PERMUTATION-INVARIANT POLAR HARMONIC POOLING FOR POINT-BASED NEURAL NETWORKS

UNIVERSITY OF LINCOLN
LINCOLN INSTITUTE FOR AGRI-FOOD TECHNOLOGY

Jaspreet Singh and Grzegorz Cielniak

BMVC 2025

INTRODUCTION

- Point cloud learning is challenging due to their unordered, irregular, sparse structure and large data volume.
- Point-based networks typically include a permutation-equivariant backbone, symmetric pooling, and a permutation-invariant head.
- Common design: equivariant backbone → handles unordered data; symmetric pooling → creates fixed-size features; invariant head → enables robust classification.

RELATED WORKS

- **Classical Pooling (Max, Avg, GeM):** Simple and efficient but captures only first-order statistics → limited expressiveness.
- **Covariance Pooling:** Uses second-order stats for richer features but is computationally heavy and unstable on small datasets.
- **Optimal Transport Methods (OTKE, PSWE):** Capture distributional info well but involve complex optimization → slow and resource-intensive.
- **Learnable/Attention Pooling (PMA, FSPool):** More flexible and adaptive, but often high cost, sorting overhead, and risk of overfitting.

CONCLUSION

- We propose Polar Harmonic Pooling (PHP): a novel symmetric global pooling method that captures both spatial and frequency-domain features for point-based neural networks.
- PHP improves accuracy and generalization, while reducing overfitting and significantly cutting model complexity.
- PHP reduces over 93% of parameters in the classification/regression head, enabling a more compact and efficient architecture.
- While gains on large datasets (e.g., ModelNet40) are modest, PHP shows strong improvements on small datasets (ModelNet10, SGCBP), making it ideal for low-data scenarios.

OUR CONTRIBUTIONS

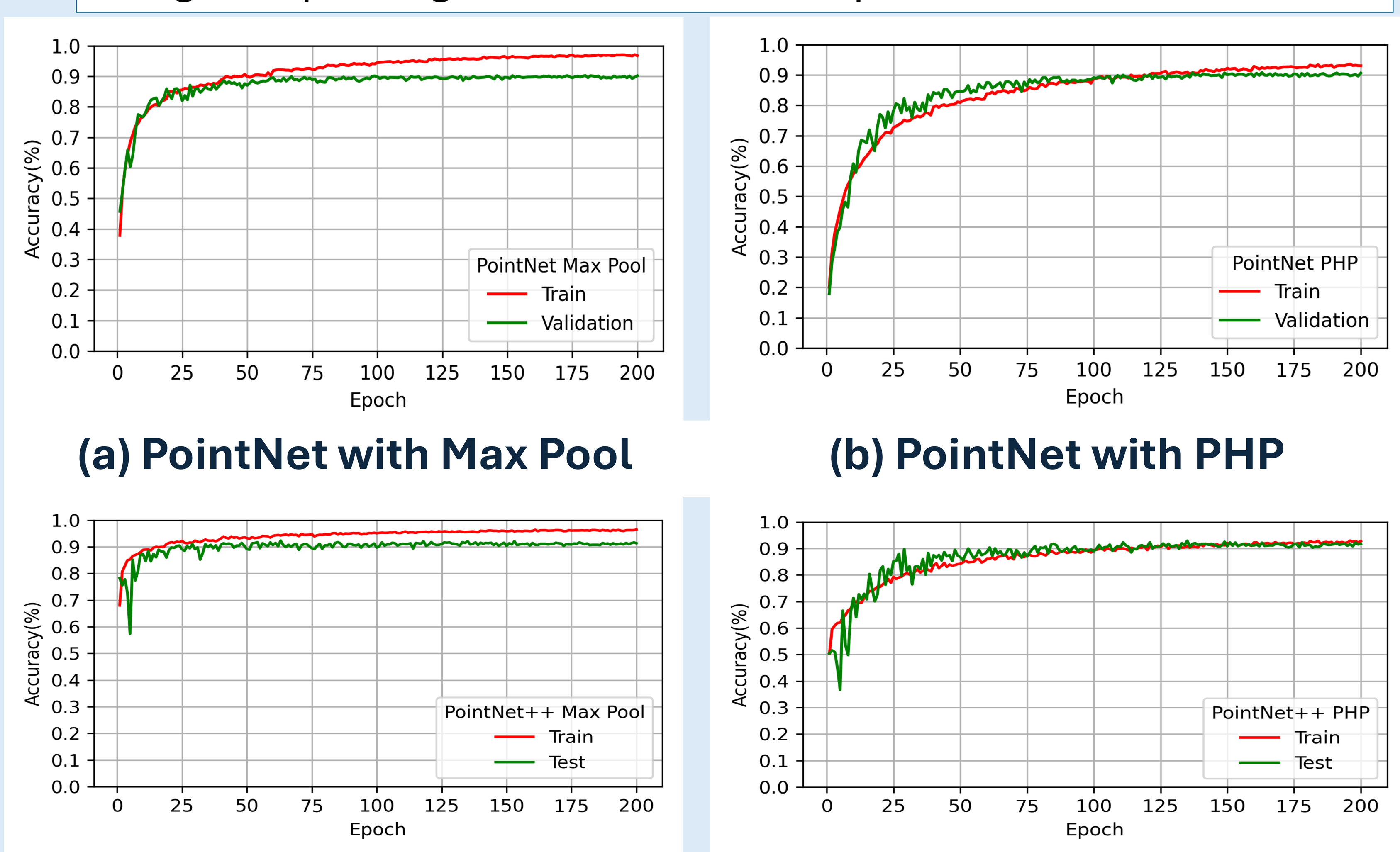
- We introduce Polar Harmonic Pooling (PHP) - a novel symmetric pooling method using polar harmonic transforms for full permutation invariance.
- PHP outperforms traditional pooling on ModelNet and SGCBP in both classification and regression tasks.
- PHP enhances generalization by reducing overfitting, improving accuracy, and lowering parameter count when integrated into PointNet and PointNet++.

RESULTS

Table 1. Classification accuracies on ModelNet40 and ModelNet10 using Max pooling, F-PSWE, and PHP with PointNet and PointNet++.

Pooling techniques	ModelNet40		ModelNet10	
	PointNet	PointNet++	PointNet	PointNet++
Max	89.2	90.6	90.6	92.2
F-PSWE	89.4	90.6	91.1	92.5
PHP (Ours)	**89.7**	**90.9**	**92.7**	**93.8**

Figure 2: Train and test accuracies of models on ModelNet40 using max pooling and PHP over 200 epochs.



(a) PointNet with Max Pool **(b) PointNet with PHP**

(c) PointNet++ with Max Pool **(d) PointNet++ with PHP**

Table 2. RMSE obtained on the SGCBP dataset using Max pooling, F-PSWE, and PHP with PointNet and PointNet++.

Pooling Techniques	PointNet	PointNet++
Max	100.2	90.7
F-PSWE	98.67	89.2
PHP (Ours)	**93.57**	**85.8**

Polar Harmonic Pooling

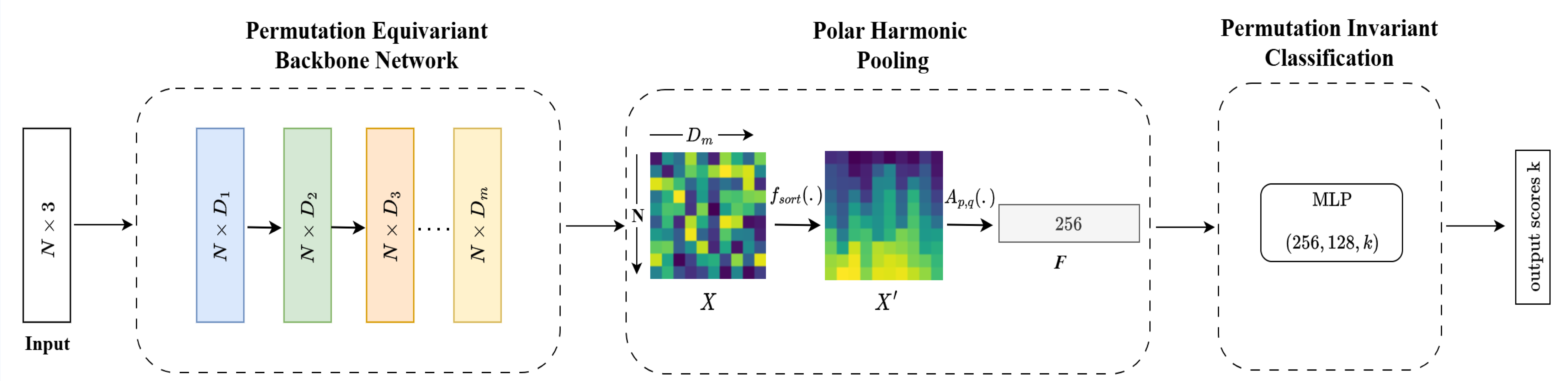


Figure 1: Polar Harmonic Pooling (PHP) pipeline illustrating the transformation and aggregation of point features in a point-based neural network.