Prototype-based Multi-view Fine-Grained 3D Shape Classification and Ad-Hoc Interpretable Analysis

Authors: Shuxian Ma, Zihao Dong, Runming Cong, Sam Kwong, Xiuli Shao **Paper ID: 195**

BMVC 2025 The 36th British Machine Vision Conference

Introduction

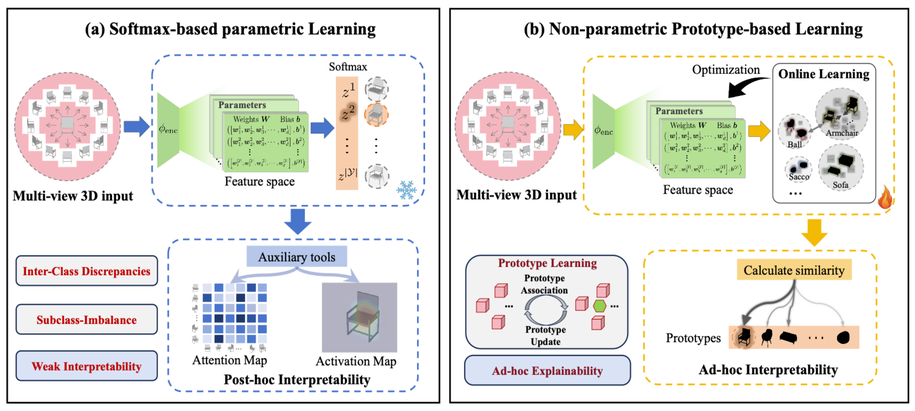
Problem: Multi-view 3D classification excels in coarse-grained tasks but struggles with fine-grained recognition.

Challenges:
·Local fine-grained inter-class discrepancies
·Severe subclass-imbalance
·Weak interpretability of visual relationships

Innovation:
First prototype-based framework for fine-grained 3D classification.

Paradigm Shift:
From parametric softmax to non-parametric prototype-based classification.

Rethinking Current Paradigm



Multi-view 3D shape classification paradigms:
(a) Parametric softmax can be interpreted as a learnable prototype-based approach, where class- and view-coupled prototypes are learned in a fully parametric manner.
(b) Non-parametric prototype learning directly identifies subcluster centers of embedded features as prototypes, enabling per-view predictions through nonparametric nearest prototype retrieval.

Visualizations

We analyze the interpretability of Proto-FG3D from both **global and local perspectives.**
The interpretable class prototypes are used to capture global 3D fine-grained shape characteristics, while the view-level class activation maps (CAMs) reveal local detail features

Table 2: Samples and corresponding prototypes respectively assigned to classes by our Proto-FG3D model during the training and testing stages.

Subcategory	Training	Prototype → Samples	Test Sample	Testing Top-3 Prototypes		
Majority class (train/test)	airliner (955/100)		 Armchair Similarity	 Armchair 0.8779	 Armchair 0.4751	 Armchair 0.1714
	helicopter (549/100)					
Minority class (train/test)	seaplane (35/20)		 Ball Similarity	Ball 0.7941	 Ball 0.4590	Straight 0.1431
	transport (15/7)					

Table 2 showcases **representative prototypes** selected from Airplane, Car, and Chair. For each dataset, including the top-3 prototypes from both majority and minority subclasses for each dataset.
The selected representative prototypes demonstrate significant diversity in both visual appearance and viewpoint variations, while maintaining clear discriminative patterns across subclasses in each dataset.
Notably, the framework consistently extracts semantically meaningful prototypes even for minority subclasses, demonstrating its robustness in addressing class imbalance challenges common in fine-grained 3D recognition tasks.
We also measure the similarity scores between input test samples and their assigned prototypes during inference. The top matched prototypes consistently achieve high similarity values across both majority and minority classes, confirming robust alignment between the learned prototype representations and the feature distributions of test instances.

Table 3: Sample explanation of predictions using top-5 representative views selected by Proto-FG3D with Grad-CAM.

Subcategory	Top-5 Representative View-level Prototypes					Prototypical Part
Airliner						
Similarity	0.6492	0.6054	0.5878	0.4653	0.3458	
Helicopter						
Similarity	0.7413	0.5759	0.5685	0.5245	0.4846	

View-Level CAMs. Table 3 demonstrates the explainability of FG3D predictions, revealing local interpretability through CAM visualization.
For each input sample, we present:
(i)the top-5 representative views with activation maps, selected by view-prototype similarity scoring, and (ii) their corresponding prototypical parts.
We observe that for majority classes such as Airliner and Helicopter, the selected views consistently capture key structural features (e.g., wing assemblies and rotor systems), highlighting the model's precise localization of discriminative class-specific regions.

Method Details

Overall Architecture

Three-stage prototype learning framework:
·Shared encoder ϕ_{enc} extracts feature embeddings $\mathbf{H} = \mathbf{h}j = 1^V \in \mathbb{R}^D$
·Dynamic prototype pool $\mathbf{P}$ with K trainable prototypes per class
·Joint optimization via optimal transport and metric learning

Prototype Association

Prototype Initialization:
·Cluster V^c multi-view representations into K class-specific sub-centroids.
· $\{\mathbf{q}_k^c\}_{k=1}^K \in \mathbb{R}^D$ stored in dynamic prototype pool.

Prototype Assignment via Optimal Transport:
Find assignment matrix $\mathbf{Z}^c \in [0,1]^{K \times V^c}$ that maximizes:
$\mathbf{Z}^{c*} = \max_{\mathbf{Z}^c} \mathrm{Tr}\left((\mathbf{Z}^c)^\top (\mathbf{Q}^c)^\top \mathbf{H}^c\right) - \mathrm{mean}_{\mathbf{Z}^c} \mathrm{Tr}\left((\mathbf{Z}^c)^\top (\mathbf{Q}^c)^\top \mathbf{H}^c\right) + \kappa \cdot H(\mathbf{Z}^c)$
Constraints: $(\mathbf{Z}^c)^\top \mathbf{1}_K = \mathbf{1}_{V^c},\ \mathbf{Z}^c \mathbf{1}_{V^c} = \frac{V^c}{K} \mathbf{1}_K$
Entropy Regularization: $H(\mathbf{Z}^c) = -\sum_{k=1}^K \sum_{v=1}^{V^c} z_{k,v}^c \log z_{k,v}^c$
Solution via Sinkhorn-Knopp: $\mathbf{Z}^{c*} = \mathrm{diag}(\mu) \exp\left(\frac{(\mathbf{Q}^c)^\top \mathbf{H}^c}{\kappa}\right) \mathrm{diag}(\nu)$

Online Prototype Update

Exponential Moving Average (EMA) Updating:
$$\mathbf{q}_k^c \leftarrow \eta^t \mathbf{q}_k^c(0) + (1-\eta) \sum_{i=1}^t \eta^{t-i} \bar{\mathbf{H}}_k^c(i)$$
Momentum Scheduling: $\eta^0 = 0.999$
for $t > T$ it dynamically adjusts as $\eta_t = \min(0.999, 1 - \frac{1}{t+1})$
Feature Aggregation: $\bar{\mathbf{H}}_k^c = \frac{1}{|\mathcal{S}_k^c|} \sum_{\mathbf{h}^c \in \mathcal{S}_k^c} \frac{1}{V^c} \sum_{i=1}^{V^c} \mathbf{h}_i^c$

Prototype-based Supervised Learning

Intra-class Alignment:
$d(\mathbf{h}_v, \mathbf{Q}^c) = \min_{1 \le k \le K} \left(1 - \mathbf{h}_v^\top \mathbf{q}_k^c\right)$
Cross-entropy Loss:
$$\mathcal{L}_{ce} = -\log \frac{\exp(-d(\mathbf{h}_v, \mathbf{Q}^c))}{\sum_{c' \in \mathcal{C}} \exp\left(-d(\mathbf{h}_v, \mathbf{Q}^{c'})\right)}$$
Inter-prototype Contrastive Learning:
$$\mathcal{L}_{pc} = -\log \frac{\exp\left(\mathbf{h}_v^\top \mathbf{q}_k^c / \tau\right)}{\exp\left(\mathbf{h}_v^\top \mathbf{q}_k^c / \tau\right) + \sum_{\mathbf{q}^- \in \mathcal{Q}^-} \exp(\mathbf{h}_v^\top \mathbf{q}^- / \tau)}$$
Total Loss:
$\mathcal{L}_{total} = \mathcal{L}_{ce} + \alpha \mathcal{L}_{pc}$

Prototype-based Inference

Nearest-Prototype Classification:
$\hat{c}_v = \mathrm{argmin}_c \left(\min_{1 \le k \le K} \left(1 - \mathbf{h}_v^\top \mathbf{q}_k\right)\right)$

Key Advantages:

•Transparent case-based reasoning

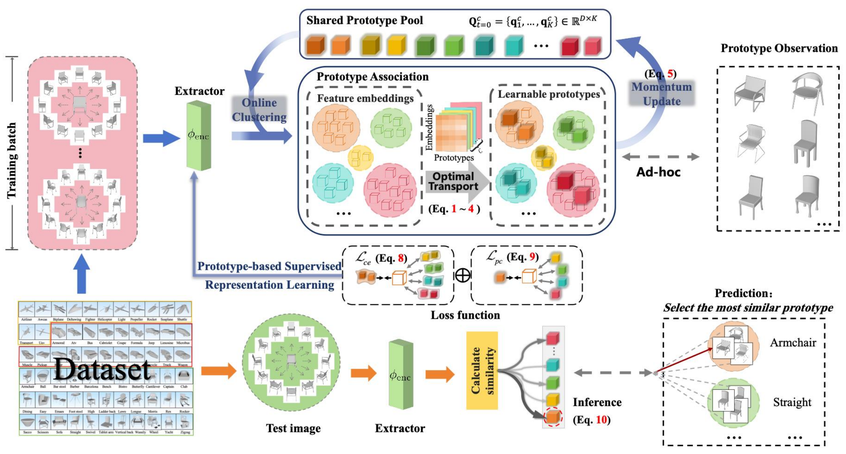
•Built-in interpretability via prototype visualization

•Robust to class imbalance through balanced clustering

Implementation Parameters:

•Prototypes per class: $K = 20$
•Temperature: $\tau = 0.1$
•Loss weight: $\alpha = 0.2$
•Initial momentum: $\eta^0 = 0.999$

Method's Architecture



Overall Architecture. As illustrated in Fig. 2, the proposed online clustering framework comprises a shared encoder ϕ_{enc}, a prototype pool $\mathbf{P}$, and a prototype association mechanism. The shared encoder extracts feature embeddings $\mathbf{H} = \{\mathbf{h}_j \in \mathbb{R}^D\}_{j=1}^V$ from V input views, where D denotes the feature dimension. To enable discriminative feature learning, we maintain a shared pool of K trainable prototypes per class, denoted as $\mathbf{Q}^c = \{q_k^c\}_{k=1}^K \in \mathbb{R}^{D \times K}$, and formulate an optimal transport problem to softly assign each feature embedding z_j to these prototypes as a probability distribution $\mathbf{q}_j \in \mathbb{R}^K$ $(\|\mathbf{q}_j\|_2 = 1)$. These class-specific prototypes establish multi-view correlations through **Prototype Association** during training, with the shared prototype pool being progressively refined via exponential moving averages of assigned features (i.e., **Prototype Update**). The model is optimized using a joint objective function that enforces feature-prototype similarity maximization while separating dissimilar pairs (i.e., **Prototype-based Supervised Representation Learning**). During inference, the final prediction for a test sample is determined by retrieving the nearest prototype in the shared pool based on the learned feature embeddings.

Evaluations

Datasets: FG3D (fine-grained) & ModelNet40 (coarse-grained)
Baselines: CNN-based (MVCNN-new, GVCNN, SMVCNN) & ViT-based (DAN, VSFormer) SOTA methods
Metrics: Dual evaluation with Average Instance Accuracy (AIA) & Average Class Accuracy (ACA)
Key Results: Comprehensive performance improvements in accuracy and efficiency

Table 1: Performance comparison of Proto-FG3D with state of the art methods on the FG3D and ModelNet40 dataset, evaluating K=20 prototypes and different backbone architectures.

Method	Airplane		Car		Chair		ModelNet40		Time
	AIA (%)	ACA (%)	AIA (%)	ACA (%)	AIA (%)	ACA (%)	AIA (%)	ACA (%)	
FG3D-Net [22]	93.99	89.44	**79.47**	74.03	83.94	80.04	-	-	-
MVCNNnew [34]	93.85	89.16	76.20	73.25	83.94	79.58	93.48	91.73	1h 53min
MVCNNnew+Ours (A)	94.40 ↑ 0.55	89.80 ↑ 0.64	76.96 ↑ 0.76	75.25 ↑ 2	84.46 ↑ 0.52	80.47 ↑ 0.89	94.57 ↑ 1.09	91.58 ↓ 0.15	-
MVCNNnew+Ours (S)	94.81 ↑ 0.96	91.54 ↑ 2.38	77.11 ↑ 0.91	**77.25** ↑ 4	84.56 ↑ 0.62	79.33 ↓ 0.25	94.57 ↑ 1.09	91.58 ↓ 0.15	**1h 23min**
GVCNN [11]	94.13	91.95	76.50	73.95	83.32	78.77	94.57	91.95	2h 11min
GVCNN+Ours (A)	**95.63** ↑ 1.5	**93.95** ↑ 2	78.02 ↑ 1.52	75.17 ↑ 1.22	84.15 ↑ 0.83	80.36 ↑ 1.59	95.26 ↑ 0.69	92.61 ↑ 0.66	-
GVCNN+Ours (S)	95.08 ↑ 0.95	93.49 ↑ 1.54	77.41 ↑ 0.91	76.88 ↑ 2.93	84.05 ↑ 0.73	80.12 ↑ 1.35	95.18 ↑ 0.61	92.50 ↑ 0.55	**1h 40min**
SMVCNN [19]	93.85	91.36	76.65	74.83	83.01	79.82	94.73	92.51	1h 57min
SMVCNN+Ours (A)	94.81 ↑ 0.96	90.98 ↓ 0.38	77.49 ↑ 0.84	76.42 ↑ 1.59	83.73 ↑ 0.72	80.09 ↑ 0.27	95.22 ↑ 0.49	92.31 ↓ 0.2	-
SMVCNN+Ours (S)	95.22 ↑ 1.37	93.10 ↑ 1.74	77.57 ↑ 0.92	74.63 ↓ 0.2	84.25 ↑ 0.52	80.07 ↑ 0.25	95.10 ↑ 0.37	92.19 ↓ 0.32	**1h 43min**
DAN [28]	94.81	91.34	75.36	72.78	83.63	80.52	95.02	92.47	3h 24min
DAN+Ours (A)	95.08 ↑ 0.27	93.05 ↑ 1.71	76.35 ↑ 0.99	71.07 ↓ 1.71	84.09 ↑ 0.46	**81.31** ↑ 0.79	**95.54** ↑ 0.52	93.19 ↑ 0.72	-
DAN+Ours (S)	94.67 ↓ 0.14	93.51 ↑ 2.17	76.96 ↑ 1.6	73.67 ↑ 0.89	83.63 ↑ 0	80.21 ↓ 0.31	94.98 ↓ 0.04	93.04 ↑ 0.57	**2h 30min**
VSFormer [35]	93.44	89.49	75.97	73.98	83.63	80.11	95.26	92.89	2h 28min
VSFormer+Ours (A)	94.81 ↑ 1.37	92.85 ↑ 3.36	77.11 ↑ 1.14	76.15 ↑ 2.17	84.31 ↑ 0.68	81.09 ↑ 0.98	95.30 ↑ 0.04	92.89 ↑ 0	-
VSFormer+Ours (S)	94.81 ↑ 1.37	92.46 ↑ 2.97	77.41 ↑ 1.44	75.60 ↑ 1.62	**84.72** ↑ 1.09	81.05 ↑ 0.94	95.50 ↑ 0.24	**93.27** ↑ 0.38	**1h 47min**

Table 4: The ablation study of (a) loss function, (b) prototype number, (c) momentum coefficient, and (d) loss weight on Proto-FG3D performance for FG3D-Airplane dataset.

(a) Loss function.

$\mathcal{L}_{ce}$	$\mathcal{L}_{pc}$	AIA (%)	ACA (%)
✓	✗	94.67	92.38
✓	✓	**95.36**	**92.95**

(b) Prototype number.

K	AIA (%)	ACA (%)
10	94.81	93.15
15	94.67	92.72
20	**95.36**	**92.87**
25	94.95	92.77
30	95.08	92.56

(c) Momentum coefficient.

η^0	AIA (%)	ACA (%)
0	95.08	92.51
0.9	95.08	92.77
0.99	95.08	93.23
0.999	**95.63**	**93.95**
0.9999	94.95	92.62

(d) Loss weight.

α	AIA (%)	ACA (%)
0.05	94.26	92.15
0.10	95.36	92.87
0.15	94.40	91.67
0.20	**95.63**	**93.95**
0.25	95.22	93.28

Conclusion & Future Work

In this paper, we proposed Proto-FG3D, **the first prototype-based framework for fine-grained 3D shape classification**. Departing from conventional softmax-based classifiers, Proto-FG3D employs prototype learning to simultaneously: **(i)** enhance inter-subcategory discrimination through online clustering that dynamically updates view-specific prototype representations, **(ii)** mitigate class imbalance via adaptive prototype allocation, and **(iii)** provide built-in interpretability through visualizable prototypes. Experimental results on both FG3D (fine-grained) and ModelNet40 (standard) benchmarks demonstrate that Proto-FG3D achieves state-of-the-art classification accuracy while providing comprehensive interpretability through global prototype visualization and local view-activation mapping.