point clouds is therefore quite natural, since point clouds are essentially sets embedded in 3D space. Currently, many transformer-based methods for 3D point clouds have been proposed to produce more precise detection results. Robert, er al. (2023) proposed a novel transformer architecture based on superpoint. Specifically, the three-dimensional point cloud is first partitioned into a hierarchical superpoint structure, which adapts to local attributes collected at multiple scales. To efficiently compute this partitioning, a novel algorithm is introduced, which is an order of magnitude faster than existing superpoint preprocessing algorithms. Next, the Superpoint Transformer (SPT) architecture is introduced, which employs sparse self-attention mechanisms to learn relationships between superpoints at multiple scales. By treating semantic segmentation of large-scale point clouds as classification of a small number of superpoints, the model can accurately classify millions of 3D points. Stratified Transformer (Lai, 2022) proposed a hierarchical transformation to capture long-range contexts effectively, which has strong generalization ability and high performance. For each query point, it densely samples neighbor points in a hierarchical manner and sparsely samples distant points as its keys, enabling the model to expand its effective receptive field at a lower computational cost and leverage long-range contexts. Additionally, Stratified Transformer utilizes first-layer point embedding to aggregate local information, enhancing convergence and performance, and employs context-relative positional encoding to adaptively capture positional information.

### 2.3 Weakly-supervised point cloud segmentation

Existing large-scale point cloud semantic segmentation methods require expensive, laborious, and error-prone manual point-wise annotations. Intuitively, weakly supervised training is a direct solution to reducing annotation costs. Generally, weakly-supervised point cloud segmentation tasks focus on sparsely labeled data: only a small number of scattered points are annotated in large point cloud scenes. There are few researches on weakly supervised point cloud semantic segmentation. Su, er al. (2023) proposed a multi-prototype classifier, abbreviated as MulPro, for weakly supervised 3D point cloud segmentation. Specifically, a multi-template memory bank is designed to store prototypes for each semantic class, where each prototype represents a subclass. In contrast to K-means clustering or offline prototype updates using sliding average, this design avoids introducing any non-differentiable operations between prototypes and loss functions, enabling end-to-end training. Additionally, a subclass-wise averaging constraint is introduced to supervise prototype learning using labeled and unlabeled data, akin to nesting K-means clustering within the classifier. Pan, er al. (2024) proposed a novel framework for weakly supervised semantic segmentation of point clouds, which includes three stages: inductive bias learning, recommendation for annotated points, and weakly supervised point cloud semantic segmentation learning. Specifically, the framework begins by introducing a point cloud upsampling task to induce inductive bias from structural information. In the recommendation stage, a cross-scene clustering strategy is proposed to generate cluster centers as recommendation points. Subsequently, a recommendation point positional attention module, LabelAttention, is introduced to model long-range dependency relationships under sparse annotation.

## 3. Methodology

This paper aims to leverage pseudo-label generation techniques to implement a point cloud segmentation network trained with a small amount of annotated points based on Transformer. Furthermore, aiming to enhance the Transformer's ability to integrate information across adjacent local regions, the LCP module is added to the Transformer. We propose an effective weakly supervised framework based on Transformer, and the overview of the framework is illustrated in Figure 2.

Our approach combines a Transformer network with LCP (Local Context Perception) modules and pseudo-label generation techniques to achieve better semantic segmentation results with only a small amount of real annotations. In Section 3.1, we provide a brief introduction to the Transformer network. Next, in Section 3.2, we present a detailed description of the LCP module's structure and analysis its principle of integrating overlapping regions. Finally, in Section 3.3, we extensively discuss the process of generating pseudo-labels and the reasons for introducing entropy regularization loss and distribution alignment loss.

### 3.1 Preliminary

The transformer consists of an encoder and a decoder. The encoder is responsible for transforming the input sequence into a series of hidden representations, while the decoder generates the target sequence based on the outputs from the encoder and the previous decoder states. Each encoder and decoder layer consists of multiple identical sub-layers, including multi-head self-attention mechanism (MHSA) and feed-forward neural network (FFN). First, we review the commonly used MHSA, which aims to enable the model to simultaneously attend to different parts of the inputs with different representations. By performing multiple attention operations in parallel, each attention head can learn different semantic information. Given a set of point cloud $X = \{x_i\}$,we denote its positions $P = \{p_i\}$ and corresponding features $F = \{f_i\}$. It can be formulated as follows:

$$F_n = \mathrm{PE}(P) + F \tag{1}$$

$$Q_s = F_n {W_s}^Q, K_s = F_n {W_s}^K, V_s = F_n {W_s}^V \tag{2}$$

$$head_s = \mathrm{Softmax}(\frac{QK^T}{\sqrt{d}})V \tag{3}$$

$$output = \mathrm{concat}(head_1, head_2, \dots, head_s) * W^O \tag{4}$$

where PE() means the position encoding function. $W_s^Q$,$W_s^K$,$W_s^V$ are projections of the $s$-th head for query, key and value respectively. $d$ is the feature dimension. $W^O$ is the projection of The FFN is a fundamental structure in neural networks. The transformation layer consists of MHSA and FFN with skip connections.

$$Y = \mathrm{MHSA}(X, F) + F \tag{5}$$

$$O = Y + \mathrm{FNN}(Y) \tag{6}$$

### 3.2 Local Context Propagation

While transformers primarily emphasize long-range dependencies, local structural information remains crucial in a transformer-based 3D point cloud model. To enable transformers to incorporate such local structural information, we utilize the LCPFormer (Huang, 2023) architecture as the backbone for a weakly supervised semantic segmentation network. LCPFormer is based on a simple observation that when dividing the whole point cloud into different local regions, naturally there is overlap among them. It works by updating point features in overlapping areas of different regions. Given a point $x_i$, we denote its corresponding local regions as $\{S_1, ..., S_m\}$. After the Transformer independently operates on these regions, for each local region $S_j$, point $x_i$ should possess corresponding features within it, denoted as $f_i^j$. To obtain features for each local region, LCP combines the results of max pooling and mean pooling. Max pooling captures important features, while mean pooling captures features from the surrounding area. We assuming that the whole point cloud contains N points grouped into C local regions. The input is $F_{\mathrm{in}} \in R^{C\times K\times D}$, where K is the number of points in each local region and D is the feature dimension of each point. Then, we obtain representations $A \in R^{C\times 2D}$ for each local region through max pooling and mean pooling, followed by a 1x1 convolution to generate the weight matrix $W \in R^{C\times D}$ corresponding to each region. Finally, we update the feature of $\boldsymbol{x_i}$ by using the weight matrix. This process can be formulated as:

$$A = MaxPool(F_{in}) \oplus AvgPool(F_{in}) \tag{7}$$

$$W = \mathrm{Softmax}(\mathrm{Conv}(A)) \tag{8}$$

$${f_i}^{new} = \sum_j w_j f_i^j \tag{9}$$

Based on the preceding discussion, the LCP Block is constructed as illustrated in Figure 3. The LCP Block consists of a grouping layer, two MHSA, and an intermediate LCP module. The grouping layer utilizes FPS sampling to select central points, and for each central point, K nearest neighbors (kNN) are gathered within the local neighborhood to construct local regions. The k of kNN is set as 16.

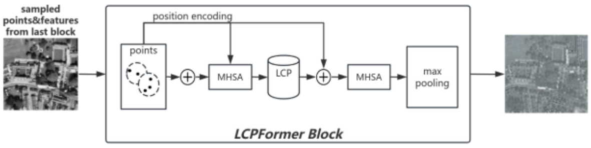

Figure 3. The specific content of the LCP Block.

### 3.3 Pseudo-Label Generation

We employ a momentum-based prototype pseudo-label generation process Xu er al. (2020). Specifically, prototypes denote the centroid of a class in the feature space, which is calculated based on labeled data, while pseudo-labels are estimated based on the feature distance between unlabeled points and class centroids. To reduce computational costs, we employ the momentum optimization algorithm for optimization, supplemented by an MLP-based projection network to aid in pseudo-label generation. The specific process is illustrated in Figure 2. We assume the input point cloud X, where the labeled point cloud is denoted as $X^l$, and the unlabeled point cloud as $X^u$. The pseudo-label generation process can be described as follows:

$$\hat{C}_k = \frac{1}{N_k^l}\sum_{x\in X^l \wedge y=k} g\circ f(x), C_k \leftarrow mC_k + (1-m)\hat{C}_k \tag{10}$$

$$\forall x \in X^u, s_k = d(g\circ f(x), C_k), p = \mathrm{Softmax}(s) \tag{11}$$

where y is the label of a labeled point, $C_k$ represents the global class centroid for the k-th class, $N_k^l$ denotes the number of labeled points of the k-th class, $g\circ f = g(f(\cdot))$ signifies the transformation through the backbone network f and the projection network $g$, $m$ represents the momentum coefficient, and cosine similarity is employed for $d(\cdot,\cdot)$to generate the scores. By default, we utilize 2-layer MLPs for the projection network $g$ and set $m$ = 0.999.

Existing pseudo-label generation methods often rely on empirical label selection strategies, such as confidence thresholds, to generate pseudo-labels that are beneficial for model training. This approach may potentially waste unlabeled points. In this paper, pseudo-labels are generated for all unlabeled points, and entropy regularization loss and distribution alignment loss are introduced to minimize the disparity between pseudo-labels and model predictions. We denote the two loss functions as $L_{ER}$ and $L_{DA}$, respectively (Li, 2023). Then, we have the overall loss of ERDA as follows:

$$L_P = \lambda L_{ER} + L_{DA} \tag{12}$$

where the $\lambda > 0$ modulates the entropy regularization.

For the entropy regularization loss, we posit that when pseudo-labels fail to provide reliable outcomes, they are more susceptible to noise interference, resulting in a high-entropy distribution within $\boldsymbol{p}$. To alleviate this issue, we propose minimizing its Shannon entropy to reduce the noise level in $\boldsymbol{p}$. By minimizing the entropy of pseudo-labels, we enhance their quality. Therefore, we have:

$$\boldsymbol{L_{ER}} = \boldsymbol{H}(\boldsymbol{p}) \tag{13}$$

where $\boldsymbol{H}(\boldsymbol{p}) = \sum_{\boldsymbol{i}} -\boldsymbol{p_i} \, \boldsymbol{log} \, \boldsymbol{p_i}$ and i iterates over the vector.

While the entropy regularization can mitigate the impact of noise in pseudo-labels, significant disparities between pseudo-labels and predictions from the segmentation network can still confound the learning process, leading to unreliable segmentation results. To address this issue, we propose a joint optimization approach for pseudo-labels and the network to narrow this gap, ensuring that generated pseudo-labels do not deviate too far from segmentation predictions. Therefore, we introduce the distribution alignment loss as follows:

$$\boldsymbol{L_{DA}} = \boldsymbol{KL}(\boldsymbol{p}||\boldsymbol{q}) \tag{14}$$

With the $\boldsymbol{L_{ER}}$ and $\boldsymbol{L_{DA}}$ formulated as above, given that $\boldsymbol{KL}(\boldsymbol{p}||\boldsymbol{q}) = \boldsymbol{H}(\boldsymbol{p},\boldsymbol{q}) - \boldsymbol{H}(\boldsymbol{p})$ where $\boldsymbol{H}(\boldsymbol{p},\boldsymbol{q})$ is the cross entropy between $\boldsymbol{p}$ and $\boldsymbol{q}$, we can have a simplified ERDA formulation as:

$$\boldsymbol{L_p} = \boldsymbol{H}(\boldsymbol{p},\boldsymbol{q}) + (\boldsymbol{\lambda} - \boldsymbol{1})\boldsymbol{H}(\boldsymbol{p}) \tag{15}$$

In particular, when $\boldsymbol{\lambda} = 1$, we obtain the final ERDA loss:

$$\boldsymbol{L_p} = \boldsymbol{H}(\boldsymbol{p},\boldsymbol{q}) = \sum_{\boldsymbol{i}} -\boldsymbol{p_i} \, \boldsymbol{log} \, \boldsymbol{q_i} \tag{16}$$

The above simplified ERDA loss differs from traditional cross-entropy loss. Traditional cross-entropy loss employs fixed labels and only optimizes the term within the logarithmic function, whereas the loss proposed above simultaneously optimizes both $p$ and $q$. Finally, with the above simplified ERDA, the final loss is given as:

$$L = \frac{1}{N^l} \sum_{x \in X^l} L_{ce}(q,y) + \alpha \frac{1}{N^u} \sum_{x \in X^u} L_p(q,p) \tag{17}$$

where $\boldsymbol{L_p}(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{L_{ce}}(\boldsymbol{q},\boldsymbol{p}) = \boldsymbol{H}(\boldsymbol{q},\boldsymbol{p})$ is the typical cross-entropy loss used for point cloud segmentation, $\boldsymbol{N^l}$ and $\boldsymbol{N^u}$ are the numbers of labeled and unlabeled points, and $\boldsymbol{\alpha}$ is the loss weight.

## 4. Experiments

To demonstrate the efficacy of our proposed PL-LCP, we evaluate 3D semantic segmentation results on both indoor and outdoor scenarios using two large-scale point cloud datasets. First, we do two ablation experiments to validate the ability of the LCP module to integrate inter-block information and the effect of pseudo-labels. Then, our method is compared with other relevant approaches, primarily to demonstrate the effectiveness of the PL-LCP network architecture. Our experimental environment is: Intel Core i7-8700 CPU (3.70GHz), 64GB RAM, NVIDIA GeForce RTX 4090 24GB GPU, 64-bit Ubuntu 22.04.3 LTS Operating System (5.4.0-149-generic). We trained the network for 200 epochs using the Adam optimizer with momentum, batch size and weight decay set to 0.9, 4 and 0.0001, respectively. The initial learning rate was set to 0.01, and decreased by a factor of 10 at 120 epochs.

### 4.1 Datasets

The S3DIS (Stanford Large-Scale 3D Indoor Spaces Dataset) is a vast collection of three-dimensional indoor space data provided by Stanford University (Armeni, 2016). It comprises six distinct indoor scenes, each containing three-dimensional reconstruction data. All points are labeled with their semantic ground truth from 13 categories including board, bookcase, chair, ceiling, beam, etc. SensatUrban (Hu, 2022) is an urban-scale photogrammetric point cloud dataset with nearly three billion richly annotated points, which is five times the number of labeled points than the existing largest point cloud dataset. Our dataset consists of large areas from two UK cities, covering about 6 km$^2$ of the city landscape. In SensatUrban, each 3D point is labeled as one of 13 semantic classes, such as ground, vegetation, car, etc.

### 4.2 LCP Network Architecture

We constructed a UNet-like (Ronneberger, 2015) network for semantic segmentation tasks using 4 LCP Blocks and 4 up-sampling layers, as depicted in Figure 2, as it requires per-point features for dense prediction. Before entering the first LCP Block, the data passes through a shared MLP. The specific structure of an LCP block is illustrated in Figure 3. For each up-sampling layer, we first employ the KNN algorithm to find the nearest neighbour point for each query point, and then perform up-sampling on the point feature set through nearest neighbour interpolation. Subsequently, the up-sampled feature maps are concatenated with the intermediate feature maps generated by the encoding layers via skip connections, followed by applying a shared MLP to the concatenated feature maps. The dimensions of each layer in the network are 128, 256, 512, and 1024, respectively. The input consists of 40960 points.

### 4.3 Evaluation Metrics

Taking into consideration simplicity and representativeness, this paper compares and analyzes various point cloud semantic segmentation methods using three evaluation metrics: Overall Accuracy (OA), mean accuracy (mAcc), and mean Intersection over Union (mIoU). For ease of description, we assume there are N semantic classes. $\boldsymbol{M}_{\mathrm{ij}}$ represents the number of units where the actual semantic type is i and the predicted type is j, and vice versa for $\boldsymbol{M}_{ji}$. $\boldsymbol{M}_{\mathrm{ii}}$ represents the number of units with both actual and predicted semantic type i.

OA is the ratio of the number of samples correctly predicted by the segmentation algorithms to the total number of samples. Its formulation is given as:

$$\mathrm{OA} = \frac{\sum_{\boldsymbol{i=0}}^{\boldsymbol{N}} \boldsymbol{M_{ii}}}{\sum_{\boldsymbol{i=0}}^{\boldsymbol{N}} \sum_{\boldsymbol{j=0}}^{\boldsymbol{N}} \boldsymbol{M_{ij}}} \tag{18}$$

mAcc represents an enhancement of OA, which computes the precision for each category individually and subsequently averages the accumulated results based on the number of categories. Its formulation is given as:

$$\mathrm{mAcc} = \frac{\boldsymbol{1}}{\boldsymbol{N+1}} \sum_{\boldsymbol{i=0}}^{\boldsymbol{N}} \frac{\boldsymbol{M_{ii}}}{\sum_{\boldsymbol{j=0}}^{\boldsymbol{N}} \boldsymbol{M_{ij}}} \tag{19}$$

mIOU stands out as the primary metric for evaluating segmentation methods' performance. It initially computes the intersection-over-union ratio for each category, reflecting the overlap between predicted and true regions of the models. Subsequently, it calculates the average value of the accumulated results based on the number of categories. Its formulation is given as:

$$\mathrm{mIOU} = \frac{1}{N+1}\sum_{i=0}^{N}\frac{M_{ii}}{\sum_{j=0}^{N}M_{ij}+\sum_{i=0}^{N}M_{ji}-M_{ii}} \qquad (20)$$

### 4.4 Experiment 1: Ablation study

| Part | LCP | OA(%) | mAcc(%) | mIOU(%) | Labels |
|---|---|---|---|---|---|
| *1* | √ | 90.2 | 74.3 | 67.6 | fully |
| *2* | × | 87.6 | 74.5 | 64.6 | fully |
| *3* | √ | 90.1 | 74.4 | 67.1 | 10% |
| *4* | √ | 89.2 | 73.2 | 65.9 | 1% |

Table 1. Ablation experiments on S3DIS

In this section, we conducted extensive ablation experiments to validate our approach, including the effectiveness of the proposed LCP module and the impact of varying degrees of ground truth annotations on PL-LCP. The results of the ablation experiments are presented in Table 1. We first examined the effectiveness of the LCP module. Through comparison, the LCP module resulted in improvements of 6.6%, 3.2%, and 9.3% on OA, mAcc, and mIOU, respectively, demonstrating the importance of integrating information across different blocks for semantic segmentation. Part 3 and part 4 respectively reduced the quantity of ground truth labels to 10% and 1% to assess the effectiveness of pseudo-labels, resulting in varying degrees of decrease compared to part 1. It is noteworthy that even with only 1% of ground truth labels, our approach achieved results that match or surpass those of some strong supervised networks. The results above demonstrate that the LCP module indeed enhances the network's local aggregation capability. The integration with weak supervision enables our network to perform remarkably well even with a limited amount of real annotations.

To visually evaluate the impact of our proposed PL-LCP, we randomly selected several point cloud scenes from the S3DIS dataset and visualized their output results, as shown in Figure 4. We present the detection results using the LCP module and compare them with the results obtained without using the LCP module, along with the ground truth labels. It can be observed that the results without the LCP module show relatively poorer handling of boundary regions, indicating that the separation sampling of local regions leads to the degradation of instance information. In contrast, our proposed LCP module effectively improves the point cloud features, providing richer information and more discriminative representations.

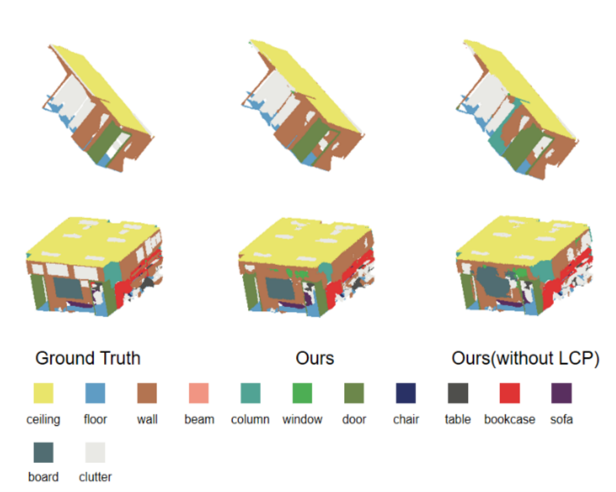

Figure 4. Visualization of segmentation results on S3DIS

| Methods | OA(%) | mIOU(%) | *ground* | *Veg.* | *buildings* | *walls* | *bridge* | *parking* | *rail* | *traffic* | *street* | *Cars* | *path* | *bikes* | *water* |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *PointNet* | 80.8 | 23.7 | 68.0 | 89.5 | 80.0 | 0.0 | 0.0 | 4.0 | 0.0 | 31.6 | 0.0 | 35.1 | 0.0 | 0.0 | 0.0 |
| *PointNet++* | 84.3 | 32.9 | 72.5 | 94.2 | 84.8 | 2.7 | 2.1 | 25.8 | 0.0 | 31.5 | 11.4 | 38.8 | 7.1 | 0.0 | 56.9 |
| *TrangenConv* | 77.0 | 33.3 | 71.5 | 91.4 | 75.9 | 35.2 | 0.0 | 45.3 | 0.0 | 26.7 | 19.2 | 67.6 | 0.0 | 0.0 | 0.0 |
| *SPGraphr* | 85,3 | 37.3 | 69.9 | 94.6 | 88.9 | 32.8 | 12.6 | 15.8 | 15.5 | 30.6 | 23.0 | 56.4 | 0.5 | 0.0 | 44.2 |
| *SparseConv* | 88.7 | 42.7 | 74.1 | 97.9 | 94.2 | 63.3 | 7.5 | 24.2 | 0.0 | 30.1 | 34.0 | 74.4 | 0.0 | 0.0 | 54.8 |
| *KPConv* | 93.2 | 57.6 | 87.1 | 98.9 | 95.3 | 74.4 | 28.7 | 41.4 | 0.0 | 56.0 | 54.4 | 85.7 | 40.4 | 0.0 | 86.3 |
| *RandLA-Net* | 89.8 | 52.7 | 80.1 | 98.1 | 91.6 | 48.9 | 40.8 | 51.6 | 0.0 | 56.7 | 33.2 | 80.1 | 32.6 | 0.0 | 71.3 |
| *PL-LCP（ours）* | **93.9** | **67.3** | **83.5** | **98.7** | **96.3** | **72.3** | **84.2** | **57.0** | **46.9** | **74.5** | **54.9** | **90.1** | **43.5** | **0.0** | **72.8** |

Table 2. Performance comparisons with existing sota methods on SensatUrban test set. overall accuracy (OA), mean IOU (mIOU), and per-class IOU scores are reported from the leaderboard of SensatUrban.

### 4.5 Experiment 2: Comparison with state-of-the art methods

The results on the S3DIS dataset are presented in Table 3. Our proposed method was evaluated by excluding Region 5 during training and using it for testing. Our method achieved an OA of 90.12%, mAcc of 74.3%, and mIOU of 67.14%.

| | OA(%) | mAcc(%) | mIOU(%) |
|---|---|---|---|
| *PointNet* | - | 23.7 | 41.1 |
| *TragenConv* | 82.5 | 63.2 | 52.8 |
| *SPGraph* | 86.4 | 66.5 | 58.0 |
| *LocalTransformer* | 87.6 | 71.9 | 64.1 |
| *RandLA-Net* | 87.2 | 71.4 | 62.4 |
| *PSNet* | 87.8 | - | 64.9 |
| *PL-LCP（ours）* | **90.2** | **74.3** | **67.6** |

Table 3. Performance comparisons with previous methods on S3DIS, evaluated on Region 5.

In Table 3, it is evident that our method outperforms certain non-Transformer architectures, such as RandLA-Net(Hu, 2020), which achieves an OA of 87.2%, mAcc of 71.4%, and mIOU of 62.4%. Our method surpasses LocalTransformer by 2.6%, 3.6%, and 3.5% in terms of OA, mAcc, and mIOU respectively, demonstrating the efficacy of our LCP module in integrating local information.

As shown in Table 2, we also evaluated our method on the challenging urban-scale segmentation dataset SensatUrban, achieving significant improvements. Compared to the previously popular methods KPConv and renowned RandLA-Net, our method demonstrated an increase in mIOU by approximately 9.7% and 14.6%, respectively. Particularly in smaller categories, our approach showcased remarkable potential. For instance, we achieved 46.9% in the railway category and 84.2% in the bridge category. Across all categories, our method decisively outperformed other approaches.

## 5. Conclusion

This work explores the combination of Transformers and weak supervision for 3D point cloud semantic segmentation, emphasizing the integration of semantic information across local regions. We introduce a novel and effective weakly supervised network, PL-LCP. In contrast to previous approaches, our method not only exploits the Transformer's capability to process sequential data but also addresses the challenge of the Transformer architecture's reliance on extensive datasets for training. Moreover, by introducing the LCP module, we effectively mitigate the issue of Transformers solely focusing on long-range dependencies while neglecting structural information. Ultimately, our proposed approach demonstrates significant improvements compared to various relevant methods and Transformer-based approaches in the dense prediction task for semantic segmentation on the S3DIS dataset.

In this paper, we combine the Transformer with an LCP module to form a network that achieves promising results even with limited true annotations. Subsequent experiments will involve the use of a custom dataset to further validate the effectiveness of our approach. Additionally, we plan to make some improvements, such as replacing precise nearest neighbour search with efficient serialized neighbour mapping organized according to specific patterns of point clouds. This enhancement will significantly increase the receptive field while accelerating processing speed and runtime efficiency, thereby enhancing its performance on outdoor point cloud datasets.

## Acknowledgement

This work was jointly supported Natural Science Foundation of Hubei Province，China (2022CFB727) and National Natural Science Foundation of China (42301507).

## References

Armeni, I., Sener, O., Zamir, A. R., et al., 2016. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1534-1543.*

Boulch, Alexandre, et al., 2018. "SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks." In *Computers & Graphics 71: 189-198*.

Cortinhal, T., Tzelepis, G., Aksoy, E. E., 2020. Salsanext: Fast semantic segmentation of lidar point clouds for autonomous driving. *arXiv preprint arXiv:2003.03653, 2020, 3(7): 2*.

Devlin, J., Chang, M. W., Lee, K., et al., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Deng, X., Zhang, W. Y., Ding, Q., et al., 2023. Pointvector: a vector representation in point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9455-9465.*

Feng, Y., Zhang, Z., Zhao, X., et al., 2018. Group-view convolutional neural networks for 3D shape recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 264-272.*

Grandvalet, Y., Bengio, Y., 2006. Entropy regularization.

Guo, M. H., Cai, J. X., Liu, Z. N., et al., 2021. Pct: Point cloud transformer. *Computational Visual Media 7: 187-199.*

Guo, Y., Wang, H., Hu, Q., et al., 2020. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence, 43(12): 4338-4364.*

Hu, H., Zhang, Z., Xie, Z., et al., 2019. Local relation networks for image recognition. In *Proceedings of the IEEE/CVF international conference on computer vision. pp. 3464-3473.*

Hu, Q., Yang, B., Khalid, S., et al., 2021. Towards semantic segmentation of urban-scale 3D point clouds: A dataset, benchmarks and challenges. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4977-4987.*

Hu, Q., Yang, B., Khalid, S., et al., 2022. Sensaturban: Learning semantics from urban-scale photogrammetric point clouds. *International Journal of Computer Vision, 130(2): 316-343.*

Hu, Q., Yang, B., Xie, L., et al., 2020. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11108-11117.*

Huang, J., You, S., 2016. Point cloud labeling using 3d convolutional neural network. In *2016 23rd International Conference on Pattern Recognition (ICPR). pp. 2670-2675*

Huang, Z., Zhao, Z., Li, B., et al., 2023. Lcpformer: Towards effective 3d point cloud analysis via local context propagation in transformers. *IEEE Transactions on Circuits and Systems for Video Technology. pp. 4985-4996*

Jung. J., Hong. S., Jeong. S., et al., 2014. Productive modeling for development of as-built BIM of existing indoor structures. *Automation in Construction, 42: 68-77*

Lai, X., Liu, J., Jiang, L., et al., 2022. Stratified transformer for 3d point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8500-8509.*

Lee, D. H., 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML, 3(2): 896.*

Liu, Z., Lin, Y., Cao, Y., et al., 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012-10022.*

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431-3440.*

Maturana, Daniel, and Sebastian Scherer., 2015. "Voxnet: A 3d convolutional neural network for real-time object recognition." In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 922-928.*

Milioto, Andres, et al., 2019. "Rangenet++: Fast and accurate lidar semantic segmentation." In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE. pp. 4213-4220.*

Pan, Z., Zhang, N., Gao, W., et al., 2024. Less Is More: Label Recommendation for Weakly Supervised Point Cloud Semantic Segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence. pp. 38(5): 4397-4405.*

Qi, C. R., Su, H., Mo, K., et al., 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652-660.*

Qi, C. R., Yi, L., Su, H., et al., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems, 30.*

Qiu, S., Anwar, S., Barnes, N., 2021. Semantic segmentation for real point cloud scenes via bilateral augmentation and adaptive fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1757-1767.*

Ramachandran, P., Parmar, N., Vaswani, A., et al., 2019. Stand-alone self-attention in vision models. *Advances in neural information processing systems, 32.*

Robert, D., Vallet, B., Landrieu, L., 2022. Learning multi-view aggregation in the wild for large-scale 3d semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5575-5584.*

Robert, D., Raguet, H., Landrieu, L., 2023. Efficient 3d semantic segmentation with superpoint transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17195-17204*

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, proceedings, part III 18. Springer International Publishing, 2015: 234-241.*

Saito, K., Ushiku, Y., Harada, T., et al., 2019. Strong-weak distribution alignment for adaptive object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6956-6965.*

Shannon, C. E., 1948. A mathematical theory of communication. *The Bell system technical journal. pp.27(3): 379-423.*

Su, H., Maji, S., Kalogerakis, E., et al., 2015. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision. pp. 945-953.*

Su, H., Jampani, V., Sun, D., et al., 2018. Splatnet: Sparse lattice networks for point cloud processing. In *Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2530-2539.*

Su, Y., Xu, X., Jia, K., 2023. Weakly supervised 3d point cloud segmentation via multi-prototype learning. *IEEE Transactions on Circuits and Systems for Video Technology. pp. 7723-7736*

Tang, L., Chen, Z., Zhao, S., et al., 2024. All Points Matter: Entropy-Regularized Distribution Alignment for Weakly-supervised 3D Segmentation. *Advances in Neural Information Processing Systems, 36.*

Tatarchenko, M., Park, J., Koltun, V., et al, 2018. Tangent convolutions for dense prediction in 3d. In *Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3887-3896.*

Vaswani, Ashish, et al., 2017. "Attention is all you need." *Advances in neural information processing systems ,30.*

Wang, C., Cho, Y. K., Kim, C., 2015. Automatic BIM component extraction from point clouds of existing buildings for sustainability applications. *Automation in Construction, 56: 1-13.*

Wu, B., Zhou, X., Zhao, S., et al., 2019. Squeezesegv2: Improved model structure and unsupervised domain adaptation for road-object segmentation from a lidar point cloud. In *2019 international conference on robotics and automation (ICRA). IEEE. pp. 4376-4382.*

Wu, F., Fan, A., Baevski, A., et al., 2019. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430.*

Xie, X., Bai, L., Huang, X., 2021. Real-time LiDAR point cloud semantic segmentation for autonomous driving. *Electronics, 11(1), 11.*

Xu, X., Lee, G. H., 2020. Weakly supervised semantic point cloud segmentation: Towards 10x fewer labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13706-13715.*

Zhang, R., Wu, Y., Jin, W., et al., 2023. Deep-Learning-Based Point Cloud Semantic Segmentation: A Survey. *Electronics, 12(17): 3642.*

Zhang, S., Li, Z., Yan, S., et al., 2021. Distribution alignment: A unified framework for long-tail visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2361-2370.*

Zhang, Y., Li, Z., Xie, Y., et al., 2021. Weakly supervised semantic segmentation for large-scale point cloud. In *Proceedings of the AAAI Conference on Artificial Intelligence. pp. 3421-3429.*

Zhao, H., Jiang, L., Jia, J., et al., 2021. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision. pp. 16259-16268.*

Zhao, H., Shi, J., Qi, X., et al., 2017. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2881-2890.*

Zhao, H., Jia, J., Koltun, V., 2020. Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10076-10085*

Zou, Z., Li, Y., 2021. Efficient urban-scale point clouds segmentation with bev projection. *arXiv preprint arXiv:2109.09074, 2021.*