ICJK

Journal of Geoscience and Earth Observation
http://dx.doi.org/10.62762/JGEO.2026.751126

RESEARCH ARTICLE

KP-Attention Net: A Point Cloud Ground Filtering Algorithm Based on Deep Learning for Urban MLS Scenes

Chaoyang Bian[1], Xiang Kang[2], Binfu Ge[1], Zhiming Liu[1], Qiaofeng Zhang[2], Yuxin Ding[1], Yifei Wang[2] and Fang Huang[1,*]

[1] School of Resources and Environment, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China
[2] College of Geographical Science, Inner Mongolia Normal University, Hohhot 010022, China

Abstract

Existing point cloud ground filtering methods rely heavily on manually selected filtering thresholds and neighborhood settings, lose information, model global features inadequately, and, in some deep-learning-based algorithms, incur high computational cost. To address these issues, this study proposes KP-Attention Net, a ground filtering network that integrates kernel point convolution (KPConv) with a position-aware self-attention mechanism. KPConv applies convolutions directly in the point cloud space to extract local features while preserving the three-dimensional structure; KPConv-Strided kernels expand the receptive field and enhance terrain perception; and a residual structure improves training stability and deep feature extraction. The position-aware self-attention mechanism strengthens the modeling of global context and long-range dependencies, which helps identify the continuity of large-scale terrain and the distribution patterns of features. The network is trained end-to-end, which reduces reliance on scene-specific manual tuning of thresholds and neighborhood settings and improves adaptability to urban mobile laser scanning (MLS) scenes. On the Toronto-3D dataset, KP-Attention Net achieves 92.99% mIoU and 98.03% OA, outperforming the comparison methods. To evaluate cross-dataset generalization, the model trained on Toronto-3D is applied without fine-tuning to a self-collected dataset (BeiQing), where it obtains 90.72% mIoU and 90.79% OA, outperforming Patchwork++ and TerraSeg. Robustness tests show stable performance under noise and density reduction, and ablation experiments confirm the contribution of the position-aware self-attention mechanism. KP-Attention Net thus provides an accurate and robust end-to-end solution for ground filtering of urban MLS point clouds.

Keywords: point cloud processing, ground filtering, kernel point convolution (KPConv), deep learning.

Submitted: 22 January 2026
Revised: 24 July 2026
Accepted: 11 November 2026
Published: 27 November 2026

Vol. 2, **No.** 1, 2027.
10.62762/JGEO.2026.751126

***Corresponding author:**
✉ Fang Huang
hfhbhzp@uestc.edu.cn

Citation
Bian, C., Kang, X., Ge, B., Liu, Z., Zhang, Q., Ding, Y., Wang, Y., & Huang, F. (2026). KP-Attention Net: A Point Cloud Ground Filtering Algorithm Based on Deep Learning for Urban MLS Scenes. *Journal of Geoscience and Earth Observation*, 2(1), 1-22.

© 2026 by the Authors. Published by Institute of Central Computation and Knowledge. This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

1 Introduction

Point cloud ground filtering is an essential step in the processing of point clouds through light detection and ranging (LiDAR) technology. It is designed to precisely separate and extract the ground point cloud from the complex distribution of objects in representations of urban areas. Point cloud filtering can not only eliminate interference and preserve core data on the given terrain, but can also subsequently ensure the efficient and accurate processing of the data to significantly enhance their usability. Such processing provides foundational data support for subsequent applications, such as extracting roads from 3D point clouds, modeling buildings, and analyzing the urban terrain.

The quality of point cloud ground filtering directly determines the effectiveness of subsequent work. On the one hand, precise ground filtering eliminates non-ground interference while preserving the core geometric and topological relationships of the terrain. This forms the foundation for generating high-precision digital elevation models (DEM) and digital terrain models (DTM). On the other hand, a reliable ground point cloud provides essential data support for urban-level applications. This is critical for accurate decision-making in scenarios such as the development of smart cities and disaster monitoring. Particularly stringent demands are placed on the accuracy and robustness of ground filtering algorithms in specific complex scenes, e.g., densely populated urban areas or locally sloped road segments, where the spatial relationships between ground and non-ground targets are highly intertwined. Optimizing the ground filtering algorithm can not only improve the precision of point cloud separation in extreme environments, but can also overcome bottlenecks in the use of LiDAR technology in complex scenarios, thus expanding its scope of application.

However, current point cloud ground filtering methods are inhibited by the following limitations: (1) Traditional approaches require manually selecting the filtering thresholds and neighborhood settings, which makes them poor at adapting to the terrain and leads to significant misclassification. (2) To address these shortcomings of strong parameter dependence and poor terrain adaptability, researchers have incorporated deep learning techniques into the filtering algorithms. While deep learning-based methods demonstrate superior capabilities of feature learning to those of traditional methods, they also have notable drawbacks. For instance, two-dimensional (2D) methods of projection lose data on elevation and 3D structural information [1]. Voxel-based methods can preserve some spatial structure, but their computational complexity increases cubically with the resolution such that they are unsuitable for large-scale point clouds. Graph convolutional approaches involve constructing additional adjacency graphs that increase their computational overhead. Furthermore, current methods based on the multi-layer perceptron (MLP) struggle to fully capture long-range dependencies within the point cloud, and cannot leverage their spatial topology due to the absence of an efficient mechanism of local-global feature interaction. This restricts their ability to accurately identify features in areas where objects overlap.

To resolve the above issues, this study proposes a deep learning-based ground filtering network called kernel point-attention net (KP-Attention Net). By combining kernel point convolution (KPConv), the residual network (ResNet) architecture, skip connections, and position-aware self-attention mechanisms, KP-Attention Net has a high computational efficiency and robustness. This model delivers a precise and efficient solution for automated urban mobile laser scanning (MLS) point cloud filtering, and outperforms prevalent methods on the main evaluation metrics.

Compared with traditional methods, the major innovation of our proposed model lies in its capacity for efficient, end-to-end point cloud ground filtering through a multi-component collaborative design that involves three stages. First, the KPConv convolutional kernel is applied directly to the point cloud to avoid the loss of information while accurately capturing local geometric details and extracting local features. Second, the introduction of a position-aware self-attention mechanism compensates for the limited early receptive field of the KPConv kernel, and enables the network to model relationships between distant points. This design enhances the model's adaptability to complex scenes, such as densely populated urban blocks and locally discontinuous ground of curbs/ramps/steps. Finally, the integration of the ResNet architecture with skip connections not only alleviates the problem of the vanishing gradient during the training of the deep network to improve the stability of the model, but also achieves the fusion of shallow local features with deep global features to extract multi-scale spatial information from the point cloud. KP-Attention Net therefore preserves point information, adapts to complex urban scenes, and makes use of multi-scale features, which supports efficient and precise ground

filtering of urban MLS point clouds.

To systematically evaluate the proposed KP-Attention Net, this study conducts four sets of experiments. Experiment I presents the comparative experiments on the Toronto-3D dataset against traditional filtering methods and the deep-learning baseline kernel point fully convolutional network (KPFCNN) to validate the effectiveness of the approach. Experiment II presents the cross-dataset experiments on the self-collected dataset to evaluate generalization ability and compares it with recently advanced models, e.g., Patchwork++ and TerraSeg. Experiment III analyzes the model's robustness under Gaussian noise and point density variation. Experiment IV conducts ablation studies to verify the contribution of each core component. Through these experiments, the effectiveness, generalization, robustness, and modular contributions of KP-Attention Net are evaluated.

The remaining part of this study is organized as follows. Section 2 gives a brief introduction to the related work on point cloud ground filtering, including traditional filtering methods and deep learning based methods. We elaborate our methodology in Section 3, i.e., the implementation framework, process, and modeling of KP-Attention Net, including the design of the KPConv convolution kernel, introduction of the position-aware self-attention mechanism, integration of the residual network structure, and the end-to-end training strategy. Section 4 describes the experiments employed for testing the performance of the complete proposed method, including the performance comparison experiments, cross-dataset generalization analysis, robustness analysis, and the ablation experiments. Finally, the conclusions of this study and prospective research in the area are provided in Section 5.

2 Related Work

The primary objective of point cloud ground filtering is to isolate and extract ground point clouds from the complex distribution of the urban terrain. In recent years, researchers have published extensive and multi-faceted studies on ground filtering containing several important algorithms. We now briefly review both traditional and deep learning-based point cloud filtering algorithms.

Traditional methods for point cloud ground filtering primarily include slope-based, morphological, and surface-based strategies. The slope-based approach was introduced by Vosselman [2] and can be used to identify non-ground points by setting thresholds with a specific gradient. However, this method is limited to individual types of terrains. Susaki [3] proposed constructing adaptive thresholds for the terrain, based on the characteristics of its gradient, by adjusting the coarse DTM generated during hierarchical processing. Morphological methods use techniques of morphological processing to remove non-ground points. For instance, Kilian et al. [4] proposed a method that applies iterative morphological filtering by using multiple sets of windows of varying sizes. The results of filtering of each window are weighted according to its size, and the final classification of ground and non-ground points is obtained by the weighted composite of all windows. To address excessive erosion in morphological filtering, Zhang et al. [5] proposed a progressive morphological filter that gradually increases the window size and uses elevation-difference thresholds. Surface-based methods iteratively select points from raw point cloud data to construct ground surfaces by using triangular irregular networks (TIN) or interpolation techniques [6]. Novel approaches have also emerged in this context. For instance, Zhang et al. [7] pioneered a cloth simulation-based filtering algorithm. This method simulates physical processes by virtually draping a cloth over inverted point clouds to filter out non-ground points from them. To enhance the accuracy and applicability of the algorithm in complex terrains, Li et al. [8] optimized it by constructing a grid index via slope-based grid segmentation. They fitted the values of elevation by using the lowest points in neighboring grids, applied normalization before implementing the cloth filtering algorithm, and finally extracted the ground points through thresholding. However, these traditional methods have a limitation in common: researchers must comprehensively understand the internal mechanisms of the algorithm and select appropriate filtering thresholds, neighborhood settings, and other scene-dependent parameters based on the characteristics of the study area. Fixed parameter settings may thus prove particularly unsuitable for complex terrains over large areas.

With the rapid advancement of deep learning, a variety of neural network models have emerged in the field of point cloud processing [9, 10]. To reduce the loss of information, some researchers have explored voxel-based approaches. Schmohl and Sörgel [11] first divided an airborne laser scanning (ALS) point cloud into voxels, and then applied sparse convolutional networks to semantically

segment it to reduce the computational cost and memory consumption. Yotsumata et al. [12] applied ground filtering to voxelized ALS data by using a 3D CNN. A major issue when using voxel-based methods is that their computational complexity and memory requirements increase cubically as the spatial resolution improves. Moreover, Nurunnabi et al. [13] applied manually extracted features that were fed into a dense neural network to ground filter an ALS point cloud. However, this approach relies on manual feature extraction and may struggle to adapt to diverse datasets.

Furthermore, some researchers have explored methods of point convolution. Janssens-Coron and Guilbert [14] made preliminary attempts to directly process an ALS point cloud for ground filtering by using PointNet, while Jin et al. [15] proposed a point-based fully convolutional network (FCN) that can learn the features of the point cloud in block-wise and point-wise configurations. Moreover, Krisanski et al. [16] used an enhanced PointNet++ model to segment a point cloud of a forest acquired from multi-sensor systems. Unlike the MLP, point convolutional models can directly process point cloud data, which are aligned with the characteristics of 3D data.

Beyond point-based methods, some studies have explored applying graph convolutions to process outdoor point clouds. For instance, Huang et al. [17] converted an MLS point cloud into super-point graphs and applied graph convolutional neural networks to filter the ground points. Although graph convolutions perform well in capturing the topology of the point cloud, converting the point cloud into graph data requires additional computational resources. An alternative approach involves using point convolution operators. For instance, Soilán et al. [18] applied the KPConv [19] architecture to segment the point cloud of a railway tunnel acquired through mobile mapping systems. Moreover, Kada and Kuramin [20] used KPConv to segment urban ALS point clouds and used additional prior knowledge to improve performance. The related studies suggest that KPConv delivers superior performance to models like PointNet and PointNet++ in terms of processing outdoor point clouds.

Recent progress in deep-learning-based ground filtering/terrain extraction has increasingly shifted toward task-aware designs that can better balance large-scene scalability, micro-topography preservation, and context modeling. The OpenGF benchmark and evaluations highlight that even strong 3D deep neural networks can suffer from scalability constraints and landform-dependent generalization issues [21]. In response, several ground-filtering-specific pipelines have been proposed, such as Terrain-Net [22] (point convolution + self-attention) for forested unmanned aerial vehicle (UAV) LiDAR, sequential ground prediction for large-scale ALS [23], and plug-in attention modules, e.g., elevation offset-attention, that inject global semantics into point-based backbones for terrain filtering [24]. More recently, hybrid convolution-Transformer networks have been explored for complex terrains. For instance, Point-SCT is a multiscale spatial convolution-Swin Transformer network [25]. Meanwhile, voxel-centric learning has also been revisited, e.g., the multi-size voxel cube (MSVC) [26] to refine rugged-terrain details. Unlike these studies, our work focuses on urban MLS ground filtering and integrates the position-aware self-attention mechanism into a KPConv encoder-decoder pipeline to improve surface continuity under occlusions and near-ground clutter.

Specifically, the differences between our work and existing studies are mainly as follows: (1) We keep an end-to-end point-based pipeline built upon KPConv-style hierarchical feature learning, avoiding profile sequencing or voxel-cube multi-pass refinement; (2) We integrate the position-aware self-attention mechanism to enhance global perception while controlling computation by inserting it only at selected encoding stages; and (3) Our algorithm mainly targets urban MLS scenes that have large man-made structures and discontinuous ground, e.g., stairs, bridges, and intersections. These scenes require global context without sacrificing local terrain detail, which complements the recent focus on forested/mountainous ALS/UAV scenarios [21, 22, 25].

In summary, traditional methods of point cloud ground filtering have some applicative value under specific conditions. However, deep learning-based approaches are emerging as the new trend in research due to the sensitivity of traditional methods to scene-dependent parameter settings and their poor adaptability. Among deep learning-based methods, point convolutional models have exhibited strong learning capabilities, and have a high computational efficiency because they directly process point cloud data. They thus have significant potential for use in processing outdoor point clouds.

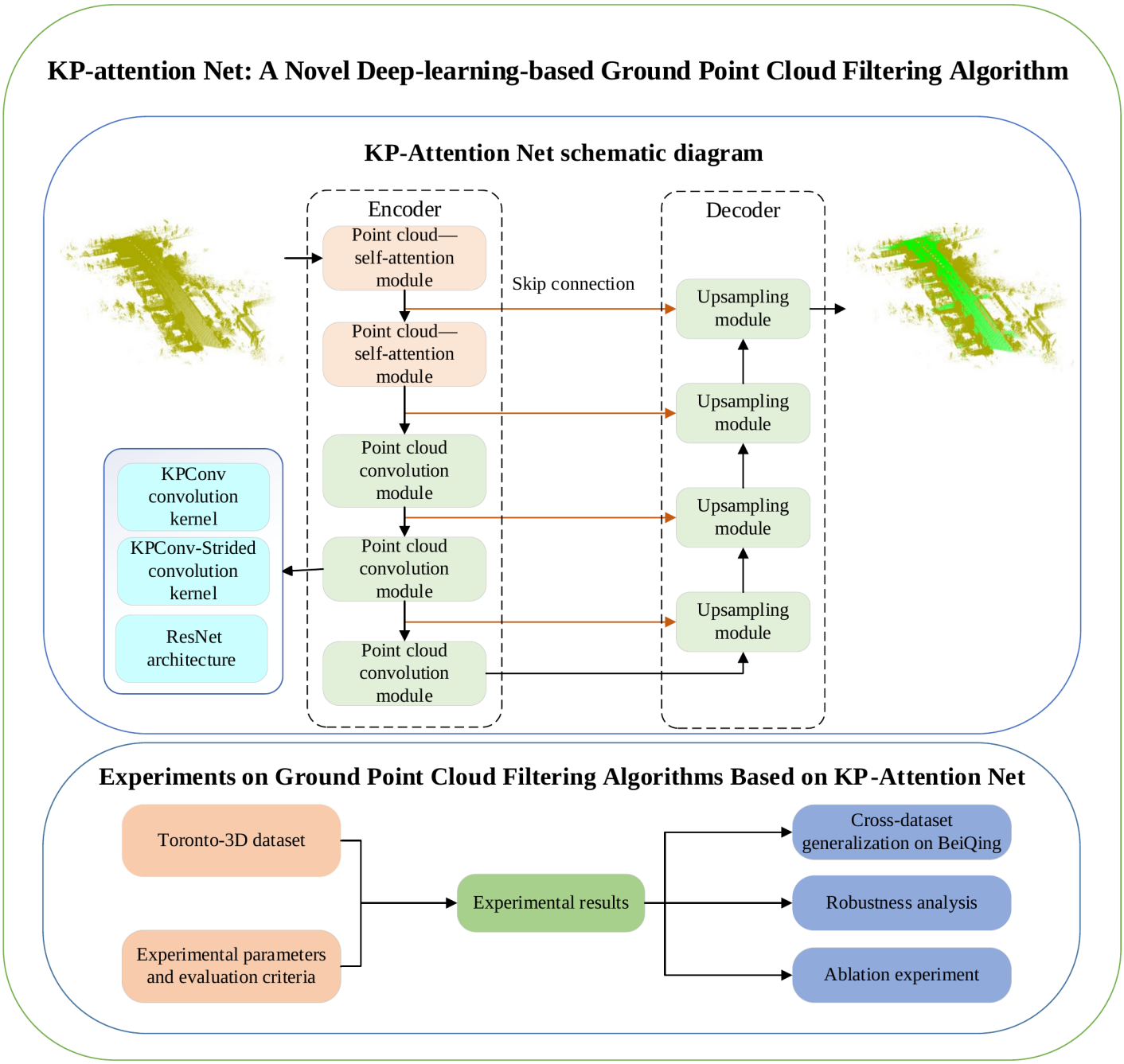


Figure 1. Overall approach of this study.

3 Methodology

KP-Attention Net adopts a KPFCNN-style KPConv encoder-decoder backbone and incorporates a local position-aware self-attention module. The attention is computed within radius-based neighborhoods, consistent with KPConv neighborhood construction, and is inserted at the encoder stage to propagate longer-range surface-consistency information. That is, the method integrates the position-aware self-attention mechanism into a KPConv pipeline in a task-driven manner for ground filtering.

3.1 Overall Approach

As summarized in Section 1, existing ground filtering methods depend strongly on manually selected thresholds and neighborhood settings, which limits their adaptability to urban MLS scenes with varying density and occlusion. Existing deep-learning-based alternatives either lose spatial information (2D projection), incur high cost (voxelization and graph construction), or fail to capture global information and the spatial topology of the point cloud (MLP-based methods).

To address the aforementioned challenges, we propose

KP-Attention Net. It is a ground filtering network that has been designed by using KPConv, the ResNet architecture, skip connections, and position-aware self-attention mechanisms. The network can enhance the capabilities of global perception of the model as well as its adaptability to the urban terrain while maintaining a high computational efficiency. Figure 2 details the network architecture, and Figure 1 outlines the workflow for experiments on urban MLS point cloud ground filtering by using KP-Attention Net.

Figure 2 shows that the KPConv convolution kernel extracts local features of the point cloud when processing it by performing convolution operations that leverage spatial information. It progressively expands the receptive field through a bottom-up approach, thereby learning deeper-level features. Moreover, to overcome the limitation of KPConv convolution kernels in capturing global features, we introduce a position-aware self-attention mechanism to enhance the expressive power of features of the point cloud by building on the standard version of the KPFCNN. We also use a ResNet architecture and skip connections to strengthen the propagation of deep information and ensure efficient gradient flow.

Figure 1 also illustrates the experimental validation framework of KP-Attention Net. The experiments are built upon the Toronto-3D dataset with unified hyperparameters and evaluation metrics. On this basis, the proposed method is systematically evaluated from four perspectives, i.e., effectiveness, generalization capability, robustness, and modular contributions.

KP-Attention Net primarily consists of two sets of components, i.e., an encoder and a decoder, and three modules: a point cloud convolution module, a point cloud position-aware self-attention module, and an upsampling and skip connection module. We now discuss them in detail. The overall architecture of the proposed network is shown in Figure 2.

3.2 Point Cloud Convolution Module

It is necessary to not only identify the geometric features of individual points in tasks of ground filtering, but also to distinguish between ground points and non-ground points in the overall environment. Although point cloud data are discrete, such features of the terrain as roads, plazas, and bridges in urban environments typically form continuous surfaces, while non-ground points like trees and vehicles have an irregular distribution. Consequently, relying solely on local geometric information is often insufficient for accurately determining the morphology of the ground, particularly in areas with significant changes in elevation, such as slopes, staircases, and complex intersections.

We use KPConv convolutional kernels to extract the local features and KPConv-Strided convolutional kernels to progressively expand the receptive field. This enables the network to perceive larger-scale terrain structures. We simultaneously integrate the ResNet architecture with the convolutional module to improve the propagation of deep information while mitigating the loss of features due to downsampling. Below, we analyze the KPConv and KPConv-Strided convolutional kernels as well as the ResNet architecture.

3.2.1 KPConv Convolution Kernel

KPConv enables the network to focus on the features of ground-level detail in the initial stage, including undulations in roads and variations in the gradient of the slope. This forms the foundation for subsequent global modeling. KPConv extracts features without altering the number of points, and instead increases the number of dimensions of the features of each point through convolution operations. This allows the information in the point cloud to assume richer representations in deeper feature spaces, as shown in Eq. (1):

$$(\mathcal{F} * g)(x) = \sum_{x_i \in \mathcal{N}_x} g(x_i - x)\, f_i, \tag{1}$$

where $g(x_i - x)$ represents the kernel function that defines the influence of the input point x_i on the query point x, $\mathcal{N}_x$ denotes the neighborhood of x and is determined by a radius-based neighborhood search, while f_i represents the feature of x_i.

The above equation shows that the feature at query point x in the KPConv convolutional kernel is computed as a weighted combination of all points x_i within its neighborhood $\mathcal{N}_x$ and their respective features f_i, which are processed through kernel function g.

Kernel function g is central to the convolution kernel. It takes as input the positions of the neighboring points around point x, denoted by $y_i = x_i - x$. As the neighborhood is defined by radius-based neighborhood searches, the domain of g is the 3D sphere $B_r^3 = \{y \in \mathbb{R}^3 \mid \|y\| \leq r\}$. Similar to image convolution kernels, the kernel function requires applying different weights to distinct regions.

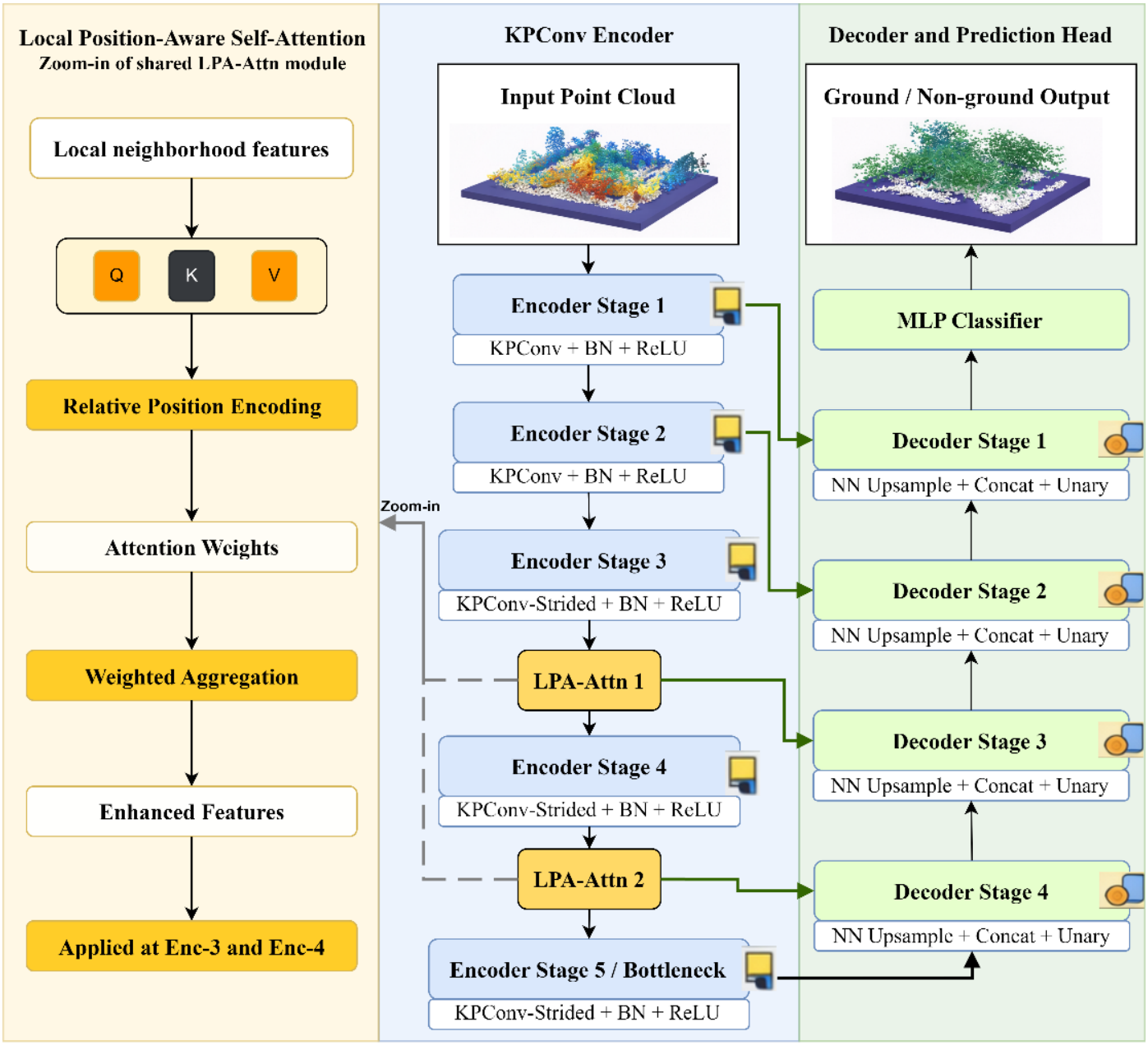


Figure 2. Overall architecture of the proposed KP-Attention Net.

Let the set of kernel points be $\{\tilde{x}_k \mid k < K\} \subset B_r^3$ and the set of weight matrices be $\{W_k \mid k < K\} \subset \mathbb{R}^{D_{in} \times D_{out}}$. They are used to map the number of dimensions of the input feature D_{in} to the number of output dimensions D_{out}, and the overall operation is as follows:

$$g(y_i) = \sum_{k<K} h(y_i, \tilde{x}_k)\, W_k, \tag{2}$$

where $h(y_i, \tilde{x}_k)$ represents the correlation between kernel point $\tilde{x}_k$ and input point y_i. The closer $\tilde{x}_k$ is to y_i, the higher is the correlation between them.

The correlation $h(y_i, \tilde{x}_k)$ is calculated as follows:

$$h(y_i, \tilde{x}_k) = \max\left(0,\ 1 - \frac{\|y_i - \tilde{x}_k\|}{\sigma}\right), \tag{3}$$

where σ represents the distance of influence of the kernel point, which is a hyperparameter that needs to be selected based on the density of the point cloud.

Similar to 2D image convolutions, the KPConv convolution kernel extracts features from a 3D spherical domain. However, it replaces pixels in the image convolutions with kernel points by assigning corresponding kernel weights to each. The position of the kernel points is formulated as an optimization problem for optimal coverage in the spherical space.

As the input points typically do not align with the kernel points, a continuous filter is defined by using correlation functions to create a continuous domain. Ultimately, the features of a given point are computed by multiplying the weights of all kernel points by their respective correlations with that point. Figure 3 illustrates the layout of the kernel points of KPConv.

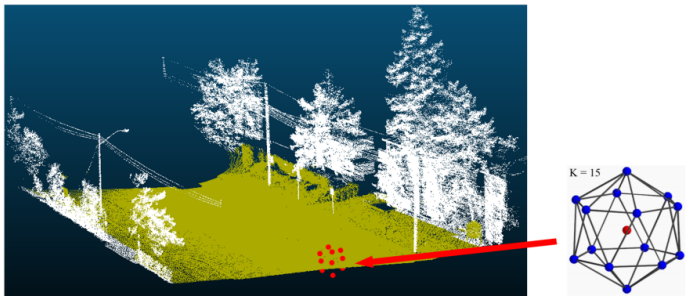


Figure 3. Interaction between the point cloud and the layout of kernel points in KPConv.

The left side of Figure 3 shows a complex outdoor 3D point cloud scene containing such elements as ground surfaces, trees, and utility poles. The set of green points represents "ground points" identified through the ground filtering algorithm, while the set of white points denotes "non-ground points," such as trees and utility poles. This visually demonstrates the core objective of ground filtering: accurately separating ground points from non-ground points within a mixed point cloud. The right side of Figure 3 depicts a schematic of constructing a local neighborhood by using a radius-based neighborhood search. The red points represent an individual point in the point cloud to be analyzed, while the blue points are its 15 example neighboring points within the radius-based local neighborhood. Together, they form the local spatial structure of the given point.

3.2.2 KPConv-Strided Convolution Kernel

KPConv convolutional kernels can capture local geometric information, but their limited receptive field makes it difficult to directly model large-scale terrain structures. We use KPConv-Strided convolutional kernels that gradually expand the receptive field by reducing the number of points through voxel subsampling during the convolution operation. This strategy is particularly suitable for MLS point clouds because such data typically encompass vast areas containing roads. Only by progressively expanding the field of view can the features of the terrain in the entire scene be fully modeled.

Compared with standard KPConv convolutional kernels, the KPConv-Strided kernel enables a progressive reduction in the number of points in the point cloud as it moves deeper into the network architecture. This approach reduces the computational demands of each layer while expanding the scope of information aggregation and the receptive field. The architecture of the encoder is shown in Figure 4.

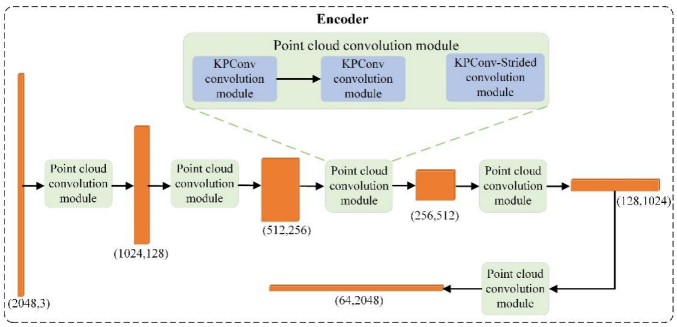


Figure 4. Structure of the encoder, where "(2048,3)" represents 2,048 points and three dimensions. The other elements are similarly represented.

Figure 4 depicts the process of bottom-up feature extraction of the point cloud. The orange blocks represent the features of the point clouds, where N is the number of points and d is the number of dimensions of their features. During the layer-by-layer convolution process, the number of points gradually decreases (as indicated by the height of the orange blocks in Figure 4), and this causes the receptive field of the convolution module to continuously expand and capture more global features. The number of dimensions of the features of the point cloud progressively increases, as indicated by the width of the orange blocks in Figure 4. The point cloud convolution module progressively maps the feature space of the point cloud in 3D coordinates into a higher-dimensional feature space. This enables the network to identify information contained in deep features of the point cloud.

3.2.3 ResNet Architecture

Layer-wise downsampling in point cloud processing inevitably leads to the loss of certain details. To address this issue, we use the ResNet architecture to enhance the stability of training of the network and its capability of extracting deep features. The geometric features of shallow layers of the point cloud can continue to influence the final results of ground filtering in deep networks through residual connections, and this enables the network to focus on small-scale details of the terrain while also understanding its large-scale morphology.

The ResNet architecture is shown in Figure 5. The

"+" symbol in the figure indicates that the original input features are directly added to the output after their dimensionality has been transformed. The MLP is designed to unify the number of dimensions of the input and output features to enable feature addition. Replacing the MLP in the residual connection with a pooling operation yields the KPConv-Strided convolution module. The pooling operation progressively reduces the number of points in the point cloud, while gradually expanding the network's receptive field.

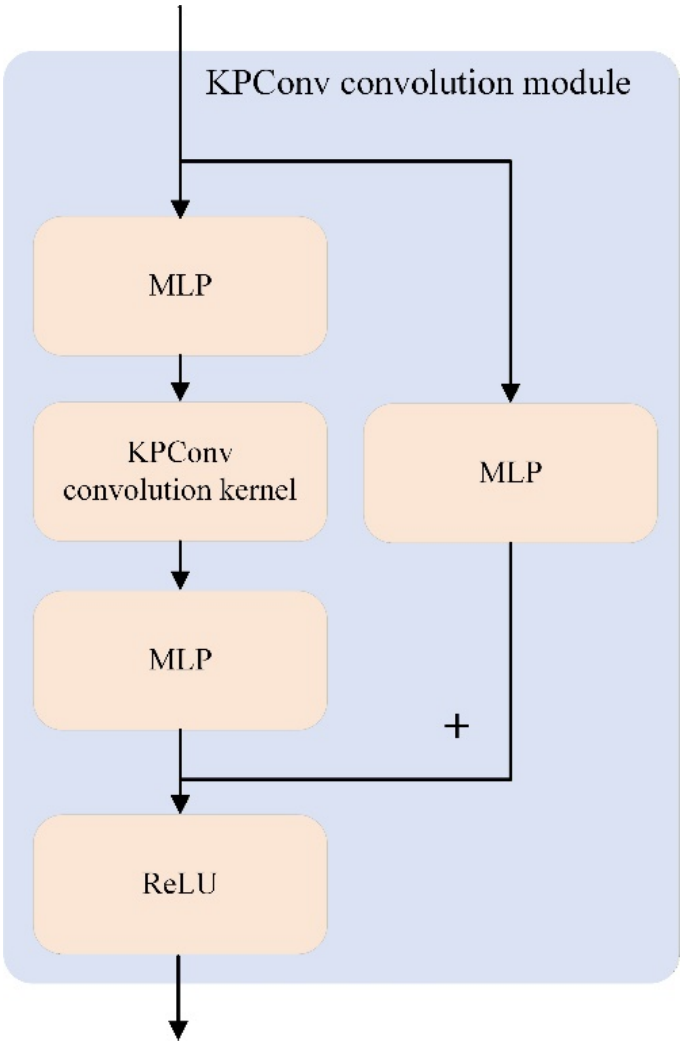


Figure 5. ResNet architecture for the KPConv convolution module.

The ResNet architecture facilitates the fusion of multi-scale information. As point cloud data contain geometric information at different scales, simple layer-by-layer convolutions may lead to the loss of certain local features during downsampling. ResNet establishes connections between the feature layers at different depths, which enables shallow features (local information) to be combined with deep features (global information) to enhance the network's capabilities of feature representation.

3.3 Point Cloud Position-aware Self-attention Module

It is essential in ground filtering tasks to not only identify local geometric features, but also to determine the overall structure of the urban environment. For instance, roads may be partially obscured by the canopies of trees, partially covered by vehicles, or may appear at varying elevations due to undulations of the terrain. Relying solely on the local information extracted by KPConv convolutional kernels thus often proves inadequate for accurately identifying these large-scale ground structures.

Moreover, the receptive field of traditional methods of point cloud convolution expands at a relatively slow pace. Even KPConv-Strided convolution kernels cannot directly model the long-range relationships between points at a global scale. Given this, we introduce a position-aware self-attention mechanism to KP-Attention Net to compensate for the limitations of early convolutional networks in modeling global information. This enables the network to learn the correlations between distant points to enhance its adaptability to complex urban terrains.

Attention mechanisms have achieved remarkable results in both language processing and image processing, particularly in large language models where the transformer architecture provides powerful capabilities of modeling global information and efficient feature representation. Therefore, research on the transformer in the context of point clouds has gradually emerged as a popular topic of research. The principle underlying the transformer is the attention mechanism. Inspired by the Point Transformer [27], this study introduces a position-aware self-attention mechanism to the encoder to resolve the issue of insufficient global information in early convolutional modules of the point cloud. This enables the network to establish a larger receptive field during the initial stage of feature extraction, which improves its ability to capture global features.

The position-aware self-attention mechanism in the Point Transformer is calculated as follows:

$$y_i = \sum_{x_j \in \mathcal{X}(i)} \operatorname{softmax}\big(\gamma(\varphi(x_i) - \psi(x_j) + \delta)\big) \odot \big(\alpha(x_j) + \delta\big), \tag{4}$$

where $\mathcal{X}(i)$ represents the local neighborhood of query point x_i; φ, ψ, and α are feature transformation

functions that respectively represent the query, key, and value in the attention mechanism, and are implemented as an MLP; δ is the positional encoding that is used to encode the spatial relationships in the point cloud; γ is the function used to calculate the attention weight and is implemented as an MLP; softmax is the normalization operation that ensures the stability of the attention weights; and $\odot$ denotes element-wise multiplication that is used for the weighted aggregation of the features.

The method used to determine the attention weights in Eq. (4) differs from traditional attention mechanisms in that it does not rely on the dot product of the query and key vectors. Instead, it adjusts the geometric information of the point cloud through vector subtraction and trainable positional encoding, as demonstrated in the computation within the function γ for calculating the attention weight. This approach avoids the limitations of traditional dot product-based attention in the face of the unordered nature of point cloud data, and enables the model to adaptively adjust and directly use information on the spatial coordinates. This, combined with the trainable positional encoding shown in Eq. (5), enhances the network's ability to perceive local geometric structures.

$$\delta = \theta(p_i - p_j), \tag{5}$$

where p_i and p_j denote the coordinates of two distinct points in the point cloud, and θ is a trainable mapping function composed of two MLP layers and a non-linear ReLU activation. This trainable positional encoding enables the attention mechanism to better adapt to the irregularity of point cloud data, and improves the model's ability to represent spatial relationships.

The point cloud position-aware self-attention module used in this study is shown in Figure 6.

The point cloud position-aware self-attention module also contains the residual structure of ResNet. The input consists of a set of feature vectors x and their corresponding spatial coordinates p. First, linear projection is applied to reduce the dimensionality of x, which accelerates computation and minimizes the amount of redundant information. Second, the features x with reduced dimensions and their spatial coordinates p exchange information through a position-aware self-attention mechanism to aggregate the features of the point cloud. Finally, the module introduces a residual connection to add the aggregated features to the original input features x, which yields the output y. This module enhances the representational capabilities of the model while ensuring the stability of gradient flow and the integrity of the original features.

The position of the position-aware self-attention module within the point cloud convolution module is shown in Figure 7. Unlike the KPConv convolution kernel, the weighting of the position-aware self-attention mechanism does not rely on fixed, position-bound kernel weights. Instead, it dynamically assigns weights based on feature differences together with the relative positional encoding. This approach yields a low inductive bias that enables the network to freely explore the intrinsic characteristics of the input data within a larger feature space, thereby enhancing the performance and robustness of the model. This dynamic weighting mechanism can capture global contextual information, and provides a different perspective on feature extraction from point cloud data. It also allows the encoder to fully leverage global information in tasks of point cloud processing, thereby improving the adaptability of the model to complex urban scenes.

3.4 Upsampling and Skip Connections

The encoder stage extracts deep features in ground filtering tasks through downsampling by using KPConv-Strided convolutional kernels. The kernels progressively expand their receptive field to learn large-scale information on the given terrain. However, during this process, the resolution of the point cloud continuously decreases, which inevitably leads to the loss of some detailed information. To ensure that the final classification of ground points preserves local geometric features while incorporating deep global features, the decoder must progressively restore the original resolution of the point cloud and fuse the multi-scale information extracted during the encoding phase.

Specifically, the decoder progressively restores the original resolution of the point cloud and maps the multi-scale features extracted during the encoding phase back to its input space. This ensures that the final output fully preserves the geometric and semantic information of the point cloud. Four upsampling modules are used within the decoder, and each comprises nearest-neighbor upsampling and a 1D convolution, as illustrated in Figure 8.

The nearest-neighbor upsampling transfers the semantic features of each nearest neighbor point in the pooled point set to the corresponding dense point

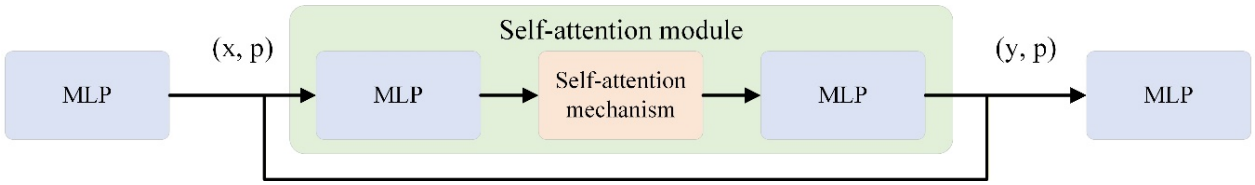


Figure 6. Point cloud position-aware self-attention module.

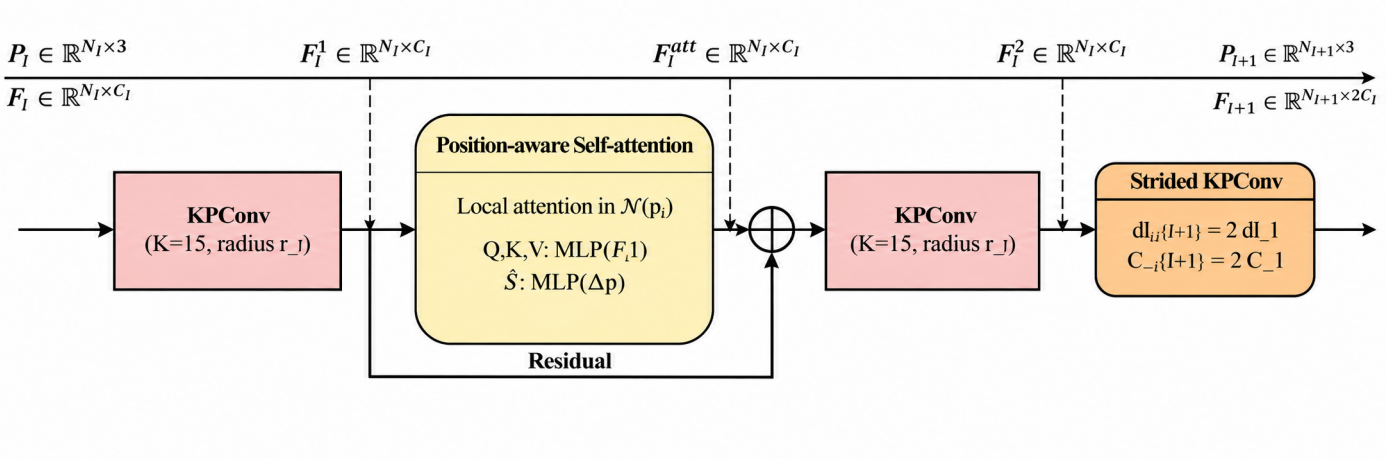


Figure 7. Point cloud convolution-self-attention module.

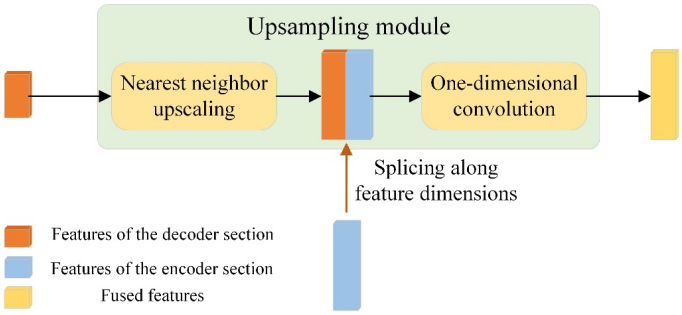


Figure 8. Upsampling module in the decoder.

set through index replication, thereby propagating the semantic information back to the original resolution. Subsequently, the upsampled features are concatenated with the corresponding encoder features through skip connections, and then fused through one-dimensional convolution (unary layer) to achieve feature mixing and dimensionality reduction.

During decoding, the number of points in the point cloud is gradually restored to the scale of the original input through nearest-neighbor upsampling, thereby imparting semantic information to all points in the original point cloud. Skip connections are subsequently introduced to enhance the continuity of information transmission, as illustrated by the orange lines in Figure 8. These connections represent the concatenation of information from the encoder and decoder along the feature dimension, and mitigate information loss during the recovery of deep features. Specifically, skip connections allow the features extracted by each layer of the encoder to be concatenated with the corresponding features from the decoder layer, which establishes the direct flow of information between them. The concatenated features undergo further feature fusion and dimension reduction through a 1D convolution layer. This ensures a compact feature representation of the point cloud, while enhancing the model's ability to perceive local structures. The overall process can be expressed as:

$$X_l^D = f_{conv}\left(f_{concat}\left(U(X_{l+1}^D),\, X_l^E\right)\right), \tag{6}$$

where X_l^D and X_l^E represent the feature representations of the lth layer in the decoder and encoder, respectively; f_{conv} and f_{concat} denote the 1D convolution operation and the operation used to concatenate the features along the feature dimension, respectively; and $U(\cdot)$ represents the nearest-neighbor upsampling operation.

Finally, the number of feature channels is reduced to the number of classes. A softmax function is then applied to calculate the probability of each point belonging to each class, and each point is assigned the class with the highest predicted probability to complete the ground filtering task.

4 Experiments and Discussion

4.1 Toronto-3D Dataset

The Toronto-3D dataset [28], shown in Figure 9, was obtained by using the Teledyne Optech Maverick 2 mobile mapping system. This system contains a 32-line LiDAR sensor, a global navigation satellite system (GNSS), and a simultaneous localization and mapping (SLAM) system. The LiDAR sensor captured point cloud data at a rate of 700,000 points per second, and featured a vertical field of view ranging from $-10°$ to $+30°$ and a data collection accuracy better than 3 cm. The acquired point cloud data were subjected to further processing within the LiDAR mapping suite (LMS) Pro 3 software, with each point assigned a natural color through an imaging camera.

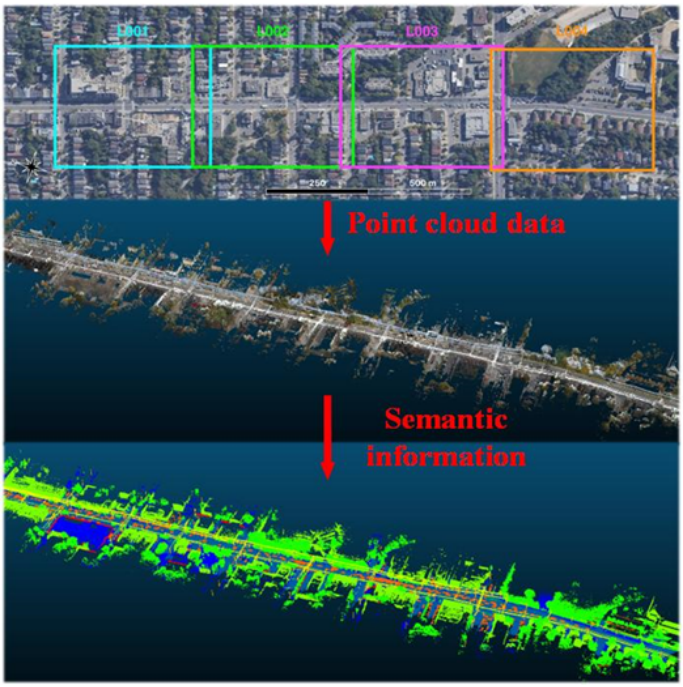


Figure 9. The Toronto-3D dataset and its semantic information.

This dataset was collected in Toronto, Canada, covered approximately 1 km of road, and contained roughly 78.3 million points. The dataset was divided into four segments, L001 through L004, each of which spanned approximately 250 m. The point cloud data were highly dense, with an average value of approximately 1,000 points per square meter. The labels of objects in different categories in this dataset covered primary elements in a typical urban environment, including roads, road markings, natural vegetation, buildings, power lines, utility poles, vehicles, and fences. Some unclassified points were retained to ensure the integrity and authenticity of the data.

For the binary ground filtering task, we remapped the original semantic labels into two classes, i.e., ground vs. non-ground, and excluded the unclassified points (label 0) from training and evaluation. On the development split (L001/L003/L004), which was further partitioned into training and validation subsets, the dataset contains 35,391,894 ground points (96.066%) and 1,449,308 non-ground points (3.934%). Thus, the imbalance ratio is approximately 24.42:1 (ground:non-ground).

4.2 Experimental Parameters and Evaluation Metrics

The code was developed by using the PyTorch [29] framework, with the primary hardware configuration consisting of a 40 GB NVIDIA A100 GPU. The KPFCNN setup was used to configure the convolution kernel, i.e., we used a KPConv convolution kernel with 15 kernel points, a downsampling grid size of 0.08 m, and a convolution radius of 2.5 times the downsampling grid size. The network was trained for a total of 400 epochs, each epoch containing 500 iterations, with a batch size of 4. To control the memory usage, each input sample was constructed as a local subset with a radius of 3.0 m. The adopted data augmentation methods include random rotation around the vertical axis, anisotropic scaling within the range of 0.9-1.1, optional symmetric flipping, and Gaussian jitter noise with a standard deviation of 0.001.

KP-Attention Net adopts a five-stage KPConv encoder and a four-stage decoder. The feature dimension of the network starts from 128 and is increased after each downsampling stage. The initial downsampling step size is set to $dl_0 = 0.08$ m, and the convolution radius of each layer is defined as $r_l = 2.5 \cdot dl_l$. At the same time, dl is doubled after each downsampling stage, so the effective convolution radii r_l of the five stages are 0.20, 0.40, 0.80, 1.60, and 3.20 m, respectively.

To mitigate the strong class imbalance described in Section 4.1, a class-weighted cross-entropy loss function was used for training, as shown in Eq. (7), while the stochastic gradient descent optimizer with momentum was used for parameter optimization with the momentum set to 0.98 and an initial learning rate of 0.01. The learning rate was divided by 10 after every 100 epochs to ensure the stable convergence of the model:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} w_c\, y_{i,c} \log\left(\hat{y}_{i,c}\right), \tag{7}$$

where N denotes the number of points in the point

cloud and C represents the number of categories (set to two for ground filtering, i.e., ground and non-ground). w_c is the weight of category c; we set the class weights inversely proportional to the class frequencies measured on the training split (excluding label 0), equivalently normalized to 1:24.42. This weighting adjusts the loss contribution of the minority class without changing the sampling distribution of the point cloud neighborhoods. $y_{i,c}$ is the label of the true category of a given point ($y_{i,c} = 1$ if point i belongs to category c), and $\hat{y}_{i,c}$ is the network's predicted probability that point i belongs to category c.

We evaluated the performance of the model based on its overall accuracy (OA), recall, precision, intersection over union (IoU), and the mean IoU (mIoU) over the two classes.

For reproducibility, Table 1 summarizes the key hyperparameters for model training. The core code and pre-trained models are also available at https://www.noda.ac.cn/datasharing/science/scientists/creator/28783 and https://github.com/fhuang80/KPConv-Based-Point-Cloud-Ground-Filtering.git.

Table 1. Key hyperparameters for model training.

Parameter	Value
Learning rate	0.01
Batch size	4
Epochs	400
Optimizer	Momentum SGD (momentum = 0.98)
Learning rate decay	Divide by 10 every 100 epochs
Voxel downsampling size	0.08 m
GPU	NVIDIA A100 (40 GB)

4.3 Experimental Results

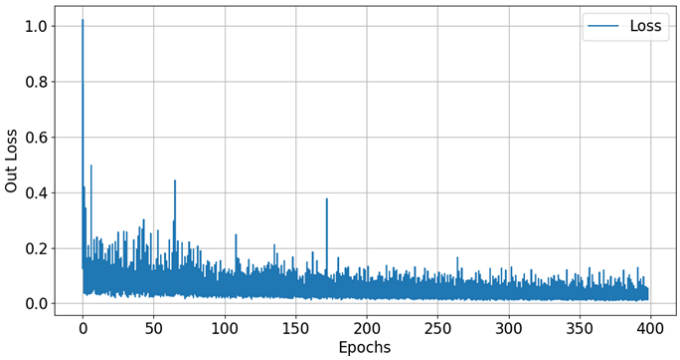


Figure 10. Curve of the loss function of the proposed model during training on the Toronto-3D dataset (stabilizes below 0.1).

The training loss of KP-Attention Net on the Toronto-3D dataset ultimately converged below 0.1, as shown in Figure 10. Its mean IoU (mIoU) on the L002 test region eventually stabilized at around 92.99%, as shown in Table 2. To visually demonstrate the effect of ground filtering, the filtering results on the test set are visualized in Figure 11. It shows that the network accurately identified and extracted the ground point clouds. Furthermore, to analyze the model's capability of discriminating between categories, the probability distribution of its predictions of the "ground" category is presented as a heatmap in Figure 12.

Higher predicted probabilities in Figure 12 correspond to deeper red colors while lower probabilities correspond to deeper blue. The network primarily focused on the ground point cloud, and offered weaker responses to other categories to exhibit a strong capability of discriminating between categories.

Our analysis of the training loss, visualization of the predicted results, and heatmaps of the probability of points belonging to one of the two categories demonstrated that KP-Attention Net performed well in ground filtering, and was able to accurately extract information from the ground point cloud.

To verify the advantages of KP-Attention Net, we conducted further experiments on the Toronto-3D dataset. We used regions L001, L003, and L004 as the development split, from which the training and validation subsets were created in proportion, while region L002 was used as the held-out test set for final evaluation. We subsequently conducted experiments on the test sets to compare the proposed method with commonly used ground filtering methods: the KPFCNN, a slope-based method, a morphology-based method, a surface-based method, and a cloth simulation filter (CSF). Table 2 shows the results.

The experimental results shown in Table 2 demonstrate that KP-Attention Net achieved the highest OA of 98.03%, indicating that it yielded the highest overall accuracy in the classification of ground and non-ground points. It also achieved the highest mIoU of 92.99%, which means that its predictions aligned most closely with the ground truth. Compared to the best performing deep-learning-based baseline model, i.e., KPFCNN, the OA and mIoU of our model improved by 2.29 percentage points (from 95.74% to 98.03%) and 5.47 percentage points (from 87.52% to 92.99%), respectively.

This advantage reflects mainly the better recovery of ground points, while ground precision decreased

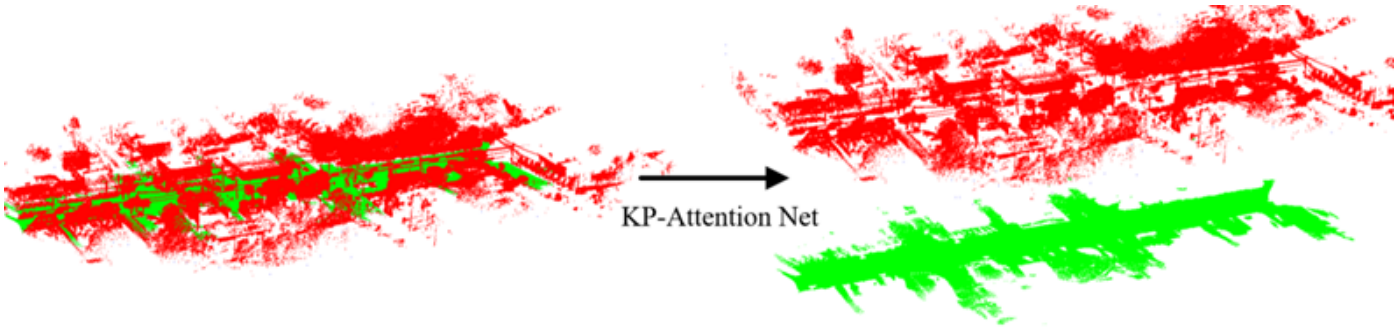


Figure 11. Visualization of the results of ground filtering by KP-Attention Net (OA = 98.03%).

Table 2. Results of experiments on ground filtering using different methods (unit: %).

Method	Ground point cloud			Non-ground point cloud			mIoU	OA
	Precision	Recall	IoU	Precision	Recall	IoU		
Slope-based	72.02	**97.32**	70.62	**99.15**	87.25	86.60	78.61	92.59
Morphology-based	79.31	96.78	77.27	97.64	92.84	90.80	84.04	92.28
Surface-based	75.47	94.27	72.16	97.82	84.52	82.96	77.56	84.32
CSF	**98.40**	71.65	70.82	72.75	**99.82**	72.65	71.74	90.14
KPFCNN	95.00	81.19	77.86	97.94	99.20	97.17	87.52	95.74
Ours	93.50	93.94	**88.18**	98.62	99.16	**97.80**	**92.99**	**98.03**

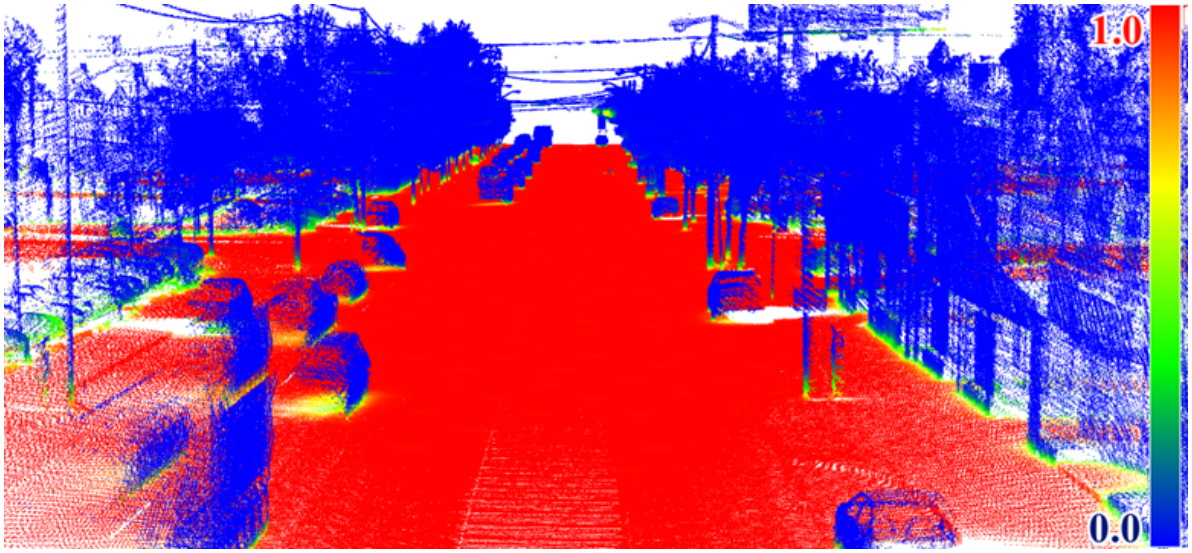


Figure 12. Heatmap of the predicted probability of the ground class by KP-Attention Net.

slightly (from 95.00% to 93.50%). More importantly, the gain is more pronounced for the ground class: the ground IoU increases by 10.32 percentage points (from 77.86% to 88.18%), mainly due to a substantial increase in ground recall of 12.75 percentage points (from 81.19% to 93.94%). This indicates that fewer ground points were missed, i.e., there were fewer false negatives. In dense urban MLS scenes, a higher recall is particularly beneficial for recovering occluded or locally discontinuous ground. However, a small portion of errors may come from slightly blurred ground-object boundaries around curbs or sidewalk transitions, low vegetation, or small near-ground objects. As the mIoU balances recall and precision, the high mIoU achieved by KP-Attention Net further attests to its stability in classification tasks. Specifically, it maintained a high recall while keeping misclassification low, which is particularly beneficial in urban MLS scenes where locally discontinuous ground (like curbs, ramps, and steps) and occlusions often cause missed ground points.

Table 2 also shows that the CSF method achieved the highest ground precision of all methods, 98.40%,

which reflects a low false-positive rate. This is consistent with its conservative cloth-fitting prior, which tends to suppress false positives but may increase false negatives in ambiguous areas that have occlusions or abrupt elevation changes. However, its recall rate was only 71.65%, suggesting a significant issue of false negatives. This means that it was unable to correctly identify many actual ground points. By contrast, the slope-based method exhibited the opposite trend: it achieved the highest ground recall of any method (97.32%), but attained the lowest ground precision (72.02%). This shows that while the method was able to detect the vast majority of ground points, it suffered from a high rate of misclassification. This means that some non-ground points were incorrectly classified as ground points (false positives), as depicted by the green areas in Figure 13. The blue color represents correct classification, green indicates false positives, and red indicates false negatives. Most false positives occur near ground-object boundaries, e.g., curbs and low vegetation, whereas false negatives are more common in occluded ground patches or regions with abrupt elevation changes.

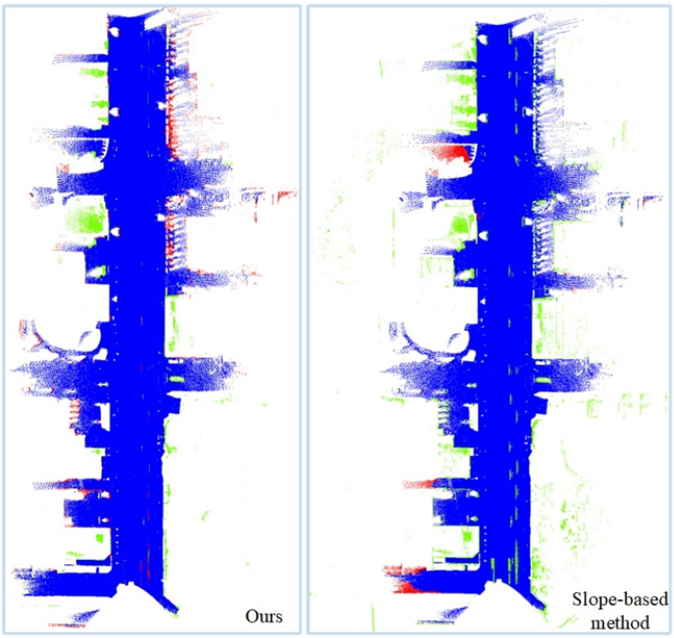


Figure 13. Visualization of results of ground filtering by KP-Attention Net and the slope-based method on the test set.

Our comparison of deep learning-based methods showed that KP-Attention Net outperformed the KPFCNN across multiple key metrics. By incorporating a position-aware self-attention mechanism, the proposed network was able to capture a broader range of global information and model the correlations between features over larger scales in earlier stages than the KPFCNN. Consequently, it achieved better mIoU and OA than the KPFCNN. These gains demonstrate the effectiveness of its position-aware self-attention mechanism in tasks of point cloud processing. It compensates for the limitation of the small early-stage receptive field of the KPConv convolutional kernel, and enables the model to better represent both global information and fine-grained features.

In summary, KP-Attention Net achieved the best overall classification performance of all methods by combining the KPConv convolutional kernel with the position-aware self-attention mechanism. Its high OA and mIoU values demonstrate its performance in terms of both accuracy and stability. They also show that the position-aware self-attention mechanism can clearly improve the model's capability of perceiving global features, thereby further improving its overall effectiveness in tasks of point cloud processing.

4.4 Cross-Dataset Generalization Experiment on Self-Collected Dataset

To evaluate the generalization capability of KP-Attention Net in real-world urban scenarios, we conduct cross-domain validation experiments on a self-collected urban road point cloud dataset, named the BeiQing dataset. The dataset features representative characteristics of urban arterial road scenes, encompassing diverse object categories including roads, vehicles, fences, poles, trees, vegetation, and ground surfaces. The dataset exhibits rich category distributions and complex spatial relationships, which reflect the intertwined distribution of semantic objects, intricate boundary relationships, and significant variations in structural scales commonly encountered in real-world urban road scenes.

The model trained on Toronto-3D is directly applied to the self-collected dataset without any fine-tuning, providing a rigorous test of its cross-domain generalization capability. Meanwhile, to verify the effectiveness of our method, we select two recent representative methods for comparison: Patchwork++ (IROS 2022) [30] as a representative of traditional geometry-based methods, and TerraSeg (CVPR 2026) [31] as a representative of self-supervised learning methods. All methods are evaluated uniformly on the BeiQing dataset to ensure a fair comparison.

Table 3. Performance comparison on the self-collected dataset (unit: %).

Method	mIoU	OA	Ground IoU	Non-ground IoU
Patchwork++	79.10	82.20	78.97	79.24
TerraSeg	88.75	90.55	88.40	89.10
Ours	**90.72**	**90.79**	**90.33**	**91.12**

From Table 3, our method achieves 90.72% mIoU and 90.79% OA on the dataset, outperforming Patchwork++ (79.10% mIoU, 82.20% OA) and TerraSeg (88.75% mIoU, 90.55% OA). Compared with TerraSeg, our method improves mIoU by 1.97 percentage points and OA by 0.24 percentage points, while the ground and non-ground IoUs are increased by 1.93 and 2.02 percentage points, respectively. This demonstrates that KP-Attention Net can effectively capture discriminative geometric features for both ground and non-ground classes, exhibiting good segmentation performance and generalization ability in complex urban scenes, thus providing technical support for terrain perception in real-world applications.

In addition to cross-dataset generalization, the Toronto-3D experiment in Section 4.3 also serves as a test of cross-scene generalization. Specifically, following the standard Toronto-3D protocol, the model is trained on L001, L003, and L004, and tested on the completely unseen L002 region. The performance on this held-out area (92.99% mIoU, Table 2) demonstrates its generalization ability across different urban street segments and scene compositions.

Overall, the experimental results on both the Toronto-3D and BeiQing datasets validate the effectiveness and generalization ability of KP-Attention Net. On the Toronto-3D dataset, the model achieves 92.99% mIoU and 98.03% OA, outperforming all compared methods; on the self-collected dataset, it also attains 90.72% mIoU and 90.79% OA, surpassing Patchwork++ (79.10% mIoU, 82.20% OA) and TerraSeg (88.75% mIoU, 90.55% OA). Despite the notable differences between the two datasets in terms of scene complexity, point cloud density, and acquisition conditions, the model retains strong segmentation performance, although its OA decreases from 98.03

4.5 Robustness Analysis

To evaluate the robustness of KP-Attention Net under challenging conditions, we designed two sets of experiments on the Toronto-3D test dataset: a Gaussian noise perturbation test and a point density downsampling test, which simulate real-world scenarios such as sensor measurement errors and point cloud sparsity caused by distance attenuation or occlusion, respectively.

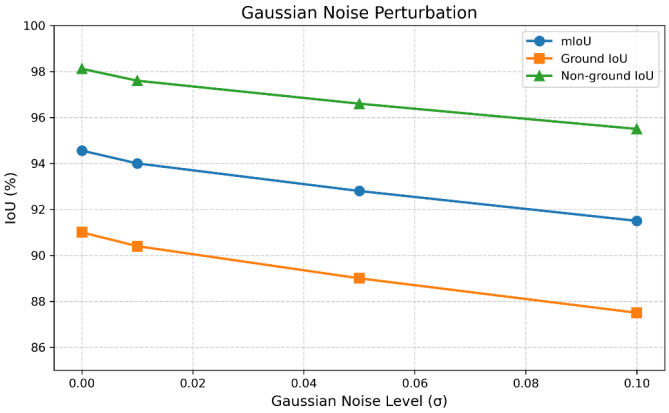


Figure 14. Performance variation of KP-Attention Net under Gaussian noise perturbation.

To evaluate noise robustness, we added Gaussian noise with standard deviations σ_n of 0.00, 0.01, 0.05, and 0.10 to the 3D coordinates of the point cloud. As shown in Figure 14, as the noise intensity increases, the model's performance exhibits a gradual decline. Even at a high noise level of $\sigma_n = 0.10$, mIoU remains at about 91.5%, while ground IoU and non-ground IoU are about 87.5% and 95.5%, respectively, representing only a limited decrease compared to the noise-free condition (about 94.5%/91.0%/98.1% for mIoU/ground IoU/non-ground IoU). This suggests that the model keeps stable feature discrimination under noise, although no attention-free baseline was tested under the same perturbation.

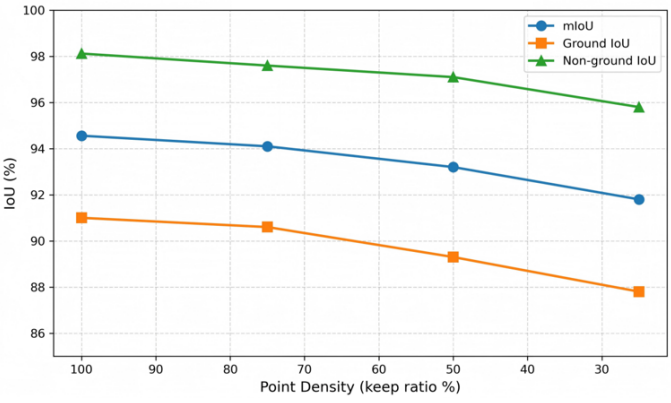


Figure 15. IoU variation of KP-Attention Net under different point cloud retention ratios in the point density downsampling experiment.

To evaluate density robustness, the test point cloud

Table 4. Results of ablation experiments on the position-aware self-attention module in KP-Attention Net (unit: %).

Method	Ground point cloud			Non-ground point cloud			mIoU	OA
	Precision	Recall	IoU	Precision	Recall	IoU		
KPFCNN [19]	95.00	81.19	77.86	97.94	99.20	97.17	87.52	95.74
KPFCNN*	**97.30**	84.57	82.63	98.12	**99.37**	97.51	90.07	96.33
Ours	93.50	**93.94**	**88.18**	**98.62**	99.16	**97.80**	**92.99**	**98.03**

was randomly downsampled to 100%, 75%, 50%, and 25% of the original number of points. As shown in Figure 15, even under extremely sparse conditions (retaining only 25% of the original points), the model still achieves about 91.8% mIoU, 87.8% ground IoU, and 95.8% non-ground IoU, with a smooth and controllable decline in performance. This suggests that the attention mechanism may still capture long-range dependencies and contextual information in sparse point clouds, thereby maintaining filtering accuracy. Overall, KP-Attention Net exhibits strong robustness against both measurement noise and density variation, making it well-suited for practical urban MLS deployment scenarios where data quality is often suboptimal.

4.6 Ablation Experiment

The core of KP-Attention Net lies in its integration of the KPConv convolutional kernel with a position-aware self-attention mechanism. By leveraging the ability of the latter to model global spatial relationships, it addresses the shortcomings of traditional convolutions in capturing long-range dependencies, and thus improves ground filtering performance under complex urban terrains. However, we need to establish that the improvement in model performance actually stems from the position-aware self-attention mechanism, rather than from the indirect effects of a deeper network. Moreover, the type, number, and embedding position of the position-aware self-attention modules directly impact the balance between performance and computational efficiency, which therefore calls for a focused investigation. We accordingly designed three targeted experiments: First, we conducted ablation experiments to isolate the independent contributions of the position-aware self-attention mechanism. Second, we compared the proposed position-aware self-attention with a standard self-attention variant to verify the necessity of adopting our design. Third, we performed experiments on the number of modules to determine the optimal configuration. The results provide empirical support for the design of the position-aware self-attention module.

4.6.1 *Validation of Position-aware Self-attention Module*

As shown in Table 4, the introduction of the position-aware self-attention mechanism to KP-Attention Net improved its performance compared with that of the original KPFCNN. This shows that the position-aware self-attention mechanism can improve the model's capability of learning in tasks of point cloud feature extraction.

However, it is widely recognized that as the depth of the network and the number of model parameters increase, deep neural networks may acquire even stronger capabilities of feature learning. To verify whether this improvement in performance stems from the introduction of the position-aware self-attention mechanism, rather than merely from increased network depth, we designed an ablation experiment. We replaced the position-aware self-attention module in KP-Attention Net with the original KPConv convolutional module (denoted by "KPFCNN*"), while keeping the rest of the network architecture unchanged, to observe the difference in performance. The results are shown in Table 4. The specific contribution of the position-aware self-attention module to the improvement in model performance is evident from them.

Table 4 shows that compared with the KPFCNN, the model that added only the KPConv convolutional module achieved a much smaller improvement (mIoU from 87.52% to 90.07%, versus 92.99% for KP-Attention Net). By contrast, KP-Attention Net achieved the best performance in terms of both mIoU and OA, indicating that it made more balanced predictions of points in the ground and non-ground categories.

The introduction of the position-aware self-attention module proved more effective than simply adding the KPConv convolutional module, likely because it provided additional information on the spatial context and a global perspective early on. From a computational perspective, the position-aware self-attention mechanism can be viewed as a

complement to convolutional operations. It expands the network's receptive field at an early stage, which enables the model to comprehensively understand the spatial relationships of the objects to be learned. In urban scenes, large objects such as buildings often render local information insufficient for accurately modeling their overall characteristics. The KPConv convolution module progressively expands the receptive field during layer-by-layer computation, whereas the position-aware self-attention mechanism can directly introduce feature-related information from a larger spatial range in the early stages of the network. This enables it to better capture large-scale structures, and enhances its power to express the features of point clouds.

Furthermore, unlike the fixed-weight distribution used by the KPConv convolutional kernel, the position-aware self-attention mechanism dynamically assigns weights based on feature similarity together with trainable relative positional encoding. This enables the weights of features of each point to be adaptively adjusted based on contextual relationships within the input data. Such a dynamic approach to weighting allows KP-Attention Net to adapt more flexibly to complex and dynamic urban environments, and improves its robustness and performance in the context of point cloud classification.

4.6.2 Comparison with Standard Self-attention Variant

Although the aforementioned experiments have shown that introducing a self-attention mechanism is beneficial, this result is still insufficient to fully demonstrate the necessity of the position-aware self-attention mechanism in this study. To further clarify this point, we compared three attention introduction schemes under the same backbone network and training settings: (1) KPFCNN without attention, (2) KPFCNN with a standard local self-attention module, and (3) KP-Attention Net with the local position-aware self-attention mechanism proposed in this study. To ensure fairness, schemes (2) and (3) both use two attention modules. The comparison results of different attention introduction schemes are shown in Table 5.

Table 5. Comparison of accuracy and efficiency indicators under different attention introduction schemes.

Metric	No attention (KPFCNN)	Standard self-attention	Ours (position-aware)
Attention form	-	Local dot-product	Local position-aware
Positional term	-	-	Trainable relative positional encoding $\delta = \theta(p_i - p_j)$
Attn modules	0	2	2
OA (%)	95.74	97.01	**98.03**
mIoU (%)	87.52	91.65	**92.99**
Ground IoU (%)	77.86	86.51	**88.18**
Non-ground IoU (%)	97.17	96.79	**97.80**
Params (M)	24.10	24.38	24.82
Peak GPU memory (GB)	0.26	0.27	0.28
Time/frame (ms)	17.38	18.18	19.30
Time/1M points (ms)	3695.69	4194.35	4723.83

As shown in Table 5, the proposed position-aware attention outperforms standard self-attention across all accuracy metrics while introducing only a limited additional cost. Compared with standard self-attention, our method's OA increases from 97.01% to 98.03%, mIoU from 91.65% to 92.99%, ground IoU from 86.51% to 88.18%, and non-ground IoU from 96.79% to 97.80%. Meanwhile, the parameter count of our method increases only from 24.38 M to 24.82 M, peak GPU memory from 0.27 GB to 0.28 GB, and inference time per frame from 18.18 ms to 19.30 ms. Since both attention variants use the same number of attention modules, these results indicate that the gain does not come merely from adding attention, but from the proposed position-aware design, which can better exploit local spatial relationships for ground filtering of urban MLS data. That is, the proposed design achieves a more favorable trade-off between accuracy and efficiency, supporting the necessity of the proposed position-aware self-attention mechanism.

4.6.3 Experiment on Number of Position-aware Self-attention Modules

Varying the number and position of the position-aware self-attention modules in KP-Attention Net may have different effects on model performance. The network comprises five layers of point cloud convolution modules, as illustrated in Figure 4. Position-aware self-attention modules can be inserted into each

Table 6. Impact of different numbers of position-aware self-attention modules on the performance of KP-Attention Net (unit: %).

Number of modules	Ground point cloud			Non-ground point cloud			mIoU	OA
	Precision	Recall	IoU	Precision	Recall	IoU		
1	**95.31**	82.46	79.24	96.41	98.93	95.42	87.33	96.18
2	93.50	**93.94**	**88.18**	**98.62**	**99.16**	**97.80**	**92.99**	**98.03**
3	93.33	93.69	87.81	97.64	96.48	94.28	91.05	98.01
4	92.91	92.44	86.35	97.31	97.96	95.38	90.87	97.97
5	93.47	92.39	86.79	97.33	98.56	95.97	91.38	98.01

module, as shown in Figure 7. To investigate their optimal configuration, we experimented by adding varying numbers of position-aware self-attention modules at different levels and analyzed their impact on model performance.

To ensure experimental comparability, all other settings were kept unchanged and only the number of modules was varied. The optimal set of results for each metric is recorded in Table 6.

The overall trend in Table 6 indicates that when the number of position-aware self-attention modules increased from one to two, the proposed method improved on almost all evaluation metrics (except ground precision). Notably, its IoUs of both the ground and non-ground classes, as well as the overall mIoU, reached their maximum values, while its OA was 98.03%. However, when the number of position-aware self-attention modules was further increased to three or more, its performance did not continue to improve, and even showed a slight decline on certain metrics. This suggests that an appropriate number of position-aware self-attention modules can improve the model's ability to capture global information, but an excessively large number of such modules may increase its complexity and computational overhead while yielding diminishing returns. Therefore, two position-aware self-attention modules are the best choice among those tested, as they gave the highest accuracy in Table 6 at a modest extra cost (Table 5).

5 Conclusion

This study proposed a deep learning-based network, called KP-Attention Net, to overcome the limitations of currently available point cloud ground filtering methods, including their strong dependence on manually selected parameters, significant loss of information, inadequate capabilities of modeling global features, and the higher computational cost of some existing deep-learning-based filtering algorithms. The proposed network integrates KPConv convolutions with a position-aware self-attention mechanism to yield the following advantages: (1) It fully leverages the strengths of KPConv in extracting local geometric features and hierarchical downsampling. (2) The position-aware self-attention mechanism enhances the model's awareness of the global context to improve its ability to express features at a modest additional computational cost. (3) Experiments on the Toronto-3D dataset confirm that the position-aware self-attention module improves global information integration, and the ablation experiment further verifies that this gain stems from the module itself rather than from an increase in network depth. Beyond Toronto-3D, cross-dataset experiments on the self-collected dataset further validate its generalization capability, with performance advantages over recent methods such as Patchwork++ and TerraSeg.

In addition, the comparison with a standard self-attention variant further showed that the observed gain does not come from the introduction of attention alone, but from the proposed position-aware self-attention mechanism. These results show that KP-Attention Net exhibits good robustness and accuracy in ground point classification, and provides a reliable end-to-end solution for filtering point clouds in urban MLS scenes.

Although this study has reported promising results for KP-Attention Net, future research should focus on further evaluating its performance in more challenging urban MLS scenarios, such as heavy occlusion, dense near-ground clutter, and weather-affected scans (e.g., snow-covered surfaces), and on conducting experiments across cities, sensors, and point densities. As urban environments grow increasingly complex, improving the model's generalization and robustness across scenarios is also an important direction of

research.

Data Availability Statement

The Toronto-3D dataset analyzed in this study is publicly available at https://github.com/WeikaiTan/Toronto-3D. The self-collected BeiQing dataset is a proprietary asset of Professor Fang Huang's research group and is subject to project agreements; therefore, it is not available for public sharing or redistribution by the authors. Inquiries regarding access to the BeiQing data should be directed to Professor Fang Huang. The source code and pre-trained models are available at https://github.com/fhuang80/KPConv-Based-Point-Cloud-Ground-Filtering.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 42271390, and in part by the Inner Mongolia Natural Science Joint General Fund under Grant 2025LHMS04004.

Conflicts of Interest

Fang Huang served as an Associate Editor of the *Journal of Geoscience and Earth Observation* at the time of manuscript submission. To ensure the integrity of the peer-review process, Fang Huang was not involved in the editorial handling, peer review, or decision-making process for this manuscript, which was handled independently by another editor. The remaining authors declare no conflicts of interest.

AI Use Statement

The authors declare that no generative AI was used in the preparation of this manuscript.

Ethical Approval and Consent to Participate

Not applicable.

References

[1] Rizaldy, A., Persello, C., Gevaert, C. M., Oude Elberink, S., & Vosselman, G. (2018). Ground and multi-class classification of airborne laser scanner point clouds using fully convolutional networks. *Remote Sensing, 10*(11), 1723. [CrossRef]

[2] Vosselman, G. (2000). Slope based filtering of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing, 33*(B3/2), 935-942. https://www.researchgate.net/publication/228719860

[3] Susaki, J. (2012). Adaptive slope filtering of airborne LiDAR data in urban areas for digital terrain model (DTM) generation. *Remote Sensing, 4*(6), 1804-1819. [CrossRef]

[4] Kilian, J., Haala, N., & Englich, M. (1996). Capture and evaluation of airborne laser scanner data. *International Archives of Photogrammetry and Remote Sensing, 31*(B3), 383-388. https://www.researchgate.net/publication/2499610

[5] Zhang, K., Chen, S. C., Whitman, D., Shyu, M. L., Yan, J., & Zhang, C. (2003). A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE Transactions on Geoscience and Remote Sensing, 41*(4), 872-882. [CrossRef]

[6] Kraus, K., & Pfeifer, N. (1998). Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing, 53*(4), 193-203. [CrossRef]

[7] Zhang, W., Qi, J., Wan, P., Wang, H., Xie, D., Wang, X., & Yan, G. (2016). An easy-to-use airborne LiDAR data filtering method based on cloth simulation. *Remote Sensing, 8*(6), 501. [CrossRef]

[8] Li, Y., Chen, Q., Li, C., & Bu, P. (2018, November). Improvement of CSF based on a wide range of urban complex scenes. In *Lidar Remote Sensing for Environmental Monitoring XVI* (Vol. 10779, pp. 191-199). SPIE. [CrossRef]

[9] Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (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). [CrossRef]

[10] Xu, Y., Fan, T., Xu, M., Zeng, L., & Qiao, Y. (2018, September). Spidercnn: Deep learning on point sets with parameterized convolutional filters. In *European Conference on Computer Vision* (pp. 90-105). Cham: Springer International Publishing. [CrossRef]

[11] Schmohl, S., & Sörgel, U. (2019). Submanifold sparse convolutional networks for semantic segmentation of large-scale ALS point clouds. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 4*(2/W5), 77-84. [CrossRef]

[12] Yotsumata, T., Sakamoto, M., & Satoh, T. (2020). Quality improvement for airborne lidar data filtering based on deep learning method. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 43*(B2), 355-360. [CrossRef]

[13] Nurunnabi, A., Teferle, F. N., Li, J., Lindenbergh, R. C., & Hunegnaw, A. (2021). An efficient deep learning approach for ground point filtering in aerial laser scanning point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 43*(B1), 31-38. [CrossRef]

[14] Janssens-Coron, E., & Guilbert, E. (2019). Ground point filtering from airborne lidar point clouds using deep learning: A preliminary study. *The International*

Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 42(2/W13), 1559-1565. [CrossRef]

[15] Jin, S., Su, Y., Zhao, X., Hu, T., & Guo, Q. (2020). A point-based fully convolutional neural network for airborne LiDAR ground point filtering in forested environments. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13*, 3958-3974. [CrossRef]

[16] Krisanski, S., Taskhiri, M. S., Gonzalez Aracil, S., Herries, D., & Turner, P. (2021). Sensor agnostic semantic segmentation of structurally diverse and complex forest point clouds using deep learning. *Remote Sensing, 13*(8), 1413. [CrossRef]

[17] Huang, S., Liu, L., Dong, J., Fu, X., & Huang, F. (2022). SPGCN: Ground filtering method based on superpoint graph convolution neural network for vehicle LiDAR. *Journal of Applied Remote Sensing, 16*(1), 016512. [CrossRef]

[18] Soilán, M., Nóvoa, A., Sánchez-Rodríguez, A., Riveiro, B., & Arias, P. (2020). Semantic segmentation of point clouds with pointnet and kpconv architectures applied to railway tunnels. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 5*(2), 281-288. [CrossRef]

[19] Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., & Guibas, L. J. (2019). KPConv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 6411-6420). [CrossRef]

[20] Kada, M., & Kuramin, D. (2021). ALS point cloud classification using PointNet++ and KPConv with prior knowledge. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 46*(4/W4), 91-96. [CrossRef]

[21] Qin, N., Tan, W., Ma, L., Zhang, D., Guan, H., & Li, J. (2023). Deep learning for filtering the ground from ALS point clouds: A dataset, evaluations and issues. *ISPRS Journal of Photogrammetry and Remote Sensing, 202*, 246-261. [CrossRef]

[22] Li, B., Lu, H., Wang, H., Qi, J., Yang, G., Pang, Y., Dong, H., & Lian, Y. (2022). Terrain-Net: A highly-efficient, parameter-free, and easy-to-use deep neural network for ground filtering of UAV LiDAR data in forested environments. *Remote Sensing, 14*(22), 5798. [CrossRef]

[23] Dai, H., Hu, X., Shu, Z., Qin, N., & Zhang, J. (2023). Deep ground filtering of large-scale ALS point clouds via iterative sequential ground prediction. *Remote Sensing, 15*(4), 961. [CrossRef]

[24] Cheng, L., Hao, R., Cheng, Z., Li, T., Wang, T., Lu, W., Ding, Y., & Hu, H. (2023). Modeling the global relationship via the point cloud transformer for the terrain filtering of airborne LiDAR data. *Remote Sensing, 15*(23), 5434. [CrossRef]

[25] Li, J., Tang, F., Ma, L., Zhu, C., Gong, Z., Ruhaiyem, N. I. R., & Li, J. (2025). Point-SCT: A multiscale spatial convolution-Swin transformer network for point cloud ground filtering in complex mountainous terrains. *IEEE Transactions on Geoscience and Remote Sensing, 63*, 1-18. [CrossRef]

[26] Štroner, M., Boušek, M., Kučera, J., Váchová, H., & Urban, R. (2025). Multi-size voxel cube (MSVC) algorithm—A novel method for terrain filtering from dense point clouds using a deep neural network. *Remote Sensing, 17*(4), 615. [CrossRef]

[27] Zhao, H., Jiang, L., Jia, J., Torr, P. H. S., & Koltun, V. (2021). Point Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 16259-16268). [CrossRef]

[28] Tan, W., Qin, N., Ma, L., Li, Y., Du, J., Cai, G., ... & Li, J. (2020, June). Toronto-3D: A large-scale mobile LiDAR dataset for semantic segmentation of urban roadways. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (pp. 797-806). IEEE. [CrossRef]

[29] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems, 32*, (pp. 8026-8037).

[30] Lee, S., Lim, H., & Myung, H. (2022, October). Patchwork++: Fast and robust ground segmentation solving partial under-segmentation using 3D point cloud. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 13276-13283). IEEE. [CrossRef]

[31] Lentsch, T., Montiel-Marín, S., Caesar, H., & Gavrila, D. M. (2026). TerraSeg: Self-Supervised Ground Segmentation for Any LiDAR. *arXiv preprint arXiv:2603.27344*. [CrossRef]



Chaoyang Bian is pursuing a master's degree in Big Data Technology and Engineering in the School of Resources and Environment, University of Electronic Science and Technology of China (UESTC). He received the B.S. degree in geographic information science from Southwest Petroleum University. (Email: biancy2023@163.com)



Xiang Kang is pursuing a master's degree in Cartography and Geographic Information System at the School of Geographical Sciences, Inner Mongolia Normal University. She received the B.S. degree in Human Geography and Urban-Rural Planning from Inner Mongolia University of Science and Technology, China, in 2024. (Email: 13357034758@163.com)



Binfu Ge is currently pursuing a master's degree in Big Data Technology and Engineering at the School of Resources and Environment, University of Electronic Science and Technology of China (UESTC). He obtained his Bachelor of Engineering degree in Communication Engineering from UESTC. (Email: 202222070206@std.uestc.edu.cn)



Zhiming Liu is currently pursuing a master's degree in Big Data Technology and Engineering at the School of Resources and Environment, University of Electronic Science and Technology of China. He holds a bachelor's degree in Geoinformation Science and Technology from China University of Mining and Technology. (Email: anjmire@163.com)



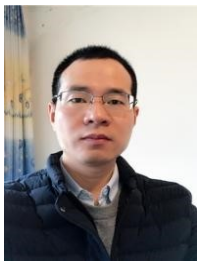
Qiaofeng Zhang received her Bachelor of Engineering in 2002 and Master of Engineering in 2005 from Taiyuan University of Technology, and Doctor of Agriculture from the Chinese Academy of Agricultural Sciences in 2016. She is currently affiliated with the College of Geographical Science, Inner Mongolia Normal University. (Email: zhangqf@imnu.edu.cn)



Yuxin Ding is pursuing a master's degree in Big Data Technology and Engineering in the School of Resources and Environment of UESTC. She received the B.S. degree in civil engineering from Chongqing University. (Email: dyx114@126.com)



Yifei Wang is currently pursuing a master's degree in Cartography and Geographic Information Systems at the School of Geographical Sciences, Inner Mongolia Normal University. She obtained a bachelor's degree in geographic information science from Baotou Normal University. (Email: 13674794320@163.com)



Fang Huang is an Associate Professor with the School of Resources and Environment, University of Electronic Science and Technology of China (UESTC), Chengdu, China. He received the B.S. degree in surveying and mapping engineering from Taiyuan University of Technology, Taiyuan, China, in 2002, the M.S. degree in photogrammetry and remote sensing from Beijing Jiaotong University, Beijing, China, in 2005, and the Ph.D. degree in cartography and geographical information system from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China. (Email: hfhbhzp@uestc.edu.cn)