Dual-Geometry Graph Network: Unifying Local and Global Priors for Few-Shot Learning

Zheng Han[1], **Xiaobin Zhu**[1], **Chun Yang**[1], **Jingyan Qin**[1,*], **Xucheng Yin**[1]

[1]University of Science and Technology Beijing, Beijing, China
han970421@foxmail.com, zhuxiaobin@ustb.edu.cn, chunyang@ustb.edu.cn, qinjingyanking@foxmail.com, xuchengyin@ustb.edu.cn

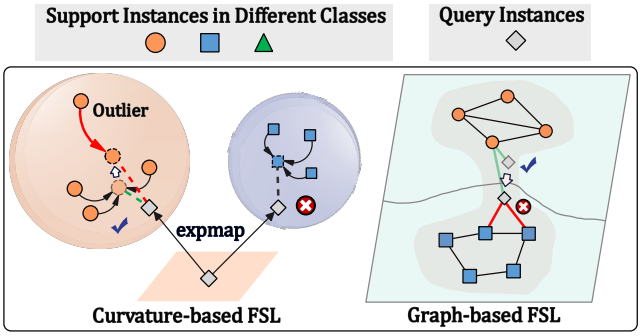
Abstract

In few-shot learning, utilizing local and global geometric priors to capture both subtle local class metrics and coarse global structures within the meta-task are important to obtain discriminative embeddings. However, existing graph-based and curvature-based few-shot approaches only focus on either one kind of geometric prior but neglect the other. To effectively utilize the pros of these two paradigms, we propose a novel Dual-Geometry Graph Network (DGGN) to adaptively integrate the local and global geometric priors via two key pathways. Specifically, the local-wise metric modeling pathway utilizes Ollivier-Ricci curvature to capture task-specific local class metrics among the instances, and the global-wise connectivity modeling pathway utilizes resistive embedding to capture global instance distributions and connectivity patterns of the entire meta-task. In addition, we introduce two new regularization loss functions to explicitly enhance the geometric representation ability of the local and global pathways respectively. We validate that DGGN's superior performance stems from its adaptively topological refinements by measuring the graph edit distance, demonstrating its ability to match the underlying data distribution. Extensive experiments show that DGGN sets a new state-of-the-art on standard, cross-domain, and semi-supervised few-shot benchmarks.

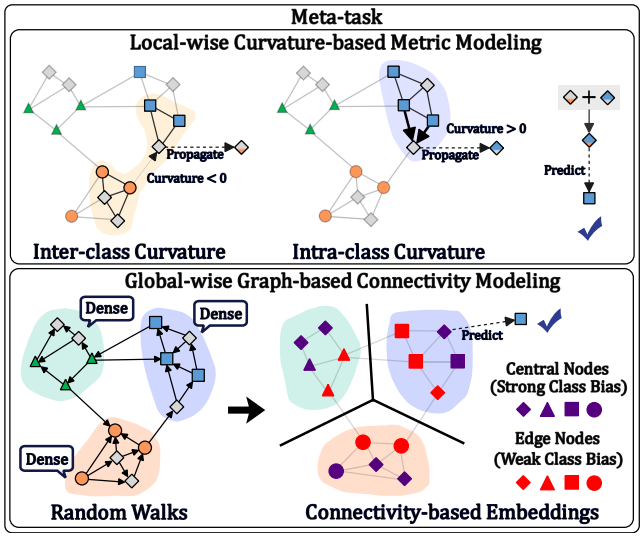
Introduction

Deep learning has achieved remarkable success across a wide range of computer vision tasks including image recognition (Fang et al. 2025), object detection (Zhang et al. 2024b), and image enhancement (Zhou et al. 2025, 2023b). However, this is mainly due to the availability of large-scale annotated datasets, which is often difficult and time-consuming to obtain in real-world scenarios. To tackle this issue, Few-Shot Learning (FSL) (Trosten et al. 2023; Zhu et al. 2023; Zhang et al. 2024a; Han et al. 2024) has emerged as an essential research field, alleviating the burden of data collection and annotation, reducing computational costs, and accelerating model generalization in low-data regimes.

Recently, most FSL approaches mainly focus on capturing global structural information across the entire meta-task, among which the Graph Neural Network (GNN) has shown substantial progress (Chen et al. 2021; Zhu et al. 2023; Yang et al. 2020). These GNN-based approaches typically construct a graph encompassing all instances within a meta-task.



(a) Existing flaws of curvature-based and graph-based FSL.



(b) Our utilization of local and global geometric prior.

Figure 1: (a) (*left*) Biased prototype caused by overlooking global-wise instance distributions. (*right*) Incorrect graph construction due to ignoring local-wise class manifolds; (b) Our method uses local and global geometric priors to provide robust class-confidence biases.

*Corresponding Author.

Message passing is then performed over this holistic graph to refine node embeddings (Yang et al. 2020; Chen et al. 2021) or propagate label information (Rodríguez et al. 2020; Zhu et al. 2023). While this global prior of the meta-task can capture broad relationships among the instances, these approaches still have a critical limitation: they implicitly assume the constructed graph structure is already optimal, which fails to introduce a fine-grained and variable class-wise bias into the message-passing process, causing wrongly connected links between samples from different classes, as shown in Figure 1 (a) (*right*).

Complementary to GNN-based methods, another line of research delves into the nuanced geometric properties in the meta-task, often by embedding instances into non-Euclidean spaces. These methods include learning task-aware embeddings in hyperbolic spaces (Gao et al. 2021), adaptive point-to-set hyperbolic metrics for better context-aware distance modeling (Ma et al. 2022), and hyperspherical embeddings to address intrinsic issues like "hubness" in high-dimensional embedding distributions (Trosten et al. 2023). While these methods significantly improved performance by emphasizing subtle local geometric priors within curved embedding spaces, a potential issue arises as they are based on the one-step nearest-neighbor prototype-based classification. This paradigm limits their ability to capture the overall instance distribution and connectivity patterns within the entire meta-task, making them sensitive to outliers that do not align well with the assumed local fine-grained geometry, as shown in Figure 1 (a) (*left*).

To effectively utilize the pros of these two paradigms, we first model them into two different pathways, aiming to capture different geometric biases: the local-wise pathway using Ollivier-Ricci Curvature (ORC) to capture nuanced, fine-grained class metrics, and the global-wise pathway based on resistive embeddings to capture coarse, long-range instance distributions and connectivity patterns. We confirm that these two pathways learned different and complementary geometric representations through empirical analysis by measuring the graph topological similarity from the learned embeddings. Based on this observation, we propose the **D**ual-**G**eometry **G**raph **N**etwork (**DGGN**), a unified framework designed to adaptively integrate the local and global geometric priors. We validate that DGGN's superior performance stems from our innovative design by showing that it performs high-adaptability topological refinements, and demonstrate that its adaptive ability sets a new state-of-the-art on standard, cross-domain, and semi-supervised FSL benchmarks. Our main contributions are threefold:

- We propose DGGN, the first framework designed to adaptively integrate the local and global geometric priors for few-shot learning, and we provide empirical evidence for its mechanism via graph topological analysis.
- We propose a new curvature-informed regularization loss function to enhance the fine-grained metric variations among the instances.
- We propose a new connectivity-informed regularization loss function to strengthen the global instance distribution and connectivity within the entire meta-task.
- DGGN achieves state-of-the-art performance across standard, cross-domain, and semi-supervised FSL benchmarks, proving its superior robustness and generalization on scarce labels and severe domain shift.

Related Works

Graph-based Few-shot Learning. Benefited from recent progress of GNNs for relation modeling (Kipf and Welling 2017; Velickovic et al. 2018), researchers proposed many graph-based approaches to address the FSL problem. GNN-FSL (Satorras and Estrach 2018) first proposed an end-to-end trainable GNN model for message passing among instances in the meta-task. DPGN (Yang et al. 2020) proposed to explicitly propagate distribution-level relationships between nodes by iteratively updating dual graphs in each meta-task. EPNet (Rodríguez et al. 2020) introduced embedding propagation as a non-parametric regularizer that interpolates between features on a similarity graph to achieve manifold smoothing. ECKPN (Chen et al. 2021) proposed to integrate semantic information from class labels into propagation to promote the quality of class prototypes. ProtoLP (Zhu et al. 2023) proposed an iterative framework that alternates between estimating class prototypes and propagating labels over a dynamically refined graph. Although these approaches show great potential of graph-based relation modeling on a global view of the meta-task, they ignore the varying and nuanced local-wise metrics among instances from different classes. In this work, we utilize ORC to model local-wise metrics among the instances, providing a nuanced class-confidence bias for relation modeling with GNN.

Curvature-based Few-shot Learning. Learning informative embeddings in curved representation spaces is an effective way for FSL. Hyperbolic geometry is a common choice due to its capacity to model inherent hierarchical relations. Gao *et al.* (Gao et al. 2021) utilized task-wise curvature-adaptive hyperbolic spaces to explore class-specific geometric spaces for each class in the meta-task. APP2S (Ma et al. 2022) proposed to learn task-specific point-to-set hyperbolic distances to fine-tune the decision boundary for outliers. HyperKT (Zhang et al. 2022) introduced prior knowledge from class hierarchies to supervise the prototype-based classification in hyperbolic spaces. Besides hyperbolic geometry, hyperspherical space were also explored. NoHub (Trosten et al. 2023) proposed to reduce the hubness problem in prototype-based classification via embedding instances into a hyperspherical space. These approaches show their effectiveness via emphasizing local-wise metrics. However, they lack a sufficient global view of the instance distributions and connectivity patterns within the entire meta-task. In this work, we introduce resistive embeddings to model the connectivity and structural roles among the instances, providing a global prior of the meta-task.

Methodology

Problem Definition

Few-shot learning is commonly formulated as a meta-learning problem, where the model is meta-trained on a distribution of meta-tasks using the training set $\mathbb{D}^{train}$ and

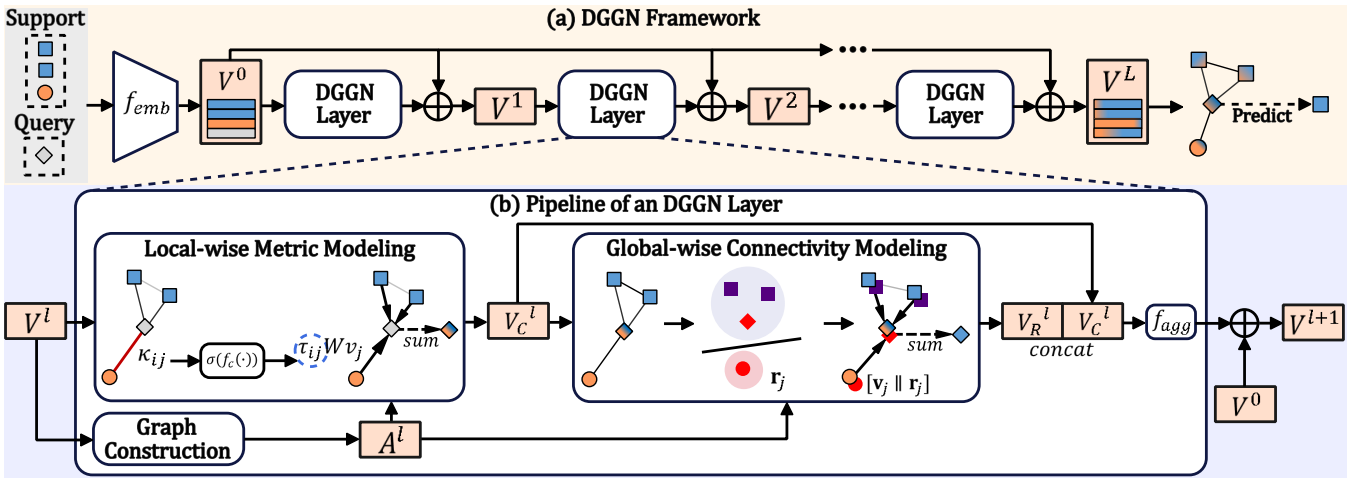


Figure 2: Illustration of DGGN framework. (a) Overall DGGN framework; (b) Details of an DGGN layer, including two key pathways: local-wise metric modeling pathway and global-wise connectivity modeling pathway. In l-th DGGN layer, a k-Nearest Neighbor (k-NN) graph is first constructed with adjacency matrix A^l using the node embedding matrix V^l. Secondly, we calculate the ORC κ_{ij} for each edge, following a curvature-guided message passing to produce the curvature-informed node embedding matrix V_C^l. Thirdly, the connectivity-based embedding $\mathbf{r}_j$ is computed for each node $\mathbf{v}_j$, then concatenated with $\mathbf{v}_j$ to perform a connectivity-guided message passing and produce the connectivity-informed node embedding matrix V_R^l. Finally, V_C^l and V_R^l are aggregated and added to the initial node embedding matrix V^0 to produce V^{l+1} of $(l+1)$-th DGGN layer.

meta-test using new unseen meta-tasks from the testing set $\mathbb{D}^{test}$. In the N-way K-shot paradigm, each meta-task consists of a support set $\mathcal{S}$ and a query set $\mathcal{Q}$. The support set $\mathcal{S}$ comprises N different classes sampled from the training class set $\mathbb{C}^{train}$, each with K labeled instances: $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{NK} \subset \mathbb{D}^{train}$, where x_i denotes the support instance and y_i is its corresponding label. The query set $\mathcal{Q}$ consists of unlabeled instances from the same N classes: $\mathcal{Q} = \{(x_j)\}_{j=1}^{T} \subset \mathbb{D}^{train}$, where x_j denotes the query instance and T is the number of query instances. The goal is to train a model on $\mathbb{D}^{train}$ and, given the support set $\mathcal{S} \subset \mathbb{D}^{test}$, the model can accurately classify the instances from the query set $\mathcal{Q} \subset \mathbb{D}^{test}$ into the N unseen classes from the testing class set $\mathbb{C}^{test}$. The class labels from $\mathbb{D}^{train}$ and $\mathbb{D}^{test}$ are mutually exclusive: $\mathbb{C}^{train} \cap \mathbb{C}^{test} = \emptyset$.

Dual-Geometry Graph Network

As shown in Figure 2 (b), each DGGN layer comprises three stages: graph construction, local-wise metric modeling, and global-wise connectivity modeling. We unify the local and global pathways in a cascading and residual manner, as shown in Figure 2 (a). For each instance image $\mathbf{x}_i$, we use the backbone $f_{emb}(\cdot)$ to extract its d-dimensional embedding: $\mathbf{v}_i^0 = f_{emb}(\mathbf{x}_i) \in \mathbb{R}^d$. Then the node embedding matrix $V^0 \in \mathbb{R}^{n \times d}$ is passed to DGGN, where $n = N \times K + T$ denotes the total number of instances in a meta-task. The final instance embeddings are generated after being processed by L layers. In the following, we describe the details of each stage in l-th DGGN layer.

Graph Construction. Given V^l, we first construct a k-NN graph $G^l = (V^l, E^l)$ to capture the relationships among all instances, where the edge set E^l is represented by the adjacency matrix $A^l \in \mathbb{R}^{n \times n}$. Following (Liu et al. 2019), we choose Gaussian similarity function with the learnable instance-wise scale parameter to construct A^l:

$$A_{ij}^l = \exp(-\frac{1}{2}\|\frac{\mathbf{v}_i^l}{\epsilon_i} - \frac{\mathbf{v}_j^l}{\epsilon_j}\|^2), \tag{1}$$

where $\epsilon_i = f_\epsilon(\mathbf{v}_i^l)$, and $f_\epsilon(\cdot)$ is a Multi-Layer Perceptron (MLP) to learn an unique ϵ_i for each $\mathbf{v}_i^l$. We only keep the k-max values in each row of A^l. In the following, we omit the superscript l for simplicity.

Local-wise Curvature-based Metric Modeling. We employ ORC (Jost and Liu 2014) to capture the local class-specific metrics around each edge. Formally, considering V and A, we calculate the ORC κ_{ij} for each edge $(\mathbf{v}_i, \mathbf{v}_j) \in E$. Denote the neighboring node set of $\mathbf{v}_i$ as $\mathcal{N}(\mathbf{v}_i) = \{\mathbf{v}_{i,1}, \mathbf{v}_{i,2}, \cdots, \mathbf{v}_{i,I}\}$ and $I = |\mathcal{N}(\mathbf{v}_i)|, J = |\mathcal{N}(\mathbf{v}_j)|$ are the numbers of nodes neighboring with $\mathbf{v}_i$ and $\mathbf{v}_j$ respectively. A probability distribution m_i over $\mathbf{v}_i$ is defined as:

$$m_i(\mathbf{v}_{i,\bar{i}}) = \frac{\exp(-\frac{1}{2}d_S(\mathbf{v}_i, \mathbf{v}_{i,\bar{i}}))}{\sum_{\mathbf{v}_{i,\bar{i}'} \in \mathcal{N}(\mathbf{v}_i)} \exp(-\frac{1}{2}d_S(\mathbf{v}_i, \mathbf{v}_{i,\bar{i}'}))}, \tag{2}$$

where $d_S(\mathbf{v}_i, \mathbf{v}_{i,\bar{i}})$ denotes the structural distance of $\mathbf{v}_i$ and $\mathbf{v}_{i,\bar{i}}$: $d_S(\mathbf{v}_i, \mathbf{v}_{i,\bar{i}}) = \| \mathbf{v}_i - \mathbf{v}_{i,\bar{i}} \|^2 + d_{set}(\mathbf{v}_{i,\bar{i}}, \mathcal{N}(\mathbf{v}_j))$, where $\mathbf{v}_{i,\bar{i}}$ is a neighbor of $\mathbf{v}_i$. Here $d_{set}(\mathbf{v}_{i,\bar{i}}, \mathcal{N}(\mathbf{v}_j))$ denotes the distance between $\mathbf{v}_{i,\bar{i}}$ and the neighboring node set of $\mathbf{v}_j$, which quantifies the node-to-set structure:

$$d_{set}(\mathbf{v}_{i,\bar{i}}, \mathcal{N}(\mathbf{v}_j)) = \frac{1}{J} \sum_{\mathbf{v}_{j'} \in \mathcal{N}(\mathbf{v}_j)} \| \mathbf{v}_{i,\bar{i}} - \mathbf{v}_{j'} \|^2 . \tag{3}$$

Then, for the edge $(\mathbf{v}_i, \mathbf{v}_j)$, we calculate the Wasserstein distance $W(m_i, m_j)$ by solving its corresponding linear programming problem, with $w_{i'j'}$ denoting the amount of mass

transported from $\mathbf{v}_{i,i'}$ to $\mathbf{v}_{j,j'}$, and $d_W(\mathbf{v}_{i,i'}, \mathbf{v}_{j,j'})$ denoting the cost per unit transported from $\mathbf{v}_{i,i'}$ to $\mathbf{v}_{j,j'}$. We define d_W as the cosine similarity between $\mathbf{v}_{i,i'}$ and $\mathbf{v}_{j,j'}$: $d_W(\mathbf{v}_{i,i'}, \mathbf{v}_{j,j'}) = 1 - ({\mathbf{v}_{i,i'}}^\top \mathbf{v}_{j,j'})/(\| \mathbf{v}_{i,i'} \| \| \mathbf{v}_{j,j'} \|)$. After the optimal transportation flow $\hat{W}$ is obtained, the Wasserstein distance is calculated as:

$$W(m_i, m_j) = \sum_{i'=1}^{I} \sum_{j'=1}^{J} (1 - d_W(\mathbf{v}_{i,i'}, \mathbf{v}_{j,j'}))\hat{w}_{i'j'}. \quad (4)$$

We use Sinkhorn distance (Cuturi 2013) to approximate $W(m_i, m_j)$ for computation efficiency. Then we obtain the ORC κ_{ij} for the edge $(\mathbf{v}_i, \mathbf{v}_j)$: $\kappa_{ij} = 1 - W(m_i, m_j)/d_W(\mathbf{v}_i, \mathbf{v}_j)$. Next, we utilize an MLP $f_c(\cdot)$ to transform κ_{ij} into a channel-wise reweighing vector $\boldsymbol{\tau}_{ij} \in \mathbb{R}^d$: $\boldsymbol{\tau}_{ij} = \sigma_{softmax}(f_c(\kappa_{ij}))$, where $\sigma_{softmax}(\cdot)$ is the channel-wise softmax activation function. Finally, a curvature-guided message passing is performed:

$$\mathbf{v}_{(\mathbf{c})_i} = \sigma_{relu}(\sum_{\mathbf{v}_j \in \mathcal{N}(\mathbf{v}_i)} diag(\boldsymbol{\tau}_{ij})\, W_C \mathbf{v}_j), \quad (5)$$

where $diag(\cdot)$ is the diagonalizing operation that transforms $\boldsymbol{\tau}_{ij}$ into the diagonal matrix $diag(\boldsymbol{\tau}_{ij})$, $\sigma_{relu}(\cdot)$ denotes the ReLU activation function, and the learnable parameter matrix $W_C \in \mathbb{R}^{d \times d}$ serves as the transformation function.

Global-wise Graph-based Connectivity Modeling. We incorporate the resistive embedding (Velingker et al. 2023) as the connectivity-based embeddings to discover meaningful structural roles and connectivity patterns in a global view of the meta-task. Formally, given V and A, we first calculate the Laplacian matrix L: $L = D - A$, where D is the degree matrix: $D_{ii} = \sum_{j=1}^{n} A_{ij}$. Then, we can express the effective resistance Ω between $\mathbf{v}_i$ and $\mathbf{v}_j$ as: $\Omega_{ij} = (\mathbf{1}_i - \mathbf{1}_j)^\top L^\dagger (\mathbf{1}_i - \mathbf{1}_j)$, where $\mathbf{1}_i$ is a n-dimensional vector specifying the indicator of $\mathbf{v}_i$, and $L^\dagger$ is the pseudoinverse of L. Afterwards, we construct an edge-node incidence matrix $B \in \mathbb{R}^{|E| \times n}$, whose entries are defined as follows: the i-th row of B denotes the i-th edge $(\mathbf{v}_{i'}, \mathbf{v}_{j'})$ of G, and $B_{ii'} = 1$, $B_{ij'} = -1$ for $i = 1, \ldots, |E|$, $i' = 1, \ldots, n$, $j' = 1, \ldots, n$. Besides, we define the diagonal matrix $C \in \mathbb{R}^{|E| \times |E|}$ with C_{ii} being the weight $A_{i'j'}$ of the i-th edge $(\mathbf{v}_{i'}, \mathbf{v}_{j'})$. Finally, we can compute the resistive embedding $\mathbf{r}_i \in \mathbb{R}^{|E|}$ of $\mathbf{v}_i$: $\mathbf{r}_i = C^{\frac{1}{2}} B L^\dagger \mathbf{1}_{\mathbf{v}_i}$. Here an important property (Baker and Faber 2006) is: $\| \mathbf{r}_i - \mathbf{r}_j \|^2 = \Omega_{ij}$. The connectivity-guided message passing is performed as:

$$\mathbf{v}_{(\mathbf{r})_i} = \sigma_{relu}(\sum_{\mathbf{v}_j \in \mathcal{N}(\mathbf{v}_i)} W_R[\mathbf{v}_j \parallel \mathbf{r}_j]), \quad (6)$$

where $W_R \in \mathbb{R}^{d \times (|E|+d)}$ is the learnable parameter matrix, and $[\cdot \parallel \cdot]$ denotes the concatenation operation.

Integration of Local and Global Pathways. We integrate the local and global pathways via a cascading and residual framework, as shown in Figure 2 (b). In the following, we re-added the superscript l for a clearer description. Formally, given V^l and A^l, we first calculate the local-wise embedding ${\mathbf{v}_{(\mathbf{c})}}_i^l$ using Equation (5). Then, the node embedding matrix $V_C^l = \left\{ {\mathbf{v}_{(\mathbf{c})}}_i^l \right\}_{i=1}^n$ and A^l are passed into the global-wise pathway to compute the global-wise embedding ${\mathbf{v}_{(\mathbf{r})}}_i^l$ using Equation (6). Finally, the local-wise embeddings are concatenated with the global-wise embeddings, and we use an MLP $f_{agg}(\cdot) : \mathbb{R}^{2 \times d} \rightarrow \mathbb{R}^d$ to aggregate the two groups of node embeddings. The node embeddings of the $(l+1)$-th layer are calculated as: $\mathbf{v}_i^{l+1} = f_{agg}([{\mathbf{v}_{(\mathbf{c})}}_i^l \parallel {\mathbf{v}_{(\mathbf{r})}}_i^l]) + \mathbf{v}_i^0$, where we find that the residual connections can improve gradient stability and mitigate over-smoothing. The node embedding matrix V^{l+1} is then passed to the graph construction stage in the next DGGN layer.

Loss Function

We train the whole DGGN framework in an end-to-end manner by optimizing the following loss function:

$$\mathcal{L} = \frac{1}{L} \sum_{l=1}^{L} \mathcal{L}_{cls}^l + \lambda_{orc} \mathcal{L}_{orc}^l + \lambda_{res} \mathcal{L}_{res}^l, \quad (7)$$

where λ_{orc}, λ_{res} are two hyper-parameters to balance the weights of $\mathcal{L}_{orc}$ and $\mathcal{L}_{res}$, respectively. We introduce $\mathcal{L}_{cls}^l$, $\mathcal{L}_{orc}^l$, and $\mathcal{L}_{res}^l$ in the following.

Classification Loss. After obtaining V^l, we compute the similarity matrix A^l using Equation (1), where each entry a_{ij}^l denotes the similarity score between $\mathbf{v}_i^l$ and $\mathbf{v}_j^l$. For each query instance node $\mathbf{v}_q^l$, its predicted class probability distribution is calculated as:

$$p(\hat{y}_q | \mathbf{v}_q^l) = \sigma_{softmax}(\sum_{s=1}^{NK} a_{qs}^l \cdot one\text{-}hot(y_s)), \quad (8)$$

where y_s is the ground-truth label of the support instance $\mathbf{x}_s$, and $one\text{-}hot(\cdot)$ is the one-hot indicator of the label. The classification loss of the l-th layer is defined as:

$$\mathcal{L}_{cls}^l = \frac{1}{T} \sum_{q=1}^{T} \mathcal{L}_{ce}(p(\hat{y}_q | \mathbf{v}_q^l), y_q), \quad (9)$$

where $\mathcal{L}_{ce}(\cdot, \cdot)$ denotes the cross-entropy loss.

Curvature-informed Regularization Loss. Intuitively, nodes connected by edges with positive curvatures should form tight embeddings, and vice versa. To utilize this principle for enhancing the local metric variations among the instances, we propose a new curvature-informed regularization loss, formulated as:

$$\mathcal{L}_{orc}^l = \frac{1}{|E^l|} \sum_{(\mathbf{v}_i^l, \mathbf{v}_j^l) \in E^l} \exp(\kappa_{ij}^l) \cdot \| {\mathbf{v}_{(\mathbf{c})}}_i^l - {\mathbf{v}_{(\mathbf{c})}}_j^l \|^2, \quad (10)$$

where κ_{ij}^l is the curvature of edge $(\mathbf{v}_i^l, \mathbf{v}_j^l)$.

Connectivity-informed Regularization Loss. To explicitly enforce tighter intra-class and looser inter-class connectivity over the entire meta-task, we propose a new connectivity-informed regularization loss, formulated as:

$$\mathcal{L}_{res}^l = \frac{1}{n} \sum_{\mathbf{v}_i^l, \mathbf{v}_j^l \in V^l} (\delta_{ij} \Omega_{ij}^l + (1 - \delta_{ij}) \max(0, 0.5 - \Omega_{ij}^l)), \quad (11)$$

where δ_{ij} is a categorical indicator that $\delta_{ij} = 1$ if $\mathbf{v}_i$ and $\mathbf{v}_j$ are in the same class otherwise $\delta_{ij} = 0$. Following (Deng et al. 2019), we set the embedding separation margin as 0.5.

	Method	Backbone	*Mini*ImageNet		*Tiered*ImageNet		CIFAR-FS	
			1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Curvature-based	Curvature(Gao et al. 2021)	ResNet-12	71.79±0.23	83.00±0.17	77.19±0.24	86.18±0.15	76.8 ±0.7	86.4 ±0.5
	OM(Qi et al. 2021)	ResNet-18	77.20±0.36	87.11±0.42	83.73±0.36	90.46±0.46	-	-
	APP2S(Ma et al. 2022)	ResNet-12	66.25±0.20	83.42±0.15	72.00±0.22	86.23±0.15	73.12±0.22	85.69±0.16
	HyperKT(Zhang et al. 2022)	ResNet-12	71.52±0.78	83.26±0.54	72.23±0.87	86.06±0.64	79.35±0.81	88.79±0.58
	noHub-S(Trosten et al. 2023)	ResNet-18	76.68±0.28	84.67±0.15	83.09±0.27	88.43±0.16	-	-
Graph-based	EPNet(Rodríguez et al. 2020)	ResNet-12	66.50±0.89	81.06±0.60	76.53±0.87	87.32±0.64	-	-
	DPGN(Yang et al. 2020)	ResNet-12	67.77±0.32	84.60±0.43	72.45±0.51	87.24±0.39	77.9 ±0.5	90.2 ±0.4
	ECKPN(Chen et al. 2021)	ResNet-12	70.48±0.38	85.42±0.46	73.59±0.45	88.13±0.28	79.2 ±0.4	91.0 ±0.5
	protoLP(Zhu et al. 2023)	ResNet-12	70.77±0.30	80.85±0.16	84.69±0.29	89.47±0.15	78.66±0.24	85.85±0.17
C+G	**DGGN (Ours)**	ResNet-12	75.27±0.20	86.28±0.15	84.77±0.23	89.80±0.17	82.29±0.23	92.27±0.15
	DGGN (Ours)	ResNet-18	**78.08±0.21**	**87.87±0.17**	**85.52±0.23**	**90.81±0.17**	**83.97±0.26**	**92.64±0.17**
Semantic	SemFew(Zhang et al. 2024a)	Swin-T	78.94±0.66	86.49±0.50	82.37±0.77	89.89±0.52	84.34±0.67	89.11±0.54
	ECER-FSL(Liu et al. 2025)	Visformer-T	81.14±0.15	-	81.81±0.51	-	86.01±0.35	-
	SemDGGN (Ours)	Swin-T	**82.16±0.19**	**89.85±0.17**	**87.36±0.21**	**91.65±0.15**	**86.71±0.24**	**94.17±0.15**

Table 1: Few-shot *5-way* classification accuracy and 95% confidence interval on *Mini*ImageNet, *Tiered*ImageNet, and CIFAR-FS. "**C+G**" denotes the incorporation of graph and curvature. The best results are shown in **bold**.

Method	Backbone	1-Shot	5-Shot
DPGN(Yang et al. 2020)	ResNet-12	75.71±0.47	91.48±0.33
EPNet(Rodríguez et al. 2020)	ResNet-12	82.85±0.81	91.32±0.41
ECKPN(Chen et al. 2021)	ResNet-12	77.43±0.54	92.21±0.41
APP2S(Ma et al. 2022)	ResNet-18	77.64±0.19	90.43±0.18
protoLP(Zhu et al. 2023)	ResNet-12	90.13±0.20	92.85±0.11
noHub-S(Trosten et al. 2023)	ResNet-18	85.81±0.24	90.52±0.12
DGGN (Ours)	ResNet-12	90.71±0.24	93.16±0.13
DGGN (Ours)	ResNet-18	**91.54±0.25**	**93.82±0.16**

Table 2: Few-shot *5-way* classification accuracy on CUB-200-2011 dataset. The best results are shown in **bold**.

Method	ChestX	ISIC	EuroSAT	CropDisease
StyleAdv(Fu et al. 2023)	22.64	33.96	70.94	74.13
LDP-net(Zhou et al. 2023a)	23.01	33.97	65.11	69.64
Dara(Zhao et al. 2023)	22.92	36.42	67.42	80.74
Exploit(Zhou et al. 2024)	22.48	34.28	69.56	84.01
SVasP(Li et al. 2025)	23.23	37.63	72.30	75.87
ECER-FSL(Liu et al. 2025)	25.12	40.13	74.13	82.13
DGGN (Ours)	23.81	38.70	72.25	81.82
SemDGGN (Ours)	**26.08**	**41.32**	**74.75**	**84.22**

Table 3: Few-shot *5-way 1-shot* classification accuracy on BSCD-FSL benchmark. The best results are shown in **bold**.

Experiments

Experimental Details

Datasets. We evaluate DGGN on two primary FSL tasks: standard FSL and cross-domain FSL. For standard FSL evaluation, we use four benchmark datasets: ***Mini*ImageNet** (Vinyals et al. 2016), ***Tiered*ImageNet** (Ren et al. 2018), **CUB-200-2011** (Wah et al. 2011), and **CIFAR-FS** (Bertinetto et al. 2019). For cross-domain FSL evaluation, we adopt the BSCD-FSL benchmark (Guo et al. 2020) including four datasets: **ChestX** (Wang et al. 2017), **ISIC** (Tschandl et al. 2018), **EuroSAT** (Helber et al. 2019), and **CropDisease** (Mohanty et al. 2016).

Network Architectures. We adopt three commonly used backbones in FSL: **ResNet-12**, **ResNet-18** (He et al. 2016), and **Swin-T** (Liu et al. 2021). The architectures of ResNet-12 and ResNet-18 are the same as (Zhu et al. 2023), and the architecture of Swin-T is the same as (Zhang et al. 2024a). To follow up recent trends of utilizing fine-grained semantic information from ChatGPT (Zhang et al. 2024a; Liu et al. 2025), we design a variant model **SemDGGN**, where we generate descriptive text for each class with GPT-3.5-Turbo and use ViT-B/16 CLIP (Radford et al. 2021) to produce the semantic embeddings, following (Zhang et al. 2024a). We first concatenate each support instance's visual embedding from Swin-T and its corresponding semantic embedding from ViT-B/16 CLIP, then we transform this concatenated embedding into the same dimension of the query nodes via a fully-connected layer to produce the initial node embedding matrix passed to the DGGN module.

Training Strategy. We first pre-train the backbone network on the training classes. For ResNet-12 and ResNet-18, we adopt the same pre-training strategy as (Zhu et al. 2023). For Swin-T and ViT-B/16 CLIP, the pre-training strategy is

Setup	*Tiered*ImageNet Averaged similarity score at l-th layer				
	0	1	2	3	4
GG	1.00±0.00	0.98±0.01	0.97±0.02	0.97±0.02	0.96±0.02
LG	1.00±0.00	0.81±0.09	0.68±0.12	0.54±0.17	0.52±0.18
LI	1.00±0.00	0.73±0.10	0.56±0.12	0.45±0.15	0.47±0.15
GI	1.00±0.00	0.92±0.07	0.85±0.07	0.87±0.05	0.86±0.06
Setup	**CUB-200-2011** Averaged similarity score at l-th layer				
	0	1	2	3	4
LL	1.00±0.00	0.98±0.01	0.98±0.01	0.97±0.02	0.97±0.02
LG	1.00±0.00	0.79±0.10	0.65±0.13	0.50±0.20	0.51±0.21
LI	1.00±0.00	0.95±0.03	0.93±0.05	0.90±0.04	0.90±0.06
GI	1.00±0.00	0.76±0.09	0.64±0.12	0.52±0.17	0.49±0.18
Setup	***Tiered*ImageNet→CUB-200-2011** Averaged similarity score at l-th layer				
	0	1	2	3	4
LL	1.00±0.00	0.98±0.02	0.97±0.03	0.96±0.02	0.96±0.03
LG	1.00±0.00	0.85±0.08	0.70±0.10	0.55±0.19	0.56±0.22
LI	1.00±0.00	0.90±0.06	0.85±0.09	0.82±0.12	0.83±0.13
GI	1.00±0.00	0.83±0.10	0.69±0.14	0.58±0.17	0.57±0.20

Table 4: Averaged similarity scores and 95% confidence interval between the k-NN ($k = 20$) graphs at each layer using a pre-trained ResNet-12 backbone.

Local	*Global*	**CUB-200-2011** 1-shot	**CUB-200-2011** 5-shot	***Tiered*ImageNet** 1-shot	***Tiered*ImageNet** 5-shot
		83.09±0.22	87.95±0.12	76.25±0.22	83.04±0.16
✓		87.92±0.23	91.21±0.13	78.12±0.24	84.16±0.15
	✓	85.11±0.22	89.29±0.13	80.37±0.22	86.12±0.15
✓	✓	**90.71±0.24**	**93.16±0.13**	**84.77±0.23**	**89.80±0.17**

Table 5: Few-shot *5-way* classification accuracy and 95% confidence interval with different combinations on local and global pathways.

the same as (Zhang et al. 2024a). During meta-training, the model is meta-trained for 10,000 iterations, with each iteration containing 8 randomly sampled meta-tasks. The Adam optimizer is used in all experiments with the learning rate of 0.001, and the backbone is frozen to prevent over-fitting.

Evaluation Protocol. We evaluate the performance under standard 5-way 1-shot and 5-way 5-shot settings. Following (Zhu et al. 2023), we randomly sample 10,000 meta-tasks from the testing set and report the average classification accuracy (%) as well as the 95% confidence interval.

Hyper-parameter Settings. We set the hyper-parameter k, L, λ_{orc}, and λ_{res} as 20, 4, 0.001, and 0.001, respectively.

Main Results

Standard FSL. We compare our approach with recent state-of-the-art graph-based methods (Rodríguez et al. 2020; Yang et al. 2020; Chen et al. 2021; Zhu et al. 2023), curvature-based methods (Gao et al. 2021; Ma et al. 2022; Zhang et al. 2022; Trosten et al. 2023; Qi et al. 2021), and semantics-based methods (Zhang et al. 2024a; Liu et al. 2025). The results are listed in Table 1, 2. It can be seen that DGGN outperforms most methods, and SemDGGN achieves the best performance, demonstrating the effectiveness and superiority of our integration with local and global geometric priors.

Cross-domain FSL. In BSCD-FSL, our model is meta-trained on *Mini*ImageNet dataset, then meta-tested on the four unrelated datasets without additional fine-tuning. The results are listed in Table 3. It can be seen that DGGN outperforms most methods, and SemDGGN achieves the highest performance, proving its effectiveness of capturing transferable knowledge across different domains.

L_{orc}	L_{res}	***Mini*ImageNet** 1-shot	***Mini*ImageNet** 5-shot	***Tiered*ImageNet** 1-shot	***Tiered*ImageNet** 5-shot
		73.39±0.22	84.30±0.15	82.13±0.24	87.34±0.15
✓		74.08±0.22	84.92±0.15	82.66±0.22	87.98±0.16
	✓	74.78±0.23	85.68±0.16	83.72±0.22	88.70±0.17
✓	✓	**75.27±0.20**	**86.28±0.15**	**84.77±0.23**	**89.80±0.17**

Table 6: Few-shot *5-way* classification accuracy and 95% confidence interval with the regularization losses.

Setups	***Mini*ImageNet** 1-shot	***Mini*ImageNet** 5-shot	***Tiered*ImageNet** 1-shot	***Tiered*ImageNet** 5-shot
$Parallel$	71.90±0.22	83.03±0.16	80.28±0.24	86.02±0.16
$Fixed\ Edges$	73.91±0.20	84.21±0.15	82.69±0.25	87.37±0.18
$Global \rightarrow Local$	73.42±0.21	84.73±0.16	81.53±0.24	87.04±0.17
$No\ residuals$	74.78±0.23	85.71±0.17	83.32±0.22	88.91±0.17
$Local \rightarrow Global$	**75.27±0.20**	**86.28±0.15**	**84.77±0.23**	**89.80±0.17**

Table 7: Few-shot *5-way* classification accuracy and 95% confidence interval with various integration architectures.

Ablation Study

Time Complexity. The time complexity of graph construction, ORC calculation, and resistive embedding derivation is $\mathcal{O}(n^2)$, $\mathcal{O}(nk^3)$, and $\mathcal{O}(n^3)$, respectively. The overall time complexity of DGGN is $\mathcal{O}(L\max(n^3, nk^3)) \approx \mathcal{O}(Ln^3)(n \gg k)$. It's worth to note that in our setting, n is very small (80 for 1-shot and 100 for 5-shot).

Graph Topological Similarity. We empirically show the complementarity between the learned embeddings from local and global pathways, and the adaption ability of DGGN on different types of data. We first meta-train a local-wise GNN, a global-wise GNN, and DGGN separately under 5-way 5-shot setting with a fixed random seed. To control randomness, we additionally retrain the local-wise and global-wise GNNs from scratch with a different random seed. For each meta-task, we construct a k-NN graph using all the embeddings from each GNN layer. We use the normalized graph edit distance (nGED) (Fisman et al.

Method	Label Ratio (%)				
	20	40	60	80	100
TPN(Liu et al. 2019)	63.4	68.5	70.7	72.1	72.8
EGNN(Kim et al. 2019)	67.0	70.0	71.9	74.4	80.4
DPGN(Yang et al. 2020)	79.4	81.0	81.8	84.6	87.6
DGGN (Ours)	**83.0**	**83.8**	**85.7**	**87.6**	**91.5**

Table 8: Semi-supervised FSL classification accuracies.

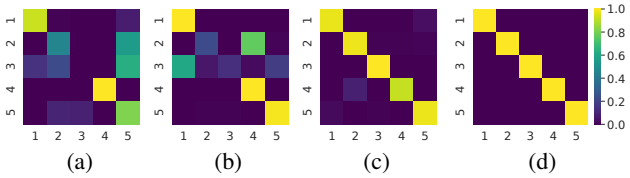


Figure 3: Visualization of the support-query similarities at each layer under *5-way 1-shot 1-query* setting.

2022) to measure the layer-wise topological similarity between the three groups of graphs, where a similarity score (defined as 1-nGED) close to 1 shows a strong structural similarity and vice versa. The similarity scores are averaged across 10,000 meta-tasks for statistical robustness. We choose *Tiered*ImageNet and CUB-200-2011, two datasets representing coarse-grained and fine-grained tasks respectively. The results are listed in Table 4. The row "*Tiered*ImageNet→CUB-200-2011" denotes that all models are trained on *Tiered*ImageNet and tested on CUB-200-2011. LL is the comparison between the two local-wise GNNs. GG is the comparison between the two global-wise GNNs. LG is the comparison between the local-wise and global-wise GNNs. LI is the comparison between the local-wise GNN and DGGN. GI is the comparison between the global-wise GNN and DGGN. It can be seen that for all LL and GG comparisons, the scores show that the graphs remain a high structural similarity, excluding the impacts of random errors. For all LG comparisons, the scores decrease to around 0.5 gradually, indicating a strong structural dissimilarity between the local and global pathways. On *Tiered*ImageNet, GI is close to GG and $LI \approx LG \ll GI$, indicating DGGN adapts to the coarse data structures. On CUB-200-2011, LI is close to LL and $LI \gg LG \approx GI$, indicating DGGN adapts to the nuanced data structures. On the cross-domain setting, LI is also close to LL and $LI \gg LG \approx GI$, indicating that DGGN's learned geometric priors are intrinsic and transferable, which supports its superior performance on BSCD-FSL benchmark.

Impacts of Local and Global Pathways. We conduct experiments with ResNet-12, comparing four models: a baseline GNN without incorporating the local and global pathways; a local-wise GNN denoted as "$Local$"; a global-wise GNN denoted as "$Global$"; the DGGN. The results are listed in Table 5. It can be seen that local-wise GNN or global-wise GNN yields higher improvements on CUB or *Tiered*ImageNet respectively, indicating that these two GNNs learned different geometric biases that suit different data granularity. Furthermore, our framework design achieves the highest performance, showing that the local and global geometric priors can facilitate between each other.

Impacts of the Regularization Loss. Table 6 lists the results with different combinations of the loss functions $\mathcal{L}_{orc}$ and $\mathcal{L}_{res}$. It can be seen that both the two regularization loss functions have a positive improvement on the classification accuracy, indicating their effectiveness.

Integration Architectures of Local and Global Pathways. To explore the integration architectures for local and global pathways, we compared five alternative integration architectures: "$Parallel$" denotes that the input node embeddings are fed separately into local and global pathways and the outputs are concatenated to obtain unified node embeddings; "$Fixed\ edges$" denotes that we only construct a k-NN graph at the first layer, then the weighted edges are fixed in the following layers; "$Global \rightarrow Local$" denotes that the node embeddings are first enhanced by the global-wise pathway, then passed through the local-wise pathway; "$No\ residuals$" denotes that all residual connections are removed; "$Local \rightarrow Global$" is the DGGN framework with all components included. The results are listed in Table 7. It can be seen that our DGGN design achieves the highest performance compared to other integration architectures.

Semi-supervised FSL Evaluation. Following the protocol of (Yang et al. 2020), we partition each meta-task's support set into labeled and unlabeled subsets, varying the ratio of labeled support instances in each class by 20%, 40%, 60%, 80%, and 100%. The experiments are conducted on *Mini*ImageNet under 5-way 10-shot with ResNet-12. The results are demonstrated in Table 8. It can be seen that DGGN surpasses all other approaches.

Visualization. We visualize the support-query similarities in each DGGN layer under 5-way 1-shot 1-query setting. The results are demonstrated as heatmaps in Figure 3. It can be seen that DGGN gradually refines the instance embeddings in each layer and make the correct classifications for the five query instances in the final layer.

Conclusion

In this work, we targeted at unifying the pros of curvature-based and graph-based FSL models in modeling geometric bias. To formalize this, we devised two different pathways, one modeled by ORC for local fine-grained metrics, and another by resistive embedding for global coarse-grained connectivity. Our experiments showed that the learned embeddings from the two pathways demonstrate complementarity in graph topological structures. Based on this key observation, we designed DGGN, a novel framework that adaptively integrates both local and global geometric priors. We confirmed our hypothesis via measuring the graph topological similarity, which demonstrates that DGGN's learned geometry well aligns with the underlying distribution from different types of data. As a result of our principled design, DGGN establishes a new state-of-the-art on standard FSL, challenging BSCD-FSL, and semi-supervised FSL benchmarks, proving its robust and discriminative power on handling limited data and domain shift.

Acknowledgments

This research is supported by National Science and Technology Major Project (2022ZD0119202), National Science Fund for Distinguished Young Scholars (62125601), National Natural Science Foundation of China (62576031), and State Key Laboratory of Multimedia Information Processing Open Fund (SKLMIP-KF-2025-03).

References

Baker, M.; and Faber, X. 2006. Metrized graphs, Laplacian operators, and electrical networks. *Contemporary Mathematics*, 415(15-34): 2.

Bertinetto, L.; Henriques, J. F.; Torr, P.; and Vedaldi, A. 2019. Meta-learning with differentiable closed-form solvers. In *ICLR*.

Chen, C.; Yang, X.; Xu, C.; Huang, X.; and Ma, Z. 2021. ECKPN: Explicit class knowledge propagation network for transductive few-shot learning. In *CVPR*, 6596–6605.

Cuturi, M. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*.

Deng, J.; Guo, J.; Xue, N.; and Zafeiriou, S. 2019. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 4690–4699.

Fang, Z.; Zhu, X.; Yang, C.; Zhou, H.; Qin, J.; and Yin, X.-C. 2025. Aligning enhanced feature representation for generalized zero-shot learning. *Science China Information Sciences*, 68(2): 122102.

Fisman, D.; Grogin, J.; Margalit, O.; and Weiss, G. 2022. The normalized edit distance with uniform operation costs is a metric. In *CPM*.

Fu, Y.; Xie, Y.; Fu, Y.; and Jiang, Y.-G. 2023. Styleadv: Meta style adversarial training for cross-domain few-shot learning. In *CVPR*, 24575–24584.

Gao, Z.; Wu, Y.; Jia, Y.; and Harandi, M. 2021. Curvature generation in curved spaces for few-shot learning. In *ICCV*, 8691–8700.

Guo, Y.; Codella, N. C.; Karlinsky, L.; Codella, J. V.; Smith, J. R.; Saenko, K.; Rosing, T.; and Feris, R. 2020. A broader study of cross-domain few-shot learning. In *ECCV*, 124–141.

Han, Z.; Zhu, X.; Yang, C.; Zhou, H.; Qin, J.; and Yin, X.-C. 2024. Exploring Stable Meta-Optimization Patterns via Differentiable Reinforcement Learning for Few-Shot Classification. In *ACM MM*, 1701–1710.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7): 2217–2226.

Jost, J.; and Liu, S. 2014. Ollivier's Ricci curvature, local clustering and curvature-dimension inequalities on graphs. *Discrete & Computational Geometry*, 51(2): 300–322.

Kim, J.; Kim, T.; Kim, S.; and Yoo, C. D. 2019. Edge-labeling graph neural network for few-shot learning. In *CVPR*, 11–20.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

Li, W.; et al. 2025. Svasp: Self-versatility adversarial style perturbation for cross-domain few-shot learning. In *AAAI*, volume 39, 15275–15283.

Liu, M.; Wu, F.; Li, B.; Lu, Z.; Yu, Y.; and Li, X. 2025. Envisioning class entity reasoning by large language models for few-shot learning. In *AAAI*, volume 39, 18906–18914.

Liu, Y.; Lee, J.; Park, M.; Kim, S.; Yang, E.; Hwang, S. J.; and Yang, Y. 2019. Learning to propagate labels: Transductive propagation network for few-shot learning. In *ICLR*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 10012–10022.

Ma, R.; Fang, P.; Drummond, T.; and Harandi, M. 2022. Adaptive poincaré point to set distance for few-shot classification. In *AAAI*, volume 36, 1926–1934.

Mohanty, S. P.; et al. 2016. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7: 215232.

Qi, G.; Yu, H.; Lu, Z.; and Li, S. 2021. Transductive few-shot classification on the oblique manifold. In *ICCV*, 8412–8422.

Radford, A.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*, 8748–8763.

Ren, M.; Triantafillou, E.; Ravi, S.; Snell, J.; Swersky, K.; Tenenbaum, J. B.; Larochelle, H.; and Zemel, R. S. 2018. Meta-Learning for Semi-Supervised Few-Shot Classification. In *ICLR*.

Rodríguez, P.; Laradji, I.; Drouin, A.; and Lacoste, A. 2020. Embedding propagation: Smoother manifold for few-shot classification. In *ECCV*, 121–138.

Satorras, V. G.; and Estrach, J. B. 2018. Few-shot learning with graph neural networks. In *ICLR*.

Trosten, D. J.; Chakraborty, R.; Løkse, S.; Wickstrøm, K. K.; Jenssen, R.; and Kampffmeyer, M. C. 2023. Hubs and hyperspheres: Reducing hubness and improving transductive few-shot learning with hyperspherical embeddings. In *CVPR*, 7527–7536.

Tschandl, P.; et al. 2018. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5(1): 1–9.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y.; et al. 2018. Graph attention networks. In *ICLR*.

Velingker, A.; Sinop, A.; Ktena, I.; Veličković, P.; and Gollapudi, S. 2023. Affinity-aware graph networks. In *NeurIPS*.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. In *NeurIPS*, volume 29.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology.

Wang, X.; Peng, Y.; Lu, L.; Lu, Z.; Bagheri, M.; and Summers, R. M. 2017. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *CVPR*, 2097–2106.

Yang, L.; Li, L.; Zhang, Z.; Zhou, X.; Zhou, E.; and Liu, Y. 2020. Dpgn: Distribution propagation graph network for few-shot learning. In *CVPR*, 13390–13399.

Zhang, B.; Jiang, H.; Feng, S.; Li, X.; Ye, Y.; and Ye, R. 2022. Hyperbolic knowledge transfer with class hierarchy for few-shot learning. In *IJCAI*, volume 7, 3723–3729.

Zhang, H.; Xu, J.; Jiang, S.; and He, Z. 2024a. Simple semantic-aided few-shot learning. In *CVPR*, 28588–28597.

Zhang, S.-X.; Yang, C.; Zhu, X.; Zhou, H.; Wang, H.; and Yin, X.-C. 2024b. Inverse-like antagonistic scene text spotting via reading-order estimation and dynamic sampling. *IEEE Transactions on Image Processing*, 33: 825–839.

Zhao, Y.; Zhang, T.; Li, J.; and Tian, Y. 2023. Dual adaptive representation alignment for cross-domain few-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(10): 11720–11732.

Zhou, F.; Wang, P.; Zhang, L.; Chen, Z.; Wei, W.; Ding, C.; Lin, G.; and Zhang, Y. 2024. Meta-exploiting frequency prior for cross-domain few-shot learning. In *NeurIPS*, volume 37, 116783–116814.

Zhou, F.; Wang, P.; Zhang, L.; Wei, W.; and Zhang, Y. 2023a. Revisiting prototypical network for cross domain few-shot learning. In *CVPR*, 20061–20070.

Zhou, H.; Zhu, X.; Qin, J.; Xu, Y.; Cesar-Jr, R. M.; and Yin, X.-C. 2025. Multi-Scale Texture Fusion for Reference-Based Image Super-Resolution: New Dataset and Solution. *International Journal of Computer Vision*, 133(10): 6971–6992.

Zhou, H.; Zhu, X.; Zhu, J.; Han, Z.; Zhang, S.-X.; Qin, J.; and Yin, X.-C. 2023b. Learning correction filter via degradation-adaptive regression for blind single image super-resolution. In *ICCV*, 12365–12375.

Zhu, H.; et al. 2023. Transductive Few-shot Learning with Prototype-based Label Propagation by Iterative Graph Refinement. In *CVPR*, 23996–24006.