

A STATE SPACE MODELS

SSMs are cyclic processes with latent states, which map a 1-D equation or sequence $x(t) \in \mathbb{R}^N$ to $y(t) \in \mathbb{R}^N$ by a latent state $h(t) \in \mathbb{R}^N$. The process is mathematically denoted as a linear ordinary differential equation as follows

$$y(t) = Ch(t), h'(t) = Ah(t) + Bx(t), \quad (7)$$

where the three parameters $A \in \mathbb{R}^{N \times N}$, $B \in \mathbb{R}^N$, and $C \in \mathbb{R}^N$ represent the state matrix, input matrix, and output matrix, respectively. Since the above SSMs run on continuous inputs and are not applicable to discrete inputs such as images and text, they cannot be introduced into deep models. Thus, it is necessary to discretize them, and the zero-order hold is commonly used as a discretization method. The discretized formulas are as follows

$$h_t = \bar{A}h_{t-1} + \bar{B}x_t, \quad y_t = Ch_t, \quad (8)$$

where \bar{A} and \bar{B} are the results of discretizing the continuous parameters A and B by a time scale Δ , denoted as

$$\bar{A} = e^{\Delta A}, \quad \bar{C} = C, \quad \bar{B} = (\Delta A)^{-1}(e^{\Delta A} - I)(\Delta B). \quad (9)$$

Since processing the input and latent state equally, previous approaches focusing on linear time-invariant SSMs (where \bar{A} and \bar{B} are invariant) may fail to capture critical information from context. Hence, Mamba proposes a novel SSM termed S6 by integrating an input-dependent selective mechanism into SSMs, where \bar{A} and \bar{B} are the functions of inputs, indicating Mamba is linear time-variant.

B DATASETS AND IMPLEMENTATION

B.1 DATASETS

ModelNet40 dataset contains 12,311 CAD models across 40 categories, with 9,843 samples in the training set and 2,468 samples in the test set. Data preprocessing follows the method of Qi et al. (2017a), where 1,024 points and their normal vectors are uniformly sampled from each sample as input. As per most relevant studies in the literature, the overall accuracy (OA) is adopted as an evaluation metric.

ShapeNet dataset comprises 16,878 samples from 50 parts across 16 categories, with 14,005 samples in the training set and 2,873 in the test set. Data preprocessing is consistent with that applied to ModelNet40 dataset. As per most relevant studies in the literature, the instance mIoU (Ins. mIoU) is adopted as an evaluation metric.

S3DIS dataset comprises 3D point cloud data from 271 indoor scenes across 6 areas, with each point annotated with one of 13 semantic labels. Data preprocessing follows the method of Qi et al. (2017a), where input features include point coordinates, RGB color, and normalized positions. As per most relevant studies in the literature, area 5 is used as the test set, while the remaining areas are used for training, and the mean IoU (mIoU) is adopted as an evaluation metric.

ScanObjectNN (PB_T50_RS variant) contains 14,450 valid samples across 15 categories, with 11,636 samples for training and 2,814 for testing. Except for using only coordinates as input, data preprocessing and evaluation metric are consistent with those used for ModelNet40 dataset.

B.2 IMPLEMENTATION

To better understand the model’s structure and implementation, we list detailed network architectures and training settings across different datasets in Tab. 11.

C MORE ABLATION STUDIES

C.1 ABLATION COMPARISON ON THE SERIALIZATION

Serialization. Based on point cloud serialization, the context-scan state space generates a continuous scanning path with inter-point structural dependencies. To validate the rationale behind selecting

Configurations	ModelNet40	ScanObjectNN	ShapeNet	S3DIS
Training epochs	500	500	600	500
Optimizer & Scheduler	Adamw & CosLR	AdamW & CosLR	Adamw & CosLR	Adamw & CosLR
Weight decay	0.01	0.01	0.01	0.01
Learning rate	8e-4	4e-4	1e-3	1e-3
Warmup epochs	10	20	10	10
Batch size	24	24	24	12
Embedding channels	48	48	96	48
KNN	8	8	8	8
IPP ratio	8	8	8	16
Encoder depth	[1, 1, 1, 1]	[1, 1, 1, 1]	[2, 2, 6, 2]	[1, 2, 3, 1]
Encoder channels	[48, 96, 192, 384]	[48, 96, 192, 384]	[96, 192, 384, 768]	[96, 192, 384, 768]
Decoder depth	-	-	[1, 1, 1, 1]	[1, 1, 1, 1]
Decoder channels	-	-	[768, 384, 192, 96]	[768, 384, 192, 96]
Downsampling stride	[4, 4, 4]	[4, 4, 4]	[4, 4, 4]	[4, 4, 4]
MLP ratio	4	4	4	4
QKV bias	True	True	True	True
Dropout	0.3	0.3	0.3	0.3
Augmentation	RandomScale RandomShift ShufflePoint	ShufflePoint RandomScale RandomRotate	RandomScale RandomShift ShufflePoint	RandomScale RandomFlip RandomJitter ChromaticAutoContrast ChromaticTranslation ChromaticJitter

Table 11: Detailed implementation configurations.

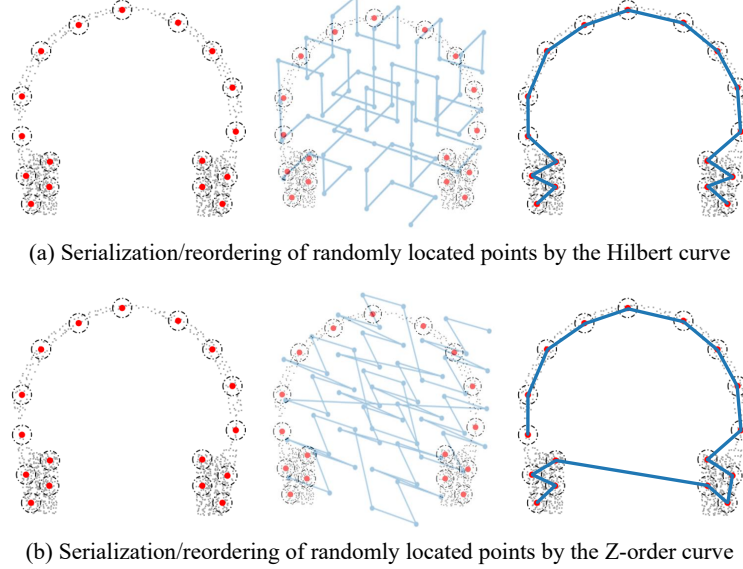


Figure 5: Comparison of the serialization of randomly located points by the Hilbert curve (top) and Z-Order curve (bottom).

the Hilbert curve, Tab. 12 compares the ablation results of various serialization strategies. When two serialization methods are employed, they are applied separately to the two directions of the bidirectional S6. Quantitative analysis indicates that while combining multiple serialization strategies can provide richer spatial information, differences in spatial relationships between these strategies may introduce confusion and interfere with the learning of spatial consistency. By leveraging its exceptional spatial locality-preserving property, as shown in Fig. 5, the Hilbert curve establishes a high-fidelity spatially adjacent scanning path with reliable inter-point structural dependencies for the state space model. This aligns with the visual search pattern of eye movements scanning continuously along spatially adjacent regions, thereby achieving an optimal balance between accuracy and efficiency. To further discuss the specific advantages of the Hilbert curve over learnable serial-

Serialization	Params	FLOPs	Throughput	OA
None	7.36M	0.610G	219FPS	91.34
Hilbert	7.36M	0.610G	163FPS	94.17
Z-Order	7.36M	0.610G	209FPS	93.06
Hilbert & Trans-Hilbert	7.36M	0.610G	133FPS	93.78
Hilbert & Z-Order	7.36M	0.610G	155FPS	93.52
Learnable Serialization	8.04M	0.723G	168FPS	92.78

Table 12: Ablation results with multiple serialization strategies.

ization strategies in terms of spatial locality preservation, continuity, and computational efficiency, we compare with the learnable serialization strategy from the latest research (Zha et al., 2025). It is intuitively observed that, compared to the learnable serialization, the Hilbert curve exhibits higher computational efficiency and superior spatial locality preservation and continuity. We attribute the poorer performance of the learnable serialization to the fact that it is an adaptive method for determining geometric correlation between points, but this approach possesses much less geometry-specific inductive biases compared to space-filling curves. In summary, the Hilbert curve introduces more precise inductive bias regarding geometric correlation compared to the Z-Order curve and learnable serialization strategies.

C.2 ABLATION COMPARISON OF IPP AND FPS

In our work, we employ the proposed Induced Point Pooling (IPP) for spatial downsampling. To investigate its ability to flexibly adapt to the non-uniform distribution of point clouds for global semantic integration, we present ablation results comparing IPP with the Farthest Point Sampling (FPS) at different sampling rates in Fig. 6, where $/N$ denotes the sampling rate relative to the input number of points, and None indicates the absence of the spatial downsampling branch. Intuitively, at low sampling rates, both methods exhibit comparable performance, indicating that FPS can obtain better global semantics with its excellent spatial coverage when sufficient sampling points are available. However, as the downsampling rate increases, the performance of FPS declines sharply. At a sampling rate of $/256$, its accuracy approaches the baseline without the downsampling branch, suggesting its inability to capture critical semantics from non-uniform point clouds at high sparsity. In contrast, IPP exhibits a more gradual performance decrease, maintaining an excellent accuracy of 93.39% even at the $/256$ sampling rate. This confirms that IPP, through trainable induced points that adaptively learn the point cloud distribution, can more flexibly and robustly integrate global semantics, providing effective coarse-grained information for the point-focused attention.

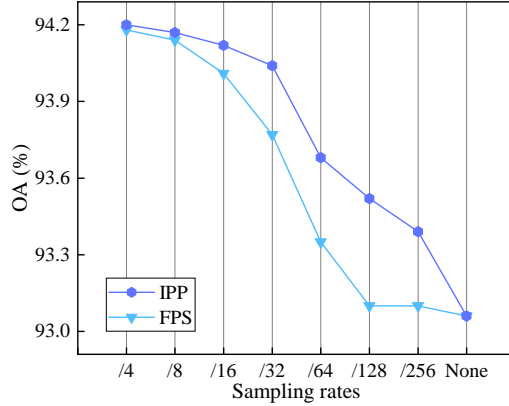


Figure 6: Quantitative results of IPP and FPS.

D COMPLEXITY ANALYSIS

Following the same settings as Eq.(6) and considering the feature transformation, the computational complexities of each module in the point-focused attention are as follows:

$$\begin{aligned}
 \Omega(\text{LNB}) &= 3ND^2 + 2NKD \\
 \Omega(\text{SSD}) &= ND^2 + 2MD^2 + 2NMD \\
 \Omega(\text{IPP}) &= 2ND^2 + 2NMD
 \end{aligned} \tag{10}$$

Finally, we have a complexity of $\Omega(\text{PFA}) = 6ND^2 + 2MD^2 + 2NKD + 4NMD$ for the point-focused attention. Since the context-scan state space inherits the linear complexity inherent in the state space model, the overall network exhibits linear complexity.

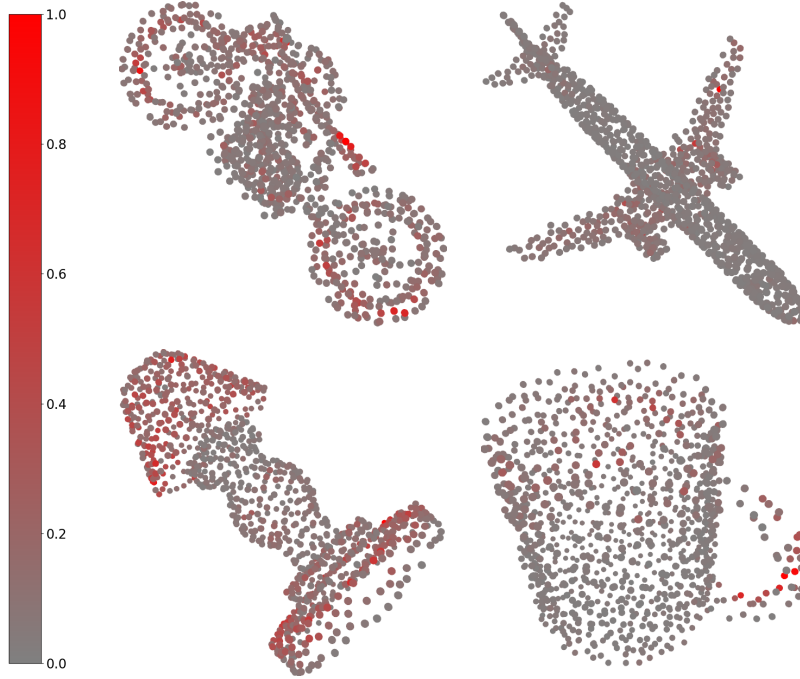


Figure 7: Visualization results of attention heatmaps.

E VISUALIZATIONS

E.1 ATTENTION HEATMAPS

To better understand the attention responses and the advantages of the proposed method, we present in Fig. 7 the attention heatmaps of PointLearner for different objects, generated from attention weights in the local neighbor branch of the last point-focused attention layer within the decoder. These attention heatmaps illustrate that, through a fully understanding of the bio-inspired visual perception, our method effectively focuses on critical information for semantic inference to achieve outstanding performance, such as the tires and seat on the motorcycle, as well as the base and cover of the lamp.

E.2 QUALITATIVE COMPARISON

To intuitively demonstrate the performance of our network, we present in Fig. 8 the visualization results of our network alongside the top-performing SSM method (PointMamba (Liang et al., 2024)) and attention method (GAD (Li et al., 2024b)) from Tab. 2 on ShapeNet dataset, where the red points denote that these points are misclassified. The comparison of the visualization results reveals that our network is able to achieve better part segmentation results at the boundaries of objects.

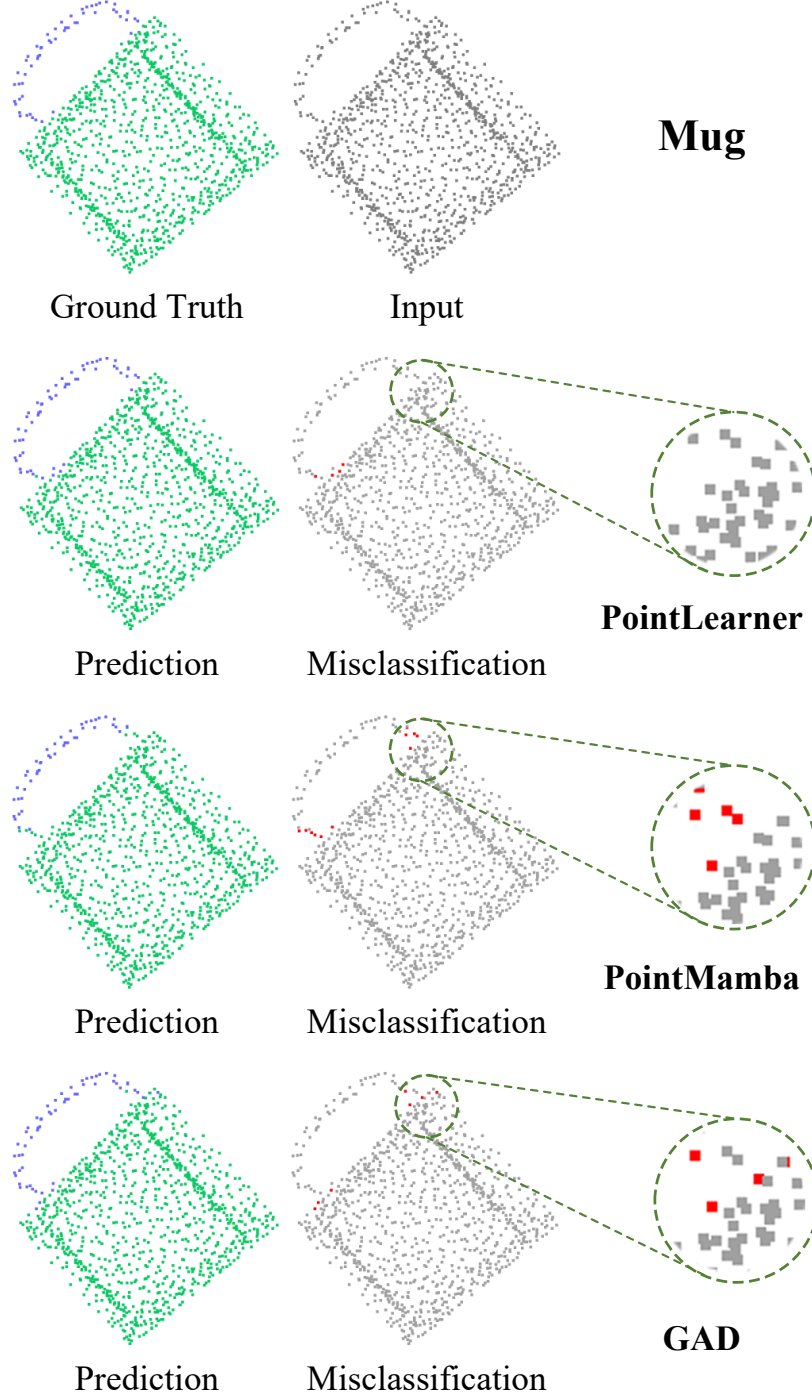
F FUTURE WORK

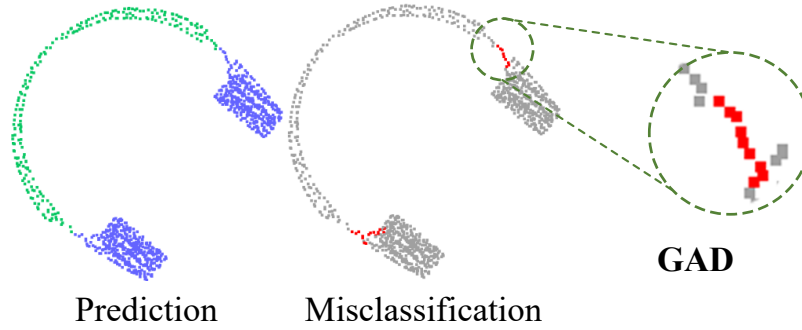
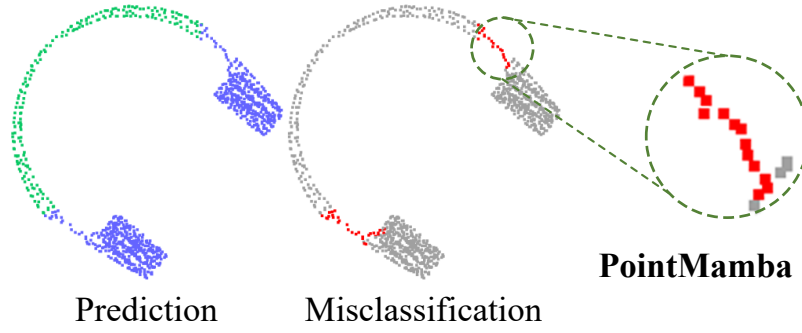
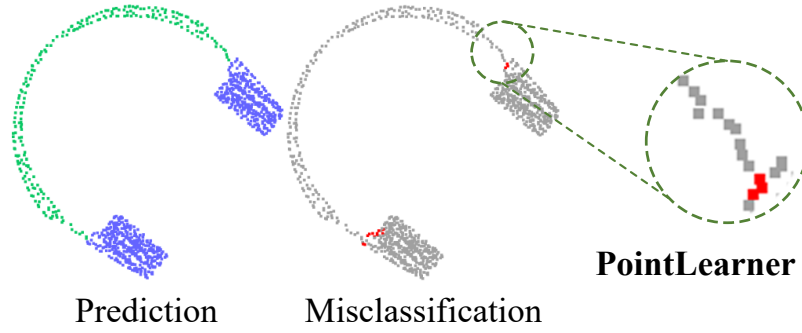
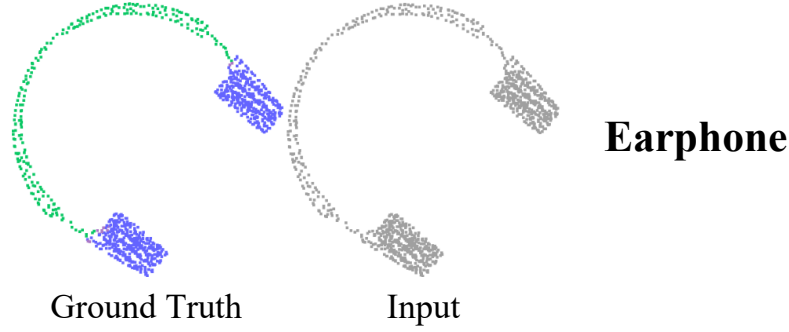
In our experimental comparisons, it is observed that most existing attention models employ pre-training methods to improve performance, with the self-supervised pre-training paradigm dominating. Self-supervised pre-training methods can leverage large amounts of unlabeled data to enhance feature modeling capabilities, as well as help Transformer models with large receptive fields achieve effective local or structural modeling by increasing data scale. Although self-supervised pre-training on large-scale point cloud datasets has been proven effective for improving the accuracy of Transformer models, the compatibility of existing Transformer self-supervised pre-training methods on hybrid architectures, as well as self-supervised pre-training strategies specifically tailored for hybrid architectures, remain underexplored. Hence, it is a promising direction for future research to collect

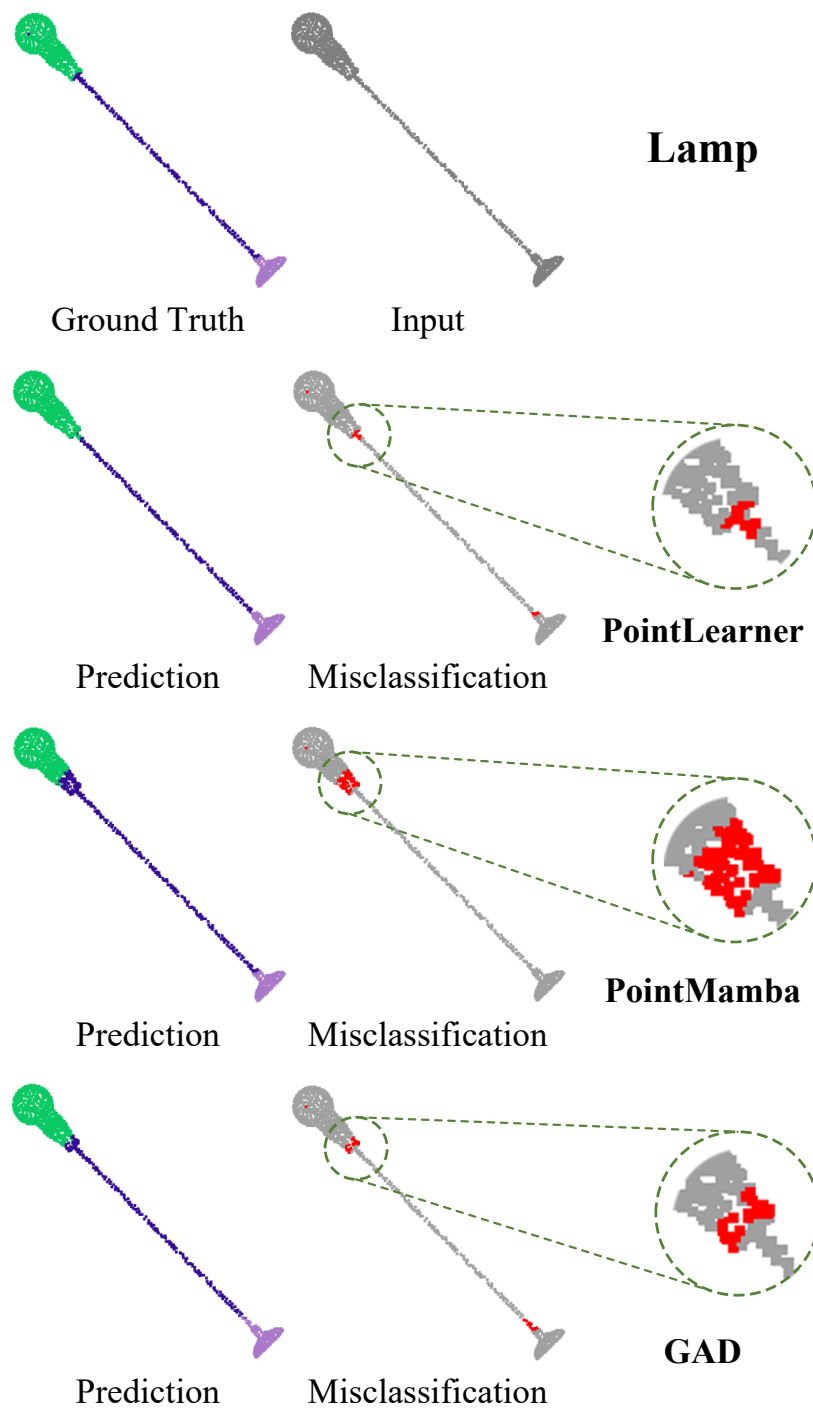
more data and design self-supervised learning methods for hybrid models, as in PPT (Wu et al., 2024b) and Sonata (Wu et al., 2025) designed for Transformer models.

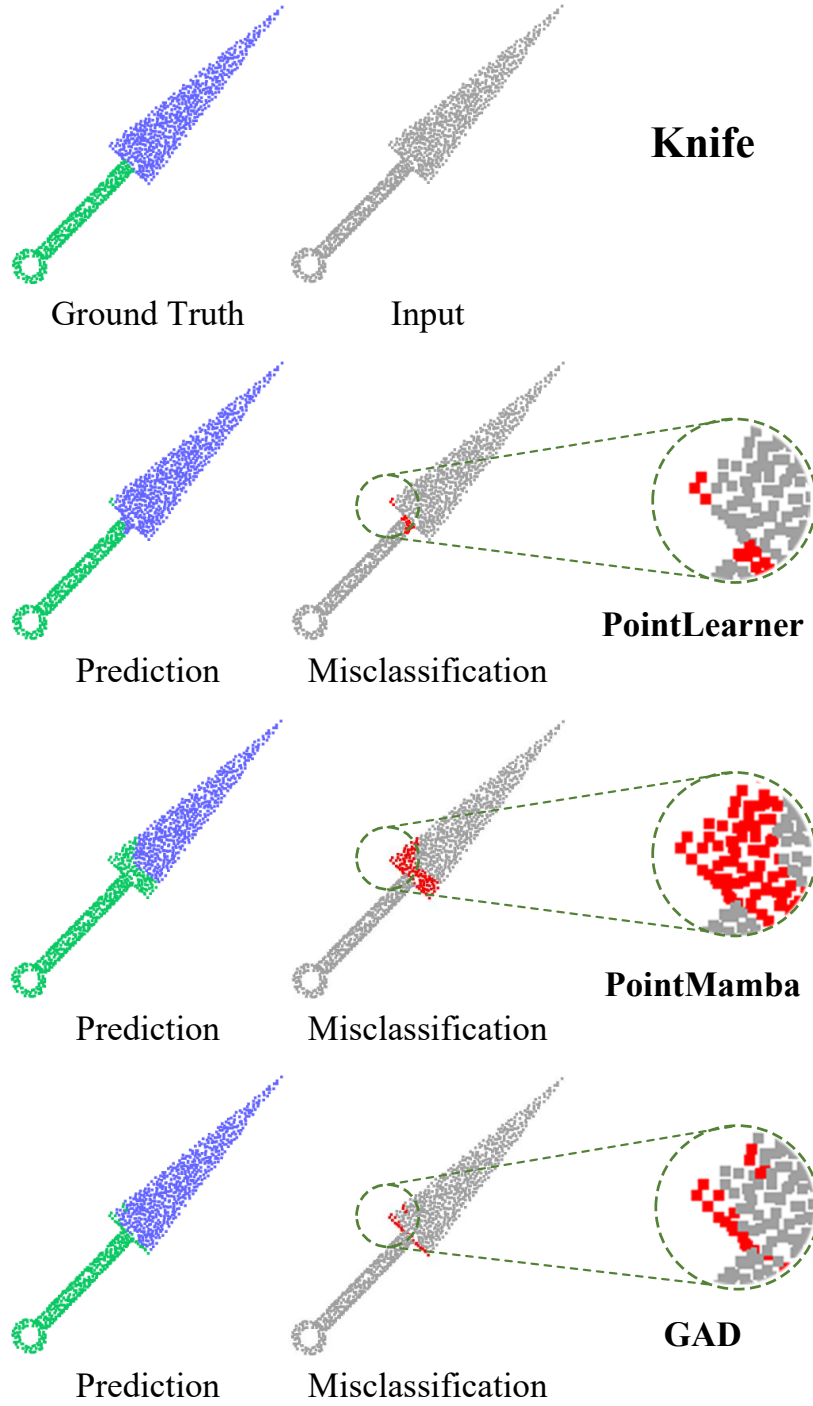
G THE USE OF LARGE LANGUAGE MODELS (LLMs)

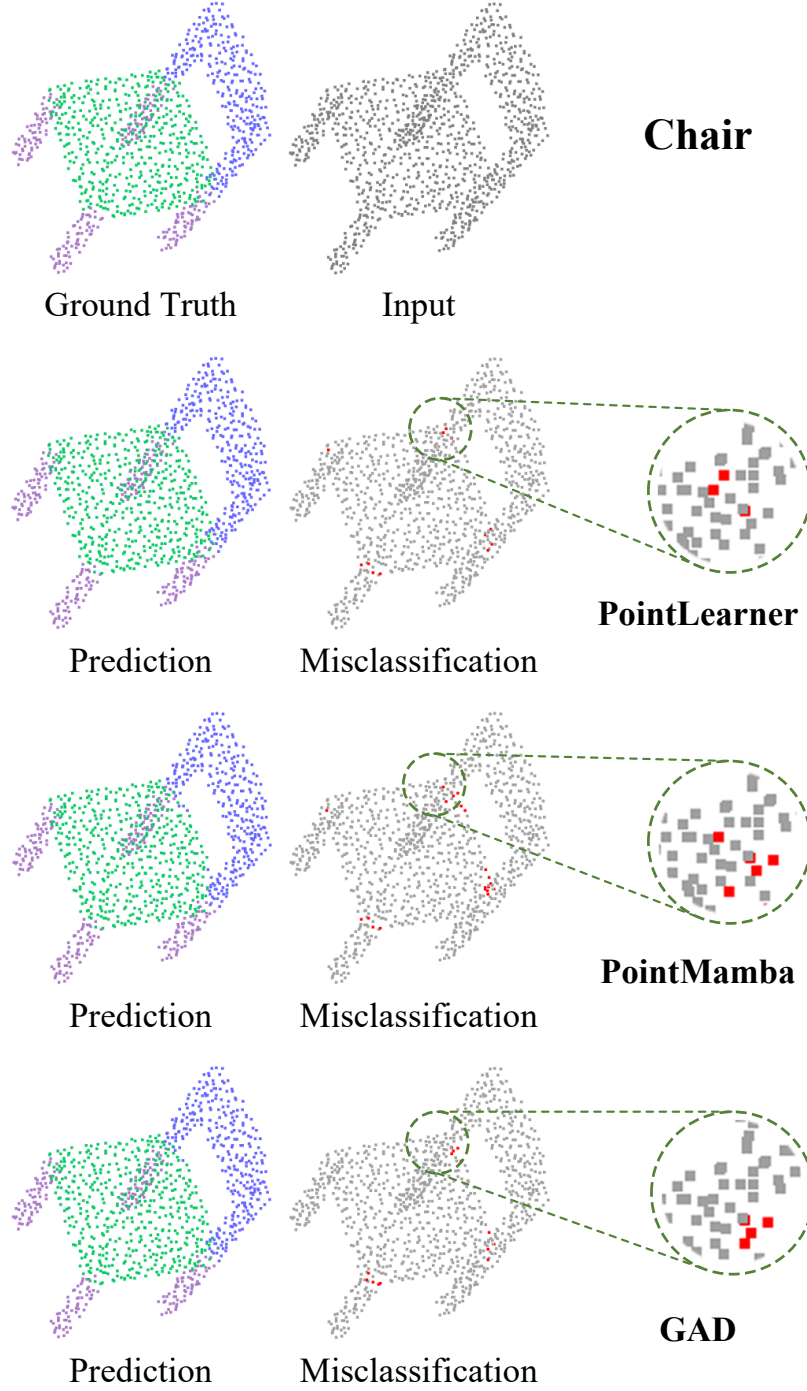
The authors utilized large language models (LLMs) to a limited extent for proofreading and improving grammatical correctness. All key aspects of the research, encompassing innovation, conceptual development, and literature discovery, were solely driven by the authors without LLM assistance.











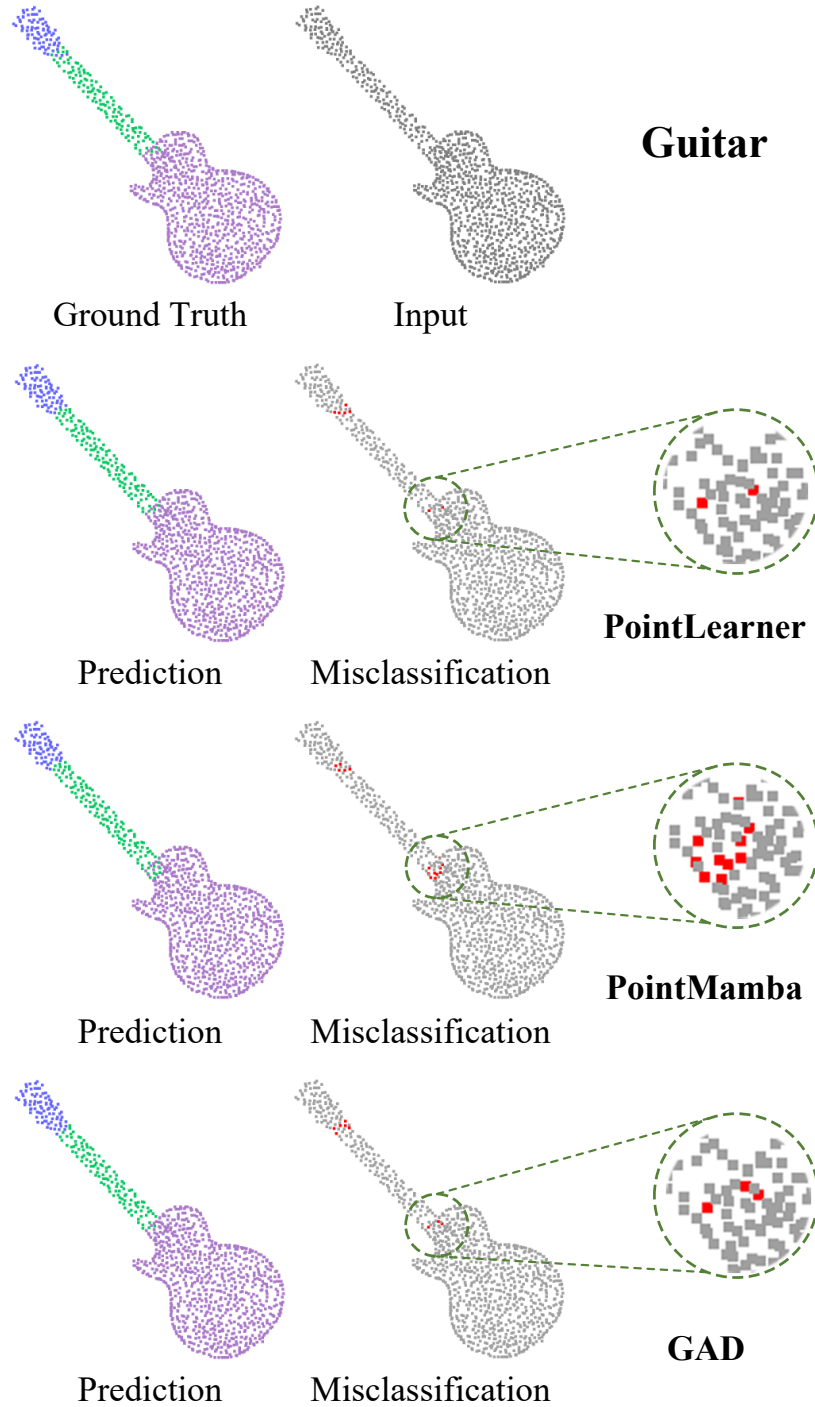


Figure 8: Visualization results of PointLearner, PointMamba, and GAD.