Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

Rotation Invariant Spatial Networks for Single-View Point Cloud Classification

Feng Luan[1,2,3], **Jiarui Hu**[2,3,4], **Changshi Zhou**[1,2,3], **Zhipeng Wang**[2,3,4], **Jiguang Yue**[2,3,4], **Yanmin Zhou**[2,3,4*] and **Bin He**[2,3,4]

[1]Shanghai Research Institute for Intelligent Autonomous Systems
[2]National Key Laboratory of Autonomous Intelligent Unmanned Systems, Tongji University
[3]Frontiers Science Center for Intelligent Autonomous Systems
[4]College of Electronics and Information Engineering, Tongji University
{fengluan, hujiarui, zhoucs,wangzhipeng, yuejiguang, yanmin.zhou, hebin}@tongji.edu.cn

Abstract

Point cloud classification is critical for three-dimensional scene understanding. However, in real-world scenarios, depth cameras often capture partial, single-view point clouds of objects with different poses, making their accurate classification a challenge. In this paper, we propose a novel point cloud classification network that captures the detailed spatial structure of objects by constructing tetrahedra, which is different from point-wise operations. Specifically, we propose a RISpaNet block to extract rotation-invariant features. A rotation-invariant property generation module is designed in RISpaNet for constructing rotation-invariant tetrahedron properties (RITPs). Meanwhile, a multi-scale pooling module and a hybrid encoder are used to process RITPs to generate integrated rotation-invariant features. Further, for single-view point clouds, a complete point cloud auxiliary branch and a part-whole correlation module are jointly employed to obtain complete point cloud features from partial point clouds. Experimental results show that this network performs better than other state-of-the-art methods, evaluated on four public datasets. We achieved an overall accuracy of 94.7% (+2.0%) on ModelNet40, 93.4% (+5.9%) on MVP, 94.7% (+6.3%) on PCN and 94.8% (+1.7%) on ScanObjectNN. Our project website is https://luxurylf.github.io/RISpaNet_project/.

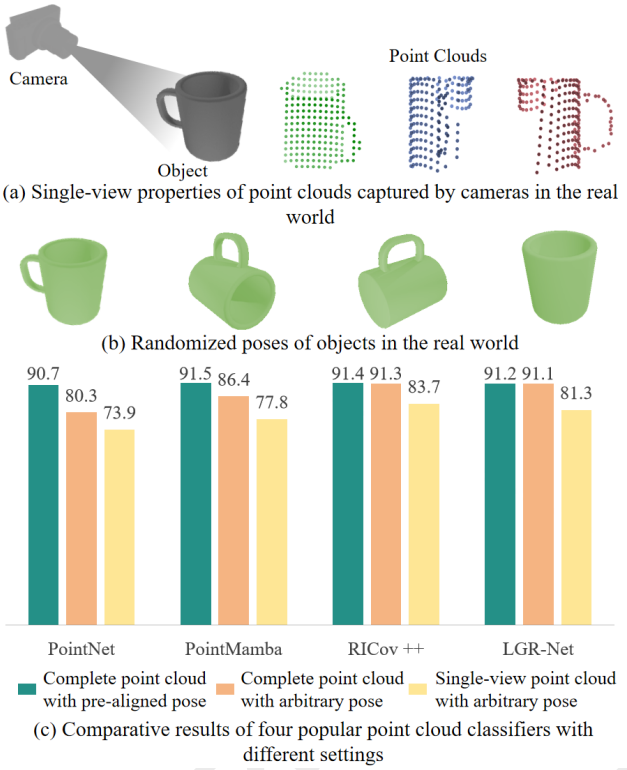


Figure 1: A single camera viewpoint and random poses of objects in the real world. (a) shows a portion of the point cloud acquired by a camera. (b) shows diverse poses of a cup. (c) shows the prediction results of four classifiers: PointNet, PointMamba, RICov++ and LGR-Net.

1 Introduction

Point cloud is three-dimensional (3D) data collected by depth devices, which provides a direct representation of the environment and objects [Qian *et al.*, 2022; Liu *et al.*, 2023]. Point cloud processing, including techniques such as classification and segmentation, is a fundamental requirement for unmanned systems to understand 3D scenes [Fei *et al.*, 2022; Fan *et al.*, 2023].

Since the introduction of PointNet [Qi *et al.*, 2017a], learning-based architectures have emerged with the ability to directly process point cloud data. Recently, an increasing number of studies have employed deep neural networks for processing point clouds to achieve high-precision classification [Wei *et al.*, 2023; Li *et al.*, 2018; Liang *et al.*, 2024; Lin *et al.*, 2022; Qiu *et al.*, 2022; Yu *et al.*, 2022]. However, most methods presume a complete, prescribed pose point cloud as network input. In the real world, the captured point cloud is usually single-viewed and partial. For example, single-view and arbitrary pose features of objects in the real world are shown in Fig. 1 (a) and (b).

*Corresponding author

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

Applying rotational enhancement during training could alleviate the impact of arbitrary poses on classifiers [Qi *et al.*, 2017a; Qi *et al.*, 2017b; Wang *et al.*, 2019]. However, due to the large number of degrees of freedom for 3D data, the effect of the above rotational enhancement is limited. Rotation-invariant models are key to accurately classifying objects for point clouds, which focus on designing features with rotational invariance [Poulenard *et al.*, 2019; Chen *et al.*, 2019; Deng *et al.*, 2018; Zhang *et al.*, 2022; Zhao *et al.*, 2022]. However, the performance of these methods tends to be suboptimal compared to translationally invariant methods, since a portion of spatial information of objects is often ignored or lost during the construction of rotation-invariant features.

The single-view point cloud data of objects with arbitrary poses results in a performance decline of classification, as presented in Fig. 1 (c). To solve this problem, we propose a rotation-invariant network for point cloud classification. Specifically, a RISpaNet block is designed for extracting rotation-invariant features from point clouds, which includes a rotation-invariant property generation module (RIPGM), a pooling module and a hybrid encoder. In RIPGM, a highly expressive rotation-invariant tetrahedron property (RITP) is developed. RITP is constructed as a local tetrahedron centered on a reference point to efficiently capture spatial structure of objects. The pooling module is used to extract multi-scale features. A hybrid encoder is designed to generate rotation-invariant features through different encoding mechanisms including convolution, attention mechanisms and state space models. For single-view point clouds, a complete point cloud auxiliary branch and a part-whole correlation module are designed to obtain complete point cloud features from partial point clouds.

To summarize, our main contributions are:

- **Rotation-invariant Tetrahedron Properties**. A highly expressive rotation-invariant property is designed by constructing local tetrahedra to capture spatial information of objects.
- **RISpaNet**. RISpaNet is a fundamental block in our model. This block includes a rotation-invariant property generation module, a multi-scale pooling module and a hybrid encoder. Different coding mechanisms are embedded in the hybrid encoder to extract the integrated rotation-invariant features from RITPs.
- **Two-branch design**. The complete point cloud auxiliary branch is designed to acquire complete point cloud features during the training phase. The part-whole correlation module is used to achieve the mapping from missing point cloud features to complete point cloud features. The two-branch structure can improve the single-view point cloud classification accuracy.

We performed experiments on the ModelNet40, MVP, PCN and ScanObjectNN datasets. Compared to state-of-the-art methods, our method achieved competitive or even the best performance.

2 Related Work

2.1 Learning-based Point Cloud Analysis

Recently, deep learning methods have become popular for processing point cloud data [Albert and Tri, 2023; Wang *et al.*, 2024; Shen *et al.*, 2024]. Initially, researchers used 3D voxel grids to process point clouds, which are easy to understand [Shi *et al.*, 2023; Li *et al.*, 2024; Singh and Yadav, 2024]. However, the voxelization process inevitably leads to information loss, which is closely related to the chosen resolution. PointNet [Qi *et al.*, 2017a] is a pioneer in point cloud feature extraction using multi-layer perceptions and global pooling operations. PointNet++ [Qi *et al.*, 2017b] was further proposed for multi-scale local feature encoding. Currently, the transformer framework has received widespread interest in point cloud processing. An efficient point self-attention layer was designed in Point Transformer [Zhao *et al.*, 2021]. The dual-channel structure has been proposed in a dual-transformer network [Han *et al.*, 2023] for efficient feature extraction. Recently, the Mamba framework based on state-space models has gained a lot of attention [Yang *et al.*, 2024; Ju *et al.*, 2024; Zhou *et al.*, 2025]. Mamba is a novel architecture that provides an efficient solution through its innovative design of a selective state-space model and linear time complexity [Albert and Tri, 2023]. Point-Mamba [Liang *et al.*, 2024] successfully applied Mamba to point cloud processing. However, the above methods are difficult to apply to the real world since the rotation of the object and the single-view features are not considered.

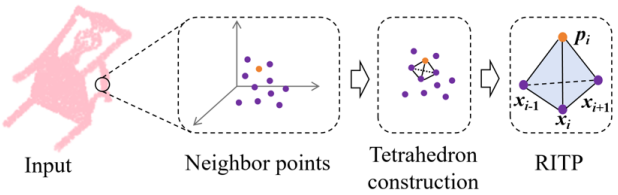
2.2 Single-view Point Cloud Analysis

Most of the inputs to current learning-based point cloud classifiers are complete point clouds. However, in the real world, the acquired 3D data are often partial point clouds from a single viewpoint. The predicted complete point cloud generated by point cloud completion methods can be used as input to classifiers [Cheng *et al.*, 2022; Sarmad *et al.*, 2019; Pan *et al.*, 2023]. Studies directly addressing single-view point cloud classification are still relatively few. A single-view point cloud classifier called SVP-classifier [Mohammadi *et al.*, 2022] was proposed for partial point cloud classification. SVP-classifier achieved classification based on global features, which were multi-view features mapped from single-view features. PAPNet [Xu *et al.*, 2023] was a single-view classification network that learned object pose changes during its implementation. However, PAPNet still has its limitations, as noted in the literature [Xu *et al.*, 2023], with inaccuracies in pose estimation and vulnerabilities to adversarial attacks, potentially leading to the failure of the perception system. In this paper, we aim to design an efficient point cloud classification network that directly processes single-view point clouds.

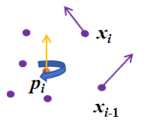
2.3 Rotation-invariant Analysis

Objects can be viewed from different perspectives and their poses are arbitrary in the real world. Therefore, rotation invariance is critical for practical applications in 3D object classification [Chen *et al.*, 2019]. Several methods [Zhang *et al.*, 2019; Zhang *et al.*, 2022] based on rotation-invariant

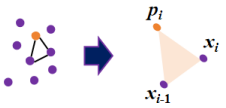
Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.



(a) 3D structure level property in RISpaNet



(b) Point level property in RIConv++



(c) Surface level property in RISurConv

Figure 2: The process of constructing rotation-invariant tetrahedron properties.

features have been proposed. RIConv [Zhang *et al.*, 2019] constructed rotation-invariant features based on the center of mass from a local point set formed by k nearest neighbor points. RIConv++ [Zhang *et al.*, 2022] designed rotation-invariant descriptors based on the theory of Local Reference Frames (LRFs). The LRF can preserve global rotation invariant features. Note that the LRF may be unstable due to changes in the centroid, especially if the shape contains disturbances. RISurConv [Zhang *et al.*, 2025] constructed a locally triangulated surface to capture detailed surface structures for extracting highly expressive rotation-invariant surface properties. However, the classification accuracies of these methods decrease in single-view point cloud classification tasks. We aim to design a point cloud classification network that is robust to single views and arbitrary rotations.

3 Method

To improve the accuracy of classifiers for arbitrary pose and single-view features in real scenarios, we propose a rotation-invariant point cloud feature extraction module and a dual-branch network architecture.

3.1 Rotation-invariant Tetrahedron Properties

RITP is used to capture detailed spatial structure by constructing local tetrahedra around a reference point p_i. The process of generating RITP is shown in Fig. 2 (a). For a reference point p_i, k nearest neighbors could be obtained to form a local point set. Then a tetrahedron is formed based on the point p_i and the nearest neighbors x_i and other two adjacent points, x_{i-1} and x_{i+1}. RITP can be expressed as:

$$RITP(x_i) = [L_0, L_1, L_2, \Phi_0, \Phi_1, \Phi_2, \theta_0, \theta_1, \theta_2, \alpha_0, \alpha_1, \beta_0, \beta_1, \gamma_0, \gamma_1, \omega_0, \omega_1, \omega_2, \omega_3] \quad (1)$$

where, L_i represents the distance from the reference point p_i to the other three vertices of the tetrahedron. Φ_i represents the relationship between the three edges in the tetrahedron with p_i as vertex. L_i and Φ_i are specified as:

$$L_0 = dist(p_i, x_i), L_1 = dist(p_i, x_{i-1}), \quad (2)$$

$$L_1 = dist(p_i, x_{i+1}), \Phi_0 = \angle(\overrightarrow{p_i x_{i-1}}, \overrightarrow{p_i x_i}), \quad (3)$$

$$\Phi_1 = \angle(\overrightarrow{p_i x_{i+1}}, \overrightarrow{p_i x_i}), \Phi_2 = \angle(\overrightarrow{p_i x_{i+1}}, \overrightarrow{p_i x_{i-1}}) \quad (4)$$

$\theta_0, \theta_1, \theta_2, \alpha_0, \alpha_1, \beta_0, \beta_1, \gamma_0, \gamma_1$ represent the relationship between point normals and edges in a tetrahedron:

$$\theta_0 = \angle(\overrightarrow{n_{p_i}}, \overrightarrow{p_i x_i}), \theta_1 = \angle(\overrightarrow{n_{p_i}}, \overrightarrow{p_i x_{i+1}}), \quad (5)$$

$$\theta_2 = \angle(\overrightarrow{n_{p_i}}, \overrightarrow{p_i x_{i-1}}), \alpha_0 = \angle(\overrightarrow{n_{x_i}}, \overrightarrow{x_i x_{i-1}}), \quad (6)$$

$$\alpha_1 = \angle(\overrightarrow{n_{x_i}}, \overrightarrow{x_i x_{i+1}}), \beta_0 = \angle(\overrightarrow{n_{x_{i+1}}}, \overrightarrow{x_{i+1} x_{i-1}}), \quad (7)$$

$$\beta_1 = \angle(\overrightarrow{n_{x_{i+1}}}, \overrightarrow{x_{i+1} x_i}), \gamma_0 = \angle(\overrightarrow{n_{x_{i-1}}}, \overrightarrow{x_{i-1} x_i}), \quad (8)$$

$$\gamma_1 = \angle(\overrightarrow{n_{x_{i-1}}}, \overrightarrow{x_{i-1} x_{i+1}}). \quad (9)$$

where $\overrightarrow{n_j}$ represents the normal of the corresponding vertex.

$\omega_0, \omega_1, \omega_2, \omega_3$ represent the relationships between the four faces that make up the tetrahedron:

$$\omega_0 = \angle(\overrightarrow{p_i x_i} \times \overrightarrow{p_i x_{i+1}}, \overrightarrow{p_i x_{i+1}} \times \overrightarrow{p_i x_{i-1}}), \quad (10)$$

$$\omega_1 = \angle(\overrightarrow{p_i x_{i+1}} \times \overrightarrow{p_i x_{i-1}}, \overrightarrow{p_i x_{i-1}} \times \overrightarrow{p_i x_i}), \quad (11)$$

$$\omega_2 = \angle(\overrightarrow{p_i x_{i-1}} \times \overrightarrow{p_i x_i}, \overrightarrow{p_i x_i} \times \overrightarrow{p_i x_{i+1}}), \quad (12)$$

$$\omega_3 = \angle(\overrightarrow{x_i x_{i+1}} \times (\overrightarrow{x_i x_{i-1}}, \overrightarrow{p_i x_{i-1}} \times \overrightarrow{p_i x_i}). \quad (13)$$

There is a significant difference in mathematical logic between RITP and other methods, which RITP exhibits high dimensional considerations. The specific difference is shown in Figs. 2 (b) and (c). The rotation-invariant property in RIConv++ is at the point level. RISurConv is at the surface level. Our RITP is at 3D structure level.

3.2 RISpaNet Block

RISpaNet is the core in our network (shown in Fig. 3), which is designed to extract the rotation-invariant features from point clouds. RISpaNet consists of three modules: the RIPGM, the pooling module and the hybrid encoder.

By constructing tetrahedra from nearest points, RIPGM generates RITP. A multi-scale pooling module (MSPM) is used to extract multi-scale features from RITP. As shown in Fig. 3, the pooling module consists of a convolution module and two pooling layers. RITP is processed by the pooling module to obtain multi-scale features $F_{RITPmul}$. MSPM enables network to contain global, local and contextual information compared to single pooling.

A hybrid encoder is designed to further process $F_{RITPmul}$ to obtain rotation-invariant features. As shown in Fig. 3, hybrid encoder consists of three different structures of encoders: Mamba, self-attention mechanism (SA) and convolutional neural network (CNN). These three components are chosen because they each represent a different encoding way. Each of them is good at extracting different information. Mamba is a novel network architecture based on state-space models (SSMs), designed to efficiently capture long-range dependencies while preserving linear computational complexity. SA is a mechanism for capturing the contextual relationships of data by calculating the correlations between queries (Q) and keywords (K) and weighting the values (V) with these correlations as weights. CNN could extract local features from

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

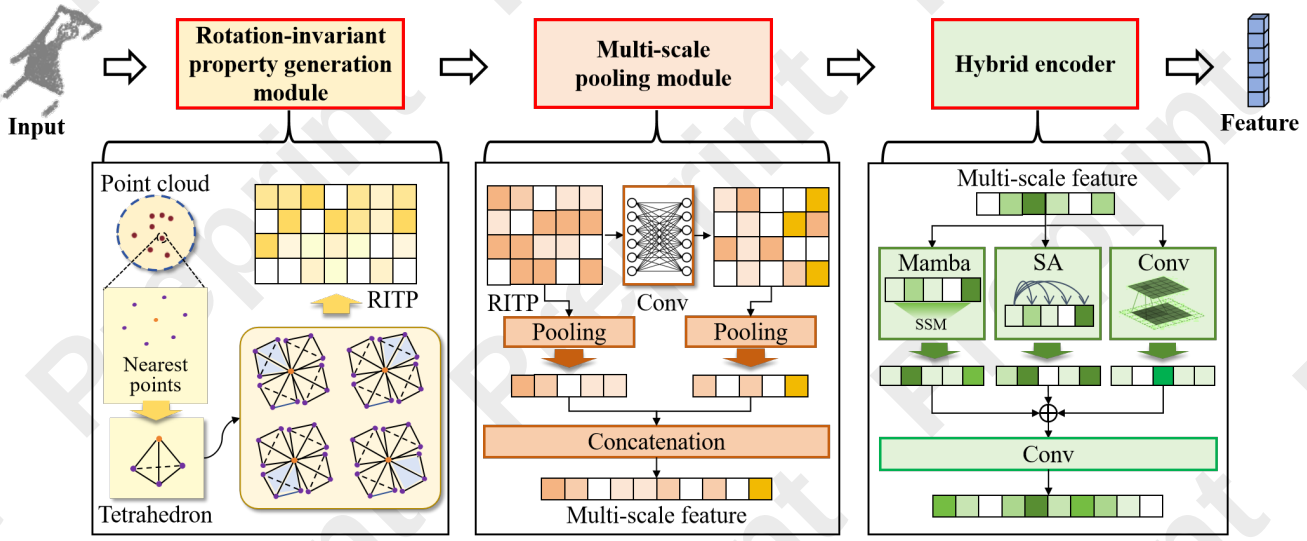


Figure 3: RISpaNet block structure. RISpaNet consists of three modules: the rotation-invariant property generation module (RIPGM), the multi-scale pooling module (MSPM) and the hybrid encoder. The RIPGM is used to generate the rotational-invariant tetrahedron properties (RITP). The multi-scale pooling module is applied to process the RITP to obtain multi-scale features. The hybrid encoder is designed to extract the integrated rotational-invariant features.

data through convolutional operations. The sum of the outputs from these three encoders is processed with a convolutional layer to obtain rotation-invariant features. The specific frameworks of Mamba and SA are shown in Fig. 4.

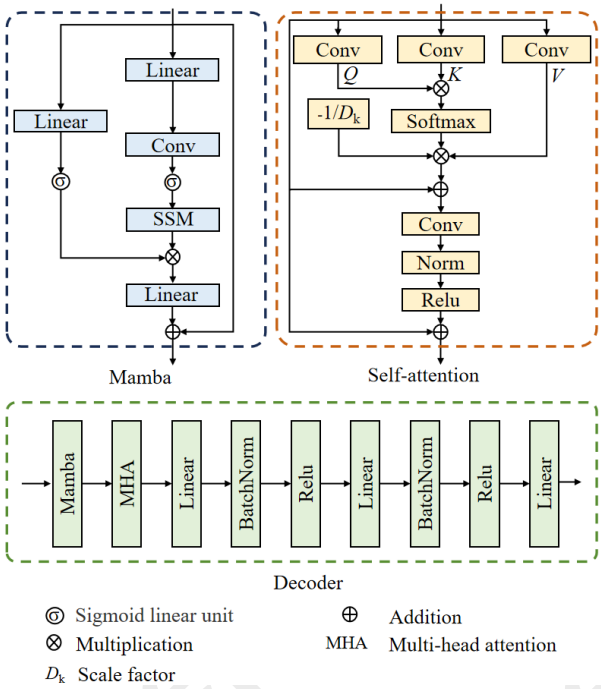


Figure 4: The structures for Mamba, SA and decoder. Mamba captures data features through state space models. SA captures data correlations through Q, K and V. Mamba and multi-head attention layers are used in decoder.

3.3 Two-branch Network Structure

To improve classification accuracy of partial point clouds, we design a complete point cloud auxiliary branch (CPCAB) to obtain complete point cloud features from partial. The network structure is shown in Fig. 5. The input to the CPCAB is the complete point cloud from the dataset used. The CPCAB is used only in the training phase. In the inference phase, we only use the partial point cloud branch. The input to the partial channel is a partial, single-view point cloud captured by the depth device. Five RISpaNet blocks are used to extract rotation-invariant features.

The parameters of RISpaNet blocks in the auxiliary and partial branch are shared. The outputs of the two branches are the complete point cloud feature f_{com} and the partial point cloud feature f_{part}. To achieve the mapping of partial point cloud features to complete point cloud features, we design a part-whole correlation module (PWCM). The structure of PWCM is similar to the multi-scale pooling module, except that there is no pooling layer. Finally, the predicted complete point cloud feature f_{compre} provided by PWCM is processed by the decoder to obtain the predicted object label y. The structure of the decoder used is shown in Fig. 4.

3.4 Loss

The loss function of our network consists of two parts: the class loss L_{cls} and the feature loss L_{fea}. The loss of the overall network is $L_{total} = L_{cls} + \lambda L_{fea}$, where λ is a trade-off parameter. The cross-entropy loss is used in L_{cls}, where $L_{cls} = F_{cross}(y_{gt}, y)$. y_{gt} is the true label. The mean square error loss is used in the feature loss L_{fea}, where $L_{fea} = F_{mse}(f_{com}, f_{compre})$.

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

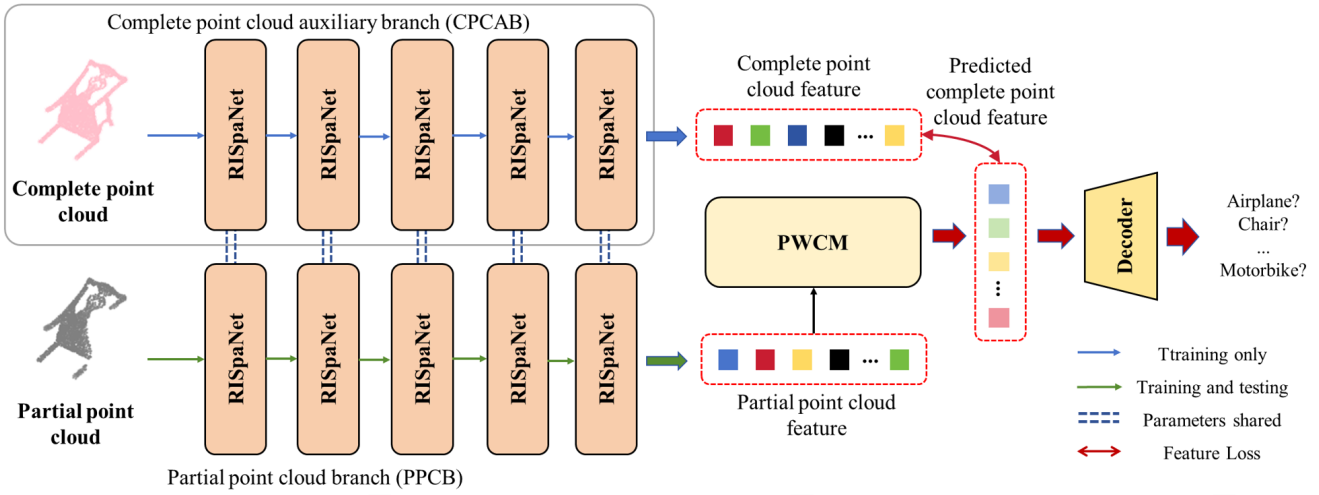


Figure 5: Two-branch structure. The proposed network contains two branches: the partial point cloud branch (PPCB) and the complete point cloud auxiliary branch (CPCAB). PPCB is used to extract the rotation-invariant features of single-view point clouds. CPCAB is used to extract the rotation-invariant features of complete point clouds. A part-whole correlation module (PWCM) is used to construct the link between the complete and partial features to get the predicted complete point cloud features. Decoder is used to output the classification results.

4 Experiment

4.1 Application Details

The network was tested on an NVIDIA RTX 3080 Ti GPU with four datasets: MoldeNet40 [Wu *et al.*, 2015], MVP [Pan *et al.*, 2023], PCN [Yuan *et al.*, 2018] and ScanObjectNN [Uy *et al.*, 2019]. The inputs to the network are 1024 3D coordinates and their corresponding point normals. It is worth noting that normals are optional for our method. In cases of no normals, we use the weighted eigenvector corresponding to the smallest eigenvalue as the normals. To evaluate the classification results, we followed the existing work on rotation invariance and performed experiments in three cases: (1) both training and test data were rotated around the z-axis (z/z), (2) training data were rotated around the z-axis and test data were arbitrarily rotated ($z/SO3$), and (3) both training and test data were arbitrarily rotated ($SO3/SO3$).

4.2 Complete Point Cloud Classification

The proposed method was initially validated on complete point clouds from ModelNet40. We trained and tested on all 40 classes in ModelNet40. We simply used the partial point cloud branch (PPCB) to connect directly to the decoder for training and testing. For simplicity, the single-branch way is called Ours-PPCB. The Adam optimizer was used. The learning rate started at 0.00033 and was reduced by 20% every 20 epochs. The batch for training was 56. We compared two types of methods: rotation-sensitive methods and rotation-invariant methods. The results are shown in Table 1. The results of PointNet [Qi *et al.*, 2017a], PointCNN [Li *et al.*, 2018], PointNet++ [Qi *et al.*, 2017b], DGCNN [Wang *et al.*, 2019], RIConv [Zhang *et al.*, 2019] and RIConv++ [Zhang *et al.*, 2022] are from RISurConv [Zhang *et al.*, 2025]. The results of the SF-CNN [Rao *et al.*, 2019], Li et al [Li *et al.*, 2021], LGR-Net [Zhao *et al.*, 2022] and Wang et al. [Wang *et al.*, 2024] are from Wang et al. [Wang *et al.*, 2024] . The results of RISurConv [Zhang *et al.*, 2025] and PointMamba [Liang *et al.*, 2024] were trained on the corresponding code provided on the official website.

Method	z/z	$SO3/SO3$	$z/SO3$
PointNet [Qi *et al.*, 2017a]	87.0	80.3	21.6
PointCNN [Li *et al.*, 2018]	91.3	84.5	41.2
PointNet++ [Qi *et al.*, 2017b]	89.3	85.0	28.6
DGCNN [Wang *et al.*, 2019]	92.2	81.1	20.6
PointMamba [Liang *et al.*, 2024]	89.3	86.4	24.1
RIConv [Zhang *et al.*, 2019]	86.5	86.4	86.4
RIConv++ [Zhang *et al.*, 2022]	91.3	91.3	91.3
Li et al. [Li *et al.*, 2021]	90.2	90.2	90.2
SF-CNN [Rao *et al.*, 2019]	91.4	90.1	84.8
Wang et al. [Wang *et al.*, 2024]	93.2	93.0	93.2
LGR-Net [Zhao *et al.*, 2022]	90.9	91.1	90.9
RISurConv [Zhang *et al.*, 2025]	93.5	92.7	93.4
Ours-PPCB (without normal)	94.5	94.5	94.5
Ours-PPCB	**94.7**	**94.7**	**94.6**

Table 1: Comparisons of the classification accuracy (%) on ModelNet40

For the rotation-sensitive method, Table 1 shows a significant drop in classification accuracy when the testing rotation differs from the training rotation. In contrast, the classification accuracy of the rotation-invariant methods decreases only slightly in this case. Our method achieves the highest classification accuracy as shown in Table 1, even without normal vectors.

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

	Method	Params(M)	MVP			PCN		
			z/z	$SO3/SO3$	$z/SO3$	z/z	$SO3/SO3$	$z/SO3$
Rotation -sensitive method	PointNet [Qi *et al.*, 2017a]	3.50	81.2	73.9	23.5	82.1	74.5	23.2
	PointCNN [Li *et al.*, 2018]	0.60	72.2	68.2	20.2	71.5	67.2	21.2
	PointMamba [Liang *et al.*, 2024]	12.3	88.9	77.8	22.5	88.2	78.2	25.4
Rotation -invariant method	RIConv++ [Zhang *et al.*, 2022]	0.40	83.8	83.6	83.7	82.2	82.0	82.2
	Li et al. [Li *et al.*, 2021]	2.91	79.4	79.3	79.4	81.1	81.1	81.0
	Wang et al. [Wang *et al.*, 2024]	1.88	85.4	85.2	85.3	86.2	86.4	86.0
	LGR-Net [Zhao *et al.*, 2022]	5.55	81.2	81.3	81.2	83.1	83.3	83.1
	RISurConv [Zhang *et al.*, 2025]	13.96	87.7	87.5	87.6	88.6	88.4	88.5
	Ours-PPCB	10.09	92.3	92.2	92.2	93.5	93.5	93.5
	Ours	10.43	**93.3**	**93.4**	**93.4**	**94.7**	**94.7**	**94.7**

Table 2: Comparisons of the classification accuracy (%) on MVP and PCN

4.3 Single-view Point Cloud Classification

We performed the single-view partial point cloud classification task on two datasets: the MVP and the PCN. MVP comprises 16 classes of point clouds captured by 26 fixed-pose depth cameras, while PCN includes 8 classes of point clouds from 8 viewpoints. For training in the two-branch way, the Adam optimizer was used. The learning rate started at 0.00044 and was reduced by 15% every 10 epochs. The batch for training was 40. As for λ, it was set to 0.001 for the first 10 epochs, 0.0001 for epochs 11 to 20 and 0.00001 for subsequent epochs.

All comparison methods were trained using the official code provided in the corresponding literature. As shown in Table 2, our method performs excellently in all cases, even in the single-channel way. This demonstrates that RISpaNet block is effective in extracting rotation-invariant features. Compared to ours-PPCB, the classification accuracy of our two-channel method is significantly improved. This demonstrates the effectiveness of CPCAB and PWCM. Our method achieves 93.4% classification accuracy in MVP and 94.7% in PCN, achieving a significant improvement on the classification results of existing methods.

Method	z/z	$SO3/SO3$	$z/SO3$
PointNet [Qi *et al.*, 2017a]	68.2	42.2	17.1
PointCNN [Li *et al.*, 2018]	78.5	51.8	14.9
PointNet++ [Qi *et al.*, 2017b]	77.9	60.1	15.8
DGCNN [Wang *et al.*, 2019]	78.1	63.4	16.1
PointMamba [Liang *et al.*, 2024]	70.1	65.6	33.8
RIConv [Zhang *et al.*, 2019]	68.1	68.3	68.3
RIConv++ [Zhang *et al.*, 2022]	80.3	80.3	80.3
Wang et al. [Wang *et al.*, 2024]	82.7	82.9	72.7
LGR-Net [Zhao *et al.*, 2022]	83.4	83.4	83.4
RISurConv [Zhang *et al.*, 2025]	93.1	93.1	93.1
Ours-PPCB	**94.8**	**94.8**	**94.8**

Table 3: Comparisons of the classification accuracy (%) on ScanObjectNN

4.4 Real World Object Classification

For classification task in the real world, we performed experiments on the ScanObjectNN dataset, which is a 3D point cloud dataset with 15 classes captured by an RGB-D camera. In our experiments, we used the processed files and choose the hardest variant PB_T50_RS to train our network with the partial point cloud branch. Due to the fact that normal were not provided in PB_T50_RS, 3D coordinate form was used as input. The results are presented in Table 3. It can be seen that the classification accuracy of our method achieves as high as 94.8%, which is significantly superior to the performances of existing methods.

4.5 Ablation Experiments

RITP. To validate the effectiveness of the designed RITP, ablation experiments were performed on rotation-invariant properties, comparing with rotation-invariant surface properties (RISP) in RISurConv [Zhang *et al.*, 2025] and informative rotation-invariant features (IRIF) in RIConv++ [Zhang *et al.*, 2022]. All three rotationally invariant features mentioned above were subsequently extracted using the pooling module and the hybrid encoder in RISpaNet. Both training and test data are arbitrarily rotated ($SO3$/$SO3$) on MVP. It can be seen from Table 4 that the RITP-based method achieves the highest classification accuracy, demonstrating its high-expressive rotation-invariant property.

RISpaNet. To validate the effectiveness of the pooling module and hybrid encoder in the RISpaNet block, ablation experiments were performed on feature extraction. Both training and test data are arbitrarily rotated ($SO3$/$SO3$) on MVP. The results are shown in Table 5. The accuracy of Model A is higher than that of Model B, indicating that the multi-scale pooling module is beneficial for classification. Hybrid encoders can efficiently extract rotation-invariant features by comparing the accuracy of model B with model D and model A with model C.

Efficiency Analysis. We performed a comprehensive analysis for network efficiency during training and testing, as shown in Table 6. We compared our method with RIConv++ [Zhang *et al.*, 2022], RISurConv [Zhang *et al.*, 2025] and LGR-Net [Zhao *et al.*, 2022]. The training and inference

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

Property	IRIF	RISP	RITP
Accuracy(%)	85.1	88.7	93.4

Table 4: Comparisons of the classification accuracy (%) on different rotation-invariant property

Model	MSPM	Hybrid encoder	Conv	Accuracy(%)
A	✓	✓		93.4
B		✓		91.8
C	✓		✓	91.2
D			✓	89.9

Table 5: Comparisons of the classification accuracy (%) on different modules in our method

Methods	Params(M)	Train(s)	Infer(ms)
RIConv++	0.40	17	0.014
RISurConv	13.96	30	0.022
LGR-Net	5.55	21	0.018
Ours	10.43	32	0.024

Table 6: Efficiency analysis of parameters, training and inferring time

λ	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
Accuracy	77.1	82.2	90.0	91.3	93.1	87.1

Table 7: Parametric analysis of λ

speeds of our network are generally comparable to existing methods, while the number of parameters is kept at a reasonable level.

Parametric Analysis. λ was obtained experimentally. We performed sensitivity tests for λ, as shown in Table 7. λ is appropriate between 10^{-3} and 10^{-5}. Based on the results of the sensitivity experiments, it is shown that the network is sensitive to λ in Loss.

5 Conclusion

In this paper, we propose a novel rotation-invariant network for point cloud classification. A highly expressive rotation-invariant property, called RITP, is designed to capture object spatial information by constructing tetrahedra. The RISpaNet block is used to extract rotation-invariant features of objects, including a rotation-invariant property generation module, a multi-scale pooling module and a hybrid encoder. Furthermore, for single-view point clouds, the joint use of a complete point cloud auxiliary branch and a part-whole correlation module is developed to obtain complete point cloud features from partial point clouds. Extensive experimental validation demonstrates the superior performance of our network for point cloud classification tasks, particularly achieving classification accuracies of 93.4% and 94.7% on the MVP and PCN benchmarks, respectively.

Our method is designed to accommodate the requirements of practical scenarios, with profound implications for future research and applications. In the future, we plan to apply this method to specific manipulation tasks to provide a cognitive foundation for robots.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 62088101), in part by the Science and Technology Commission of Shanghai Municipality (No. 2021SHZDZX0100, 22ZR1467100), and the Fundamental Research Funds for the Central Universities (No. 22120240291).

References

[Albert and Tri, 2023] Gu Albert and Dao Tri. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[Chen *et al.*, 2019] Chao Chen, Guanbin Li, Ruijia Xu, Tianshui Chen, Meng Wang, and Liang Lin. Clusternet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4994–5002, 2019.

[Cheng *et al.*, 2022] Ming Cheng, Guoyan Li, Yiping Chen, Jun Chen, Cheng Wang, and Jonathan Li. Dense point cloud completion based on generative adversarial network. *IEEE Transactions on Geoscience and Remote Sensing*, 60, 2022.

[Deng *et al.*, 2018] Haowen Deng, Tolga Birdal, and Slobodan Ilic. PPF-FoldNet: Unsupervised learning of rotation invariant 3d local descriptors. In *Proceedings of the European Conference on Computer Vision*, pages 602–618, 2018.

[Fan *et al.*, 2023] Zhaoxin Fan, Yulin He, Zhicheng Wang, Kejian Wu, Hongyan Liu, and Jun He. Reconstruction-aware prior distillation for semi-supervised point cloud completion. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 726–735, 2023.

[Fei *et al.*, 2022] Ben Fei, Weidong Yang, Wen-Ming Chen, Zhijun Li, Yikang Li, Tao Ma, Xing Hu, and Lipeng Ma. Comprehensive review of deep learning-based 3d point cloud completion processing and analysis. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):22862–22883, Dec. 2022.

[Han *et al.*, 2023] XianFeng Han, YiFei Jin, HuiXian Cheng, and GuoQiang Xiao. Dual transformer for point cloud analysis. *IEEE Transactions on Multimedia*, 25:5638–5648, 2023.

[Ju *et al.*, 2024] Mingye Ju, Siying Xie, and Fuping Li. Improving skip connection in u-net through fusion perspective with mamba for image dehazing. *IEEE Transactions on Consumer Electronics*, 70(4):7505–7514, 2024.

[Li *et al.*, 2018] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen. Pointcnn: Convolution on x-transformed points.

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

In *Proceedings of the Advances in Neural Information Processing Systems*, pages 820–830, 2018.

[Li *et al.*, 2021] Feiran Li, Kent Fujiwara, Fumio Okura, and Yasuyuki Matsushita. A closer look at rotation-invariant deep point cloud analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16218–16227, 2021.

[Li *et al.*, 2024] Liangliang Li, Guihua Liu, Feng Xu, and Lei Deng. Carvingnet: Point cloud completion by stepwise refining multi-resolution features. *Pattern Recognition*, 156:110780, 2024.

[Liang *et al.*, 2024] Dingkang Liang, Xin Zhou, Wei Xu, Xingkui Zhu, Zhikang Zou, Xiaoqing Ye, Xiao Tan, and Xiang Bai. Pointmamba: A simple state space model for point cloud analysis. In *Proceedings of the 2024 Conference on Neural Information Processing Systems*, 2024.

[Lin *et al.*, 2022] ZhiHao Lin, ShengYu Huang, and YuChiang Frank Wang. Learning of 3d graph convolution networks for point cloud analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4212–4224, Aug 2022.

[Liu *et al.*, 2023] Quan Liu, Yunsong Zhou, Hongzi Zhu, Shan Changd, and Minyi Guo. Apr: Online distant point cloud registration through aggregated point cloud reconstruction. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1204–1212, 2023.

[Mohammadi *et al.*, 2022] Seyed Saber Mohammadi, Yiming Wang, Matteo Taiana, Pietro Morerio, and Alessio Del Bue. Svp-classifier: Single-view point cloud data classifier with multi-view hallucination. In *Proceedings of the 2022 International Conference on Image Analysis and Processing*, volume 13232, pages 15–26, 2022.

[Pan *et al.*, 2023] Liang Pan, Xinyi Chen, Zhongang Cai, Junzhe Zhang, and Haiyu Zhao. Variational relational point completion network for robust 3d classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11340–11351, Sept 2023.

[Poulenard *et al.*, 2019] Alexis Poulenard, Marie-Julie Rakotosaona, Yann Ponty, and Mikhail Ovsjanikov. Effective rotation invariant point cnn with spherical harmonics kernels. In *Proceedings of the International Conference on 3D Vision*, pages 47–56, 2019.

[Qi *et al.*, 2017a] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017.

[Qi *et al.*, 2017b] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 5105–5114, 2017.

[Qian *et al.*, 2022] Rui Qian, Xin Lai, and Xirong Li. 3d object detection for autonomous driving: A survey. *Pattern Recognition*, 130(108796), 2022.

[Qiu *et al.*, 2022] Shi Qiu, Saeed Anwar, and Nick Barnes. Geometric back-projection network for point cloud classification. *IEEE Transactions on Multimedia*, 24:1943–1955, 2022.

[Rao *et al.*, 2019] Yongming Rao, Jiwen Lu, and Jie Zhou. Spherical fractal convolutional neural networks for point cloud recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–460, 2019.

[Sarmad *et al.*, 2019] Muhammad Sarmad, Hyunjoo Jenny Lee, and Young Min Kim. Rl-gan-net: A reinforcement learning agent controlled gan network for real-time point cloud shape completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5891–5900, Long Beach, CA, USA, 2019.

[Shen *et al.*, 2024] Wenxiang Shen, BaoYe Zhang, Hao Xu, XiaoHan Li, and Jun Wu. Multi-space point geometry compression with progressive relation-aware transformer. *IEEE Transactions on Multimedia*, 26:8969–8980, 2024.

[Shi *et al.*, 2023] Guangsheng Shi, Ke Wang, Ruifeng Li, and Chao Ma. Real-time point cloud object detection via voxel-point geometry abstraction. *IEEE Transactions on Intelligent Transportation Systems*, 24(6):5971–5982, June 2023.

[Singh and Yadav, 2024] Dheerendra Pratap Singh and Manohar Yadav. Deep learning-based semantic segmentation of three-dimensional point cloud: a comprehensive review. *International Journal of Remote Sensing*, 45(2):532–586, 2024.

[Uy *et al.*, 2019] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[Wang *et al.*, 2019] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics*, 2019.

[Wang *et al.*, 2024] Zhaoxuan Wang, Yunlong Yu, and Xianzhi Li. Rethinking local-to-global representation learning for rotation-invariant point cloud analysis. *Pattern Recognition*, 154(110624), 2024.

[Wei *et al.*, 2023] Mingqiang Wei, Zeyong Wei, Haoran Zhou, Fei Hu, Huajian Si, and Zhilei Chen. Agconv: Adaptive graph convolution on 3d point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):9374–9392, Aug. 2023.

[Wu *et al.*, 2015] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.

Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

[Xu *et al.*, 2023] Zelin Xu, Kangjun Liu, Ke Chen, Changxing Ding, Yaowei Wang, and Kui Jia. Classification of single-view object point clouds. *Pattern Recognition*, 135:109137, 2023.

[Yang *et al.*, 2024] Aitao Yang, Min Li, Yao Ding, Leyuan Fang; Yaoming Cai, and Yujie He. Graphmamba: An efficient graph structure learning vision mamba for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–14, 2024.

[Yu *et al.*, 2022] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, LA, USA, 2022.

[Yuan *et al.*, 2018] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *Proceedings of the International Conference on 3D Vision*, pages 728–737, 2018.

[Zhang *et al.*, 2019] Zhiyuan Zhang, Binh-Son Hua, David W. Rosen, and Sai-Kit Yeung. Rotation invariant convolutions for 3d point clouds deep learning. In *Proceedings of the International Conference on 3D Vision*, pages 204–213, Quebec City, QC, Canada, 2019.

[Zhang *et al.*, 2022] Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. Riconv++: Effective rotation invariant convolutions for 3d point clouds deep learning. *International Journal of Computer Vision*, 130:1228–1243, 2022.

[Zhang *et al.*, 2025] Zhiyuan Zhang, Licheng Yang, and Zhiyu Xiang. Risurconv: Rotation invariant surface attention-augmented convolutions for 3d point cloud classification and segmentation. In *Proceedings of the European Conference on Computer Vision*, volume 15086. Springer, Cham, 2025.

[Zhao *et al.*, 2021] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the International Conference on Computer Vision*, pages 16259–16268, 2021.

[Zhao *et al.*, 2022] Chen Zhao, Jiaqi Yang, Xin Xiong, Angfan Zhu, Zhiguo Cao, and Xin Li c. Rotation invariant point cloud analysis: Where local geometry meets global topology. *Pattern Recognition*, 127(108626), 2022.

[Zhou *et al.*, 2025] Weilian Zhou, Sei ichiro Kamata, Haipeng Wang, Man Sing Wong, and Huiying (Cynthia) Hou. Mamba-in-mamba: Centralized mamba-cross-scan in tokenized mamba model for hyperspectral image classification. *Neurocomputing*, 613:128751, 2025.