



AME: Auxiliary Model Enhancement for cross-modal few-shot learning

Chengqian Yu ^{a,1}, Yihong Chen ^{b,1}, Yuwu Lu ^{a,1,*}

^a School of Artificial Intelligence, South China Normal University, Foshan, China

^b Maynooth International Engineering College, Fuzhou University, Fuzhou, China

ARTICLE INFO

Keywords:

Few-shot classification

Model fine-tuning

Pre-trained vision-language models

Self-supervised learning

ABSTRACT

Few-shot classification aims to recognize novel categories from limited labeled samples. In this work, we study this task in a cross-modal setting by leveraging both visual and textual cues from pre-trained models. Although recent approaches based on pre-trained vision-language models have demonstrated strong performance, their visual branches typically rely on instance-level contrastive paradigms such as CLIP, which can be limited in fine-grained visual representation. Moreover, existing frameworks often underexplores the integration of high-quality features from external vision-only models, leaving room for improvement in low-shot regimes. To address these challenges, we propose Auxiliary Model Enhancement (AME), a framework that enriches the CLIP visual embedding space through the collaborative fusion of self-supervised auxiliary models. AME introduces a feature-initialized predictor to absorb complementary visual information and further refines the fused representation via soft-label knowledge distillation. In addition, we develop a cross-modal adaptation strategy that incorporates semantically enriched class information as supplementary one-shot samples, thereby improving the use of scarce multimodal supervision. Extensive experiments on ImageNet and multiple standard benchmarks validate the effectiveness of AME, demonstrating competitive performance on challenging few-shot classification tasks.

1. Introduction

Few-shot learning, namely the ability to adapt to new tasks with minimal supervision, is a critical capability for intelligent systems. Traditional few-shot learning approaches [1] often rely on single-modality data and predefined class labels, which limit their ability to generalize across diverse and complex tasks. Recently, the emergence of large-scale pre-trained vision-language models [2–5], such as CLIP [6], has opened new avenues for few-shot learning by leveraging rich multimodal embeddings. CLIP aligns images and texts in a shared representation space, enabling zero-shot and few-shot classification with minimal prompt engineering. However, while CLIP-based methods have demonstrated strong performance, they often struggle with task-specific adaptation, especially in low-data regimes [7–9].

To address these limitations, as illustrated in Fig. 1, researchers have explored strategies such as prompt tuning [7,8], feature adaptation [9], and auxiliary model integration [10,11]. Prompt tuning methods, like context optimization, aim to optimize continuous vectors as task-specific prompts, but their reliance on text-only modifications limits their versatility. Feature adaptation approaches, such as CLIP-Adapter [9] and Tip-Adapter [12], introduce additional layers for

residual blending of pre-trained features, enhancing task-specific representations. Auxiliary model integration goes a step further by leveraging multiple foundation models to provide diverse prior knowledge. For instance, CaFo [10] incorporates vision-contrastive, vision-generative, and language-generative models to improve few-shot learning through a combination of synthetic data generation and cache-based prediction blending [12]. AMU-Tuning [11] optimizes logit bias using auxiliary features and uncertainty-based fusion to enhance CLIP-based few-shot learning performance.

Despite significant progress in few-shot learning, most visual encoders still rely on contrastive pre-training with image-text pairs, such as CLIP. Current studies reveal that such approaches suffer from the “CLIP-blind pairs” phenomenon [13], where CLIP erroneously perceives visually distinct images as similar samples. Recent adapter-based studies have also highlighted the need for stronger task-specific refinement under few-shot settings [14]. For semantic representation, existing methods primarily rely on manually designed templates or learnable tokens, limiting semantic scalability. Regarding cross-modal fusion strategies, mainstream approaches typically employ additional

* Corresponding author.

E-mail address: luyuwu2008@163.com (Y. Lu).

¹ These authors contributed equally to this work.

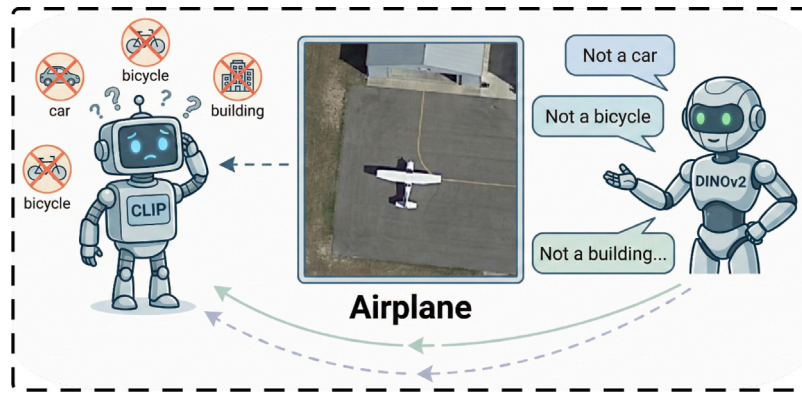


Fig. 1. Examples of CLIP-based few-shot learning are as follows: (a) **CoOp** learns prompt embeddings but lacks visual adaptation. (b) **CLIP-Adapter/Tip-Adapter** introduces learnable MLPs to better adapt CLIP features. (c) **AMU-Tuning/CaFo** enriches visual representations with frozen auxiliary encoders but overlooks semantic cues. (d) **AME (Ours)** fuses dual visual encoders through a feature-initialized MLP and cross-modal adaptation for improved few-shot performance.

projection layers to align image-text category prototypes, yet still struggle to obtain discriminative cross-modal prototypes under limited supervision.

To address these challenges, we propose Auxiliary Model Enhancement (AME), a unified framework that integrates multi-source visual representation fusion, cross-modal feature alignment, and soft-label-based knowledge distillation. Different from Tip-Adapter, which mainly adapts CLIP features, AME introduces a DINOv2 auxiliary branch to provide complementary fine-grained visual cues. Compared with CaFo, which relies more on generated data or prediction-level fusion, AME explicitly coordinates CLIP-based semantic adaptation with auxiliary visual refinement. Compared with AMU-Tuning, which focuses on logit-bias correction, AME learns feature-initialized branch adapters and further refines the DINOv2 branch through soft-label distillation. Specifically, AME combines DINOv2 [15] and CLIP [6] through parallel pathways with residual adapters initialized from pre-trained features. The CLIP branch adopts a cross-modal joint training strategy [16], where semantic features serve as auxiliary support samples to enrich the conventional n -shot setting. The DINOv2 branch serves as an independent visual auxiliary encoder and is optimized with teacher-student distillation. An overview of AME is illustrated in Fig. 2. Compared with the evaluated baselines, AME maintains competitive performance in the low-shot regime and benefits more clearly as the number of shots increases, suggesting improved scalability across few-shot scenarios.

To demonstrate the effectiveness of the proposed AME method, we evaluate its performance across eleven diverse downstream benchmarks [17–27], utilizing multiple backbone architectures including ResNet [28] and Vision Transformer (ViT) [29]. The main contributions of this work are outlined as follows:

- We propose Auxiliary Model Enhancement (AME), a unified method that integrates a vision-language model with an auxiliary self-supervised visual model. This design enriches multi-modal representations and improves downstream performance by effectively leveraging heterogeneous prior knowledge.
- AME incorporates a cross-modal adaptation strategy, an initialized feature-guided MLP, and a knowledge distillation mechanism to improve learning from very limited data while mitigating overfitting.
- Extensive experiments on ImageNet and multiple benchmarks show that AME achieves competitive results in few-shot classification, maintaining strong accuracy in 1- to 4-shot settings and delivering clear gains from 4-shot to 16-shot.

The remainder of this article is organized as follows. Section 2 reviews the related work in few-shot classification and multimodal adaptation. Section 3 details our proposed method, including visual

feature fusion, cross-modal adaptation, and knowledge distillation. Experimental results and analysis are presented in Section 4, followed by the conclusion in Section 5.

2. Related work

This section reviews the literature on three topics central to our work: few-shot classification, strategies for vision model pre-training, and the ecosystem of large-scale pre-trained models.

2.1. Few-shot classification

Few-shot learning aims to recognize novel categories from only a few labeled examples and has been widely studied through metric learning [1], meta-learning [30], and fine-grained discrepancy modeling [31]. With the emergence of vision-language models such as CLIP [6], recent few-shot methods increasingly exploit pre-trained multimodal knowledge. Representative approaches include prompt tuning methods such as CoOp [7] and CoCoOp [8], as well as adapter-based methods such as CLIP-Adapter [9] and Tip-Adapter [12]. Recent studies further show that CLIP-based few-shot learning can benefit from more targeted adaptation strategies: ProLIP [32] adapts the CLIP visual projection layer with regularization, AT-Adapter [33] introduces attribute-level semantic knowledge into category-specific adapters, and Cross-Modal Mapping [34] mitigates the image-text modality gap by aligning visual features with textual prototypes. These works suggest that efficient adaptation, semantic enrichment, and cross-modal alignment remain important for few-shot recognition. In this context, AME integrates CLIP with an auxiliary self-supervised visual model and uses multi-branch adaptation to combine semantic and visual cues for more robust few-shot prediction.

2.2. Pre-training of vision models

Pre-training has become a cornerstone in computer vision, with early models leveraging supervised pre-training on datasets like ImageNet [19] to achieve strong transferability to downstream tasks such as classification, object detection [35], semantic segmentation [36], and 3D recognition [37]. Self-supervised learning, which eliminates the need for labeled data, has advanced significantly through techniques such as contrastive learning [38], self-distillation without labels [39], and masked image modeling [40]. These approaches substantially enhance the transferability of pre-trained models. Among vision-language models, CLIP [6] emerged as a breakthrough, using contrastive pre-training on large-scale image-text pairs to achieve zero-shot and few-shot generalization. Building on these advances, recent works such as CaFo [10] combine multiple pre-trained models (e.g., CLIP [6],

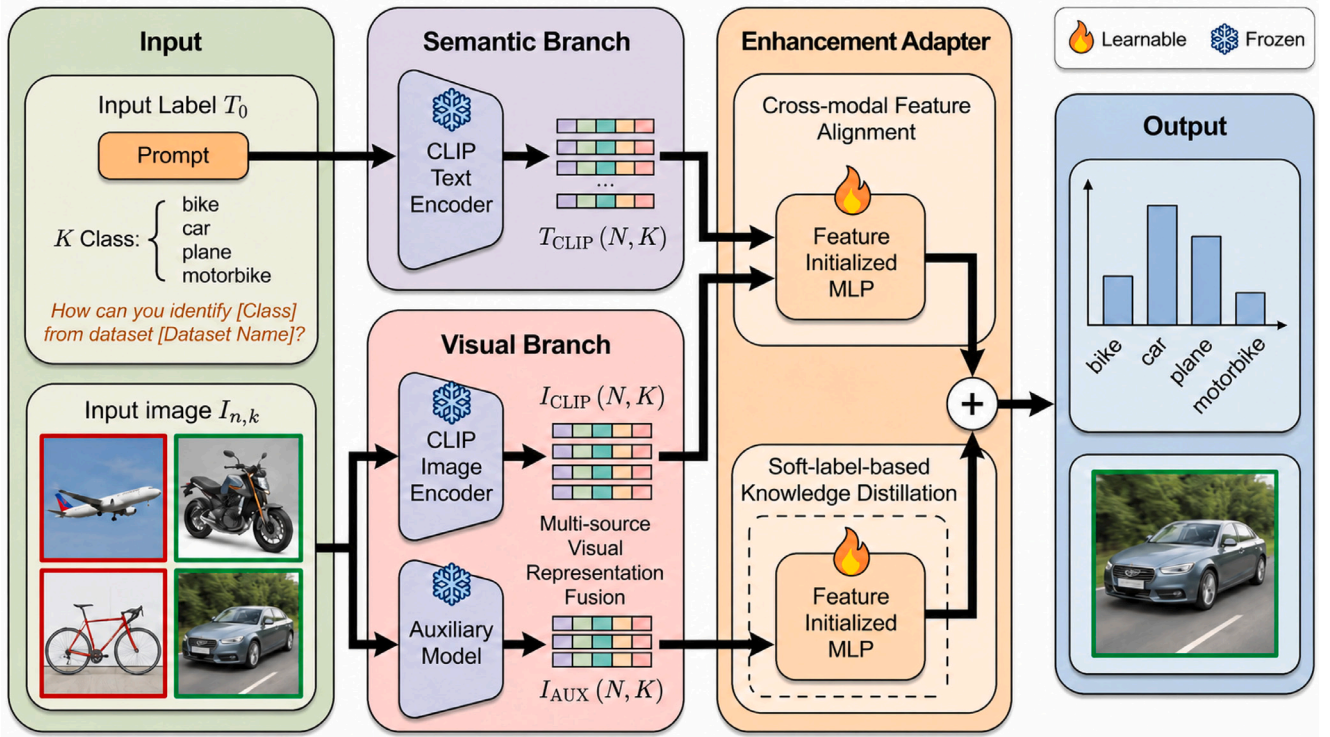


Fig. 2. Overview of AME. Class-name prompts and support images are encoded by frozen CLIP and DINOv2 encoders, then adapted by feature-initialized MLPs. The CLIP branch uses cross-modal alignment, the DINOv2 branch is refined by soft-label distillation, and their logits are fused for prediction.

DINO [39], and DALL-E [41]) to improve few-shot learning performance. However, such auxiliary-model methods often emphasize generated data or prediction-level fusion, while the relationship between semantic adaptation and auxiliary visual refinement is less explicitly modeled. In AME, we extend this idea by incorporating a residual MLP structure with knowledge distillation after both the CLIP and auxiliary branches. This design encourages better complementarity and enhances feature representation for few-shot learning tasks.

2.3. Large-scale pre-trained models

Large-scale pre-trained models have become fundamental to visual representation learning, with foundational architectures evolving from Convolutional Neural Networks like ResNet [28] to more scalable Vision Transformers (ViT) [29] and Swin Transformers [42]. Building on these powerful backbones, models such as CLIP [6], SLIP [43], and CoCa [44] further enhanced generalization by pre-training on massive image-text datasets to learn an aligned and multimodal embedding space. In parallel, advances in self-supervised learning, including contrastive methods like MoCo [45], and masked autoencoders like MAE [40], have unlocked the ability to learn highly adaptable features from visual data alone. Recent work has also shown that pre-trained representations can be refined for fine-grained class-incremental scenarios through pseudo-set frequency modeling [46]. Our work, AME, is built on the insight that these different pre-training paradigms yield complementary strengths. For instance, we harness auxiliary features from both a vision-language model like CLIP, which excels at high-level semantics, and a powerful self-supervised model like DINOv2 [15], known for its rich structural and textural features. By integrating these diverse features into a residual adapter guided by knowledge distillation, our method provides a principled framework to align and enhance embeddings across models, fostering stronger and more robust generalization for few-shot learning.

3. Proposed method

AME consists of three coupled components: a dual-branch CLIP-DINOv2 fusion framework, a cross-modal adaptation strategy for the CLIP branch, and a soft-label knowledge distillation strategy for the DINOv2 auxiliary branch. The fusion framework defines how complementary semantic and visual logits are combined, while cross-modal adaptation and knowledge distillation serve as branch-specific training mechanisms that refine the two logits before final prediction. In this section, we first introduce the CLIP-DINOv2 feature fusion design, then describe cross-modal adaptation and DINOv2-branch distillation. The overall architecture is illustrated in Fig. 2.

3.1. Fusion of CLIP and self-supervised visual embeddings

We propose a multi-branch training framework that integrates the pre-trained models CLIP [6] and DINOv2 [15] to enhance representation learning in few-shot classification tasks. This design is motivated by the observation that contrastive vision-language models such as CLIP often suffer from the “blind pair” problem [13], in which visually distinct samples are incorrectly mapped to similar embeddings due to their reliance on image-text alignment rather than fine-grained visual discrimination. In contrast, DINOv2, trained purely through self-supervised objectives, excels at capturing structural and textural cues, thereby providing complementary and more reliable visual evidence. Leveraging this complementarity, AME employs two independent multilayer perceptrons (MLPs), corresponding to CLIP and DINOv2 respectively, trained in parallel to jointly refine and align their representations.

For each model branch ($\text{model} \in \{\text{CLIP}, \text{DINOv2}\}$), the weight matrix of the linear projection layer is initialized using the class-wise mean feature:

$$W_{\text{model}} = [\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_C]^T, \quad \mathbf{m}_i = \frac{1}{K_i} \sum_{j=1}^{K_i} f_{\text{model}, i, j}, \quad (1)$$

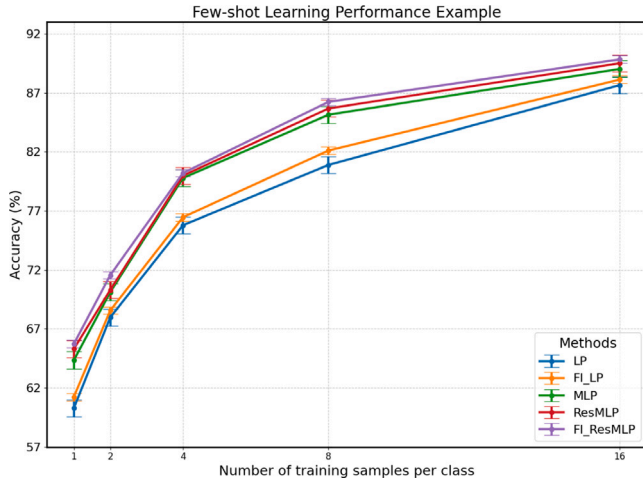


Fig. 3. Few-shot accuracy (%) versus the number of training samples per class (1, 2, 4, 8, and 16). Comparison among five prediction heads (LP, FI_LP, MLP, ResMLP, and FI_ResMLP) with error bars shows that FI_ResMLP achieves strong performance, especially under low-shot settings.

where $f_{\text{model},i,j}$ denotes the feature of the j th sample from class i , and K_i denotes the number of support samples in that class.

To enhance feature alignment, a residual adapter $g_{\text{model}}(\cdot)$ with a learnable weighting parameter β is applied before the classifier:

$$\mathbf{h}_{\text{model}} = \beta g_{\text{model}}(f_{\text{model}}) + (1 - \beta) f_{\text{model}}, \quad \mathbf{s}_{\text{model}} = W_{\text{model}} \mathbf{h}_{\text{model}}. \quad (2)$$

For inference, the final prediction is obtained by aggregating the two branches through a weighted fusion controlled by a hyperparameter:

$$\mathbf{s} = \lambda \mathbf{s}_{\text{CLIP}} + (1 - \lambda) \mathbf{s}_{\text{DINOv2}}, \quad (3)$$

where λ controls the contribution of each encoder during logit fusion at test time. This fusion allows DINOv2 to compensate for CLIP's blind-pair weaknesses, while CLIP provides high-level semantic priors that complement DINOv2's structural cues.

During training, each branch is optimized through its own objective rather than through a supervised loss directly imposed on the fused logits in Eq. (3). The branch-wise supervised objective is written as:

$$\mathcal{L}_{\text{model}} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(\text{softmax}(s_{\text{model},i})_c), \quad (4)$$

where $\text{softmax}(s_{\text{model},i})_c$ is the predicted probability for class c , $y_{i,c}$ is the ground-truth indicator, C is the number of classes, and N is the batch size. In AME, this generic form is instantiated as cross-modal supervision on the CLIP branch and cross-entropy plus distillation on the DINOv2 branch, while the fused logits are used only for final prediction (see Fig. 3).

3.2. Cross-modal adaptation via text-augmented support

To address the limitations of unimodal few-shot learning, which often struggles to form discriminative decision boundaries under scarce supervision, we adopt a cross-modal adaptation strategy on the CLIP branch that explicitly leverages the complementary nature of visual and textual modalities. Specifically, textual embeddings produced by the CLIP text encoder are incorporated as additional support cues alongside visual embeddings from the CLIP image encoder, conceptually enriching each n -shot class with one additional class-level textual support prototype. Built upon the pre-trained vision-language model CLIP, this strategy maps both images and texts into a shared d -dimensional feature space, enabling more robust prototype formation and mitigating

the representation bias commonly observed in purely visual few-shot models.

In our implementation, the textual support features are constructed in a simple and fixed manner. For each class c , we insert the corresponding class name into the prompt template “a photo of a [CLASS]”. The resulting sentence is encoded with the frozen CLIP text encoder. No learnable prompt vectors, prompt ensemble, or text generation module is used. The obtained text embedding is treated as a fixed class-level semantic support feature and is not updated during training. During mini-batch training, this class-level text feature is indexed according to the label of each visual support sample. Thus, the CLIP branch is optimized with both image-support and text-support features. Before computing logits, both CLIP image embeddings and CLIP text embeddings are ℓ_2 -normalized. Therefore, the dot product is equivalent to cosine similarity in the shared CLIP embedding space.

The cross-modal learning objective on the CLIP branch is defined over the class-prototype logits:

$$\mathcal{L}_{\text{cross-modal}}^{\text{CLIP}} = \sum_i \mathcal{H} \left(y_i, \{w_c^\top \phi_{m_i}^{\text{CLIP}}(x_i)\}_{c=1}^C \right), \quad (5)$$

where $\mathcal{H}(y, \cdot)$ denotes the softmax loss, $m_i \in \{\text{image}, \text{text}\}$ indicates the modality type, $\phi_{m_i}^{\text{CLIP}}(x_i)$ represents the corresponding CLIP modality-specific encoder, and w_c denotes the class prototype/classifier weight of class c in the CLIP branch. All embeddings are aligned within the same shared feature space:

$$w_y, \phi_m^{\text{CLIP}}(\cdot) \in \mathbb{R}^d. \quad (6)$$

During inference, the prediction for a test sample x is given by:

$$\hat{y} = \arg \max_y w_{y'} \cdot \phi_m^{\text{CLIP}}(x). \quad (7)$$

By introducing CLIP-based textual embeddings as auxiliary support samples, the proposed cross-modal strategy enriches visual representations and improves generalization under low-shot conditions. Compared with unimodal baselines, this unified CLIP cross-modal design provides additional semantic cues that stabilize prototype learning and effectively alleviate performance degradation when training data are extremely limited.

3.3. Knowledge distillation for auxiliary model compression

In contrast to prototype-based or symmetric distillation schemes, we apply soft-label knowledge distillation to the auxiliary DINOv2 visual branch, aiming to reduce the adapter-side parameter overhead while preserving its complementary discriminative capability. Although incorporating a self-supervised encoder such as DINOv2 provides stronger fine-grained visual representations, it increases the number of learnable parameters in the adapter module. To mitigate this issue, we adopt a teacher-student distillation framework in which a stronger DINOv2-based model with a wider adapter acts as the teacher, and a lightweight adapter-enhanced model with a reduced hidden dimension is trained as the student. This asymmetric design allows the teacher to provide more expressive supervisory signals during training, while the compact student is retained for inference to reduce adapter-side parameter overhead. Through soft-label distillation, knowledge from the auxiliary branch is transferred into a compact representation, enabling efficient inference with limited accuracy degradation.

Specifically, we adopt a soft-label distillation loss based on Kullback-Leibler (KL) divergence between the temperature-scaled outputs of the teacher and student models. For compactness, let the superscript D denote the DINOv2 branch, and define $p_T^r = \text{softmax}(z_T^D/\tau)$ and $p_S^r = \text{softmax}(z_S^D/\tau)$. The KL term is:

$$\mathcal{L}_{\text{KL}}^D = D_{\text{KL}}(p_T^r \parallel p_S^r), \quad (8)$$

where z_T^D and z_S^D are the logits produced by the DINOv2 teacher and student, and τ is the temperature parameter. In the main experiments, we set $\tau = 5$.

The overall distillation objective and final training objective are:

$$\mathcal{L}_{\text{KD}}^D = \alpha \mathcal{L}_{\text{CE}}^D + (1 - \alpha) \tau^2 \mathcal{L}_{\text{KL}}^D, \quad (9)$$

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{cross-modal}}^{\text{CLIP}} + \mathcal{L}_{\text{KD}}^D, \quad (10)$$

where $\mathcal{L}_{\text{CE}}^D$ denotes the cross-entropy loss between the student predictions and ground-truth labels, and $\alpha \in [0, 1]$ balances hard-label supervision and soft-label distillation. $\mathcal{L}_{\text{final}}$ is used to optimize the two branches during training, whereas Eq. (3) is used at inference time to combine the resulting CLIP and DINOv2 logits. We do not impose an additional loss on the fused logits, because our goal is to preserve branch-specific learning signals and perform a lightweight test-time ensemble after both branches have been optimized.

The complete training procedure is summarized in Algorithm 1. The pipeline uses fixed class texts (no text generation) together with CLIP and DINOv2 features, and optimizes the final objective in Eq. (10).

Algorithm 1 AME Training with Cross-Modal Adaptation and DINOv2 Distillation

- 1: **Input:** Few-shot support set $D = \{(x_i, y_i)\}$, class-name texts $\mathcal{T} = \{t_c\}_{c=1}^C$, frozen CLIP image/text encoders, frozen DINOv2 encoder.
 - 2: Extract and cache CLIP image features, CLIP text features, and DINOv2 visual features.
 - 3: Initialize the CLIP branch head, a wider DINOv2 teacher head, and a compact DINOv2 student head with class-wise mean features.
 - 4: Set distillation temperature τ and soft-label distillation weight α .
 - 5: **Stage 1: DINOv2 teacher training.**
 - 6: **for** each teacher-training iteration **do**
 - 7: Sample DINOv2 support features and labels.
 - 8: Train the wider DINOv2 teacher with cross-entropy loss on ground-truth labels.
 - 9: **end for**
 - 10: Freeze the DINOv2 teacher and compute teacher logits z_T^D .
 - 11: **Stage 2: Joint CLIP-branch and DINOv2-student training.**
 - 12: **for** each joint-training iteration **do**
 - 13: Sample CLIP image features, matched text features, DINOv2 features, and labels.
 - 14: Compute $\mathcal{L}_{\text{cross-modal}}^{\text{CLIP}}$ using CLIP image/text features with Eq. (5).
 - 15: Compute student logits z_S^D and the cross-entropy loss $\mathcal{L}_{\text{CE}}^D$; compute $\mathcal{L}_{\text{KL}}^D$ using Eq. (8), and then compute $\mathcal{L}_{\text{KD}}^D$ using Eq. (9).
 - 16: Optimize $\mathcal{L}_{\text{final}}$ in Eq. (10) to update the CLIP branch and DINOv2 student.
 - 17: **end for**
 - 18: **Output:** Trained AME model; discard the teacher and fuse CLIP and DINOv2 student logits using Eq. (3) at inference.
-

4. Experiments and discussions

4.1. Implementation details

To validate the proposed method, we evaluate its performance on a suite of 11 standard classification datasets, including Food-101 [17], DTD [18], ImageNet-1 K [19], Caltech-101 [20], EuroSAT [21], Stanford Cars [22], FGVC-Aircraft [23], Oxford Flowers-102 [24], Oxford Pets [25], UCF-101 [26], and SUN397 [27]. Unless otherwise specified, all main experiments use CLIP RN50 as the default vision-language backbone, while DINOv2-Large serves as the auxiliary visual encoder. Only the dedicated backbone-comparison experiment in Table 2 additionally reports results with RN101, ViT-B/32, and ViT-B/16.

In terms of architecture, both CLIP and DINOv2 encoders are frozen during training, and we optimize only the residual adapter layers built on top of the extracted features. To enhance expressive capacity, the teacher network adopts a wider MLP bottleneck, while the student network uses a more compact bottleneck selected according to validation

performance. This setting is also consistent with recent discrepancy-based transfer strategies for few-shot image classification [47]. Based on the sensitivity study in Table 5, we use a hidden dimension of 4D by default for the adapter bottleneck in the main experiments. For each dataset, we evaluate the model under the [1, 2, 4, 8, 16]-shot settings per class and report results on the full test sets.

During training, we adopt a knowledge distillation strategy in which a wider teacher network is first introduced to provide soft targets, and a more compact student network is then optimized for final deployment. The training loss is a weighted combination of cross-entropy loss and KL divergence loss, with the temperature set to $\tau = 5$ in the main experiments and the soft-label distillation weight fixed to $\alpha = 0.6$ according to the sensitivity analysis in Fig. 6. For the inference-time logit fusion in Eq. (3), we set $\lambda = 0.6$ when the number of shots is fewer than 4 and $\lambda = 0.4$ otherwise, following the trend observed in Fig. 5. This design combines soft and hard supervision signals while reducing the number of adapter parameters required at inference time.

Since both CLIP and DINOv2 encoders are frozen, the additional trainable complexity of AME comes mainly from the residual MLP heads. Under the default RN50/DINOv2-Large setting on ImageNet-1 K, the CLIP head has 17.81M parameters and the compact DINOv2 student has 9.42M, totaling 27.23M inference-time head parameters; the 17.81M-parameter DINOv2 teacher is training-only. Thus, AME adds one frozen DINOv2 forward pass relative to CLIP-only methods, while cached-feature training optimizes only the heads.

For all experiments, we adopt the PyTorch framework and train for 100 epochs using the Adam optimizer, with an initial learning rate of 1e-4 and a batch size of 1024 for cached feature training. All parameters of the pre-trained models remain fixed during training to preserve their prior knowledge. In comparison to strong baselines such as CoOp [7], CLIP-Adapter [9], and Tip-Adapter-F [12], our approach delivers clear gains in the 2-, 4-, 8-, and 16-shot settings while remaining competitive in the 1-shot regime. Furthermore, under 8- and 16-shot conditions, the use of a refined MLP bottleneck results in more discriminative embeddings and consistent performance gains. All experiments are executed on RTX 4090 GPUs to ensure high computational efficiency and experimental reproducibility.

4.2. Performance

As illustrated in Fig. 4, the proposed AME method demonstrates competitive classification performance across the majority of downstream tasks when compared with existing approaches. Fig. 4 shows that AME performs favorably against AMU-Tuning [11] and other strong baselines across the 10 downstream tasks shown, highlighting its generalization ability in diverse few-shot scenarios. However, it is worth noting that on some datasets, such as EuroSAT, SUN397, and Caltech-101 under the 1-shot setting, AME does not surpass CaFo [10]. A likely reason is that CaFo leverages generated or retrieved similar samples to augment the training set, which directly increases visual diversity in the extremely low-shot regime.

As shown in Table 1, under the extremely low-shot setting, especially 1-shot, the performance improvements of AME are relatively limited. In this setting, each class has only one labeled image, so the class-wise feature initialization degenerates into single-sample initialization and becomes highly sensitive to sample noise. Although the CLIP text feature provides useful semantic information, it cannot fully compensate for the lack of intra-class visual diversity. Similarly, the DINOv2 student branch has very limited visual variation from which to learn complementary cues. Therefore, it is intrinsically difficult to obtain large gains in the 1-shot regime for adapter-based methods with additional trainable heads. As more support samples become available, the adapters can form more stable class representations, cross-modal alignment receives more reliable supervision, and the auxiliary branch can better exploit fine-grained visual variations. This explains why AME benefits more clearly in the 4-, 8-, and 16-shot settings. Such

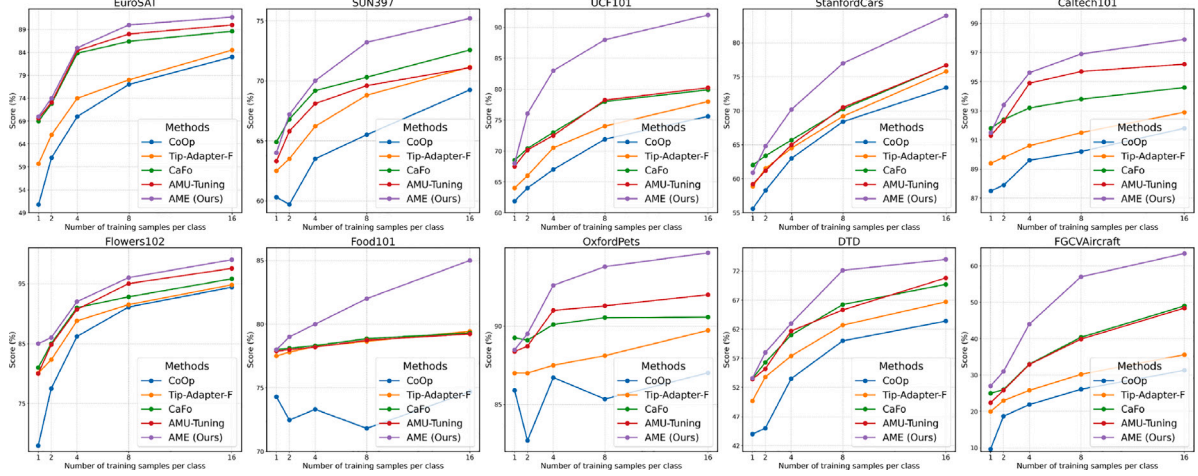


Fig. 4. Accuracy (%) on 10 benchmark datasets. Our approach achieves competitive performance against prior methods under various few-shot scenarios, demonstrating robust and transferable performance.

Table 1

ImageNet-1 K performance (%) and trainable head/adaptor parameters (M), excluding frozen encoders.

Methods	Params (M)	Acc (%)				
		1-shot	2-shot	4-shot	8-shot	16-shot
CoOp [7]	8.20	57.15	57.81	59.99	61.56	62.95
CLIP-Adapter [9]	18.72	61.20	61.52	61.84	62.68	63.59
Tip-Adapter-F [12]	16.38	61.32	61.69	62.52	64.00	65.51
CaFo [10]	49.15	63.80	64.34	65.64	66.86	68.79
AMU-Tuning [11]	2.05	62.60	64.25	65.92	68.25	70.02
AME (Ours)	27.23	62.85	68.18	72.13	76.11	80.30

challenges can become more pronounced when the label distribution is additionally imbalanced, as discussed in prior long-tailed few-shot studies [48].

Table 1 also compares AME on ImageNet-1 K against several strong CLIP-based methods for few-shot classification, including CoOp [7], CLIP-Adapter [9], Tip-Adapter-F [12], CaFo [10], and AMU-Tuning [11]. For transparency, we report inference-time trainable head/adaptor parameters, excluding frozen CLIP and auxiliary encoders. The baseline results in Tables 1 and 2 are quoted from the corresponding papers or their officially reported results. For consistency, the comparisons follow the same ImageNet-1 K few-shot evaluation setting. The support set contains the specified number of labeled images per class, and accuracy is evaluated on the ImageNet-1 K validation set. Unless otherwise specified, Table 1 uses CLIP RN50 as the default visual backbone. In Table 2, the shot number is fixed to 16, and the CLIP visual backbone is varied consistently across methods. For AME, DINOv2-Large is kept as the fixed auxiliary visual encoder, while the CLIP branch follows the backbone specified in each column. The results show that AME achieves substantial improvements in accuracy under the 4-shot, 8-shot, and 16-shot conditions. In these settings, AME effectively leverages the availability of additional training samples to learn more comprehensive feature representations, resulting in significant performance gains. Specifically, compared with AMU-Tuning, AME yields absolute accuracy improvements of 6.21%, 7.86%, and 10.28% in the 4-shot, 8-shot, and 16-shot settings, respectively. These results further demonstrate the performance improvement of AME in few-shot classification.

4.3. Results on different vision backbones

We further benchmark AME against strong few-shot classification methods across multiple CLIP visual backbones, including RN50,

Table 2

Performance (%) of compared methods with various CLIP visual backbones on ImageNet-1 K under the 16-shot setting.

Methods	Backbone			
	RN50	RN101	ViT-B/32	ViT-B/16
CoOp [7]	62.95	65.53	63.80	68.73
CLIP-Adapter [9]	63.59	65.39	66.19	71.13
Tip-Adapter-F [12]	65.51	68.56	68.65	73.69
CaFo [10]	68.79	70.82	70.82	74.48
AMU-Tuning [11]	70.02	71.58	71.65	74.98
AME (Ours)	80.30	81.56	81.54	81.92

RN101, ViT-B/32, and ViT-B/16, as shown in Table 2. In our method, auxiliary features are derived from DINOv2-Large, providing complementary information to the CLIP encoder during inference. AME achieves the best performance among the compared methods across all encoder variants, highlighting the benefit of integrating self-supervised and vision-language representations within a unified learning paradigm.

4.4. Ablation study

Our AME method comprises three key components: auxiliary features (AUX), knowledge distillation (KD), and cross-modal adaptation (CMA). To evaluate the contribution of each module, we conduct ablation studies on EuroSAT, DTD, and FGVC-Aircraft. The baseline is built on CLIP auxiliary features with a simple linear probe followed by β -fusion, without AUX, KD, or CMA. Table 3 reports the results under different component settings.

From the results, we observe that the auxiliary DINOv2 branch brings clear gains over the baseline on all three datasets, especially in the 4- and 16-shot settings. KD provides additional improvements in most higher-shot cases through softened supervisory signals, while CMA improves the use of semantic information by introducing textual features as additional support cues. When these modules are combined, the full AME model achieves the best overall results across EuroSAT, DTD, and FGVC-Aircraft. The improvements are particularly pronounced on DTD and FGVC-Aircraft in the 16-shot setting, indicating that the proposed components become more effective when more support samples are available.

To further validate the high ImageNet-1 K results, we report the mean accuracy and standard deviation over three random seeds in Table 4. The small standard deviations indicate that the reported performance is stable across different seeds. Full AME achieves the best mean accuracy in the 4- and 16-shot settings and remains comparable

Table 3

Ablation results of AME on EuroSAT, DTD, and FGVC-Aircraft with various modules.

Component			EuroSAT			DTD			FGVC-Aircraft		
AUX	KD	CMA	1-shot	4-shot	16-shot	1-shot	4-shot	16-shot	1-shot	4-shot	16-shot
Baseline			65.60	81.40	89.10	33.13	44.78	57.40	23.13	25.89	27.59
✓			67.20	83.20	90.10	48.62	61.42	73.45	25.87	41.95	63.12
✓	✓		67.80	83.80	90.20	48.53	61.55	73.58	25.74	42.19	63.28
✓		✓	69.30	84.50	91.40	52.72	62.71	73.76	26.86	43.82	63.35
✓	✓	✓	70.10	85.10	92.10	53.05	62.85	73.95	27.00	44.00	63.50

Table 4Statistical validation on ImageNet-1 K over three random seeds. Results are reported as mean accuracy (%) $\pm$ standard deviation.

Component			Acc (%)		
AUX	KD	CMA	1-shot	4-shot	16-shot
Baseline			61.01 $\pm$ 0.50	62.33 $\pm$ 0.29	65.34 $\pm$ 0.20
✓			61.14 $\pm$ 0.55	70.48 $\pm$ 0.12	79.66 $\pm$ 0.10
✓	✓		61.19 $\pm$ 0.54	70.52 $\pm$ 0.11	79.61 $\pm$ 0.09
✓		✓	62.55 $\pm$ 0.67	72.10 $\pm$ 0.05	80.18 $\pm$ 0.05
✓	✓	✓	62.85 $\pm$ 0.61	72.13 $\pm$ 0.08	80.30 $\pm$ 0.06

Table 5

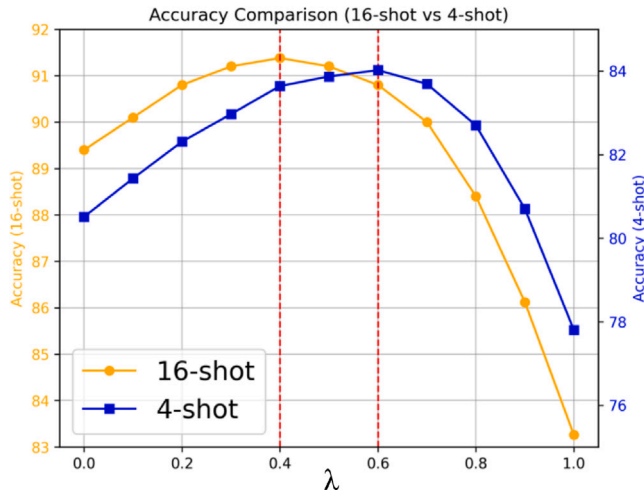
Sensitivity study of different feature dimensions on EuroSAT and ImageNet in the 16-shot setting.

Dimension	$D/8$	$D/4$	D	$4D$	$8D$
EuroSAT [21]	89.71	91.00	91.77	92.10	92.07
ImageNet [19]	81.50	81.53	81.05	81.92	81.30

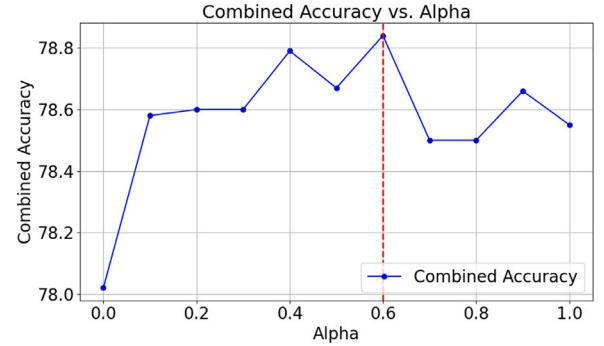
Table 6

Sensitivity study of different distillation temperatures on EuroSAT and ImageNet in the 16-shot setting.

Temperature	1	3	5	7	9
EuroSAT [21]	91.43	91.49	92.10	91.47	91.45
ImageNet [19]	81.20	81.80	81.92	81.77	81.28

**Fig. 5.** Comparison of AME performance with different values of λ , which balances the contribution between cross-modal (CLIP) and visual-only (DINOv2) logits.

to the best variant in the 1-shot setting, which is consistent with the observation that the 1-shot regime is more sensitive to sample noise and limited visual diversity.

**Fig. 6.** Comparison of KD performance with different values of α , where α controls the balance between the cross-entropy loss on ground-truth labels and the KL divergence from the teacher's soft labels.

4.5. Parameter analysis

Unless otherwise stated, the key hyperparameters used in the main experiments are selected according to the following sensitivity analyses. They are then kept fixed across datasets. Specifically, we set the distillation temperature to $\tau = 5$ and the soft-label distillation weight to $\alpha = 0.6$ for all datasets. The logit fusion weight is not tuned separately for each dataset. Instead, we set $\lambda = 0.6$ for settings with fewer than 4 shots and $\lambda = 0.4$ for settings with 4 shots or more, following the trend observed in Fig. 5.

Parameter Analysis of the Feature Dimension Scaling Factor:

To investigate the impact of the scaling factor applied to the feature dimension extracted from the pre-trained model, we conduct experiments on the EuroSAT [21] and ImageNet [19] datasets under the 16-shot setting. The feature dimension is scaled in multiples ranging from one-eighth ($1/8$) of the original dimension to eight times ($8\times$) the original dimension. As shown in Table 5, both EuroSAT and ImageNet achieve the best performance at $4\times$ the original feature dimension, reaching 92.10% and 81.92%, respectively. This indicates that increasing the feature dimension can improve feature expressiveness up to an appropriate scale, while an overly large dimension may introduce redundancy and slightly degrade performance. Therefore, we use $4D$ as the default feature dimension scaling factor in the main experiments.

Parameter Analysis of the KD with Different Temperature τ :

To investigate the effect of the temperature parameter τ in the distillation process, we conduct experiments on the EuroSAT and ImageNet

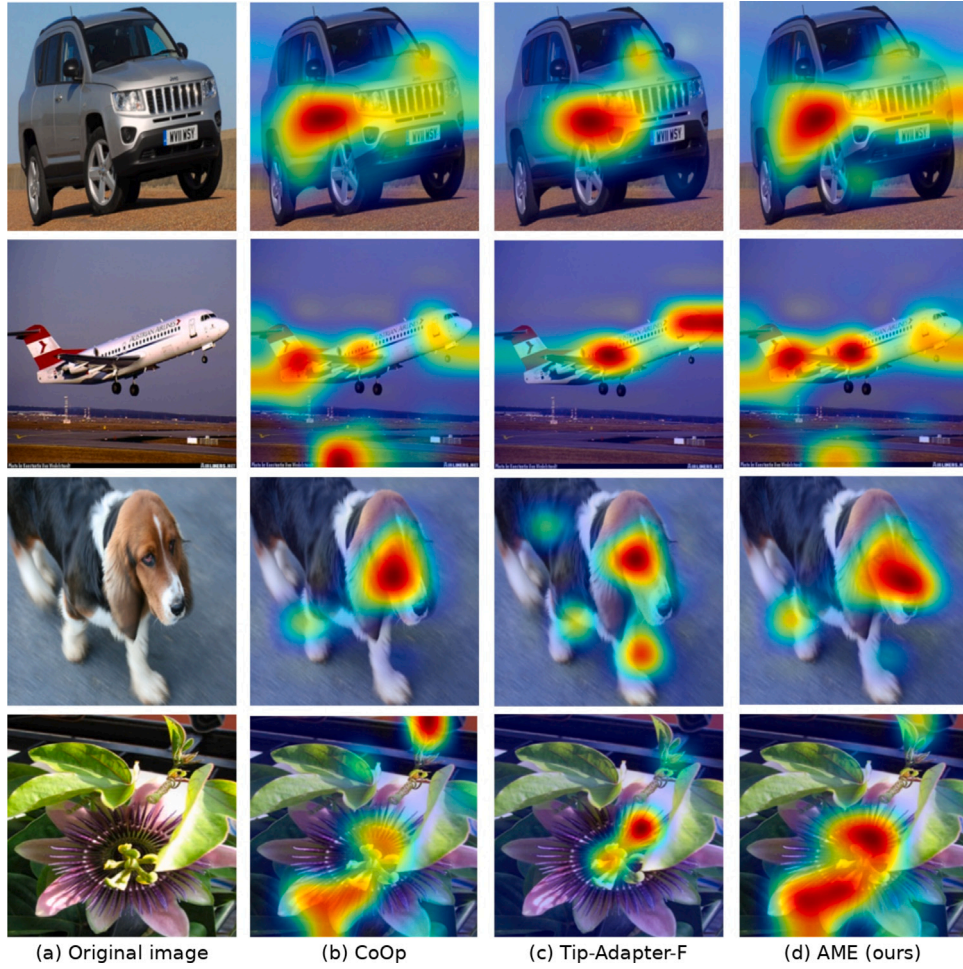


Fig. 7. Grad-CAM visualization of CoOp [7], Tip-Adapter-F [12], and our AME. AME is trained in a 1-shot setting. Examples are from Stanford Cars [22], FGVC-Aircraft [23], Oxford Pets [25], and Flowers-102 [24].

datasets across a range of temperature values. As reported in Table 6, setting $\tau = 5$ yields the most favorable performance, indicating an effective compromise between maintaining informative soft predictions and avoiding over-smoothing. When τ is too low, the softened outputs approximate hard labels, diminishing the value of distillation. In contrast, overly large τ values lead to excessively uniform distributions, diluting the teacher's guidance. Therefore, we adopt $\tau = 5$ in the main experiments to enable stable and effective knowledge transfer.

Parameter Analysis of the Logit Fusion Weight λ : In Eq. (3), we introduce a hyper-parameter λ to balance the contributions of the CLIP and DINOv2 logits during inference. In Fig. 5, we evaluate the performance of the AME method on the EuroSAT [21] dataset under 4-shot and 16-shot settings with various values of λ . The results indicate that the performance of the AME method remains stable when λ is within the range of 0.2 to 0.6, with fluctuations below 0.05%. Further analysis suggests that CLIP is more effective in extremely low-shot scenarios (fewer than 4 shots), while DINOv2 becomes more helpful with larger sample sizes (4 shots or more). Based on these observations, we set $\lambda = 0.6$ in scenarios where the number of shots is fewer than 4, and $\lambda = 0.4$ in those where it is 4 or more, thereby improving adaptation to tasks with different sample sizes.

Parameter Analysis of the Soft Label Weight α : To evaluate the influence of the distillation weight α in Eq. (9), which balances the standard cross-entropy loss $\mathcal{L}_{CE}$ and the KL divergence term $\mathcal{L}_{KL}$, we perform experiments on the EuroSAT [21] and Oxford Pets [25] datasets under both 4-shot and 16-shot settings. As illustrated in Fig. 6, setting α to around 0.6 yields the best performance, suggesting

an effective trade-off between ground-truth supervision and the soft targets provided by the teacher model. Extremely small α values lead to over-reliance on potentially noisy soft labels, while excessively large values diminish the advantages of knowledge distillation. Accordingly, we use $\alpha = 0.6$ in the main experiments.

4.6. Visualization of attention map

Fig. 7 illustrates the Grad-CAM visualizations on the test sets of four benchmark datasets: Stanford Cars [22], FGVC-Aircraft [23], Oxford Pets [25], and Flowers-102 [24]. The first column displays the original input images. The second to fourth columns present the attention maps generated by CoOp [7], Tip-Adapter-F [12], and our AME method under the 1-shot setting, respectively.

The qualitative results suggest that AME can focus more accurately on the target objects than the baseline methods, CoOp and Tip-Adapter-F. As observed in the examples across all four datasets, the attention of CoOp and Tip-Adapter-F sometimes focuses on local regions of the target or disperses into the background. In contrast, AME tends to produce a more complete focus on the target object, thereby extracting more discriminative features. These visualizations suggest that AME helps alleviate feature-localization limitations in the compared baselines. This advantage is also consistent with the performance improvements observed across these four datasets.

In summary, our AME method introduces cross-modal learning and leverages foundation models to enrich the few-shot feature space. By incorporating enhanced adapters and a knowledge distillation strategy,

the classifier effectively utilizes limited data and jointly benefits from the pre-trained text and image encoders. This comprehensive approach improves object recognition and few-shot classification performance in the evaluated settings.

5. Conclusion

To further advance few-shot learning with CLIP, this paper proposes a unified Auxiliary Model Enhancement (AME) method that integrates a visual branch leveraging self-supervised DINOv2 for fine-grained visual embeddings, together with a cross-modal fusion mechanism and soft-label knowledge distillation. Extensive experiments demonstrate that AME achieves competitive performance against strong baselines across 11 datasets under various few-shot settings. Nevertheless, AME still has limitations. Although the DINOv2 teacher is discarded during inference, the training stage requires auxiliary DINOv2 feature extraction and teacher-student distillation, making the pipeline more complex than single-branch CLIP adaptation methods. Moreover, its gains in the extremely low-shot setting, especially 1-shot, remain relatively limited due to the scarcity of support samples. Future work will explore lighter auxiliary-model integration and more robust low-shot adaptation strategies.

CRedit authorship contribution statement

Chengqian Yu: Writing – review & editing, Writing – original draft.
Yihong Chen: Visualization, Writing – review & editing.
Yuwu Lu: Writing – review & editing, Writing – original draft, Supervision.

Declaration of competing interest

We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript

Data availability

Data will be made available on request.

References

- [1] J. Snell, K. Swersky, R.S. Zemel, Prototypical networks for few-shot learning, in: *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4077–4087.
- [2] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al., Flamingo: A visual language model for few-shot learning, in: *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 21338–21353.
- [3] X. Chen, X. Wang, S. Changpinyo, A.J. Piergiovanni, P. Padlewski, D. Salz, et al., PaLI: A jointly-scaled multilingual language-image model, in: *Proc. Int. Conf. Learn. Represent.*, 2023.
- [4] Z. Jin, M. Hayat, Y. Yang, Y. Guo, Y. Lei, Context-aware alignment and mutual masking for 3d-language pre-training, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 10984–10994.
- [5] T.B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, in: *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1877–1901.
- [6] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, in: *Proc. Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 8748–8763.
- [7] K. Zhou, J. Yang, C.C. Loy, Z. Liu, Learning to prompt for vision-language models, *Int. J. Comput. Vis.* 130 (9) (2022) 2337–2348.
- [8] K. Zhou, J. Yang, Conditional prompt learning for vision-language models, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16795–16804.
- [9] P. Gao, S. Geng, R. Zhang, T. Ma, R. Fang, Y. Zhang, H. Li, Y. Qiao, CLIP-Adapter: Better vision-language models with feature adapters, *Int. J. Comput. Vis.* 132 (2) (2024) 581–595.
- [10] R. Zhang, X. Hu, B. Li, S. Huang, H. Deng, H. Li, Y. Qiao, P. Gao, Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 15211–15222.
- [11] Y. Tang, Z. Lin, Q. Wang, P. Zhu, Q. Hu, AMU-Tuning: Effective Logit Bias for CLIP-based Few-shot Learning, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 23323–23333.
- [12] R. Zhang, Z. Wei, R. Fang, P. Gao, K. Li, J. Dai, Y. Qiao, H. Li, Tip-Adapter: Training-free adaption of CLIP for few-shot classification, in: *Proc. Eur. Conf. Comput. Vis.*, vol. 13695, 2022, pp. 493–510.
- [13] S. Tong, Z. Liu, Y. Zhai, Y. Ma, Y. LeCun, S. Xie, Eyes wide shut? Exploring the visual shortcomings of multimodal LLMs, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 9568–9578.
- [14] J. Sun, J. Li, Few-shot classification with Fork Attention Adapter, *Pattern Recognit.* 156 (2024) 110805.
- [15] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, I. Laptev, L. Najman, M. Rabbat, P. Bojanowski, A. Joulin, et al., DINOv2: Learning robust visual features without supervision, *Trans. Mach. Learn. Res. (ISSN: 2835-8856)* (2023).
- [16] Z. Lin, S. Yu, Z. Kuang, D. Pathak, D. Ramanan, Multimodality helps unimodality: Cross-modal few-shot learning with multimodal models, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 27040–27049.
- [17] L. Bossard, M. Guillaumin, L. Van Gool, Food-101—mining discriminative components with random forests, in: *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 446–461.
- [18] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, A. Vedaldi, Describing textures in the wild, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3606–3613.
- [19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
- [20] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshop*, 2004, pp. 178–178.
- [21] P. Helber, B. Bischke, A. Dengel, D. Borth, EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification, *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* 12 (7) (2019) 2217–2226.
- [22] J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3d object representations for fine-grained categorization, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshop*, 2013, pp. 554–561.
- [23] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, A. Vedaldi, Fine-grained visual classification of aircraft, 2013, arXiv preprint arXiv:1306.5151.
- [24] M.-E. Nilsback, A. Zisserman, Automated flower classification over a large number of classes, in: *Proc. Indian Conf. Comput. Vis., Graphics Image Process.*, 2008, pp. 722–729.
- [25] O.M. Parkhi, A. Vedaldi, A. Zisserman, C.V. Jawahar, Cats and dogs, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3498–3505.
- [26] K. Soomro, A.R. Zamir, M. Shah, UCF101: A dataset of 101 human actions classes from videos in the wild, 2012, arXiv preprint arXiv:1212.0402.
- [27] J. Xiao, J. Hays, K.A. Ehinger, A. Oliva, A. Torralba, SUN database: Large-scale scene recognition from abbey to zoo, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3485–3492.
- [28] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [29] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, in: *Proc. Int. Conf. Learn. Represent.*, 2021.
- [30] Y. Chen, Z. Liu, H. Xu, T. Darrell, X. Wang, Meta-baseline: Exploring simple meta-learning for few-shot learning, in: *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9062–9071.
- [31] X. Li, Z. Guo, R. Zhu, Z. Ma, J.-H. Xue, A simple scheme to amplify inter-class discrepancy for improving few-shot fine-grained image classification, *Pattern Recognit.* 155 (2024) 110674.
- [32] M. Fahes, T.-H. Vu, A. Bursuc, P. Perez, R. De Charette, CLIP’s visual embedding projector is a few-shot cornucopia, in: *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2026, pp. 3254–3264.
- [33] Y. Jo, J. Kim, C. Park, S. Choi, J.-W. Lee, J. Park, AT-Adapter: Leveraging attribute knowledge of CLIP for few-shot classification, *Neurocomputing* 671 (2026) 132627.
- [34] X. Yang, W. Xie, P. Peng, J. Wen, X. Lu, Cross-modal mapping: Mitigating the modality gap for few-shot classification, *Pattern Recognit.* (2026).
- [35] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
- [36] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, *IEEE Trans. Pattern Anal. Mach. Intell.* 40 (4) (2017) 834–848.

- [37] R. Zhang, L. Wang, Y. Qiao, P. Gao, H. Li, Learning 3D representations from 2D pre-trained models via image-to-point masked autoencoders, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 21769–21780.
- [38] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
- [39] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9650–9660.
- [40] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, R. Girshick, Masked autoencoders are scalable vision learners, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16000–16009.
- [41] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, I. Sutskever, Zero-shot text-to-image generation, in: *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8821–8831.
- [42] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012–10022.
- [43] N. Mu, A. Kirillov, D. Wagner, S. Xie, SLIP: Self-supervision meets language-image pre-training, in: *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 529–544.
- [44] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, Y. Wu, CoCa: Contrastive captioners are image-text foundation models, in: *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 29834–29848.
- [45] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9729–9738.
- [46] Z. Pan, D. Wang, Y. Chen, H. Yan, X. Hu, J. Wang, Pseudo-set frequency refinement architecture for fine-grained few-shot class-incremental learning, *Pattern Recognit.* 155 (2024) 110686.
- [47] J. Wu, S. Wang, J. Sun, Attentive maximum mean discrepancy for few-shot image classification, *Pattern Recognit.* 155 (2024) 110680.
- [48] H. Zhang, S. Chen, L. Luo, J. Yang, Few-shot learning with long-tailed labels, *Pattern Recognit.* 156 (2024) 110806.