

Article

Hypergraph-Augmented Reconstruction and Adaptive Metric Aggregation Network for Few-Shot Fine-Grained Image Classification

Zhen Zheng, Zhen Wang , Xinzhong Wang, Yang Liu, Mingzhe Yang and Yichen Yu

School of Computer Science and Technology, Shandong University of Technology, 266 Xincun West Road, Zhangdian District, Zibo 255000, China; 24505040721@stumail.sdut.edu.cn (Z.Z.); 24505020694@stumail.sdut.edu.cn (X.W.); 25505020700@stumail.sdut.edu.cn (Y.L.); 25505040751@stumail.sdut.edu.cn (M.Y.); 24110507021@stumail.sdut.edu.cn (Y.Y.)

* Correspondence: wzh@sdut.edu.cn

Abstract

Few-shot fine-grained image classification aims to recognize unseen fine-grained categories from very few labeled examples. The task is difficult because visually similar classes may differ only in subtle details, while images from the same class can vary considerably. Useful evidence is therefore often localized. Reconstruction-based metric methods reduce local misalignment by reconstructing query features from class-wise support features, but a small support set may still provide an incomplete reconstruction basis. In addition, spatial reconstruction errors are usually combined with fixed or nearly uniform weights, and intermediate features receive no direct reconstruction-oriented supervision. We develop HARA, a Hypergraph-Augmented Reconstruction and Adaptive Metric Aggregation Network, to address these limitations. HARA forms learnable soft hyperedges over local support features and uses the resulting aggregated features to enrich the class-wise reconstruction basis. It then combines class-conditional reconstruction with query self-reconstruction to obtain position-wise joint errors, which are aggregated using query-dependent weights. A training-only auxiliary branch applies reconstruction-based supervision to intermediate backbone features. For 5-way evaluation, HARA obtains 84.09%/93.48% on CUB-200-2011, 76.91%/88.69% on Stanford Dogs and 88.53%/95.87% on Stanford Cars under the 1-shot/5-shot settings, respectively. Ablation and diagnostic analyses further characterize the contributions and behavior of the proposed framework. Overall, HARA provides an effective extension to reconstruction-based few-shot fine-grained classification.

Keywords: few-shot learning; fine-grained image classification; reconstruction-based metric learning; hypergraph feature augmentation; adaptive metric aggregation; computer vision



Academic Editor: Ferdinando Di Martino

Received: 6 September 2026

Revised: 5 October 2026

Accepted: 7 October 2026

Published: 10 October 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Fine-grained image classification distinguishes visually similar subcategories within the same broad category. It has important applications in bird species recognition, dog breed recognition, vehicle model recognition, and species monitoring. Unlike coarse-grained recognition, fine-grained classification often depends on small visual differences in texture, color, shape, and characteristic object parts. Accurate annotation of such categories therefore requires considerable manual effort and domain expertise [1]. In practice, many fine-grained categories contain only a limited number of labeled images, which can weaken

the transfer of models trained with large-scale supervision to novel categories. Few-shot fine-grained image classification addresses this setting by recognizing unseen categories using only a few labeled support samples, providing an important approach to fine-grained recognition when annotations are scarce [2,3].

Fine-grained few-shot recognition places greater demands on local discrimination than general few-shot classification. Different categories may have very similar global appearances, with useful differences restricted to a few local patterns, such as feather texture or head morphology in birds, facial structure in dogs, and local appearance details in vehicles. Meanwhile, samples from the same category may vary substantially in pose, viewpoint, scale, occlusion, and background. A few support samples are therefore unlikely to cover these intra-class variations adequately. Under such conditions, a model must not only extract representative local features from limited support images but also distinguish class-relevant discriminative regions from background and other non-discriminative regions. Otherwise, irrelevant information may interfere with class matching. Making reliable use of these local cues from only a few labeled samples therefore remains a central challenge [1,2].

Metric learning is widely used for few-shot recognition because it provides transferable representations and similarity functions for novel categories. These methods classify a query sample using class prototypes or support–query similarities. They generally do not require complex parameter updates for novel categories during testing [3,4]. However, early metric-based methods mainly rely on image-level global representations or overall similarity measures, which may mix subtle local differences with background, pose, viewpoint, and other non-discriminative factors. Subsequent studies have therefore introduced local feature matching, region alignment, and local relation modeling to establish finer correspondences between support and query images [5,6]. These approaches improve local correspondence. However, features extracted from only a few support images may still provide insufficient coverage of intra-class variations, especially under large changes in pose, viewpoint, or local appearance. Therefore, it remains important to explore intra-class relations among local support features and to construct a richer class-wise support representation.

Reconstruction offers another way to perform local class comparison. For each candidate class, its support features form the reconstruction basis, and class compatibility is assessed from the error obtained when reconstructing the query features. Multiple support features can jointly represent the same query region, reducing the need for exact spatial correspondence between support and query images [7,8]. Nevertheless, reconstruction-based metrics still face three limitations in few-shot fine-grained classification. First, a reconstruction basis formed from only a few support images covers only a limited range of intra-class local variations. This limitation may restrict class-conditional reconstruction when a query presents unseen poses, viewpoints, or local appearances. Second, reconstruction errors at different spatial positions are commonly aggregated using fixed or nearly uniform weights, although discriminative object parts, background regions, and occluded regions contribute differently to fine-grained classification. Third, intermediate backbone features retain useful local structural information. However, they are usually optimized only indirectly through the final classification objective and therefore lack direct supervision aligned with reconstruction-based classification. These limitations motivate improvements in class-wise support representation, adaptive aggregation of local reconstruction errors, and reconstruction-oriented supervision of intermediate features.

Based on these observations, we develop HARA (Hypergraph-Augmented Reconstruction and Adaptive Metric Aggregation Network) for few-shot fine-grained image classification. HARA consists of a main reconstruction-based classification branch and a

training-only auxiliary supervision branch. In the main branch, the Hypergraph-Based Feature Augmentation (HFA) treats local support features from the same class as hypergraph nodes. Learnable soft hyperedges then aggregate intra-class relations across support images and spatial positions. The resulting aggregated features supplement the original support features to construct an augmented class-wise reconstruction basis. HARA then combines class-conditional reconstruction with query self-reconstruction to obtain position-wise joint reconstruction errors. Instead of uniformly aggregating these errors, the Adaptive Metric Aggregation (AMA) module generates query-dependent spatial weights from local query features. For a given query, the same spatial weights are shared across all candidate classes. These weights are used to aggregate the position-wise errors into class scores. During training, the Reconstruction-Based Auxiliary Supervision Branch (RASB) provides an additional reconstruction-oriented constraint for intermediate backbone features. RASB is removed during testing, and final predictions are produced only by the main classification branch.

The main contributions are as follows:

- (1) We introduce the Hypergraph-Based Feature Augmentation module to enrich the class-wise reconstruction basis when support data are scarce. Within each class, local support features are treated as hypergraph nodes, and learnable soft hyperedges aggregate relations that span support images and spatial positions. The aggregated features are appended to the original support features and then used for class-conditional reconstruction.
- (2) We propose an Adaptive Metric Aggregation module that derives query-dependent spatial weights from local query features and uses them to combine position-wise joint reconstruction errors. For each query, the same weights are shared across candidate classes, allowing AMA to adjust the contribution of different query positions without introducing class-specific attention.
- (3) We design a Reconstruction-Based Auxiliary Supervision Branch for intermediate backbone features. During training, RASB introduces an additional reconstruction-oriented objective that encourages intermediate local features to better support the final reconstruction-based classification task. The branch is removed during testing and therefore does not introduce an additional inference branch.
- (4) We conduct systematic experiments on CUB-200-2011, Stanford Dogs, and Stanford Cars under 5-way 1-shot and 5-way 5-shot settings. HARA achieves competitive performance across the three fine-grained benchmarks. We further report complete factorial ablations on all three datasets, a soft-hyperedge sensitivity study on CUB-200-2011 and Stanford Dogs, three independent training runs on CUB-200-2011, and an additional CUB evaluation without bounding-box cropping. Varying-shot, hard-episode, reconstruction-mechanism, support-coverage, query-position, visualization, and complexity analyses provide further evidence for the behavior of the proposed framework.

2. Related Work

We organize the related literature around four topics that are directly relevant to HARA: few-shot fine-grained classification, local feature matching and calibration, reconstruction-based metrics, and graph/hypergraph relation modeling. This organization places HARA in the context of both reconstruction-based classifiers and structural relation models. Broader surveys of few-shot learning and few-shot fine-grained recognition are available in [2,3].

2.1. Few-Shot Fine-Grained Image Classification

Few-shot classifiers are typically learned through episodes that mimic the small classification tasks encountered at test time. Each training episode samples an N-way K-shot problem, allowing the model to repeatedly learn from changing class subsets rather than

from a single fixed classification task [3,9]. Two widely used families of methods are optimization-based and metric-based approaches. MAML [10], for example, learns an initialization from which the model can adapt to a new task with only a few update steps. Metric-based approaches instead compare support and query representations in a learned embedding space. Matching Networks [11], Prototypical Networks [12], and Relation Networks [13] implement this idea through sample-level matching, class prototypes, and learned relation scores, respectively.

When few-shot learning is applied to fine-grained recognition, the model must address both limited labeled data and the difficulty of distinguishing visually similar subcategories. Since class differences are often concentrated in local textures, shape characteristics, or key object parts, image-level global representations may overlook important discriminative regions. Recent studies have therefore shifted from global feature metrics toward local feature modeling, support–query relation modeling, and task-related feature selection. For example, DN4 [5] represents images with local descriptors and predicts a query through image-to-class nearest-neighbor matching. CDN4 [14] further improves the modeling of fine-grained discriminative regions through cross-view local metrics. These studies reflect a general shift in few-shot fine-grained classification from image-level global metrics to local feature modeling and task-related metric learning.

2.2. Local Feature Matching, Alignment, and Feature Calibration

Fine-grained category differences are often concentrated in local regions. Therefore, many few-shot fine-grained classification methods improve discriminative ability through local feature matching, region alignment, and feature calibration. Local matching and alignment methods explicitly establish correspondences between local regions in support and query images. This helps address the limited ability of image-level global representations to capture subtle category differences. DeepEMD [6], for example, formulates local feature matching as an optimal transport problem and establishes correspondences between local regions in support and query images. OLSA [15] learns spatial offsets for local patterns so that support and query regions can be aligned more precisely. TOAN [16] derives cross-correlation attention from support and query features to encode their spatial correspondence, followed by local compositional matching. HelixFormer [17] uses a Transformer to exchange semantic information across images and strengthen interactions between local support and query representations.

Background suppression and feature calibration provide another way to improve local discrimination. These methods generally generate spatial or channel-wise weights from foreground regions, task information, or support–query relations. The resulting weights emphasize regions that are informative for the current task. BSFA [18] suppresses irrelevant background content while aligning foreground features. MFSI-Net [19] identifies discriminative regions through multi-scale feature selection and interaction. TDM [20] generates task-related channel weights by maximizing task discrepancy, which encourages the model to focus on discriminative details in the current task. C2-Net [21] refines support and query representations by combining cross-layer enhancement with cross-sample adjustment. ATR-Net [22] learns representations conditioned on the current task while adaptively attenuating background noise.

Related vision tasks have also explored robustness and foreground guidance under complex scenes. Liu et al. [23] combine high-resolution representation, auxiliary guidance, and a domain transformation strategy to improve human parsing under complex and corrupted scenes. Wang et al. [24] use mask-guided constraints to emphasize vehicle-related regions during visible–infrared feature fusion for aerial detection. Although these studies address different tasks from few-shot fine-grained classification, their use of auxiliary guid-

ance and foreground-aware feature enhancement provides relevant insights for reducing background and occlusion interference in complex visual scenes.

Together, these methods show the value of local correspondence, background suppression, and task-conditioned feature selection when labeled examples are limited. However, they mainly focus on local matching between support and query samples or on the calibration of existing feature representations. For reconstruction-based metric methods, the construction of class-wise support representations and the aggregation of reconstruction errors across spatial positions are also important factors in the final classification decision.

2.3. Reconstruction-Based Few-Shot Classification

Pose, viewpoint, and spatial layout can differ substantially between support and query images, making direct global similarity less reliable for fine-grained matching. Reconstruction-based methods instead evaluate each candidate class according to how well its support representation can reconstruct the query features. Unlike prototype or nearest-neighbor metrics, this formulation allows several local support features to jointly represent a query region, so exact spatial correspondence is not required.

FRN [7] casts few-shot classification as feature-map reconstruction. For each candidate class, the support features reconstruct the query through a closed-form ridge-regression solution, and the resulting error is used as the class score. Later studies extended this reconstruction-based framework by considering reconstruction direction, local content enhancement, query feature constraints, and relational feature modeling. BiFRN [25] performs reconstruction in both directions to model within-class variation together with differences between classes. LCCRN [26] combines local content enhancement with cross-reconstruction so that the original appearance cues and locally enhanced cues are used jointly. SRM [8] adds query self-reconstruction to support-based query reconstruction and combines it with a restrained cross-entropy loss to reduce overfitting in few-shot fine-grained classification. AFRN [27] improves fine-grained discrimination by increasing inter-class differences, while SRML [28] strengthens feature representation through mutual learning between structural and relational features.

Additional reconstruction variants have also been studied for fine-grained few-shot recognition. Liu et al. [29] combine bidirectional reconstruction with mixed attention and multi-scale feature fusion in a dual-reconstruction framework. Ou and Zou [30] introduced channel-wise attention-enhanced feature mutual reconstruction, where channel-wise attention is applied to support and query features before mutual reconstruction.

Reconstruction-based methods provide an effective class-matching strategy by representing query features as linear combinations of local support features and evaluating the resulting reconstruction errors. However, they often use only the original local features extracted from a few support samples as the reconstruction basis. The resulting class-wise support representations may therefore be limited by the coverage of the available samples. In addition, local reconstruction errors from different spatial positions must be aggregated into a class score. Since these positions contribute differently to fine-grained category discrimination, fixed or nearly uniform aggregation may reduce the influence of key local regions. HARA does not redefine the reconstruction solver and retains query self-reconstruction as part of the existing reconstruction framework. Instead, it focuses on two stages surrounding reconstruction: HFA enriches the class-wise support representation before class-conditional reconstruction, while AMA adaptively aggregates position-wise joint reconstruction errors after reconstruction. RASB further provides reconstruction-based supervision for intermediate backbone features during training.

2.4. Graph-Structured and Hypergraph-Based Relation Modeling

Graph-structured models represent relationships among samples, regions, or features using nodes and edges. They provide a useful framework for relation modeling in visual tasks. Graph convolutional networks, such as GCN [31], aggregate information from neighboring nodes through neighborhood propagation. GAT [32] further introduces attention weights for different neighbors, making graph-based relation modeling more flexible. However, an edge in an ordinary graph generally represents a pairwise relationship between two nodes. When several local features jointly form a class-related visual pattern, pairwise connections may not fully represent the relations among all participating features.

A hypergraph generalizes ordinary graph edges to hyperedges that can connect multiple nodes simultaneously. It is therefore suitable for representing high-order relations among multiple samples, regions, or features [33]. Classical hypergraph learning has been applied to clustering, classification, and embedding. Several surveys have reviewed its structures, training strategies, and applications [34,35]. Within deep learning frameworks, HGNN [36] introduces hypergraph structures into deep representation learning and aggregates high-order information through hypergraph convolution. HyperGCN [37] builds on hypergraph spectral theory and converts hypergraph learning into a form that can be processed using graph convolution. DHGNN [38] dynamically constructs and updates hypergraph structures so that relation modeling can adapt to the feature distribution. Bai et al. [39] combine hypergraph convolution with hypergraph attention to model the importance of different nodes within each hyperedge. HGNN+ [40] further extends general hypergraph neural networks through spatial-domain hypergraph convolution and hyperedge-group fusion. Hypergraph modeling has also been introduced into visual tasks. For example, Hyper-YOLO [41] maps visual features into a semantic space and constructs a hypergraph to model relations across feature levels and spatial positions. These studies show that hypergraphs provide a useful representation for complex relations involving multiple nodes.

Under the few-shot fine-grained setting, only a small support set is available for each class. Nevertheless, local features from different support images and spatial positions may jointly describe stable class-related visual patterns. Existing local matching, region alignment, and feature calibration methods mainly focus on correspondences between support and query samples. Relations among multiple local support features within the same class are less explicitly modeled in these approaches.

Hypergraph structures have also been explored in few-shot learning. DMH-FSL [42] constructs feature-modal and label-modal hypergraphs to model high-order relations among few-shot samples and uses hypergraph convolution for feature aggregation. MSH-FSL [43] constructs multi-semantic hypergraphs from different semantic mappings and refines sample representations through iterative node–hyperedge message passing. These methods use hypergraphs mainly as task-level relational learners for propagating and refining information among few-shot samples.

HFA differs from both conventional hypergraph neural networks and the above hypergraph-based few-shot methods in its objective and operating granularity. It is applied independently within each candidate class and treats local support features, rather than whole support/query samples, as nodes. Learnable soft node-to-hyperedge assignments form multiple latent hyperedges, and a local feature can contribute to more than one hyperedge. Each hyperedge produces a separate aggregated support feature that is appended to the original class-wise reconstruction basis. Although this aggregation contains a weighted pooling operation, HFA does not collapse all support features into a single pooled representation; the multiple overlapping soft hyperedges retain several relation-specific aggregation features. Moreover, HFA does not perform iterative hyperedge-to-node updates

or multilayer node propagation. Its role is therefore to construct additional features for the class-wise reconstruction basis from support-local relations rather than to refine node representations through general-purpose hypergraph propagation.

3. Method

Given an N -way K -shot episode, HARA first extracts local representations from the support and query images using a shared backbone. Within each candidate class, HFA treats the local support features as hypergraph nodes and forms additional support features through learnable soft hyperedges while preserving the original features. The augmented support representation is then used for class-conditional reconstruction, and the query representation is also reconstructed from itself. Their position-wise reconstruction errors are combined and subsequently weighted by AMA using spatial coefficients derived from the query features. During training, RASB provides an additional reconstruction-oriented constraint on intermediate backbone features; this branch is removed at inference. Figure 1 summarizes the complete HARA pipeline.

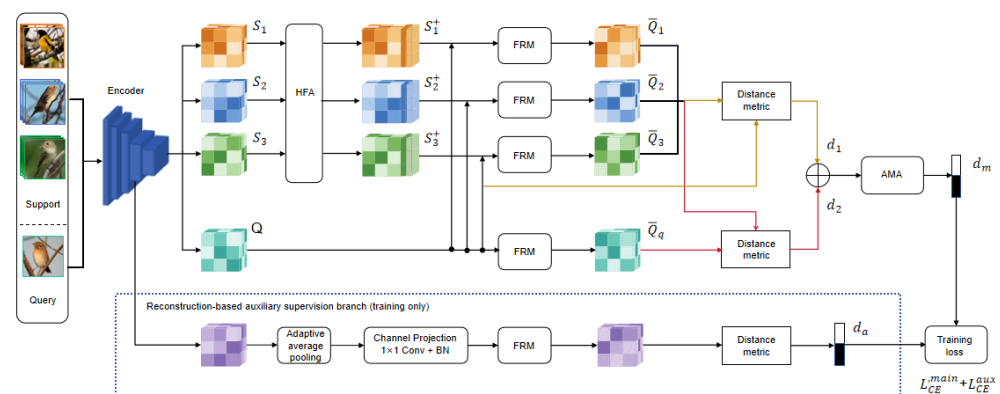


Figure 1. Architecture of HARA. The main classification branch consists of hypergraph-based feature augmentation, class-conditional reconstruction, query self-reconstruction, and adaptive metric aggregation. RASB is used only during training to provide reconstruction-based supervision for intermediate backbone features. FRM denotes the ridge-regression-based feature reconstruction module.

3.1. Problem Definition and Overall Framework

Few-shot learning is organized as a sequence of episodic classification tasks. The dataset classes are divided into non-overlapping training, validation, and test subsets. Episodes sampled from the training classes are used to learn the feature representation and classification metric, whereas evaluation episodes are formed from previously unseen test classes. This protocol measures how well the learned model transfers to novel categories.

Each episode contains N sampled classes, with K labeled examples from every class forming the support set:

$$S = \left\{ \left(x_{c,k}^s, c \right) \mid c = 1, \dots, N; k = 1, \dots, K \right\}, \quad (1)$$

Here, $x_{c,k}^s$ is the k -th support image belonging to class c . The query set contains T additional images from each of the same N classes:

$$Q = \left\{ \left(x_{c,u}^q, c \right) \mid c = 1, \dots, N; u = 1, \dots, T \right\}, \quad (2)$$

where $x_{c,u}^q$ denotes the u -th query image of class c . Query labels contribute to the classification objective during training. At test time, the class representations are formed solely from the episode support set and are then used to predict the query labels.

Let $f_\theta(\cdot)$ denote the shared backbone. Applied to support image $x_{c,k}^s$, it produces R local descriptors of dimension d . The descriptors from the K support images of class c are stacked row-wise to construct $\mathbf{S}_c \in \mathbb{R}^{KR \times d}$. Likewise, the local descriptors of query image x_q form $\mathbf{Q}_q \in \mathbb{R}^{R \times d}$. A row in either matrix corresponds to one location of the associated feature map. With the ResNet-12 backbone used here, the main branch outputs a 5×5 feature map, giving $R = 25$ and $d = 640$.

Based on these definitions, HARA contains a main classification branch and a reconstruction-based auxiliary supervision branch. The main branch first applies HFA to the support feature matrix $\mathbf{S}_c$ and obtains the augmented support representation $\mathbf{S}_c^+$. For each candidate class, $\mathbf{S}_c^+$ is used for class-conditional reconstruction of the query feature matrix $\mathbf{Q}_q$, while $\mathbf{Q}_q$ is also used for query self-reconstruction. The two reconstruction results are then used together to compute position-wise joint reconstruction errors. AMA then predicts position weights from the query features and combines these errors into class scores. During training, RASB provides additional reconstruction-based supervision using intermediate backbone features. During testing, only the main branch is retained for prediction.

3.2. Hypergraph-Based Feature Augmentation Module

In reconstruction-based few-shot classification, the class-wise support representation serves as the basis for class-conditional reconstruction and directly affects reconstruction quality. Feature map reconstruction retains local features from different spatial positions in the support images. However, when only a few support images are available, these features may cover only a limited range of intra-class variations in pose, viewpoint, and local appearance. If a query image contains local patterns that are not adequately represented by the support samples, using only the original support features may limit how well the query features can be reconstructed.

To improve the class-wise support representation when only a few samples are available, we introduce the Hypergraph-Based Feature Augmentation (HFA) module. HFA treats local support features from the same class as hypergraph nodes. Learnable soft hyperedges aggregate these features across support images and spatial positions according to their intra-class relations. The resulting class-augmented features supplement the original support features. Unlike a general hypergraph convolutional network, HFA does not perform multilayer node propagation. Instead, it directly constructs supplementary support features through soft hyperedge aggregation for subsequent reconstruction.

HFA first summarizes each support image by averaging its R local features. Let $\mathbf{s}_{c,k,t}$ represent the descriptor at spatial location t of support image k in class c . The resulting image-level feature is

$$\mathbf{p}_{c,k}^{\text{img}} = \frac{1}{R} \sum_{t=1}^R \mathbf{s}_{c,k,t}, \quad (3)$$

Thus, $\mathbf{p}_{c,k}^{\text{img}}$ is the mean local representation of support image k and supplies sample-level context to the reconstruction basis.

The KR local features in $\mathbf{S}_c$ are treated as hypergraph nodes. The total number of nodes is $V = KR$. Let $\mathbf{s}_{c,v}$ denote the feature of the v -th node. To reduce the effect of feature-magnitude differences on soft hyperedge association, each node feature is first normalized as follows:

$$\hat{\mathbf{s}}_{c,v} = \frac{\mathbf{s}_{c,v}}{\|\mathbf{s}_{c,v}\|_2 + \varepsilon}, \quad (4)$$

The constant ε prevents numerical instability when the feature norm approaches zero. For L latent soft hyperedges, the association weight between node v and the l -th soft hyperedge is defined as:

$$H_c(v, l) = \frac{\exp(\mathbf{w}_l^T \hat{\mathbf{s}}_{c,v})}{\sum_{j=1}^L \exp(\mathbf{w}_j^T \hat{\mathbf{s}}_{c,v})}, \quad (5)$$

Here, $\mathbf{w}_l \in \mathbb{R}^d$ parameterizes latent hyperedge l , while $H_c(v, l)$ gives the soft membership of node v in that hyperedge. Collecting these memberships yields $\mathbf{H}_c \in \mathbb{R}^{V \times L}$. Because soft association is used, each local feature can contribute to multiple hyperedges with different weights. This allows HFA to model overlapping relations among multiple intra-class local features. Equation (5) defines a soft incidence matrix rather than a discrete binary incidence matrix. The softmax is applied across the L hyperedges for each node, so that $\sum_{l=1}^L H_c(v, l) = 1$ for every node v . Each row of $\mathbf{H}_c$ therefore describes the normalized memberships of one support-local node over the latent hyperedges.

For latent hyperedge l , HFA aggregates the original support-local features according to their memberships $H_c(v, l)$:

$$\mathbf{p}_{c,l}^{\text{sem}} = \frac{\sum_{v=1}^V H_c(v, l) \mathbf{s}_{c,v}}{\sum_{v=1}^V H_c(v, l) + \varepsilon}, \quad (6)$$

where $\mathbf{p}_{c,l}^{\text{sem}}$ denotes the aggregation feature of the l -th soft hyperedge in class c . It summarizes the local support features assigned to this hyperedge. Importantly, the normalized node features are used only to compute the association weights. The weighted aggregation in Equation (6) is performed on the original local support features, thereby preserving the feature magnitudes required for subsequent reconstruction. The denominator in Equation (6) normalizes each hyperedge by its total incidence mass, $\sum_v H_c(v, l)$. Thus, HFA performs a normalized node-to-hyperedge aggregation: the soft incidence matrix determines how strongly each local support node contributes to each latent hyperedge, and each hyperedge produces one relation-specific aggregated feature. HFA stops at this node-to-hyperedge stage and does not propagate the aggregated hyperedge features back to the nodes.

The image-level aggregation features and soft hyperedge aggregation features are concatenated to form the class-augmented feature matrix:

$$\mathbf{P}_c = \begin{bmatrix} \mathbf{p}_{c,1}^{\text{img}} \\ \vdots \\ \mathbf{p}_{c,K}^{\text{img}} \\ \mathbf{p}_{c,1}^{\text{sem}} \\ \vdots \\ \mathbf{p}_{c,L}^{\text{sem}} \end{bmatrix} \in \mathbb{R}^{(K+L) \times d}. \quad (7)$$

The class-augmented features are then concatenated with the original support feature matrix to obtain the augmented class-wise support representation:

$$\mathbf{S}_c^+ = \begin{bmatrix} \mathbf{S}_c \\ \mathbf{P}_c \end{bmatrix} \in \mathbb{R}^{(KR+K+L) \times d}, \quad (8)$$

Here, $\mathbf{P}_c$ stacks the image-level and hyperedge-derived features, and $\mathbf{S}_c^+$ is the reconstruction basis after augmentation. The original local support features are retained, while the image-level and soft hyperedge aggregation features encode sample-level context and intra-class local relations, respectively. The resulting $\mathbf{S}_c^+$ is used for class-conditional re-

construction of the query features. For later use, we denote the number of rows in the augmented reconstruction basis by $B_s = KR + K + L$.

3.3. Ridge-Regression-Based Feature Reconstruction

After constructing the augmented support representation, the model evaluates how well each candidate class represents the local features of a query image. For class-conditional reconstruction, we adopt the ridge-regression-based feature reconstruction formulation of FRN [7]. Query self-reconstruction is retained from the reconstruction framework used in SRM [8] and provides an internal reference based on relations among the query features. The outputs of the two reconstruction paths are used to compute position-wise joint reconstruction errors.

3.3.1. Ridge Regression Basics

Ridge regression augments the least-squares objective with an L_2 penalty, which improves estimation stability when observations are limited or feature variables are strongly correlated. Given a response vector $\mathbf{y}$, a design matrix $\mathbf{X}$, and coefficient vector $\boldsymbol{\beta}$, the ridge estimate is obtained from

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta}} \left(\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2 \right). \quad (9)$$

In Equation (9), the residual term controls data fit and the L_2 penalty limits the coefficient norm. The parameter $\lambda \geq 0$ balances these two terms. Setting $\lambda = 0$ recovers ordinary least squares, whereas increasing λ suppresses excessively large coefficients and improves conditioning in correlated or under-constrained systems.

For feature map reconstruction, the vector-valued target is extended to a feature matrix. Let $\mathbf{Y} \in \mathbb{R}^{m \times d}$ denote the target feature matrix, $\mathbf{X} \in \mathbb{R}^{n \times d}$ denote the reconstruction basis, and $\mathbf{A} \in \mathbb{R}^{m \times n}$ denote the reconstruction coefficient matrix. The matrix form of ridge regression is:

$$\hat{\mathbf{A}} = \arg \min_{\mathbf{A}} \left(\|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F^2 + \lambda \|\mathbf{A}\|_F^2 \right), \quad (10)$$

where $\|\cdot\|_F$ is the Frobenius norm. This objective trades reconstruction fidelity against the magnitude of the coefficient matrix $\mathbf{A}$. Its minimizer has the closed form

$$\hat{\mathbf{A}} = \mathbf{Y}\mathbf{X}^T \left(\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I}_n \right)^{-1}. \quad (11)$$

Using the matrix inversion identity, Equation (11) can be equivalently written as:

$$\hat{\mathbf{A}} = \mathbf{Y} \left(\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I}_d \right)^{-1} \mathbf{X}^T. \quad (12)$$

In Equations (11) and (12), $\mathbf{I}_n$ and $\mathbf{I}_d$ are identity matrices of size n and d , respectively. The two equivalent forms involve inverses of size $n \times n$ and $d \times d$. We use Equation (12) for both reconstruction paths.

The regularization and scale parameters are parameterized by unconstrained learnable scalars and exponentiated to ensure positivity. Let α_s, β_s denote the parameters of class-conditional reconstruction and α_q, β_q those of query self-reconstruction. The effective coefficients used in our implementation are

$$\lambda_s = \frac{KR}{d} \exp(\alpha_s) + 10^{-6}, \quad \rho_s = \exp(\beta_s), \quad (13)$$

$$\lambda_q = \frac{R}{d} \exp(\alpha_q) + 10^{-6}, \quad \rho_q = \exp(\beta_q). \quad (14)$$

Although HFA increases the number of class-wise basis features from KR to $B_s = KR + K + L$, the scaling factor of λ_s remains KR/d , where KR is the number of original local support features. The appended HFA features therefore do not implicitly increase the regularization strength merely by enlarging the reconstruction basis.

3.3.2. Class-Conditional Reconstruction and Query Self-Reconstruction

Class-conditional reconstruction measures how well each candidate class represents the query image. For a query image x_q , let $\mathbf{Q}_q \in \mathbb{R}^{R \times d}$ denote its local feature matrix. For candidate class c , let $\mathbf{S}_c^+ \in \mathbb{R}^{(KR+K+L) \times d}$ denote the support feature matrix augmented by HFA. Substituting $\mathbf{Q}_q$ and $\mathbf{S}_c^+$ into Equation (12) yields the class-conditional reconstruction:

$$\overline{\mathbf{Q}}_c = \rho_s \mathbf{Q}_q \left[(\mathbf{S}_c^+)^T \mathbf{S}_c^+ + \lambda_s \mathbf{I}_d \right]^{-1} (\mathbf{S}_c^+)^T \mathbf{S}_c^+, \quad (15)$$

where λ_s controls the regularization strength of class-conditional reconstruction and ρ_s adjusts the magnitude of the reconstructed features. Both are learnable positive parameters optimized jointly with the network. Here, $\mathbf{S}_c^+ \in \mathbb{R}^{B_s \times d}$, so $(\mathbf{S}_c^+)^T \mathbf{S}_c^+ \in \mathbb{R}^{d \times d}$. The regularized inverse and the resulting reconstruction operator are therefore both $d \times d$, and multiplication by $\mathbf{Q}_q \in \mathbb{R}^{R \times d}$ produces $\overline{\mathbf{Q}}_c \in \mathbb{R}^{R \times d}$.

Equation (15) does not assign each query position to a single nearest support feature. Instead, it represents each query feature using a linear combination of the augmented support features from class c . These features may come from different support images and spatial positions, so strict spatial correspondence between support and query images is not required. A candidate class that better matches the query image is expected to produce a smaller reconstruction error.

The structure among query-local features provides a second reference that does not depend on the candidate class. We obtain this reference by reconstructing the query representation from itself:

$$\overline{\mathbf{Q}}_q = \rho_q \mathbf{Q}_q \left(\mathbf{Q}_q^T \mathbf{Q}_q + \lambda_q \mathbf{I}_d \right)^{-1} \mathbf{Q}_q^T \mathbf{Q}_q, \quad (16)$$

where λ_q and ρ_q are the regularization and scale parameters of the query self-reconstruction path, respectively. They are also learnable positive parameters. The parameters λ_q , ρ_q , λ_s , and ρ_s are independent so that the two reconstruction paths can adapt to their different targets. For query self-reconstruction, $\mathbf{Q}_q^T \mathbf{Q}_q \in \mathbb{R}^{d \times d}$, and the corresponding reconstruction operator is also $d \times d$. Hence, $\overline{\mathbf{Q}}_q$ has the same dimension as the original query feature matrix, i.e., $\overline{\mathbf{Q}}_q \in \mathbb{R}^{R \times d}$.

The self-reconstructed feature matrix $\overline{\mathbf{Q}}_q$ is formed by linear combinations of local features within the query image and captures its internal local structure. The class-conditional reconstruction reflects how well the support representation of a candidate class represents the query, whereas query self-reconstruction provides an internal query reference. The two reconstruction results are then used to compute the joint reconstruction error described next. The class-conditional and query self-reconstruction procedure is summarized in Algorithm 1.

Algorithm 1: Class-Conditional and Query Self-Reconstruction

Require: $\mathbf{Q}_q \in \mathbb{R}^{R \times d}$; $\{\mathbf{S}_c^+ \in \mathbb{R}^{B_s \times d}\}_{c=1}^N$, with $B_s = KR + K + L$.

Ensure: $\{\bar{\mathbf{Q}}_c\}_{c=1}^N$ and $\bar{\mathbf{Q}}_q$.

1: $\lambda_s \leftarrow \frac{KR}{d} \exp(\alpha_s) + 10^{-6}$, $\rho_s \leftarrow \exp(\beta_s)$

2: $\lambda_q \leftarrow \frac{R}{d} \exp(\alpha_q) + 10^{-6}$, $\rho_q \leftarrow \exp(\beta_q)$

3: **for** $c = 1, \dots, N$ **do**

4: $\mathbf{G}_c \leftarrow (\mathbf{S}_c^+)^T \mathbf{S}_c^+$

5: $\bar{\mathbf{Q}}_c \leftarrow \rho_s \mathbf{Q}_q (\mathbf{G}_c + \lambda_s \mathbf{I}_d)^{-1} \mathbf{G}_c$

6: **end for**

7: $\mathbf{G}_q \leftarrow \mathbf{Q}_q^T \mathbf{Q}_q$

8: $\bar{\mathbf{Q}}_q \leftarrow \rho_q \mathbf{Q}_q (\mathbf{G}_q + \lambda_q \mathbf{I}_d)^{-1} \mathbf{G}_q$

9: **return** $\{\bar{\mathbf{Q}}_c\}_{c=1}^N$, $\bar{\mathbf{Q}}_q$

3.3.3. Joint Reconstruction Error

To retain spatial differences in reconstruction quality, we compute the reconstruction errors separately at each local position. Let $\mathbf{q}_{q,t}$, $\bar{\mathbf{q}}_{c,t}$, and $\bar{\mathbf{q}}_{q,t}$ denote the local features at the t -th spatial position of $\mathbf{Q}_q$, $\bar{\mathbf{Q}}_c$, and $\bar{\mathbf{Q}}_q$, respectively.

At position t , the original-query error is defined as the squared Euclidean distance between the class-conditional reconstruction and the corresponding query feature:

$$e_{q,c,t}^{\text{ori}} = \|\bar{\mathbf{q}}_{c,t} - \mathbf{q}_{q,t}\|_2^2. \quad (17)$$

This term measures how closely the class-conditional reconstruction matches the original query feature at the t -th spatial position. A candidate class that better matches the query image is expected to produce a smaller value.

The self-reconstruction consistency error compares the class-conditional reconstruction with the self-reconstructed query feature at the same position:

$$e_{q,c,t}^{\text{self}} = \|\bar{\mathbf{q}}_{c,t} - \bar{\mathbf{q}}_{q,t}\|_2^2. \quad (18)$$

This error uses the self-reconstructed feature as an internal reference for the query and measures its agreement with the class-conditional reconstruction.

The two terms are added to define the joint reconstruction error for query x_q , class c , and spatial position t :

$$e_{q,c,t}^{\text{joint}} = e_{q,c,t}^{\text{ori}} + e_{q,c,t}^{\text{self}}. \quad (19)$$

The original-query reconstruction error measures agreement with the original query feature, whereas the self-reconstruction consistency error provides an additional reference based on internal query relations. Unlike a single error computed over the entire feature matrix, Equations (17)–(19) retain the reconstruction information at each spatial position. These position-wise errors provide the metric basis for AMA.

3.4. Adaptive Metric Aggregation Module

The two reconstruction paths provide a joint error for every query position and candidate class. Conventional reconstruction-based classifiers usually average these errors uniformly, implicitly assuming that all local regions contribute equally to class discrimination. In fine-grained images, however, discriminative differences are often concentrated in a few local regions, while background regions, occluded regions, or regions shared across classes may contribute less. Uniform aggregation may therefore weaken the influence of discriminative regions on the final class score.

We therefore introduce Adaptive Metric Aggregation (AMA), which derives spatial weights from the query representation and applies them when combining the local joint errors. Rather than modifying class-conditional reconstruction itself, AMA operates only when the local errors are combined into class scores. It therefore models the contribution of each spatial position to the final class decision.

For query position t , the local descriptor $\mathbf{q}_{q,t}$ is first L_2 normalized to obtain $\hat{\mathbf{q}}_{q,t}$. A linear scorer then converts the normalized descriptor into a scalar response, from which we define the residual gating coefficient:

$$g_{q,t} = 1 + 0.5 \tanh(\xi) \tanh(\mathbf{w}_g^T \hat{\mathbf{q}}_{q,t}), \quad (20)$$

Here, the learnable scorer $\mathbf{w}_g \in \mathbb{R}^d$ is initialized as $\mathbf{w}_g \sim \mathcal{N}(0, 0.01^2)$, and the initial value of the learnable strength ξ is 0.10. The scoring vector maps each normalized query feature to a scalar position response, while ξ controls the range of position-wise adjustment. The residual form keeps the gating coefficient positive and uses uniform aggregation as the reference for bounded position-wise adjustment.

The gating coefficients of all spatial positions are normalized to obtain the position weights of the query image:

$$\omega_{q,t} = \frac{g_{q,t}}{\sum_{r=1}^R g_{q,r}}. \quad (21)$$

After normalization, each weight satisfies $\omega_{q,t} > 0$, and the weights over all R positions sum to 1. For a given query image, the same set of position weights is used when comparing it with all candidate classes. The weights in Equation (21) are query-dependent but class-shared. AMA therefore does not generate a separate attention map for each candidate class. Class-specific information remains in $e_{q,c,t}^{\text{joint}}$, which is computed independently for each candidate class c . The shared weight $\omega_{q,t}$ only changes how strongly the t -th query position contributes when these class-dependent reconstruction errors are aggregated. Using the same weights across candidate classes also keeps the spatial weighting scale consistent during class comparison.

The main-branch score for candidate class c is obtained by applying these weights to the corresponding position-wise joint errors:

$$z_{q,c}^m = -\gamma_m \sum_{t=1}^R \omega_{q,t} e_{q,c,t}^{\text{joint}}, \quad (22)$$

where γ_m is a learnable scale parameter. Lower weighted reconstruction error corresponds to greater compatibility between the query and candidate class. Equation (22) therefore uses the negative weighted error as the class score. When all position weights equal $1/R$, the formulation reduces to uniform error aggregation.

The scores over the N candidate classes are normalized using the softmax function:

$$p_{q,c}^m = \frac{\exp(z_{q,c}^m)}{\sum_{j=1}^N \exp(z_{q,j}^m)}, \quad (23)$$

The resulting class probability vector for query image x_q is denoted by $\mathbf{p}_q^m = [p_{q,1}^m, \dots, p_{q,N}^m]^T$. Prediction selects the candidate class with the largest entry in $(\mathbf{p}_q^m)$. In this way, AMA adjusts the contribution of each spatial position without changing the class-conditional reconstruction process. Accordingly, AMA should be interpreted as query-side reweighting of position-wise reconstruction evidence rather than class-specific attention.

3.5. Reconstruction-Based Auxiliary Supervision Branch and Training Objective

The main classification branch performs reconstruction-based classification using the deepest features produced by the backbone. These features contain strong category-level semantics but have relatively low spatial resolution. Intermediate backbone features retain richer local structural information, yet they are supervised mainly through the final classification loss. Relying only on this loss may be insufficient to encourage intermediate features to support the same reconstruction-based objective.

We therefore introduce the Reconstruction-Based Auxiliary Supervision Branch (RASB). RASB extracts the output of the third residual stage of ResNet-12, which has 320 channels, and applies adaptive average pooling to obtain a 5×5 feature map. The pooled features are then processed by a 1×1 convolution, batch normalization, and ReLU activation, with the output channel dimension kept at 320. The 1×1 convolution adjusts the channel representation without mixing neighboring spatial positions, while batch normalization stabilizes the feature distribution. The resulting features retain local structural information and are suitable for reconstruction-based auxiliary classification.

RASB uses the same episodic support–query split as the main branch and performs class-conditional reconstruction and query self-reconstruction with its own regularization and scale parameters. Because the branch is intended only to supervise intermediate feature learning, HFA and AMA are not included. Instead, its position-wise joint reconstruction errors are uniformly aggregated to obtain the auxiliary class score:

$$z_{q,c}^a = -\frac{\gamma_a}{R} \sum_{t=1}^R e_{q,c,t}^a \quad (24)$$

where $e_{q,c,t}^a$ denotes the joint reconstruction error at the t -th spatial position in the auxiliary feature space, and γ_a is the learnable scale parameter of RASB.

The auxiliary class scores are normalized using the softmax function:

$$p_{q,c}^a = \frac{\exp(z_{q,c}^a)}{\sum_{j=1}^N \exp(z_{q,j}^a)} \quad (25)$$

The resulting auxiliary probability vector is denoted by $\mathbf{p}_q^a = [p_{q,1}^a, \dots, p_{q,N}^a]^T$. RASB is used only during training and does not participate in prediction during testing.

For an episode with $|Q|$ queries, let $\mathbf{y}_i$ be the label vector of query i , and let $\mathbf{p}_i^m$ and $\mathbf{p}_i^a$ be its probability vectors from the main and auxiliary branches. The main-branch loss is:

$$\mathcal{L}_{\text{CE}}^{\text{main}} = -\frac{1}{|Q|} \sum_{i=1}^{|Q|} \mathbf{y}_i^T \log(\mathbf{p}_i^m) \quad (26)$$

The auxiliary branch is trained with the analogous cross-entropy term:

$$\mathcal{L}_{\text{CE}}^{\text{aux}} = -\frac{1}{|Q|} \sum_{i=1}^{|Q|} \mathbf{y}_i^T \log(\mathbf{p}_i^a) \quad (27)$$

The overall training objective is:

$$\mathcal{L} = \mathcal{L}_{\text{CE}}^{\text{main}} + \lambda_{\text{aux}} \mathcal{L}_{\text{CE}}^{\text{aux}}, \quad (28)$$

where λ_{aux} controls the contribution of the auxiliary objective. During training, the main loss updates the shared feature extractor, HFA, the reconstruction parameters, and AMA. The auxiliary loss additionally supervises the intermediate backbone features and updates the auxiliary reconstruction head. During testing, RASB is removed, and predictions are based only on the class probabilities produced by the main branch.

4. Experiments and Analysis

We evaluate HARA from both performance and mechanism perspectives. After describing the datasets and implementation protocol, we report comparisons with published few-shot fine-grained methods, complete factorial ablations, and computational complexity. Additional experiments examine the number of soft hyperedges, varying-shot behavior, hard episodes, variability across independent training runs, and sensitivity to bounding-box preprocessing. We further analyze joint-reconstruction discrimination, support coverage, query-position contributions, and the effect of RASB on intermediate representations, followed by spatial-response and foreground-evidence analyses.

4.1. Experimental Setup

Experiments use three established fine-grained benchmarks: CUB-200-2011 [44], Stanford Dogs [45], and Stanford Cars [46]. CUB-200-2011 contains visually similar bird categories whose differences are often concentrated in feather textures, color patterns, head shapes, and other local structures. Stanford Dogs contains multiple dog breeds with substantial variations in pose, scale, and background. Stanford Cars contains different vehicle categories that mainly differ in body structure and local appearance. Together, these datasets cover several representative fine-grained recognition scenarios and allow us to evaluate the ability of HARA to model local discriminative differences from limited labeled samples.

All experiments follow the episodic learning paradigm for both training and testing. The three datasets use disjoint class-level training, validation, and test splits. CUB-200-2011 contains 100 training, 50 validation, and 50 test classes; Stanford Dogs contains 60, 30, and 30 classes, respectively; and Stanford Cars contains 130, 17, and 49 classes, respectively. Unless stated otherwise, evaluation uses 5-way episodes with either one or five support images per class. Accuracy is measured on the associated query set. Final results are averaged over 10,000 episodes sampled from the held-out test classes, and 95% confidence intervals are calculated from the episode-level accuracies. ResNet-12 [47] serves as the backbone, matching the architecture used by the methods reported in Table 1. All backbone parameters are learned from scratch; no external pretrained weights are used.

Table 1. Literature-level 5-way 1-shot and 5-way 5-shot accuracy (%) on CUB-200-2011, Stanford Dogs, and Stanford Cars. The listed methods use ResNet-12. Values are reported as mean accuracy $\pm$ 95% confidence interval.

Methods	CUB		Dogs		Cars	
	1-Shot	5-Shot	1-Shot	5-Shot	1-Shot	5-Shot
ProtoNet (NeurIPS-17) [12]	79.64 $\pm$ 0.20	91.15 $\pm$ 0.11	72.85 $\pm$ 0.22	86.54 $\pm$ 0.13	82.29 $\pm$ 0.20	93.11 $\pm$ 0.10
FRN (CVPR-21) [7]	82.72 $\pm$ 0.16	92.42 $\pm$ 0.14	74.37 $\pm$ 0.15	85.42 $\pm$ 0.12	86.39 $\pm$ 0.13	94.74 $\pm$ 0.06
HelixFormer (MM-22) [17]	81.66 $\pm$ 0.30	91.83 $\pm$ 0.17	65.92 $\pm$ 0.49	80.65 $\pm$ 0.36	79.40 $\pm$ 0.43	92.26 $\pm$ 0.15
TDM (CVPR-22) [20]	82.53 $\pm$ 0.26	92.82 $\pm$ 0.14	74.84 $\pm$ 0.27	86.29 $\pm$ 0.17	86.94 $\pm$ 0.21	96.16 $\pm$ 0.05
BiFRN (AAAI-23) [25]	82.73 $\pm$ 0.19	92.54 $\pm$ 0.10	75.90 $\pm$ 0.22	87.59 $\pm$ 0.12	88.12 $\pm$ 0.16	95.60 $\pm$ 0.07
LCCRN (TCSVT-23) [26]	82.97 $\pm$ 0.19	93.63 $\pm$ 0.10	75.84 $\pm$ 0.20	86.72 $\pm$ 0.11	87.04 $\pm$ 0.17	96.19 $\pm$ 0.07
BSFA (TCSVT-23) [18]	82.27 $\pm$ 0.46	90.76 $\pm$ 0.26	69.58 $\pm$ 0.50	82.59 $\pm$ 0.33	88.93 $\pm$ 0.38	95.20 $\pm$ 0.20
C2-Net (AAAI-24) [21]	83.77 $\pm$ 0.43	92.57 $\pm$ 0.25	75.50 $\pm$ 0.49	87.65 $\pm$ 0.28	88.96 $\pm$ 0.37	95.16 $\pm$ 0.20
AFRN (PR-24) [27]	83.95 $\pm$ 0.18	93.17 $\pm$ 0.10	77.01 $\pm$ 0.21	88.60 $\pm$ 0.12	89.27 $\pm$ 0.16	96.89 $\pm$ 0.06
SRM (PR-24) [8]	83.82 $\pm$ 0.18	93.45 $\pm$ 0.10	76.54 $\pm$ 0.21	88.52 $\pm$ 0.12	88.02 $\pm$ 0.16	96.23 $\pm$ 0.07
TST_MFL (IF-25) [48]	80.79 $\pm$ 0.84	91.15 $\pm$ 0.40	67.84 $\pm$ 0.82	81.72 $\pm$ 0.61	82.82 $\pm$ 0.84	93.61 $\pm$ 0.35
MFSI-Net (KBS-25) [19]	80.70 $\pm$ 0.42	91.30 $\pm$ 0.24	76.35 $\pm$ 0.46	88.17 $\pm$ 0.26	85.60 $\pm$ 0.37	95.26 $\pm$ 0.16
ATR-Net (TCSVT-25) [22]	81.37 $\pm$ 0.42	91.26 $\pm$ 0.24	75.68 $\pm$ 0.46	87.77 $\pm$ 0.26	85.60 $\pm$ 0.38	95.24 $\pm$ 0.16
SRML (PR-25) [28]	83.05 $\pm$ 0.43	92.74 $\pm$ 0.23	72.97 $\pm$ 0.47	86.01 $\pm$ 0.30	87.49 $\pm$ 0.36	95.34 $\pm$ 0.16
HARA (Ours)	84.09 $\pm$ 0.18	93.48 $\pm$ 0.09	76.91 $\pm$ 0.21	88.69 $\pm$ 0.13	88.53 $\pm$ 0.16	95.87 $\pm$ 0.09

Images are resized to 84×84 pixels before being passed to the network. CUB-200-2011 and Stanford Dogs are preprocessed using the official bounding-box annotations, with a padding ratio of 0.25 used for Stanford Dogs, whereas Stanford Cars is evaluated using the original uncropped images. For CUB-200-2011, the official bounding boxes are used to crop images in the training, validation, and test splits. They are used only for image preprocessing and are never provided to the classifier as model inputs. An additional evaluation without bounding-box cropping is reported in Section 4.5.8. The main classification branch uses the 5×5 feature map produced by the final stage of the backbone, with a local feature dimension of 640. Training episodes are 10-way 5-shot, validation uses 5-way 5-shot episodes, and test episodes use the 5-way protocol in both the 1-shot and 5-shot cases. Unless otherwise specified, each class contains 16 query images per episode. Optimization uses SGD for 1200 epochs with an initial learning rate of 0.1 and momentum of 0.9; the learning rate is reduced at epochs 400 and 800. We use an auxiliary-loss coefficient of 0.1 and set the default number of soft hyperedges to $L = 4$. Both the main classification branch and RASB are optimized during training. During testing, RASB is removed, and predictions are produced only by the main classification branch. Unless otherwise stated, all experiments use these settings. The checkpoint with the highest validation accuracy is used for final testing. Unless otherwise stated, random seed 42 is used.

4.2. Comparison with Existing Methods

Table 1 places HARA alongside representative methods from metric learning, local matching and alignment, reconstruction-based classification, and feature calibration. Results are shown for the 5-way 1-shot and 5-way 5-shot settings on all three benchmarks. All comparator results in Table 1 are taken from previously published studies and were not reproduced in this work. The source studies do not necessarily use identical data splits, pretraining, or episode construction. Table 1 should therefore be interpreted as a literature-level comparison.

Across the six reported dataset/shot combinations, HARA remains competitive with the methods in Table 1. It gives the largest reported 1-shot value on CUB-200-2011 and the largest reported 5-shot value on Stanford Dogs. Section 4.3 examines the individual contributions of HFA, AMA, and RASB under matched experimental settings.

HARA reports higher accuracies than ProtoNet across the three datasets and both shot settings. For CUB-200-2011, the corresponding HARA values are 84.09% and 93.48%, which are 4.45 and 2.33 percentage points above the ProtoNet entries in Table 1.

The strongest relative results in Table 1 occur on CUB-200-2011 and Stanford Dogs. On CUB-200-2011, it achieves the highest 1-shot accuracy of 84.09% and the second-highest 5-shot accuracy of 93.48%, which is only 0.15 percentage points below LCCRN. On Stanford Dogs, HARA achieves the highest 5-shot accuracy of 88.69%. Its 1-shot accuracy reaches 76.91%, second only to AFRN among the methods listed in Table 1. These two benchmarks therefore account for HARA's strongest relative rankings in Table 1.

Relative to the reported FRN results in Table 1, HARA achieves higher accuracies across all three datasets. The 1-shot accuracies increase by 1.37, 2.54, and 2.14 percentage points on CUB-200-2011, Stanford Dogs, and Stanford Cars, respectively. For 5-shot evaluation, the differences are 1.06, 3.27, and 1.13 percentage points, respectively. These literature-level comparisons indicate that HARA remains competitive relative to the basic FRN reconstruction framework across the three datasets. The specific contributions of HFA, AMA, and RASB are examined by the ablation study in Table 2.

Table 2. Complete factorial ablation of HFA, AMA, and RASB on CUB-200-2011, Stanford Dogs, and Stanford Cars (%). ✓ indicates that a module is used, and × indicates that it is removed. All experiments use ResNet-12 as the backbone.

HFA	AMA	RASB	CUB		Dogs		Cars	
			1-Shot	5-Shot	1-Shot	5-Shot	1-Shot	5-Shot
×	×	×	83.04 ± 0.18	93.04 ± 0.09	76.44 ± 0.21	88.35 ± 0.13	87.93 ± 0.16	95.54 ± 0.09
✓	×	×	83.25 ± 0.18	93.13 ± 0.09	76.54 ± 0.21	88.47 ± 0.13	88.16 ± 0.16	95.74 ± 0.09
×	✓	×	83.32 ± 0.18	93.17 ± 0.09	76.51 ± 0.21	88.42 ± 0.13	88.12 ± 0.16	95.68 ± 0.09
×	×	✓	83.19 ± 0.18	93.14 ± 0.09	76.47 ± 0.21	88.39 ± 0.13	88.07 ± 0.16	95.64 ± 0.09
×	✓	✓	83.53 ± 0.18	93.28 ± 0.09	76.59 ± 0.21	88.56 ± 0.13	88.35 ± 0.16	95.79 ± 0.09
✓	×	✓	83.39 ± 0.18	93.24 ± 0.09	76.58 ± 0.21	88.40 ± 0.13	88.27 ± 0.16	95.76 ± 0.09
✓	✓	×	83.52 ± 0.18	93.43 ± 0.09	76.58 ± 0.21	88.51 ± 0.13	88.39 ± 0.16	95.81 ± 0.09
✓	✓	✓	84.09 ± 0.18	93.48 ± 0.09	76.91 ± 0.21	88.69 ± 0.13	88.53 ± 0.16	95.87 ± 0.09

For Stanford Cars, HARA records 88.53% in the 1-shot case and 95.87% in the 5-shot case. The former exceeds the FRN and SRM entries in Table 1, indicating that HARA remains competitive when only one support example is available. Although its 5-shot accuracy is lower than those of AFRN, SRM, LCCRN, and TDM, it remains competitive with the other methods listed in Table 1. Overall, HARA shows particularly strong relative performance on CUB-200-2011 and Stanford Dogs while maintaining competitive results on Stanford Cars.

4.3. Ablation Study

We perform a complete 2^3 factorial ablation of HFA, AMA, and RASB on CUB-200-2011, Stanford Dogs, and Stanford Cars. The baseline uses the same backbone, reconstruction framework, and training setting as HARA, with all three modules removed. Within each dataset, the eight configurations use the same experimental protocol and differ only in the combination of HFA, AMA, and RASB.

Each module improves the matched baseline when used alone. Across the six evaluation settings, HFA provides gains of 0.09–0.23 percentage points. AMA provides gains of 0.07–0.28 percentage points. RASB provides gains of 0.03–0.15 percentage points. These results confirm that all three modules provide independent benefits.

The interactions among the three modules are limited. A factorial contrast analysis shows that the absolute pairwise interaction effects are below 0.13 percentage points in all settings. The three-way interaction effects are below 0.10 percentage points. These results indicate that no strong interaction dominates the overall improvement.

Full HARA achieves the highest accuracy in all six settings. Compared with the baseline, it improves CUB-200-2011 by 1.05/0.44 percentage points. The gains on Stanford Dogs are 0.47/0.34 percentage points. The gains on Stanford Cars are 0.60/0.33 percentage points. HFA enriches the class-wise support representation. AMA reweights position-wise reconstruction evidence. RASB provides reconstruction-oriented supervision during training. Together, the three modules provide complementary improvements within HARA.

4.4. Model Complexity and Efficiency

Beyond classification accuracy, we report parameter count, FLOPs, and training time for HARA and the comparison methods in Table 3. We also profile inference latency and peak GPU memory for HARA and the matched baseline under identical hardware and implementation conditions.

Table 3. Complexity comparison of the evaluated methods. Params denotes parameter count; FLOPs denotes the theoretical computational cost per episode; training time denotes the mean profiled time per episode.

Methods	Params (M)	FLOPs (G) 1-Shot	FLOPs (G) 5-Shot	Train Time (s) 1-Shot	Train Time (s) 5-Shot
FRN	12.43	563.68	704.60	0.060	0.086
TDM	17.35	563.96	704.88	0.064	0.091
BiFRN	16.12	578.46	723.07	0.054	0.083
C2-Net	18.49	574.10	754.46	0.337	0.495
ATR-Net	22.82	568.17	719.41	0.107	0.146
SRML	12.69	574.20	734.85	0.297	0.495
HARA (Ours)	12.94	564.13	705.17	0.185	0.210

HARA contains 12.94 M parameters, which is only slightly higher than FRN with 12.43 M parameters and SRML with 12.69 M parameters. Its parameter count is lower than those of TDM, BiFRN, C2-Net, and ATR-Net. These results show that introducing HFA, AMA, and RASB causes only a small increase in model size.

HARA uses 564.13 GFLOPs for a 1-shot episode and 705.17 GFLOPs for a 5-shot episode. These values are close to those of FRN and TDM and are lower than those of BiFRN, C2-Net, ATR-Net, and SRML. Therefore, although HARA introduces support-side feature augmentation and query-side adaptive aggregation, its FLOPs remain close to those of FRN and TDM.

The profiled training time averages 0.185 s per 1-shot episode and 0.210 s per 5-shot episode. These values are higher than those of FRN, TDM, BiFRN, and ATR-Net but lower than those of C2-Net and SRML. The additional training cost is associated with the extra computations introduced by HFA, AMA, and the training-only RASB. We further profile test-time efficiency using the matched baseline and HARA. Both models are evaluated on an NVIDIA GeForce RTX 3090 in FP32. We use 20 warm-up runs followed by 100 timed forward passes. Data loading is excluded from the measurement. RASB is removed during HARA inference.

For 1-shot evaluation, the baseline requires 0.0744 ± 0.0043 s per episode, while HARA requires 0.0725 ± 0.0041 s. Their peak allocated GPU memory is 787.96 MiB and 789.91 MiB, respectively. For 5-shot evaluation, the corresponding inference times are 0.0805 ± 0.0064 s and 0.0807 ± 0.0027 s. The peak memory values are 927.25 MiB and 929.21 MiB, respectively. The measured inference times are similar for the two models. HARA increases peak GPU memory by approximately 2 MiB in both settings.

Overall, HARA introduces only modest increases in parameter count and theoretical computation, although it requires additional training time. At test time, its latency and peak GPU memory remain close to those of the matched baseline.

4.5. Further Analysis

4.5.1. Analysis of the Number of Soft Hyperedges

The number of soft hyperedges, denoted by L , is an important hyperparameter of HFA. It controls the number of latent soft hyperedges used to aggregate relations among local support features from the same class. To examine sensitivity to L , we evaluate CUB-200-2011 and Stanford Dogs with both 1-shot and 5-shot 5-way episodes; Figure 2 summarizes the results.

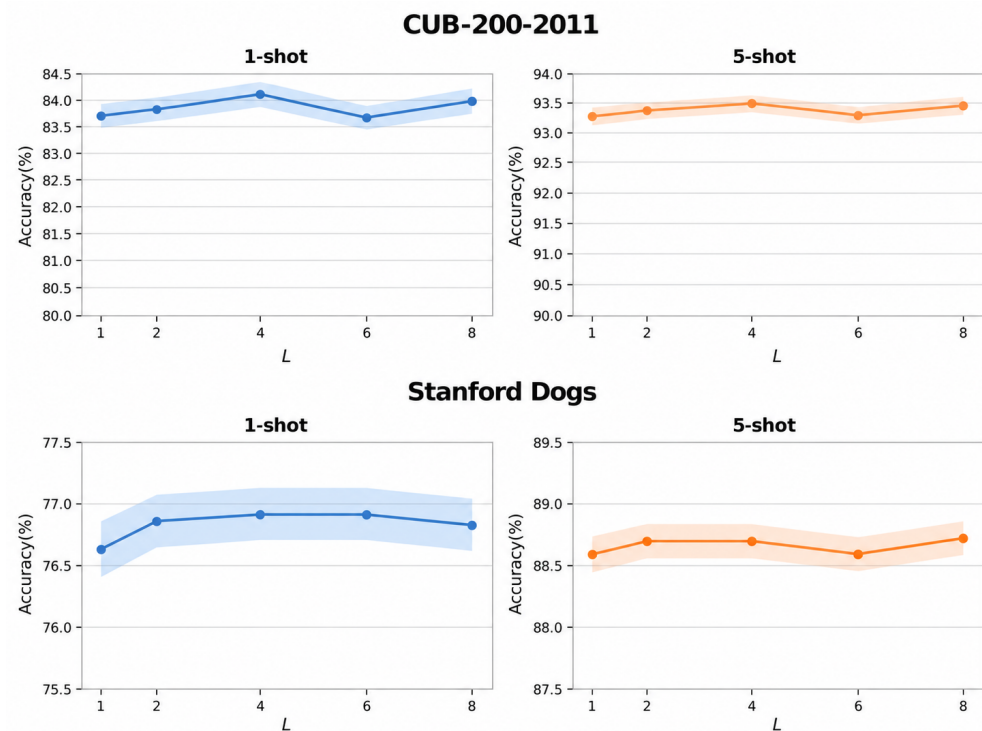


Figure 2. Sensitivity to the number of soft hyperedges L on CUB-200-2011 and Stanford Dogs. Curves correspond to 1-shot and 5-shot 5-way evaluation, with shaded 95% confidence intervals computed from episode-level accuracies.

On CUB-200-2011, the accuracy increases as L grows from 1 to 4. The highest accuracy is obtained at $L = 4$ for both 1-shot and 5-shot evaluation. Larger values of L lead only to small fluctuations.

Stanford Dogs shows a similar level of stability. In the 1-shot setting, $L = 4$ and $L = 6$ obtain the same highest accuracy of 76.91%. In the 5-shot setting, $L = 8$ gives the highest accuracy of 88.71%. This is only 0.02 percentage points higher than the result at $L = 4$.

These results show that HARA is not highly sensitive to moderate changes in L . The setting $L = 4$ was fixed as the default configuration before the present sensitivity analysis. The experiments in Figure 2 were conducted to evaluate the robustness of this fixed setting and were not used to select L based on test performance. The results show that HARA remains stable under moderate changes in the number of soft hyperedges.

4.5.2. Analysis Under Different Numbers of Support Samples

To study the effect of support-set size, we evaluate HARA on CUB-200-2011 with 1, 3, 5, and 7 support images per class in 5-way episodes; Figure 3 reports the resulting accuracies.

Accuracy rises from 84.10% with one support image to 92.00% with three, then reaches 93.59% and 94.21% with five and seven support images, respectively. The largest increase occurs from one to three shots; later gains are smaller, indicating diminishing returns as the support set grows.

These results show that HARA can effectively exploit the additional local information provided by larger support sets. A larger support set provides broader coverage of local appearance variations for reconstruction. The varying-shot results characterize how HARA behaves as the support-set size changes.

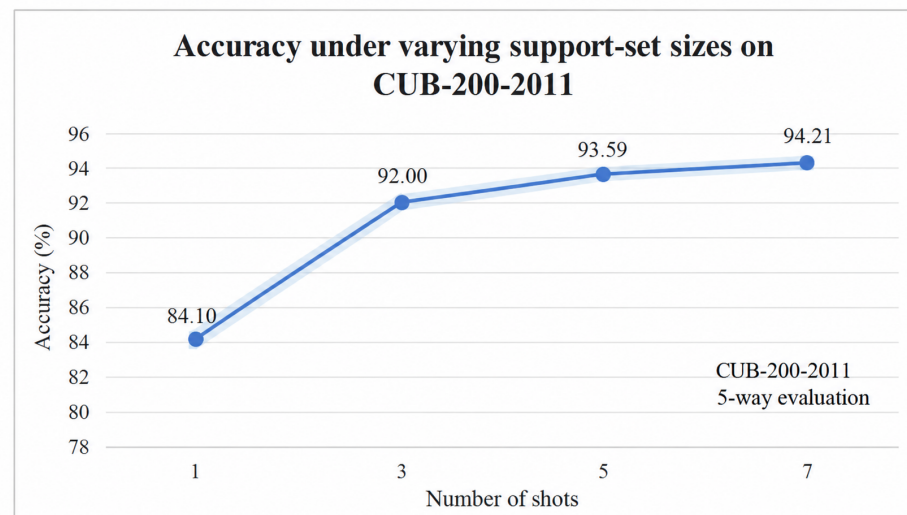


Figure 3. HARA accuracy on CUB-200-2011 as support-set size varies. All results use 5-way episodes. The horizontal axis gives support-set size, the vertical axis gives accuracy, and the shaded region shows the episode-level 95% confidence interval.

4.5.3. Analysis of Hard Episodes

To evaluate HARA on difficult episodes while reducing selection bias, we conduct a 5-way 1-shot analysis on CUB-200-2011 using 10,000 test episodes. Each episode contains 16 query images per class. Using a fixed random split, we divide them into eight calibration queries and eight held-out evaluation queries per class. The matched baseline is evaluated on the calibration queries, and its accuracy is used to determine episode difficulty. We then select the hardest 5%, 10%, 15%, and 25% of the episodes. Performance is measured only on the held-out evaluation queries. The same episodes and query splits are used for all compared models. Thus, the query observations used for difficulty ranking are disjoint from those used for performance evaluation. We report the gains of HFA-only, AMA-only, and full HARA over the matched baseline. The results are shown in Figure 4.

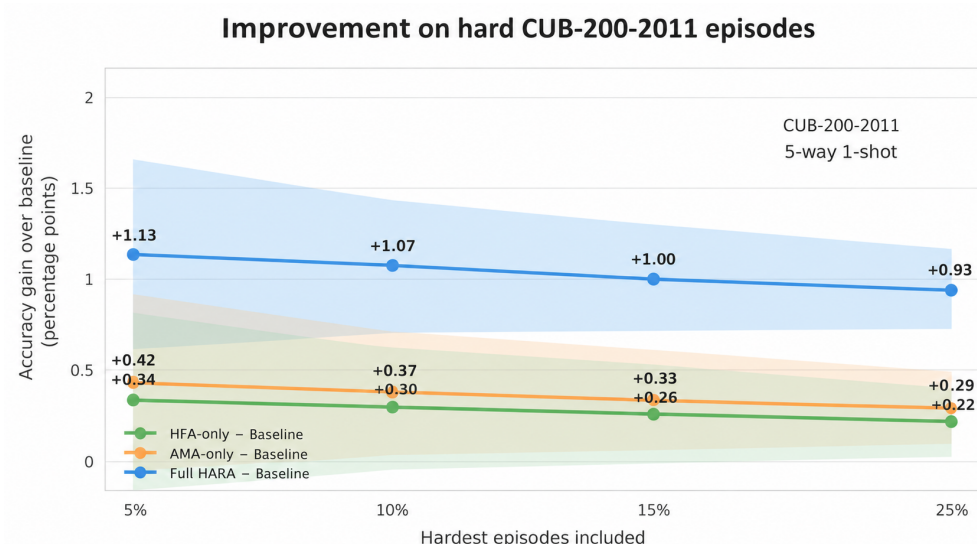


Figure 4. Performance gains on hard episodes of CUB-200-2011 under the 5-way 1-shot setting. Episode difficulty is determined using baseline accuracy on calibration queries, while performance is measured on disjoint held-out evaluation queries. The curves show the gains of HFA-only, AMA-only, and full HARA over the matched baseline. The shaded regions indicate 95% confidence intervals obtained by paired episode-level bootstrap resampling.

Full HARA consistently improves the baseline across all four hard-episode subsets. The gain is 1.13 percentage points on the hardest 5% of episodes. The improvements are 1.07, 1.00, and 0.93 percentage points on the hardest 10%, 15%, and 25% subsets, respectively. The gain remains close to one percentage point as a larger proportion of episodes is included.

The point estimates of the single-module configurations are also positive across all hard subsets. HFA-only improves the baseline by 0.34, 0.30, 0.26, and 0.22 percentage points on the hardest 5%, 10%, 15%, and 25% subsets, respectively. AMA-only provides corresponding gains of 0.42, 0.37, 0.33, and 0.29 percentage points. The confidence intervals are wider for the smaller hard subsets, and some single-module intervals overlap zero. Nevertheless, the positive point estimates are consistent with the overall factorial ablation in Table 2. Full HARA shows the largest gain across all four subsets.

4.5.4. Analysis of Class Discrimination in Joint Reconstruction

Section 3.3 uses two reconstruction errors for class measurement. The original-query reconstruction error compares the class-conditional reconstruction with the original query feature. The self-reconstruction consistency error uses the query self-reconstruction as an internal reference. The two terms describe different aspects of reconstruction quality. We therefore analyze the original-query reconstruction error, the self-reconstruction consistency error, and their joint error separately.

For a query image x_q with ground-truth class y_q , Equations (17)–(19) define the three errors at each spatial position. To preserve the inference path of the trained model, the same query-dependent position weights produced by AMA are applied to the original-query, self-reconstruction consistency, and joint reconstruction components. This allows the three components to be compared under the same spatial aggregation rule. For component a , the class-level reconstruction error for candidate class c is defined as:

$$E_{q,c}^a = \sum_{t=1}^R \omega_{q,t} e_{q,c,t}^a, \quad a \in \{\text{ori}, \text{self}, \text{joint}\}, \quad (29)$$

Here, $\omega_{q,t}$ is the query-dependent position weight defined in Equation (21). The same weights are used for all candidate classes and for all three reconstruction components.

Based on the class-level error, the hardest non-target class for query image x_q is defined as:

$$c_q^* = \arg \min_{c \neq y_q} E_{q,c}^{\text{joint}}, \quad (30)$$

Here, $E_{q,c}^{\text{joint}}$ is the joint reconstruction error in Equation (29). The class c_q^* has the smallest joint error among all non-target classes. It is therefore the most confusing non-target class for the current query. The same c_q^* is used for all three reconstruction components. The class-discriminative margin of component a is defined as:

$$M_q^a = E_{q,c_q^*}^a - E_{q,y_q}^a, \quad (31)$$

A positive M_q^a means that the ground-truth class has a smaller reconstruction error than the hardest non-target class. A larger margin indicates stronger error separation. Equation (29) uses the same spatial aggregation for all three components. The resulting margins can therefore be compared on the same basis. For CUB-200-2011, the analysis uses 10,000 episodes for each of two 5-way protocols, with either one or five support images per class.

Table 4 reports the class-discriminative margins of the three reconstruction components. The original-query reconstruction term has negative margins under both shot settings. The values are −0.168 for 1-shot and −0.183 for 5-shot. Thus, for the hardest

non-target class selected by the joint error, this term alone does not consistently produce positive class separation. By contrast, the self-reconstruction consistency term yields positive margins of 0.713 and 0.904. The self-reconstruction consistency term therefore provides the dominant positive contribution to the joint margin in this analysis.

Table 4. Class-discriminative margins of different reconstruction components. Each entry reports the mean margin and its 95% confidence interval obtained by episode-level bootstrap resampling. The hardest non-target class is determined by Equation (30) and is kept fixed across the three components. The margin is computed by Equation (31). A larger value indicates stronger error separation between the ground-truth class and the hardest non-target class.

Evaluation Setting	Original-Query Reconstruction $\uparrow$	Self-Reconstruction Consistency $\uparrow$	Joint Reconstruction $\uparrow$
5-way 1-shot	−0.168 [−0.170, −0.166]	0.713 [0.709, 0.718]	0.545 [0.542, 0.549]
5-way 5-shot	−0.183 [−0.184, −0.182]	0.904 [0.900, 0.908]	0.721 [0.718, 0.724]

After the two errors are combined, the joint margins reach 0.545 in 1-shot and 0.721 in 5-shot. Both values remain positive. The joint formulation is not intended to maximize either component separately. Instead, it combines two complementary reconstruction relations. The self-reconstruction term compensates for insufficient separation from the original-query term on confusing classes. Both the self-reconstruction and joint margins are larger under the 5-shot setting than under the 1-shot setting, indicating stronger class separation when more support samples are available.

Average results may hide differences among query images. We therefore rank all queries by the joint reconstruction error of their ground-truth class. The full query set and nested highest-error subsets of 50%, 25%, 10%, and 5% are then evaluated. A larger ground-truth joint error indicates a more difficult query under the current support set. Figure 5 reports the self-reconstruction consistency margin and the joint margin across these subsets.

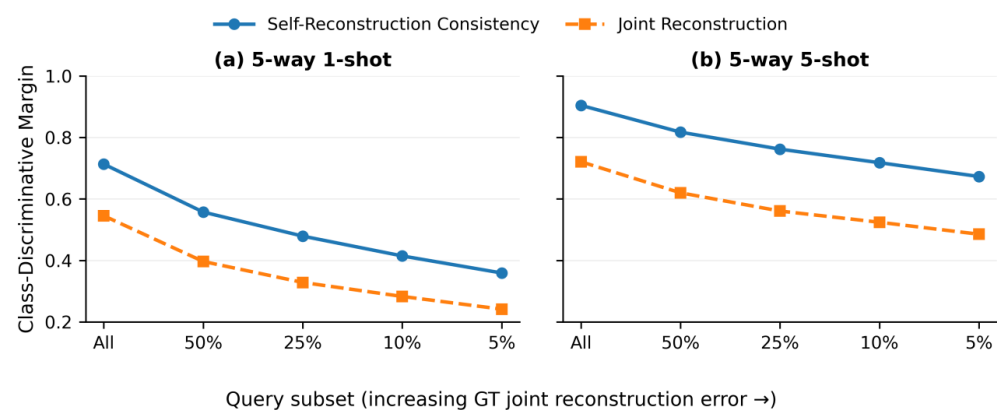


Figure 5. Class-discriminative margins as reconstruction difficulty increases on CUB-200-2011. Results use 5-way episodes with one or five support images per class. Queries are ordered according to the joint reconstruction error associated with their ground-truth class. “All” denotes the complete query set, while the 50%, 25%, 10%, and 5% subsets retain the corresponding highest-error queries; subsets farther to the right are therefore progressively more difficult.

Both margins decrease as reconstruction difficulty increases. In the 1-shot setting, the joint margin falls from 0.545 on all queries to 0.242 on the highest-error 5% subset. In the 5-shot setting, it decreases from 0.721 to 0.485. The self-reconstruction consistency margin

follows the same trend. Queries that are harder to reconstruct from their ground-truth class therefore show weaker separation from the hardest non-target class.

Correlation analysis further characterizes this relationship. For 1-shot, the Pearson correlation between the ground-truth joint reconstruction error and the joint class-discriminative margin is -0.317 , with an episode-cluster bootstrap 95% confidence interval of $[-0.320, -0.313]$. The corresponding Spearman correlation is -0.325 $[-0.328, -0.321]$. For 5-shot, the Pearson and Spearman correlations are -0.251 $[-0.255, -0.248]$ and -0.260 $[-0.263, -0.256]$, respectively. Episodes are resampled as clusters so that query observations sharing the same support set remain together during bootstrap resampling. These negative associations show that larger ground-truth joint reconstruction errors tend to accompany smaller class-discriminative margins. We therefore use the ground-truth joint reconstruction error as an indicator of reconstruction difficulty in this diagnostic analysis.

4.5.5. Analysis of Local Feature Coverage in the Support Set

Class-conditional reconstruction has access to only a few local support features in the few-shot setting. A query image may contain a pose, viewpoint, or local appearance that is poorly covered by the support set. The coverage of the original support features can therefore affect reconstruction reliability. HFA is designed to supplement this support representation through image-level and soft-hyperedge aggregation. We next examine whether limited local support coverage is observable in the test episodes.

Consider local query feature $\mathbf{q}_{q,t}$ at spatial position t of query image x_q . Its ground-truth class is y_q . Let $\mathbf{s}_{y_q,k,r}$ denote the original local feature at position r of the k -th support image from that class. The local support coverage of the query position is defined as:

$$C_{q,t} = \max_{1 \leq k \leq K, 1 \leq r \leq R} \frac{\mathbf{q}_{q,t}^T \mathbf{s}_{y_q,k,r}}{\|\mathbf{q}_{q,t}\|_2 \|\mathbf{s}_{y_q,k,r}\|_2}, \quad (32)$$

Equation (32) compares the query feature with all KR original local support features from the ground-truth class. The largest cosine similarity is retained. A larger $C_{q,t}$ means that a closer support feature is available for the current query pattern. We then average over the R query positions to obtain query-level support coverage:

$$C_q = \frac{1}{R} \sum_{t=1}^R C_{q,t}, \quad (33)$$

A larger C_q indicates more complete local support coverage for the current query image.

For CUB-200-2011, the coverage analysis uses 10,000 episodes for each 5-way protocol, with either one or five support images per class. Queries are partitioned into five equal-frequency groups according to C_q . For each group, we compute the mean support coverage and its corresponding joint class-discriminative margin. Only the original local support features enter this diagnostic; the contribution of HFA itself is assessed separately in Table 2. Figure 6 summarizes the grouped results.

The grouped results exhibit a positive association between support coverage and the joint class-discriminative margin. For 1-shot, the Pearson and Spearman correlations are 0.645 $[0.642, 0.647]$ and 0.636 $[0.633, 0.639]$, respectively. For 5-shot, the corresponding values are 0.535 $[0.531, 0.538]$ and 0.521 $[0.517, 0.524]$. All intervals are 95% confidence intervals obtained by episode-cluster bootstrap resampling. All queries from the same episode are kept together during resampling. The joint margin increases monotonically across the five coverage groups under both shot settings.

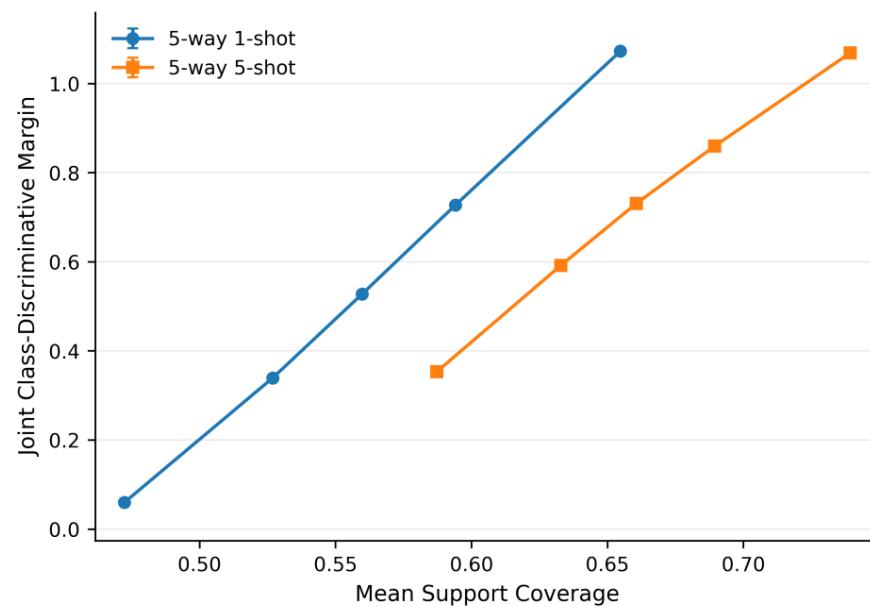


Figure 6. Local support coverage versus joint class-discriminative margin on CUB-200-2011. Queries are partitioned into five equal-frequency groups by support coverage. Points show the mean coverage and mean margin of each group; error bars are 95% confidence intervals obtained by episode-level clustered bootstrap resampling.

In the 1-shot setting, the lowest-coverage 20% group has an average joint margin of only 0.060. The highest-coverage 20% group reaches 1.073. The difference is 1.013, with a 95% confidence interval of [1.007, 1.020]. In the 5-shot setting, the corresponding margin rises from 0.353 to 1.069. The difference is 0.716, with a 95% confidence interval of [0.710, 0.722]. The mean support coverage also increases from 0.562 in 1-shot to 0.662 in 5-shot. Under this coverage measure, the larger support set provides closer local matches for the query features.

These results show a close relationship between local support coverage and reconstruction-based class separation. Poor coverage is associated with a smaller margin between the ground-truth class and confusing classes. The margin increases as support coverage improves. Thus, limited support coverage is observable in the test episodes rather than only being a structural assumption. HFA is designed to address this limitation by supplementing the reconstruction basis with image-level and soft-hyperedge aggregation features. Together with the performance improvement provided by HFA in Table 2, this analysis provides additional empirical support for support-side representation augmentation.

4.5.6. Analysis of Discriminative Contributions Across Query Positions

Section 3.4 replaces fixed averaging of position-wise joint reconstruction errors with query-dependent aggregation. This design assumes that different spatial positions do not contribute equally to class discrimination. We quantify this difference by examining the local discriminative contribution of each query position.

For query image x_q , the hardest non-target class c_q^* is determined by Equation (30). At spatial position t , the local class-discriminative margin is defined as:

$$m_{q,t} = e_{q,c_q^*,t}^{\text{joint}} - e_{q,y_q,t}^{\text{joint}} \quad (34)$$

Here, $e_{q,c,t}^{\text{joint}}$ is the position-wise joint reconstruction error defined in Equation (19). A positive $m_{q,t}$ means that the ground-truth class has the smaller local reconstruction error.

The position therefore supports correct class separation. A negative $m_{q,t}$ indicates that the local reconstruction relation favors the confusing class.

Each query image contains 25 local positions. We rank these positions by $m_{q,t}$ from largest to smallest. The mean margin at each rank is then computed over all test queries. This analysis does not rely on predefined spatial coordinates. It measures the relative discriminative strength of local positions within each query. Figure 7 reports the ranking curves for the two 5-way test protocols, using one or five support images per class.

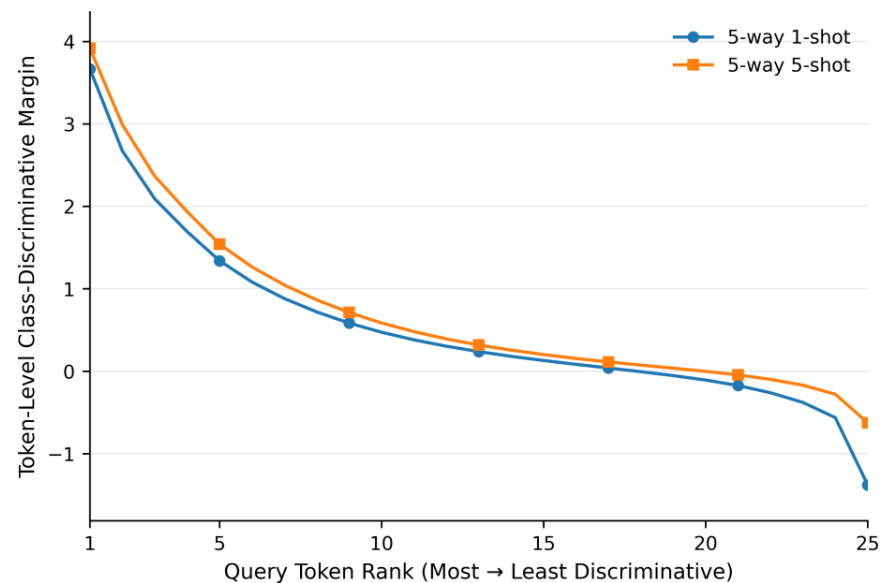


Figure 7. Discriminative contribution of query positions on CUB-200-2011. For each query, the 25 positions are ordered by their position-wise joint-reconstruction margins, from most to least discriminative. The curves show the mean margin at each rank for the 1-shot and 5-shot 5-way protocols; shaded regions are episode-level bootstrap 95% confidence intervals.

Discriminative contribution varies substantially across query positions (Figure 7). The mean margin decreases steadily from high to low ranks. It becomes negative near the end of the ranking. For 1-shot, the mean margin is 3.666 at rank 1 and -1.380 at rank 25. For 5-shot, the corresponding values are 3.916 and -0.627 . Both shot settings show the same overall pattern.

The difference also remains clear when the top five and bottom five positions are compared. For 1-shot, their mean margins are 2.293 and -0.552 . The gap is 2.844, with an episode-level bootstrap 95% confidence interval of [2.836, 2.852]. For 5-shot, the corresponding means are 2.550 and -0.244 . The gap is 2.794, with a 95% confidence interval of [2.785, 2.803].

The negative margins at low ranks are also informative. Some local regions do not support the ground-truth class in the hardest class comparison. Uniform averaging mixes these weak or misleading errors with strongly discriminative ones. This can reduce the separation created by informative regions. The observed heterogeneity provides quantitative motivation for query-dependent weighting in AMA. Together with the performance improvement provided by AMA in Table 2, these results support adaptive modeling of local contributions instead of fixed uniform aggregation.

4.5.7. Robustness Across Independent Training Runs

To evaluate the sensitivity of HARA to training randomness, we conduct three independent training runs on CUB-200-2011 using random seeds 42, 2026, and 3407. All runs use the same data split, network architecture, training protocol, and model-selection

procedure. Table 5 reports the 5-way 1-shot and 5-way 5-shot test results. The confidence interval reported for each individual run is computed from episode-level accuracies, while the variation across the three runs is summarized separately using the mean and standard deviation of the independently trained models.

Table 5. Results of three independent training runs of HARA on CUB-200-2011. The $\pm$ values in the first three rows denote episode-level 95% confidence intervals. The final row reports the mean $\pm$ standard deviation across independent training runs.

Training Seed	1-Shot	5-Shot
42	84.09 $\pm$ 0.18	93.48 $\pm$ 0.09
2026	83.89 $\pm$ 0.18	93.45 $\pm$ 0.09
3407	84.17 $\pm$ 0.18	93.52 $\pm$ 0.10
Mean $\pm$ SD	84.05 $\pm$ 0.14	93.48 $\pm$ 0.04

The results are stable across the three independent runs. The 1-shot accuracy ranges from 83.89% to 84.17%, while the 5-shot accuracy ranges from 93.45% to 93.52%. The standard deviations across runs are 0.14 and 0.04 percentage points for the 1-shot and 5-shot settings, respectively. These results indicate that the reported performance is not strongly affected by the training seed.

4.5.8. Analysis Without Bounding-Box Cropping

The main CUB-200-2011 experiments use the official bounding-box annotations during image preprocessing. The bounding boxes are used only to crop the input images and are never provided to the classifier as model inputs. To examine the dependence of HARA on this preprocessing step, we additionally train and evaluate the full model using the original uncropped CUB images. The class split and all other experimental settings are kept unchanged. The quantitative comparison is reported in Table 6.

Table 6. Effect of bounding-box cropping on HARA on CUB-200-2011.

Preprocessing	1-Shot	5-Shot
Bounding-box crop	84.09 $\pm$ 0.18	93.48 $\pm$ 0.09
No crop	83.30 $\pm$ 0.19	92.99 $\pm$ 0.10

Without bounding-box cropping, HARA achieves 83.30% and 92.99% accuracy under the 1-shot and 5-shot settings, respectively. Compared with the cropped setting, the accuracies decrease by 0.79 and 0.49 percentage points. These results indicate a moderate benefit from bounding-box preprocessing on CUB-200-2011, while HARA remains applicable to the original uncropped images.

4.5.9. Intermediate Feature Analysis of RASB

To further examine the effect of RASB on intermediate feature learning, we compare Full HARA with the no-RASB variant on CUB-200-2011. We extract the output of the third residual stage before the RASB projection head. The analysis uses 3000 randomly sampled 5-way 1-shot episodes with 16 queries per class. For each query image, global average pooling is applied to the intermediate feature map, followed by L_2 normalization. We then compute the intra-class cosine dispersion, the inter-class centroid separation, and their ratio. The same episodes are used for both models, and paired 95% confidence intervals are obtained by bootstrap resampling at the episode level.

As shown in Table 7, the intra-class dispersion increases slightly from 0.1440 to 0.1501 after introducing RASB. In contrast, the inter-class separation increases from 0.1437 to

0.1713. The paired increase is 0.0277, with a 95% confidence interval of [0.0274, 0.0280]. Consequently, the separation ratio increases from 1.0029 to 1.1483, with a paired difference of 0.1454 and a 95% confidence interval of [0.1429, 0.1478]. These results indicate that RASB mainly affects the intermediate representation by enlarging between-class separation rather than by reducing within-class dispersion. The resulting increase in the separation ratio suggests that the auxiliary reconstruction objective encourages intermediate features with stronger class discrimination.

Table 7. Intermediate feature statistics of Full HARA and the variant without RASB on CUB-200-2011. The columns report intra-class cosine dispersion, inter-class centroid separation, and their ratio.

Model	Intra-Class Dispersion ↓	Inter-Class Separation ↑	Separation Ratio ↑
no-RASB	0.1440	0.1437	1.0029
Full HARA	0.1501	0.1713	1.1483

4.6. Visualization and Foreground Response Analysis

To examine the local image regions contributing to class discrimination, we visualize the position-wise positive margin contributions produced by the baseline model and HARA for several query images from CUB-200-2011. For each query image, both methods use the ground-truth class and the same hardest non-target class determined by the baseline model. This keeps the class pair identical for the two response maps. Representative response maps are provided in Figure 8.

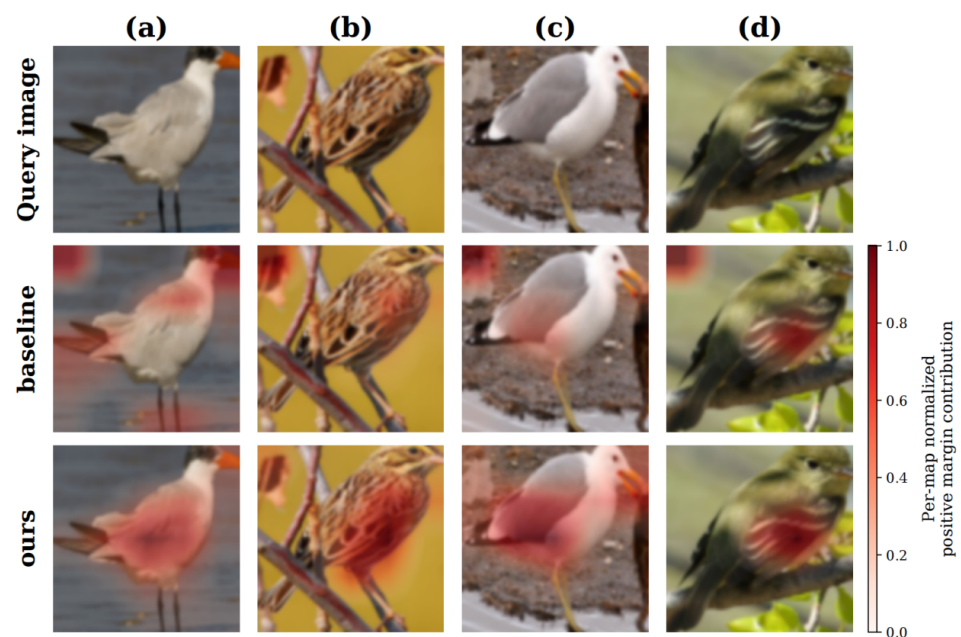


Figure 8. Position-wise positive margin contributions for the matched baseline and HARA on CUB-200-2011. (a) First representative query example; (b) second representative query example; (c) third representative query example; (d) fourth representative query example. For each example, the top row shows the query image, the middle row shows the baseline response map, and the bottom row shows the HARA response map. Each query is evaluated against its ground-truth class and the same baseline-selected hardest non-target class. Each heatmap is normalized by its own maximum positive margin contribution; color intensity therefore represents relative response strength within a map and should not be compared directly across different maps.

At each spatial position, the visualization retains the positive contribution to the decision margin between the ground-truth class and the selected non-target class. Within each heatmap, darker colors indicate larger normalized positive margin contributions. Because

the heatmaps are normalized independently, color intensity should not be compared directly across the baseline and HARA maps. For several examples, the baseline responses extend across both object and background regions. Responses are also visible near image boundaries and in surrounding contextual regions.

In contrast, HARA produces more concentrated responses in the visualized examples. Strong responses are mainly located on the bird body and, in several examples, around the head and neck regions. This response pattern is consistent with the design of HARA: HFA supplements class-related local information on the support side, while AMA adaptively aggregates the position-wise joint reconstruction errors on the query side. In the visualized examples, this design is accompanied by more concentrated responses in fine-grained discriminative regions.

To complement the qualitative visualization, we further quantify the allocation of positive decision evidence to the bird foreground using the official CUB-200-2011 segmentation masks. The masks are used only for post-hoc evaluation and are never provided to the classifier. We evaluate the matched baseline and full HARA on 3000 randomly sampled 5-way 1-shot episodes, with 16 query images per class. For each query, the positive margin contributions on the final 5×5 spatial grid are normalized to sum to one. The aligned segmentation mask is pooled to the same grid, and the foreground evidence allocation is computed according to the foreground occupancy of each position. All valid queries are included without correctness-based filtering. The baseline obtains a foreground evidence allocation of 0.768, whereas HARA obtains 0.780. The paired difference is 0.0120, with a 95% confidence interval of [0.0107, 0.0133]. To avoid treating queries within the same episode as independent observations, the confidence interval is obtained by paired bootstrap resampling at the episode level. These results indicate a modest but stable increase in the proportion of positive discriminative evidence allocated to the bird foreground.

The qualitative visualizations and the foreground evidence analysis provide complementary evidence of the spatial response behavior of HARA. The ablation results in Table 2 provide component-level evidence, while the analyses in Sections 4.5.5 and 4.5.6 further show limited support coverage and unequal query-position contributions. Together, these analyses indicate that the classification improvements of HARA are accompanied by a modest but consistent shift of positive discriminative evidence toward the bird foreground.

5. Conclusions

This study presents HARA, a Hypergraph-Augmented Reconstruction and Adaptive Metric Aggregation Network for few-shot fine-grained image classification. HFA expands the class-wise reconstruction basis by treating local support features as hypergraph nodes and aggregating intra-class relations through learnable soft hyperedges. The augmented basis is used for class-conditional reconstruction, while query self-reconstruction provides an internal query reference. AMA then reweights the resulting position-wise joint errors using query-dependent spatial weights. During training, RASB adds reconstruction-oriented supervision to intermediate backbone features.

Across CUB-200-2011, Stanford Dogs, and Stanford Cars, HARA remains competitive in both 5-way protocols with one or five support images per class. Factorial ablations support the contributions of HFA, AMA, and RASB. The diagnostic analyses further show positive class separation from self-reconstruction consistency, smaller margins when support coverage is limited, and substantial variation in the usefulness of individual query positions. HARA also retains positive gains on hard episodes. On CUB-200-2011, the foreground-response analysis shows a modest increase in the proportion of positive discriminative evidence assigned to bird foreground regions. These findings support HARA as an extension of reconstruction-based metrics for few-shot fine-grained recognition.

Several limitations should also be noted. First, the main CUB-200-2011 protocol uses official bounding-box annotations during image preprocessing. Although the additional no-crop experiment shows that HARA remains effective without bounding-box-assisted preprocessing, the cropped setting benefits from annotation-assisted localization. Second, all experiments use ResNet-12 as the backbone. The behavior of HARA with other feature extractors therefore remains to be evaluated. Third, HFA, AMA, and the training-only RASB introduce additional computation during training, although the measured test-time latency and peak GPU memory remain close to those of the matched baseline. Finally, the current evaluation is limited to three public fine-grained benchmarks. Further evaluation on more diverse datasets and practical recognition scenarios is needed to examine the generality of the method.

Author Contributions: Conceptualization, Z.Z.; methodology, Z.Z.; software, Z.Z.; validation, M.Y. and Y.Y.; formal analysis, Y.Y.; investigation, Z.Z., X.W., Y.Y. and Y.L.; data curation, Y.L.; writing—original draft preparation, Z.Z.; writing—review and editing, Z.W., X.W., M.Y. and Y.Y.; visualization, Z.Z.; supervision, Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: CUB-200-2011, Stanford Dogs, and Stanford Cars are publicly available datasets, and their original sources are cited in Section 4.1 and References [44–46]. The implementation code, dataset split files, experiment configurations, a representative CUB-200-2011 checkpoint, inference-profiling code, and analysis utilities are publicly available at <https://github.com/laorensiji/HARA-FSFG> (accessed on 5 October 2026).

Acknowledgments: Sincere appreciation is extended to Lujia Yang for her contributions to data organization, quantitative analysis, and code optimization in this research. Her technical support has significantly enhanced the robustness of the empirical part of the paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wei, X.-S.; Song, Y.-Z.; Mac Aodha, O.; Wu, J.; Peng, Y.; Tang, J.; Yang, J.; Belongie, S. Fine-grained image analysis with deep learning: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 8927–8948. <https://doi.org/10.1109/TPAMI.2021.3126648>.
2. Lim, J.M.; Lim, K.M.; Lee, C.P.; Lim, J.Y. A review of few-shot fine-grained image classification. *Expert Syst. Appl.* **2025**, *275*, 127054. <https://doi.org/10.1016/j.eswa.2025.127054>.
3. Wang, Y.; Yao, Q.; Kwok, J.T.; Ni, L.M. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.* **2020**, *53*, 63. <https://doi.org/10.1145/3386252>.
4. Li, X.; Yang, X.; Ma, Z.; Xue, J.-H. Deep metric learning for few-shot image classification: A review of recent developments. *Pattern Recognit.* **2023**, *138*, 109381. <https://doi.org/10.1016/j.patcog.2023.109381>.
5. Li, W.; Wang, L.; Xu, J.; Huo, J.; Gao, Y.; Luo, J. Revisiting local descriptor based image-to-class measure for few-shot learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 7260–7268. <https://doi.org/10.1109/CVPR.2019.00743>.
6. Zhang, C.; Cai, Y.; Lin, G.; Shen, C. DeepEMD: Few-shot image classification with differentiable Earth Mover’s distance and structured classifiers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 12203–12213. <https://doi.org/10.1109/CVPR42600.2020.01222>.
7. Wertheimer, D.; Tang, L.; Hariharan, B. Few-shot classification with feature map reconstruction networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual Conference, 19–25 June 2021; pp. 8012–8021. <https://doi.org/10.1109/CVPR46437.2021.00792>.
8. Li, X.; Li, Z.; Xie, J.; Yang, X.; Xue, J.-H.; Ma, Z. Self-reconstruction network for fine-grained few-shot classification. *Pattern Recognit.* **2024**, *153*, 110485. <https://doi.org/10.1016/j.patcog.2024.110485>.

9. Chen, W.-Y.; Liu, Y.-C.; Kira, Z.; Wang, Y.-C.F.; Huang, J.-B. A closer look at few-shot classification. In Proceedings of the 7th International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019. Available online: <https://openreview.net/forum?id=HkxLXnAcFQ> (accessed on 27 September 2026).
10. Finn, C.; Abbeel, P.; Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In Proceedings of the 34th International Conference on Machine Learning (ICML), Sydney, Australia, 6–11 August 2017; Proceedings of Machine Learning Research; Volume 70, pp. 1126–1135. Available online: <https://proceedings.mlr.press/v70/finn17a.html> (accessed on 27 September 2026).
11. Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; Wierstra, D. Matching networks for one shot learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 3630–3638. Available online: <https://proceedings.neurips.cc/paper/2016/hash/90e1357833654983612fb05e3ec9148c-Abstract.html> (accessed on 27 September 2026).
12. Snell, J.; Swersky, K.; Zemel, R.S. Prototypical networks for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4077–4087. Available online: <https://proceedings.neurips.cc/paper/2017/hash/cb8da6767461f2812ae4290eac7cbc42-Abstract.html> (accessed on 27 September 2026).
13. Sung, F.; Yang, Y.; Zhang, L.; Xiang, T.; Torr, P.H.S.; Hospedales, T.M. Learning to compare: Relation network for few-shot learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 1199–1208. <https://doi.org/10.1109/CVPR.2018.00131>.
14. Li, X.; Ding, S.; Xie, J.; Yang, X.; Ma, Z.; Xue, J.-H. CDN4: A cross-view deep nearest neighbor neural network for fine-grained few-shot classification. *Pattern Recognit.* **2025**, *163*, 111466. <https://doi.org/10.1016/j.patcog.2025.111466>.
15. Wu, Y.; Zhang, B.; Yu, G.; Zhang, W.; Wang, B.; Chen, T.; Fan, J. Object-aware long-short-range spatial alignment for few-shot fine-grained image classification. In Proceedings of the 29th ACM International Conference on Multimedia (ACM MM), Virtual Event, China, 20–24 October 2021; pp. 107–115. <https://doi.org/10.1145/3474085.3475532>.
16. Huang, H.; Zhang, J.; Yu, L.; Zhang, J.; Wu, Q.; Xu, C. TOAN: Target-oriented alignment network for fine-grained image categorization with few labeled samples. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 853–866. <https://doi.org/10.1109/TCSVT.2021.3065693>.
17. Zhang, B.; Yuan, J.; Li, B.; Chen, T.; Fan, J.; Shi, B. Learning cross-image object semantic relation in transformer for few-shot fine-grained image classification. In Proceedings of the 30th ACM International Conference on Multimedia (ACM MM), Lisbon, Portugal, 10–14 October 2022; pp. 2135–2144. <https://doi.org/10.1145/3503161.3547961>.
18. Zha, Z.; Tang, H.; Sun, Y.; Tang, J. Boosting few-shot fine-grained recognition with background suppression and foreground alignment. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 3947–3961. <https://doi.org/10.1109/TCSVT.2023.3236636>.
19. Yu, L.; Guan, Z.; Zhao, W.; Yang, Y.; Tan, J. Few-shot fine-grained image classification based on multi-scale selecting and feature interacting. *Knowl.-Based Syst.* **2025**, *330*, 114743. <https://doi.org/10.1016/j.knosys.2025.114743>.
20. Lee, S.; Moon, W.; Heo, J.-P. Task discrepancy maximization for fine-grained few-shot classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 19–24 June 2022; pp. 5331–5340. <https://doi.org/10.1109/CVPR52688.2022.00526>.
21. Ma, Z.-X.; Chen, Z.-D.; Zhao, L.-J.; Zhang, Z.-C.; Luo, X.; Xu, X.-S. Cross-layer and cross-sample feature optimization network for few-shot fine-grained image classification. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 4136–4144. <https://doi.org/10.1609/aaai.v38i5.28208>.
22. Yu, L.; Guan, Z.; Zhao, W.; Yang, Y.; Tan, J. Adaptive task-aware refining network for few-shot fine-grained image classification. *IEEE Trans. Circuits Syst. Video Technol.* **2025**, *35*, 2301–2314. <https://doi.org/10.1109/TCSVT.2024.3499937>.
23. Liu, Y.; Wang, C.; Lu, M.; Yang, J.; Gui, J.; Zhang, S. From simple to complex scenes: Learning robust feature representations for accurate human parsing. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 5449–5462. <https://doi.org/10.1109/TPAMI.2024.3366769>.
24. Wang, S.; Wang, C.; Shi, C.; Liu, Y.; Lu, M. Mask-guided Mamba fusion for drone-based visible-infrared vehicle detection. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 1–12. <https://doi.org/10.1109/TGRS.2024.3452550>.
25. Wu, J.; Chang, D.; Sain, A.; Li, X.; Ma, Z.; Cao, J.; Guo, J.; Song, Y.-Z. Bi-directional feature reconstruction network for fine-grained few-shot image classification. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; Volume 37, pp. 2821–2829. <https://doi.org/10.1609/aaai.v37i3.25383>.
26. Li, X.; Song, Q.; Wu, J.; Zhu, R.; Ma, Z.; Xue, J.-H. Locally-enriched cross-reconstruction for few-shot fine-grained image classification. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 7530–7540. <https://doi.org/10.1109/TCSVT.2023.3275382>.
27. Li, X.; Guo, Z.; Zhu, R.; Ma, Z.; Guo, J.; Xue, J.-H. A simple scheme to amplify inter-class discrepancy for improving few-shot fine-grained image classification. *Pattern Recognit.* **2024**, *156*, 110736. <https://doi.org/10.1016/j.patcog.2024.110736>.
28. Li, X.; Wang, L.; Zhu, R.; Ma, Z.; Cao, J.; Xue, J.-H. SRML: Structure-relation mutual learning network for few-shot image classification. *Pattern Recognit.* **2025**, *168*, 111822. <https://doi.org/10.1016/j.patcog.2025.111822>.
29. Liu, S.; Zhong, W.; Guo, F.; Cong, J.; Gu, B. Fine-grained few-shot image classification based on feature dual reconstruction. *Electronics* **2024**, *13*, 2751. <https://doi.org/10.3390/electronics13142751>.

30. Ou, Q.; Zou, J. Channel-wise attention-enhanced feature mutual reconstruction for few-shot fine-grained image classification. *Electronics* **2025**, *14*, 377. <https://doi.org/10.3390/electronics14020377>.
31. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. In Proceedings of the 5th International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017. Available online: <https://openreview.net/forum?id=SJU4ayYgl> (accessed on 27 September 2026).
32. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph attention networks. In Proceedings of the 6th International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018. Available online: <https://openreview.net/forum?id=rjXMPikCZ> (accessed on 27 September 2026).
33. Zhou, D.; Huang, J.; Schölkopf, B. Learning with hypergraphs: Clustering, classification, and embedding. *Adv. Neural Inf. Process. Syst.* **2006**, *19*, 1601–1608. Available online: https://papers.nips.cc/paper_files/paper/2006/hash/dff8e9c2ac33381546d96deea9922999-Abstract.html (accessed on 27 September 2026).
34. Gao, Y.; Zhang, Z.; Lin, H.; Zhao, X.; Du, S.; Zou, C. Hypergraph learning: Methods and practices. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 2548–2566. <https://doi.org/10.1109/TPAMI.2020.3039374>.
35. Kim, S.; Lee, S.Y.; Gao, Y.; Antelmi, A.; Polato, M.; Shin, K. A survey on hypergraph neural networks: An in-depth and step-by-step guide. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Barcelona, Spain, 25–29 August 2024; pp. 6534–6544. <https://doi.org/10.1145/3637528.3671457>.
36. Feng, Y.; You, H.; Zhang, Z.; Ji, R.; Gao, Y. Hypergraph neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 3558–3565. <https://doi.org/10.1609/aaai.v33i01.33013558>.
37. Yadati, N.; Nimishakavi, M.; Yadav, P.; Nitin, V.; Louis, A.; Talukdar, P. HyperGCN: A new method for training graph convolutional networks on hypergraphs. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 1509–1520. Available online: <https://proceedings.neurips.cc/paper/2019/hash/1efa39bcaec6f3900149160693694536-Abstract.html> (accessed on 27 September 2026).
38. Jiang, J.; Wei, Y.; Feng, Y.; Cao, J.; Gao, Y. Dynamic hypergraph neural networks. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), Macao, China, 10–16 August 2019; pp. 2635–2641. <https://doi.org/10.24963/ijcai.2019/366>.
39. Bai, S.; Zhang, F.; Torr, P.H.S. Hypergraph convolution and hypergraph attention. *Pattern Recognit.* **2021**, *110*, 107637. <https://doi.org/10.1016/j.patcog.2020.107637>.
40. Gao, Y.; Feng, Y.; Ji, S.; Ji, R. HGNN+: General hypergraph neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 3181–3199. <https://doi.org/10.1109/TPAMI.2022.3182052>.
41. Feng, Y.; Huang, J.; Du, S.; Ying, S.; Yong, J.-H.; Li, Y.; Ding, G.; Ji, R.; Gao, Y. Hyper-YOLO: When visual object detection meets hypergraph computation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2025**, *47*, 2388–2401. <https://doi.org/10.1109/TPAMI.2024.3524377>.
42. Xu, R.; Liu, B.; Lu, X.; Zhang, K.; Liu, W. DMH-FSL: Dual-Modal Hypergraph for Few-Shot Learning. *Neural Process. Lett.* **2022**, *54*, 1317–1332. <https://doi.org/10.1007/s11063-021-10684-7>.
43. Chen, H.; Li, L.; Hu, F.; Lyu, F.; Zhao, L.; Huang, K.; Feng, W.; Xia, Z. Multi-semantic hypergraph neural network for effective few-shot learning. *Pattern Recognit.* **2023**, *142*, 109677. <https://doi.org/10.1016/j.patcog.2023.109677>.
44. Wah, C.; Branson, S.; Welinder, P.; Perona, P.; Belongie, S. *The Caltech-UCSD Birds-200-2011 Dataset*; Technical Report CNS-TR-2011-001; California Institute of Technology: Pasadena, CA, USA, 2011. Available online: <https://authors.library.caltech.edu/records/cvm3y-5hh21> (accessed on 27 September 2026).
45. Khosla, A.; Jayadevaprakash, N.; Yao, B.; Li, F.-F. Novel dataset for fine-grained image categorization: Stanford Dogs. In Proceedings of the First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011. Available online: <https://people.csail.mit.edu/khosla/papers/fgvc2011.pdf> (accessed on 27 September 2026).
46. Krause, J.; Stark, M.; Deng, J.; Li, F.-F. 3D object representations for fine-grained categorization. In Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW), Sydney, Australia, 2–8 December 2013; pp. 554–561. <https://doi.org/10.1109/ICCVW.2013.77>.
47. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. <https://doi.org/10.1109/CVPR.2016.90>.
48. Sun, Z.; Zheng, W.; Guo, P.; Wang, M. TST_MFL: Two-stage training based metric fusion learning for few-shot image classification. *Inf. Fusion* **2025**, *113*, 102611. <https://doi.org/10.1016/j.inffus.2024.102611>.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.