DAT: Dual-Branch Adapter-Tuning for Few-Shot Recognition

Junxi Chen, Guangxing Wu, Hongxiang Li, Jiankang Chen, Wentao Zhang, Wei-Shi Zheng, and Ruixuan Wang

***Abstract*— Parameter-Efficient Fine-Tuning methods based on vision-language models (such as CLIP) for few-shot learning have recently received considerable attention. However, previous works only fine-tune either the image or text branch, breaking the alignment of the original two branches, meanwhile fine-tuning both branches of the CLIP would inevitably introduce more trainable parameters and likely cause more severe over-fitting due to the limited training data. In this study, we propose a novel Dual-branch Adapter-Tuning framework (DAT), which collaboratively trains the visual adapter and textual adapter added to the two branches of the original CLIP with multiple consistency constraints. By effectively utilizing the semantically detailed class-specific prompts and outputs of the original CLIP to guide the fine-tuning of both branches, our method gains exceptional adaptation ability to the downstream few-shot learning tasks and alleviates the over-fitting issue, meanwhile maximally preserving the generalization ability of the original CLIP model. Our proposed framework has achieved superior performance on diverse datasets under various few-shot learning settings compared to the existing approaches. The source code is available at https://github.com/SandyXi/DAT.**

***Index Terms*— Vision-language model, parameter-efficient fine-tuning, few-shot learning.**

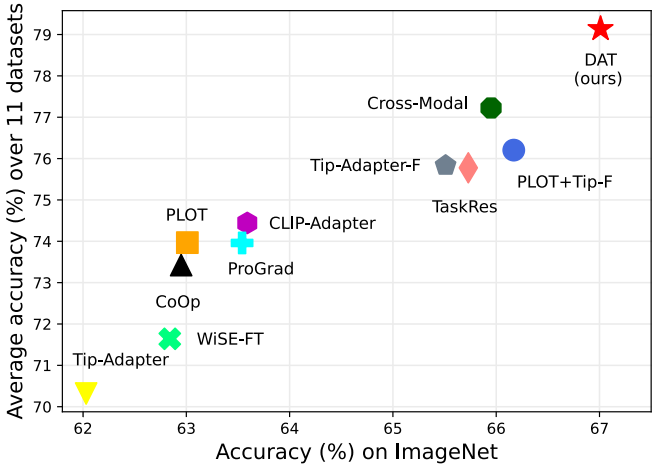

Fig. 1. Performance comparison between different methods (in different colors) under the 16-shot setting. X-axis: accuracy on the ImageNet dataset; Y-axis: averaged performance over 11 datasets.

I. Introduction

FEW-SHOT learning [1], [2], [3], [4], [5], [6], [7], [8], [9] aims to enable the model to well perform a new task after training the model with limited task-specific samples. For image classification tasks, traditional approaches to few-shot learning rely on training the model with training images and class labels, which fails to fully utilize the rich semantic textual information of each class and limits the trained model's effectiveness to a close-set of categories. Recently, large-scale foundational vision-language models (VLMs) [10], such as CLIP (Contrastive Language-Image Pre-training) [11], [12], have shown exceptional performance on zero-shot and few-shot learning tasks. CLIP leverages a collection of 400 million text-image pairs gathered from the internet and trains the model by contrastive learning, which enables the model to encode both text and images into a unified semantic space without retraining, even in an open-vocabulary situation. However, due to the massive scale of the VLMs, it is impractical to fully fine-tune the model in few-shot scenarios.

For that reason, subsequent methods known as Parameter-Efficient Fine-Tuning (PEFT) [13], [14] have emerged with lightweight fine-tuning of VLMs. During the fine-tuning process, the parameters of the pre-trained model are fixed, while only a small number of newly added parameters are tuned for any downstream task. Currently, the PEFT approach based on CLIP for few-shot learning is generally categorized into the prompt-tuning paradigm and the adapter-tuning paradigm. The former paradigm adds learnable tokens called "prompt" to the inputs of the model to enable better alignment between the text and image modalities, such as CoOp [15] and CoCoOp [16], while the latter paradigm adds lightweight trainable network modules called "adapter" somewhere in the

Received 30 April 2024; revised 6 September 2024 and 23 November 2024; accepted 2 February 2025. Date of publication 13 February 2025; date of current version 4 July 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62071502, in part the Major Key Project of Peng Cheng Laboratory (PCL) under Grant PCL2023A09, and in part Guangdong Excellent Youth Team Program under Grant 2023B1515040025. This article was recommended by Associate Editor S. Li. *(Corresponding author: Ruixuan Wang.)*

Junxi Chen, Guangxing Wu, Jiankang Chen, Wentao Zhang, and Wei-Shi Zheng are with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China, and also with the Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510275, China.

Hongxiang Li is with the School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China.

Ruixuan Wang is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510275, China, also with the Peng Cheng Laboratory, Shenzhen 518066, China, and also with the Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou 510275, China (e-mail: wangruix5@mail.sysu.edu.cn).

Digital Object Identifier 10.1109/TCSVT.2025.3541960

pre-trained CLIP, achieving better adaptation to the downstream tasks, such as CLIP-Adapter [6] and Tip-Adapter [17].

However, these methods are faced with a dilemma. On the one hand, to increase the adaptation ability of CLIP to the downstream tasks, it is expected to fine-tune both the text branch and the image branch of CLIP, since fine-tuning only one of the two modalities would break the original alignment of the two branches [18]. On the other hand, fine-tuning both modalities would inevitably introduce more trainable parameters or modules, which in turn would likely lead to more severe over-fitting of the fine-tuned model to the limited training data. For example, researchers attempted adding both textual and visual adapters into CLIP but observed a performance drop compared to the single visual adapter fine-tuning [6]. Similarly, adding trainable parameters to both text and image features also caused decreased performance compared to adding trainable parameters to the text features only [19].

To address this dilemma, we propose a novel **D**ual-branch **A**dapter-**T**uning (DAT) framework with the help of multiple consistency constraints. This framework contains an image-modality branch and a text-modality branch, where the trainable visual adapter and the textual adapter are added to the visual branch and the textual branch, respectively. Compared to relevant studies [6], our approach alleviates the over-fitting issue and significantly improves the model's performance in the case of dual-branch fine-tuning condition, thanks to the whole well-designed framework. When learning to recognize a set of visual classes with limited training images, more descriptive prior knowledge generated from GPT-3 for each visual class is encoded in the text-modality branch to more effectively guide the visual learning. To help remain well aligned between the two modalities and alleviate the over-fitting issue during dual-branch training process, two types of consistency constraints are proposed for training of the textual adapter and the visual adapter. Specifically, the textual adapter and the visual adapter are trained collaboratively with feature consistency constraint enforcing that the visual feature representations is well aligned with the textual feature representations, such that the rich semantic knowledge of each class can be effectively absorbed into the trained model. To alleviate the over-fitting issue, the original relationship between the two outputs from the pre-trained CLIP image encoder and the text encoder for each training image is utilized as logit consistency constraint to guide the training of both adapters, such that the visual feature representations and the textual feature representations after adapter tuning are largely similar to the two embedding features from the original CLIP. In this way, the desired generalization ability of the original CLIP would be largely preserved after adapter tuning. Extensive empirical evaluations on 11 diverse image classification datasets and under various few-shot learning settings (see Figure 1 for an example) confirm the effectiveness and generalizability of our proposed DAT framework. The contributions of this study are summarized below.

- A novel dual-branch adapter-tuning framework for few-shot learning. To the best of our knowledge, this is the first learning framework via which visual and textual adapters together can be added to CLIP to effectively improve few-shot learning performance.
- Multiple consistency constraints to help train the model with the guidance of class-specific prior knowledge and the original CLIP. In particular, more descriptive knowledge of each visual class and the desired generalization ability of the original CLIP can be well absorbed into the trained model.
- Extensive empirical evaluations on multiple datasets confirm the effectiveness of our proposed framework, with state-of-the-art few-shot learning performance achieved.

II. Related Work

A. Vision-Language Models

Vision-Language Models (VLMs), such as CLIP [11], ALIGN [20] and FILIP [21], are pre-trained using large-scale datasets of image-text pairs from the internet. Typically, VLMs employ contrastive learning to bring matched image-text pairs closer into a unified embedding space and push unmatched pairs further apart, containing rich multi-modal representations. While VLMs have achieved impressive performance, fine-tuning VLMs in few-shot scenarios still remains a challenge to strike a balance between downstream tasks and prior knowledge. Our approach tackles this challenge by employing multiple consistency constraints during the model training process, which helps reduce over-fitting to limited training data in few-shot learning while fully leveraging rich prior knowledge encoded by CLIP.

B. Few-Shot Learning Based on CLIP

Few-shot learning involves training the model on a small set of samples per class and aiming for better generalization to unseen samples. Methods such as meta-learning [22], [23], [24], [25], [26], [27], data augmentation [28], [29] and metric learning [30], [31] have been widely applied in few-shot learning. For example, LCCRN [27] introduces a local content-enriched module to learn the discriminative local features and a cross-reconstruction module to fully engage the local features with the appearance details obtained from a separate embedding module, with both modules working together to better classify fine-grained images. TADRNet [26] proposes a task-aware dualrepresentation network for few-shot action recognition, which learns how to adapt video representations to novel tasks in a meta-learning manner. DSD [25] introduces a regularized dense-sparse-dense fine-tuning flow for regularizing the capacity of pre-trained networks and achieving efficient few-shot domain adaptation. These works use a meta-learning approach to train models and only test the performance of the models in a specific and single scenario. Recently, the emergence of CLIP has opened up new possibilities for few-shot learning. For example, Cross-Modal [32] enhances the performance of image modality by mapping different modalities information into the same representation space. CALIP [5] utilizes a parameter-free attention mechanism to guide the interaction between the image and text modalities. CLAP [33] introduces a class-adaptive linear probe objective, whose balancing term is optimized via an adaptation

of the general Augmented Lagrangian method tailored to this context. CLIP4STR [34] transforms CLIP into a scene text reader, which is a simple yet effective STR method built upon image and text encoders of CLIP. However, due to the large scale of CLIP and the reality that full fine-tuning of CLIP with small-scale data is impractical, the Parameter-Efficient Fine-Tuning (PEFT) approach [13], which was originally applied in NLP, has been gradually applied to the image classification. The prevailing PEFT methods based on CLIP for few-shot learning tasks can be categorized into two groups, i.e., prompt-tuning based and adapter-tuning based.

1) Prompt-tuning on CLIP: CoOp [15] and CoCoOp [16] optimize continuous learnable prompts in the text branch, enabling better adaptation of the text encoder to downstream tasks. TaskRes [19] introduces learnable parameters to the text features while keeping the pre-trained CLIP parameters frozen, enabling more flexible task-specific knowledge exploration. Descriptor and Word Soups [35] greedily selects a small set of textual descriptors and assembles a chain of words using generic few-shot training data, then calculates robust class embeddings using the selected descriptors to increase out-of-distribution target accuracy.

2) Adapter-tuning on CLIP: CLIP-Adapter [6] and SgVA-CLIP [36] add an adapter in the image branch, while Tip-Adapter [17] treats images and labels as key-value pairs stored in a cache model and initializes the key parameters as an adapter. However, these adapter-tuning methods only fine-tune the image branch. In contrast, our approach involves adding adapters on both the text and image sides and fine-tuning with the help of multiple consistency constraints, ensuring complete adaptation of the multi-modality model as a whole and providing greater flexibility in aligning visual and language representations.

III. Method

In this section, we first revisit CLIP and overview the framework of our method, and then introduce the details of each modality and the inference process.

A. Preliminary

CLIP consists of an image encoder and a text encoder, which are used to encode images and text into the same embedding space respectively, and has shown promising performance for zero-shot and few-shot classification tasks. For example, for the zero-shot classification task which involves totally C classes, any test image $\mathbf{x}$ can be passed through the image encoder to obtain the image feature vector representation $\mathbf{f} \in \mathbb{R}^{D\times 1}$, where D is the dimensionality of the feature vector space. Then each of the C class name is respectively filled into the vanilla category prompt templates, such as "A photo of a [CLASS]", which in turn is fed to the text encoder to obtain the text feature vector representation $\mathbf{t}_c, c \in \{1, 2, \ldots, C\}$. The degree of the image $\mathbf{x}$ belonging to the c-th class can be measured by the cosine similarity $s(\mathbf{f}, \mathbf{t}_c) = \frac{\mathbf{f}\cdot\mathbf{t}_c}{\|\mathbf{f}\|\cdot\|\mathbf{t}_c\|}$ between the image feature vector $\mathbf{f}$ and the text feature vector $\mathbf{t}_c$ of the c-th class. Then, the probability of $\mathbf{x}$ belonging to the c-th class can be estimated by

$$p(c|\mathbf{x}) = \frac{\exp(s(\mathbf{f}, \mathbf{t}_c)/\tau)}{\sum_{j=1}^{C} \exp(s(\mathbf{f}, \mathbf{t}_j)/\tau)}, \tag{1}$$

where τ represents the learned temperature of CLIP.

B. The Few-Shot Learning Framework

1) Framework Overview: Our few-shot learning framework is illustrated in Figure 2. It consists of a text-modality branch (Figure 2, upper half) and an image-modality branch (Figure 2, lower half). In the image-modality branch, a pre-trained image encoder is adapted with a lightweight learnable visual adapter, leading to the adapted visual encoder ('Adapted Visual Encoder'). The output of the adapted visual encoder is then fed into a multilayer perceptron ('MLP') as the classifier head for category prediction of the input image. Along the text-modality branch, prior knowledge of each visual category is initially represented in the form of textual descriptions of the category name and its various properties, and such descriptive prior knowledge is then encoded by a pre-trained and fixed text encoder ('CLIP Text Encoder'). Considering that the alignment between the original pre-trained text encoder and image encoder has been impaired due to the adapted visual encoder, a learnable textual adapter ('T.A.') is proposed to be attached on top of the fixed text encoder, such that the encoded prior knowledge is transformed to be more easily aligned with the visual representation of images from the same category. The output of the textual adapter for each visual category can be considered as the learnable textual prototype for that category.

To the best of our knowledge, this is the first few-shot learning framework in which both the text branch and the visual branch contain learnable adapter modules. Since more model parameters need to be learned in our dual-branch adapter-tuning framework compared to existing frameworks where only one (either text or image) branch contains learnable parameters [6], [17], [19], it becomes more challenging to alleviate the over-fitting issue during optimizing these dual-branch modules. In this study, a synergistic training strategy is proposed to collaboratively optimize the textual adapter in the text-modality branch and the visual adapter together with the classifier head in the image-modality branch. In particular, to fully utilize the generalizability of the pre-trained text and image encoders, the similarity between the outputs of the original text and image encoders is used to guide the training of the learnable modules in both branches, such that severe over-fitting of the learnable modules to the limited training images can be largely prevented.

2) Text-Modality Branch: The text-modality branch is used to encode the prior knowledge of each visual category and to help guide the training of the image-modality branch. While most previous studies simply feed the vanilla text prompts for each visual class (e.g., "A photo of a [CLASS]") to the pre-trained and fixed text encoder to obtain the prior knowledge representation, such prior knowledge is solely based on the class name and therefore often lacks detailed class-specific information (e.g., various properties and characteristics of the

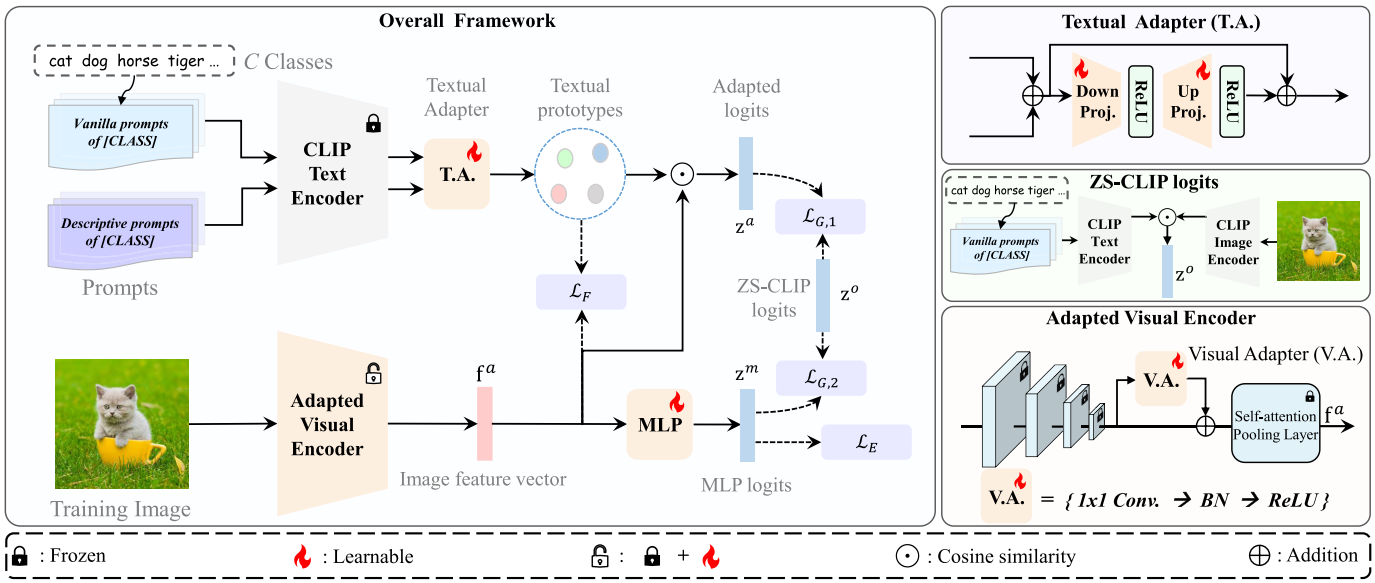

Fig. 2. An overview of our framework for few-shot image classification. In the text-modality branch (Left, upper half), the vanilla prompts and a set of descriptive prompts corresponding to each class are used to obtain the textual prototypes through the CLIP's frozen text encoder and the trainable textual adapter. In the image-modality branch (Left, lower half), an image is fed to the adapted visual encoder and the multilayer perceptron (MLP) to obtain the image feature vector $\mathbf{f}^a$ and the logit vector $\mathbf{z}^m$ respectively. The textual adapter, the visual adapter within the adapted visual encoder, and the MLP are jointly trained by the feature consistency loss $\mathcal{L}_F$, the logit consistency losses $\mathcal{L}_{G,1}$ and $\mathcal{L}_{G,2}$, and the cross-entropy loss $\mathcal{L}_E$. The architectures of the textual adapter and the adapted visual encoder are illustrated on the upper right and the lower right respectively, while the process of obtaining zero-shot CLIP logit vector $\mathbf{z}^o$ is drawn in the right center.

visual class). In order to encode more prior knowledge for each class, multiple descriptive prompts with more detailed class information are generated based on the recently proposed strategy in CuPL [37], where each descriptive prompt corresponds to the output from the GPT-3 [38] by enquiring it in different ways (e.g., "Describe what a(n) [CLASS] looks like" or "How can you identify a(n) [CLASS]?"). Here 50 generated descriptive prompts per class (Table I) are respectively fed to the pre-trained text encoder, and for simplicity, the 50 output vectors from the text encoder are averaged to represent the detailed prior knowledge for the class. In contrast, the encoding of the vanilla prompts from the fixed text encoder can be considered to represent rough prior knowledge of the class. It is important to note that the structure of CLIP text encoder is the same as that of BERT [39], which means that both the detailed and the rough representations are extracted from the [cls] token of the pre-trained CLIP text encoder.

Both the detailed and the rough representations of the prior knowledge for each class are fed to the learnable textual adapter module ('T.A.'; Figure 2, top right). The two prior knowledge representations are fused simply with the addition operator, considering that concatenation or additional learnable fusion layer would cause more learnable parameters. The simply fused prior knowledge representation is then transformed by the trainable textual adapter. The textual adapter consists of a trainable down-projection layer and a trainable up-projection layer, each of which is followed by a rectified linear unit (ReLU) activation [40]. To be consistent with the dimension D of the visual encoder output, the output dimension of the up-projection layer is set to D, while the output dimension of the down-projection layer is simply set to $D/2$. A skip connection as in ResNet [41] is included in the trainable textual adapter, such that the two projection layers are trained to learn just the residual between the fused prior knowledge representation and image encodings of the same class, such residual learning is expected to reduce the risk of over-fitting of the textual adapter to the limited training images in the few-shot scenario, as confirmed in the relevant ablation study (See Addition ablation study).

The output of the textual adapter for each class can be considered as the learnable textual prototype of the class. Let $\boldsymbol{\mu}_c$ denote the learnable textual prototype of the c-th class. For each input image $\mathbf{x}$ from the image-modality branch, the output $\mathbf{f}^a$ of the adapted visual encoder can be compared with each textual prototype in the form of cosine similarity $\cos(\mathbf{f}^a, \boldsymbol{\mu}_c) = \frac{\mathbf{f}^a \cdot \boldsymbol{\mu}_c}{\|\mathbf{f}^a\| \cdot \|\boldsymbol{\mu}_c\|}$. Such cosine similarities over all the C classes are collected to form a vector $\mathbf{z}^a \in \mathbb{R}^{C\times 1}$ which will be used to help train both the textual adapter and the visual adapter.

3) Image-Modality Branch: The image-modality branch is used not only to extract visual features from each input image and predict the class of the input image, but also to help guide the training of the textual adapter in the text-modality branch. The pre-trained image encoder by default is from CLIP. When the backbone of the pre-trained CLIP image encoder is CNN followed by a self-attention pooling layer (Figure 2, lower right, blue), the image encoder is adapted by including a parallel visual adapter on top of the last convolutional layer of the pre-trained image encoder ('V.A.'; Figure 2, lower right, orange). The pre-trained image encoder is frozen and only the visual adapter is trainable. Specifically, the input of the visual adapter is the feature maps of the last convolutional layer rather than the feature vector output of the final self-attention pooling layer, considering that more visual information exists in the feature maps than in the pooled feature vector, and

TABLE I
DEMONSTRATIVE TEXTUAL DESCRIPTIONS OF DIFFERENT CLASSES. THE TEXTUAL DESCRIPTION OF EACH CLASS IS OBTAINED BY ASKING GPT-3 IN DIFFERENT WAYS

class name	Textual description from GPT-3
goldfish	A goldfish has a long, gold body with back fins.
abbey	The abbey is a large, old building made of stone.
tiger shark	Tiger sharks are large, predatory sharks.
gnocchi	Gnocchi are small, doughy dumplings.

therefore potentially discriminative features specifically for the few-shot learning task can be more likely preserved through the self-attention pooling layer with the help of the visual adapter. Here, the visual adapter is simply designed as a convolutional layer with kernel size 1×1 to preserve the spatial size of the feature maps. Batch normalization (BN) [42] and ReLU activation are adopted on top of the convolutional layer as usual. The output of the visual adapter, containing the same number (i.e., D) of feature channels as that of the input to the visual adapter, is combined (by addition) with the feature maps of the last convolutional layer from the pre-trained image encoder, which is then spatially pooled by the pre-trained and fixed self-attention pooling layer to generate the feature vector output $\mathbf{f}^a$ of the adapted visual encoder. Note that, compared to using the only output of the visual adapter as the input to the final self-attention pooling layer, combining the two sets of feature maps together ensures that the extracted feature information from the pre-trained CNN is always considered in the self-attention pooling layer, and therefore can help alleviate over-fitting of the overall adapted visual encoder to the limited training images. Also note that the pre-trained image encoder can be the ViT version [43] of the CLIP. With such an image encoder, the trainable visual adapter becomes a fully connected layer followed by BN and ReLU activation, with the output dimension being the same as that of the input. The visual adapter is inserted in front of the final projection head of ViT to fine-tune the class token.

The output of the adapted visual encoder is finally fed to the trainable MLP which consists of two fully connected (FC) layers with the architecture FC→BN→ReLU→FC. The output dimensions from the two FC layers are respectively $2D$ and C, considering that the lifted dimension from the first FC may provide better learning ability. Although more model parameters need to be learned in the MLP, the effectively designed loss function will largely alleviate the over-fitting issue (See relevant description below). The MLP module allows our framework to achieve a large performance improvement compared to the original CLIP on a variety of datasets from different domains, but at the same time our framework becomes a closed-set classification model. The combinatio of few-shot and open-set recognition is an interesting and challenging task and we will consider as part of the future work.

4) Model Training and Inference: Our few-shot learning framework aims to well adapt the pre-trained CLIP encoders to the specific few-shot learning tasks with limited training data, and meanwhile to use the generalizability of the pre-trained encoders to alleviate the over-fitting issue of the final classifier. This goal is achieved with the following four loss terms.

a) Feature consistency loss: As for the training of the CLIP model, both the textual adapter and the visual adapter in our framework are collaboratively optimized such that the visual representation of any input image from one class is well aligned with the learnable textual prototype of the same class. This can be achieved by minimizing the contrastive loss function $\mathcal{L}_F$,

$$\mathcal{L}_F = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C}\mathbb{1}(y_n=c)\log\frac{\exp(s(\mathbf{f}_n^a,\boldsymbol{\mu}_c)/\tau)}{\sum_{j=1}^{C}\exp(s(\mathbf{f}_n^a,\boldsymbol{\mu}_j)/\tau)}, \tag{2}$$

where $\mathbf{f}_n^a$ denotes the feature vector output from the adapted visual encoder for the n-th training image, and $\boldsymbol{\mu}_c$ is the textual prototype of the c-th class. $\mathbb{1}(\cdot)$ is the indicator function and $y_n \in \{1, 2, \ldots, C\}$ denotes the ground-truth class label for the n-th image. N denotes the total number of training images, and τ represents the temperature hyper-parameter. $s(\cdot,\cdot)$ represents the cosine similarity.

b) Logit consistency loss: Both the visual adapter and the textual adapter would be likely over-fitted to the limited training images without further constraint. To alleviate such possible over-fitting, the knowledge in the original CLIP model is utilized to guide the training of the two adapters. Specifically, considering the better generalization ability of the original CLIP model, the relational vision-language knowledge between each image and the corresponding vanilla prompt is captured by the original CLIP, and then such knowledge is distilled to the visual adapter and textual adapter. Formally, let $\mathbf{z}_n^o = [s(\mathbf{f}_n^o,\mathbf{t}_1), s(\mathbf{f}_n^o,\mathbf{t}_2), \ldots, s(\mathbf{f}_n^o,\mathbf{t}_C)]^\intercal$ denote the relational vision-language knowledge for the n-th training image, where $s(\mathbf{f}_n^o,\mathbf{t}_c)$ represents the cosine similarity between the image feature vector output $\mathbf{f}_n^o$ from the pre-trained image encoder of the original CLIP and the text feature vector output $\mathbf{t}_c$ from the pre-trained text encoder of the original CLIP. Actually, $\mathbf{z}_n^o$ can also be considered as the logit vector of the classifier head when CLIP is used for zero-shot classification (see Equation 1). Similarly, let $\mathbf{z}_n^a = [s(\mathbf{f}_n^a,\boldsymbol{\mu}_1), s(\mathbf{f}_n^a,\boldsymbol{\mu}_2), \ldots, s(\mathbf{f}_n^a,\boldsymbol{\mu}_C)]^\intercal$ denote the logit vector based on the outputs from the adapted visual encoder and the textual adapter. Then, knowledge distillation can be achieved by minimizing the logit consistency loss $\mathcal{L}_{G,1}$, i.e.,

$$\mathcal{L}_{G,1} = \frac{1}{N\cdot C}\sum_{n=1}^{N}\sum_{c=1}^{C}|z_{n,c}^a - z_{n,c}^o|, \tag{3}$$

where $z_{n,c}^a$ is the c-th element of $\mathbf{z}_n^a$, and similarly for $z_{n,c}^o$.

With a similar rationale, to alleviate the potential over-fitting of the MLP module in the image-modality branch, the relational vision-language knowledge from the original CLIP can also be used to guide MLP training. Let $\mathbf{z}_n^m$ denote the output logit vector from the MLP for the n-th training image. Then, knowledge distillation can be achieved by minimizing the logit consistency loss $\mathcal{L}_{G,2}$, i.e.,

$$\mathcal{L}_{G,2} = \frac{1}{N\cdot C}\sum_{n=1}^{N}\sum_{c=1}^{C}|z_{n,c}^m - z_{n,c}^o|, \tag{4}$$

where $z_{n,c}^m$ is the c-th element of $\mathbf{z}_n^m$.

TABLE II
SUMMARY OF 11 DATASETS FOR FEW-SHOT LEARNING AND FOUR TARGET DATASETS OF DOMAIN GENERALIZATION

Name	Number of Classes	Size (Train / Val / Test)	Description
ImageNet	1000	1.28M / - /50000	Recognition of generic objects
Caltech101	100	4128 / 1649 / 2465	Recognition of generic objects
OxfordPets	37	2944 / 736 / 3669	Fine-grained classification of pets
StanfordCars	196	6509 / 1635 / 8041	Fine-grained classification of cars
Flowers102	102	4093 / 1633 / 2463	Fine-grained classification of flowers
Food101	101	50500 / 20200 / 30300	Fine-grained classification of foods
FGVCAircraft	100	3334 / 3333 / 3333	Fine-grained classification of aircrafts
SUN397	397	15880 / 3970 / 19850	Scene classification
DTD	47	2820 / 1128 / 1692	Texture classification
EuroSAT	10	13500 / 5400 / 8100	Land use & cover classification with satellite images
UCF101	101	7639 / 1898 / 3783	Action recognition
ImageNet-V2	1000	- / - / 10000	New test data for ImageNet
ImageNet-Sketch	1000	- / - / 50889	Sketch-style images of ImageNet classes
ImageNet-A	200	- / - / 7500	Natural adversarial examples of 200 ImageNet classes
ImageNet-R	200	- / - / 30000	Renditions of 200 ImageNet classes

In summary, the textual adapter, the visual adapter, and the MLP module can be jointly trained by minimizing the combined loss $\mathcal{L}$,

$$\mathcal{L} = \mathcal{L}_F + \lambda_1(\mathcal{L}_{G,1} + \mathcal{L}_{G,2}) + \lambda_2\mathcal{L}_E\,, \tag{5}$$

where $\mathcal{L}_E$ is the conventional cross-entropy loss based on the final classifier output in the image-modality branch. Coefficients λ_1 and λ_2 are used to balance these loss terms.

c) Model inference: Once the model is well trained, it can be used to predict the class of any new image. Since three logit vectors $\mathbf{z}^a, \mathbf{z}^m, \mathbf{z}^o$ are generated for any input image, the three vectors can be assembled together for class prediction as follows,

$$\mathbf{z} = \mathbf{z}^a + \alpha_1\mathbf{z}^m + \alpha_2\mathbf{z}^o\,. \tag{6}$$

where coefficients α_1 and α_2 are used to balance the contributions of the three logit vectors. The class corresponding to the logit element with the largest value in $\mathbf{z}$ is the final prediction.

IV. EXPERIMENTS

A. Experimental Setup

1) Datasets: We conduct our few-shot evaluation on 11 datasets which cover a wide range of distinct recognition tasks, including two datasets of generic objects (i.e., ImageNet [44] and Caltech101 [45]), five datasets for fine-grained classifications of pets, cars, flowers, food and aircraft (i.e., OxfordPets [46], StanfordCars [47], Flowers102 [48], Food101 [49] and FGVCAircraft [50]), one dataset for scene classification (i.e., SUN397 [51]), one dataset for texture classification (i.e., DTD [52]), one dataset for satellite images (i.e., EuroSAT [53]), and one dataset for action recognition (i.e., UCF101 [54]). For domain generalization evaluation, we use ImageNetV2 [55], ImageNet-Sketch [56], ImageNet-A [57], and ImageNet-R [58] as out-of-distribution datasets and ImageNet as in-distribution dataset. Please see Table II for more dataset details.

2) Settings: To evaluate our framework in extremely scarce data situations, we train the model using 1/2/4/8/16-shot images per class and then evaluate the model on the test set of all images. For domain generalization, we train our model on 16-shot ImageNet images per class and evaluate the model on the different out-of-distribution test sets without any fine-tuning technique.

TABLE III
AVERAGE ACCURACY FROM DIFFERENT METHODS OVER 11 DATASETS

Shot Setup	1	2	4	8	16
Zero-shot CLIP : 58.77					
Linear-probe CLIP	36.67	47.61	57.19	64.98	71.10
CoOp	59.59	62.32	66.77	69.89	73.42
WiSE-FT	59.09	61.80	65.29	68.43	71.64
ProGrad	62.61	64.90	68.45	71.41	73.96
CLIP-Adapter	62.67	65.55	68.61	71.40	74.44
Tip-Adapter	62.33	64.62	66.54	68.50	70.32
Tip-Adapter-F	64.62	66.65	69.67	72.45	75.83
PLOT	62.59	65.23	68.60	71.23	73.94
Tip-Adapter-F + PLOT	65.45	68.63	71.23	73.49	76.20
Cross-Modal	64.66	67.68	70.59	73.97	77.22
Cross-Modal Linear-probe	64.13	66.95	70.31	72.96	75.97
Cross-Modal Adapter	64.40	67.57	70.78	73.35	75.94
TaskRes	64.28	67.55	70.28	73.35	75.78
CLAP	62.79	66.07	69.13	72.08	74.57
DAT (ours)	**65.99**	**68.69**	**72.49**	**75.57**	**79.14**

3) Implementation Details: By default, we employ ResNet-50 as the backbone for the CLIP image encoder, along with the corresponding Transformer [59] for the text encoder. During the training period, we apply random cropping and random horizontal flipping to each image, then resize it to 224×224 pixels for all datasets. The batch size is set to 128 for the ImageNet and 64 for other datasets. On all datasets, the model is trained for 200 epochs with an initial learning rate of 0.001. We utilize the AdamW optimizer [60] with a weight decay of $1e^{-5}$ and cosine annealing learning rate scheduling. The hyper-parameter τ in $\mathcal{L}_F$ is set to 0.1 for all experiments. By default, λ_1 is set to 0.5 and λ_2 is set to 3.0. α_1 and α_2 are tuned on the validation sets. During the inference stage, only center cropping and resizing are applied to each test image.

B. Performance Comparison

1) Few-Shot Learning: Table III presents a comparison between our method and other approaches in terms of average performance across 11 datasets for the 1/2/4/8/16-shot settings. The compared methods include CLIP [11], CoOp [15], WiSE-FT [61], ProGrad [62], CLIP-Adapter [6], Tip-Adapter [17], Tip-Adapter-F [17], PLOT [63], Cross-Modal [32], CLAP [33]

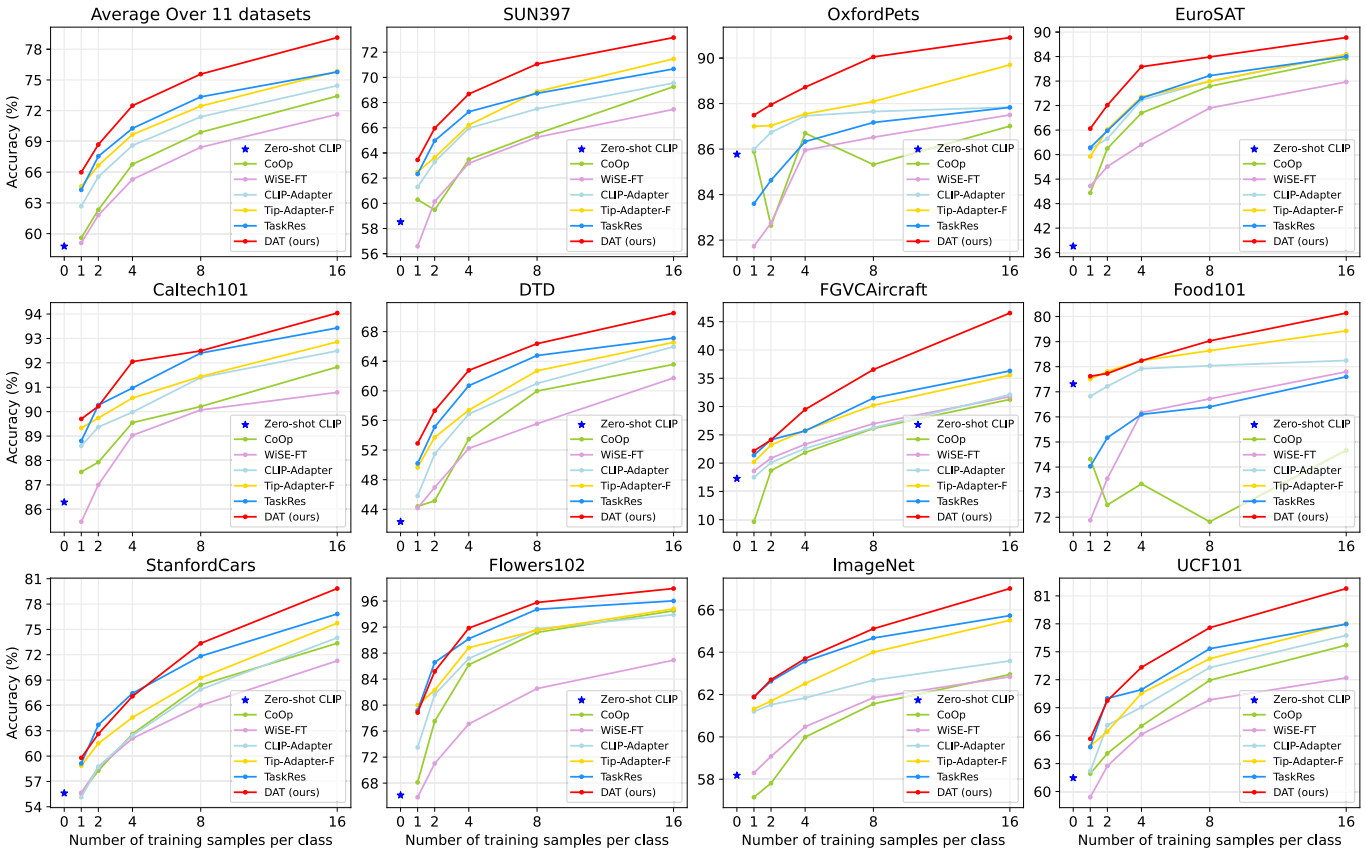

Fig. 3. The performance comparison in few-shot learning on 11 datasets. Averaged performance is presented in the upper-left figure, which demonstrates the superiority of our approach. Detailed numerical results can be found in Table IX.

and TaskRes [19], and our method exhibits state-of-the-art performance. For example, we achieve an average performance gain of nearly 2 percentage points over 11 datasets under the 16-shot setting. Even in scenarios with extremely limited data (e.g., 1/2-shot), we still achieve the best average performance compared to other methods. Figure 3 illustrates the performance comparison of our method against other approaches across all datasets, and Table IX lists the full numerical results of our method under various few-shot settings for each dataset. Notably, on the FGVCAircraft dataset, our method achieves a performance gain of over 10 percentage points under the 16-shot setting compared to the TaskRes. Similarly, on the UCF101 dataset, our method achieves a performance gain of nearly 4 percentage points under the same setting compared to the TaskRes. Overall, our approach achieves optimal performance in various settings, demonstrating that our DAT framework for few-shot learning is able to better align the text branch and image branch of the CLIP model, and achieves better adaptation ability on downstream tasks.

TABLE IV

DOMAIN GENERALIZATION PERFORMANCE COMPARISON. MODELS ARE TRAINED ON THE AVERAGE OF THREE RANDOM SEEDS OF SOURCE DATASET WITH 16-SHOT SETTING AND TESTED ON TARGET DATASETS

Methods	Target				
	-V2	-Sketch	-A	-R	Average
Zero-shot CLIP	51.34	33.32	21.65	56.00	40.58
Linear-probe CLIP	45.97	19.07	12.74	34.86	28.16
CoOp	55.40	34.67	**23.06**	56.60	42.43
CLIP-Adapter	52.67	32.04	20.12	54.75	39.90
Tip-Adapter-F	57.11	36.00	20.60	57.98	42.92
PLOT	55.11	33.00	21.86	55.61	41.40
Cross-Modal Linear-probe	55.30	33.10	20.00	56.40	41.20
Cross-Modal + WiSE-FT	56.60	35.60	22.60	59.50	43.58
TaskRes	57.00	34.43	21.50	58.13	42.77
Word soup (ViT-B/16)	63.00	49.00	**50.40**	77.20	59.90
CLAP (ViT-B/16)	64.06	47.66	48.40	76.70	59.21
DAT (ours, RN50)	**58.51±0.12**	**37.05±0.15**	23.01±0.16	**60.79±0.05**	**44.84**
DAT (ours, ViT-B/16)	**64.21±0.07**	**49.51±0.20**	48.59±0.23	**77.70±0.14**	**60.00**

2) Generalization Ability: Table IV demonstrates the domain generalization ability of our method, where the model is trained on the source dataset and tested on the target datasets. The model is trained on the average of three random seeds of the 16-shot ImageNet images per class and evaluated on four out-of-distribution datasets. Our method overall shows stronger capability in domain generalization, confirming its robustness and transferring ability. On the ImageNet-A dataset, our method slightly underperforms compared to CoOp [15] and Word soup [35], which may be attributed to the fact that ImageNet-A consists of real-world adversarially filtered images that can fool current ImageNet classifiers. However, we still achieve the best average performance on the four out-of-distribution datasets.

In addition, our learning framework is not limited to the image encoder backbone ResNet-50. As Table V shows,

TABLE V
FEW-SHOT LEARNING PERFORMANCE ON IMAGENET WITH DIFFERENT BACKBONES UNDER THE 16-SHOT SETTING

Backbone	RN50	RN101	ViT-B/32	ViT-B/16
Zero-shot CLIP	58.18	61.62	62.05	66.73
CoOp	62.95	66.60	66.85	71.92
CLIP-Adapter	63.59	65.39	66.19	71.13
Tip-Adapter-F	65.51	68.56	68.65	73.69
TaskRes	65.73	68.73	69.17	73.90
DAT (ours)	**67.01**	**70.08**	**69.42**	**74.54**

TABLE VI
FEW-SHOT LEARNING COMPUTATION EFFICIENCY BETWEEN DIFFERENT METHODS ON IMAGENET UNDER THE 16-SHOT SETTING

Methods	Training	Epochs	Parameters	Accuracy
Zero-shot CLIP	-	-	-	58.18
CALIP (training free)	-	-	-	60.57
Tip-Adapter (training free)	-	-	-	62.03
CoOp	14h	200	0.01M	62.95
CLIP-Adapter (single-branch fine-tuning)	50min	200	0.52M	63.59
CLIP-Adapter (dual-branch fine-tuning)	50min	200	1.04M	60.70
Tip-Adapter-F	5min	20	16.3M	65.51
DAT (ours)	40min	200	9.38M	**67.01**

TABLE VII
ABLATION STUDY OF OUR METHOD ON THREE REPRESENTATIVE DATASETS UNDER THE 16-SHOT SETTING

Components					**Datasets**		
$\mathcal{L}_F$	$\mathcal{L}_{G,1}$	$\mathcal{L}_{G,2}$	V.A.	T.A.	ImageNet	Caltech101	OxfordPets
					65.13	92.78	89.51
			✓		64.82	92.65	88.17
✓			✓		65.76	93.06	90.31
✓				✓	65.23	92.86	88.99
✓			✓	✓	65.83	93.27	90.45
✓	✓	✓		✓	65.92	93.23	90.02
✓	✓	✓	✓		66.40	93.26	90.40
	✓	✓	✓	✓	65.64	92.09	89.51
✓		✓	✓	✓	66.50	93.43	90.60
✓	✓		✓	✓	66.83	93.71	90.54
✓	✓	✓	✓	✓	**67.01**	**94.04**	**90.90**

TABLE VIII
ABLATION STUDY OF DOMAIN GENERALIZATION CLASSIFICATION ON FOUR OUT-OF-DISTRIBUTION DATASETS UNDER THE 16-SHOT SETTING

Components		**Datasets**			
V.A.	T.A.	-V2	-Sketch	-A	-R
✓		58.26	36.37	21.51	59.98
	✓	57.89	36.76	22.19	60.29
✓	✓	58.60	37.01	22.85	60.87

when replacing it with ResNet-101, ViT-B/32, and ViT-B/16 respectively, our method still achieves optimal performance compared to current state-of-the-art methods.

Furthermore, we also use other vision-language model to demonstrate the effectiveness of our approach. As a self-supervised pre-training version of CLIP, SLIP [64] can achieve higher accuracy in DAT framework, the performance gain on the ImageNet under the 16-shot SLIP condition is around 7% in accuracy compared to the 16-shot CLIP condition.

C. Computation Efficiency

We compare the computing efficiency between DAT and existing methods in Table VI. We test by an NVIDIA GeForce RTX 3090 GPU and report the performance on 16-shot ImageNet classification. As shown in Table VI, CLIP-Adapter dual-branch fine-tuning has more trainable parameters compared to single-branch fine-tuning, however, it results in a performance decrease. Tip-Adapter-F, on the other hand, has a shorter training time but has more trainable parameters and poor performance. In contrast, our method also employs dual-branch fine-tuning, and better overcomes the over-fitting problem caused by the increase of training parameters and obtains better classification performance under few-shot samples conditions.

D. Ablation Study

Ablation study is performed on three representative datasets. As shown in Table VII, removing any one of the key components (feature consistency loss $\mathcal{L}_F$, logit consistency losses $\mathcal{L}_{G,1}$ and $\mathcal{L}_{G,2}$, visual adapter V.A., and textual adapter T.A.) will cause degraded performance (rows 6th-10th vs. last row), confirming the role of each component in improving the few-shot learning performance. Without the feature consistency loss $\mathcal{L}_F$ and logit consistency loss $\mathcal{L}_{G,1} + \mathcal{L}_{G,2}$, solely inclusion of the visual adapter will cause decreased performance (2nd row vs. 1st row), probably because more trainable parameters in the model without any consistency loss would likely cause more severe over-fitting in few-shot learning. However, the performance is significantly improved by incorporating the feature consistency loss and logit consistency loss during model training (rows 3rd and 7th vs. 2nd row), even with the incorporation of the textual adapter compared to including the visual adapter only (5th row and rows 8th-10th vs. 2nd row), highlighting the importance and effectiveness of these consistency constraints. Meanwhile, when removing T.A. from the framework (7th row), the model shows a clear performance drop on all three datasets (7th row vs. 11th row), demonstrating the importance of T.A.. The small difference between rows 3rd and 5th indicates that T.A. would not work effectively without the help of the logit consistency losses $L_{G,1}$ and $L_{G,2}$, which confirms that the proposed logit consistency losses are essential for T.A. (and V.A.).

Additionally, we place the results of domain generalization ablation experiments containing only V.A. or T.A. in Table VIII. It can be observed that in the domain generalization classification, except for the ImageNet-V2 dataset which is just the new test data for ImageNet, and V.A. is playing a more dominant role. For the remaining three out-of-distribution datasets, due to the training images of ImageNet being of the same class as the test images of three out-of-distribution datasets but with a large difference in image styles, it is the text-branch of T.A. that plays a more critical factor in the categorization process.

Overall, each component helps and the performance gain comes from the combination of the model's structure and the losses functions. From the perspective of model structure,

TABLE IX
FULL NUMERICAL RESULTS OF PERFORMANCE COMPARISON ON FEW-SHOT LEARNING

Method	Setting	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	Average
Zero-Shot CLIP		58.18	86.29	85.77	55.61	66.14	77.31	17.28	58.52	42.32	37.56	61.46	58.77
Linear-probe CLIP	full-shot	73.30	89.60	88.20	78.30	96.10	86.40	49.10	73.30	76.40	95.20	81.60	80.68
Linear-probe RN50	full-shot	74.30	90.80	92.40	49.90	90.80	71.30	48.50	60.50	72.30	96.70	71.20	74.43
Linear-probe CLIP		22.07	70.62	30.14	24.64	58.07	30.13	12.89	32.80	29.59	51.00	41.43	36.67
CoOp		57.15	87.53	85.89	55.59	68.12	74.32	9.64	60.29	44.39	50.63	61.92	59.59
CLIP-Adapter		61.20	88.60	85.99	55.13	73.49	76.82	17.49	61.30	45.80	61.40	62.20	62.67
Tip-Adapter-F	1-shot	61.13	89.33	87.00	58.86	**79.98**	77.51	20.22	62.50	49.65	59.53	64.87	64.62
TaskRes		**61.90**	88.80	83.60	59.13	79.17	74.03	21.40	62.33	50.20	61.70	64.77	64.28
CLAP		58.50	88.38	83.64	56.35	79.90	73.00	20.62	61.15	47.46	59.21	62.48	62.79
DAT (Ours)		61.88	**89.70**	**87.49**	**59.79**	78.85	**77.62**	**22.17**	**63.45**	**52.90**	**66.36**	**65.66**	**65.99**
Linear-probe CLIP		31.95	78.72	43.47	36.53	73.35	42.79	17.85	44.44	39.48	61.58	53.55	47.61
CoOp		57.81	87.93	82.64	58.28	77.51	72.49	18.68	59.48	45.15	61.50	64.09	62.32
CLIP-Adapter		61.52	89.37	86.73	58.74	81.61	77.22	20.10	63.29	51.48	63.90	67.12	65.55
Tip-Adapter-F	2-shot	61.69	89.74	87.03	61.50	82.30	**77.81**	23.19	63.64	53.72	66.15	66.43	66.65
TaskRes		62.63	**90.27**	84.63	**63.70**	**86.57**	75.17	**24.13**	64.97	55.13	65.83	**70.00**	67.55
CLAP		58.50	89.79	84.93	61.40	84.22	74.94	23.21	63.31	53.05	65.63	67.77	66.07
DAT (Ours)		**62.70**	90.22	**87.95**	62.62	85.18	77.73	24.09	**65.96**	**57.33**	**72.09**	69.76	**68.69**
Linear-probe CLIP		41.29	84.34	56.35	48.42	84.80	55.15	23.57	54.59	50.06	68.27	62.23	57.19
CoOp		59.99	89.55	86.70	62.62	86.20	73.33	21.87	63.47	53.49	70.18	67.03	66.77
CLIP-Adapter		61.84	89.98	87.46	62.45	87.17	77.92	22.59	65.96	56.86	73.38	69.05	68.61
Tip-Adapter-F	4-shot	62.52	90.56	87.54	64.57	88.83	78.24	25.80	66.21	57.39	74.12	70.55	69.67
TaskRes		63.57	90.97	86.33	**67.43**	90.20	76.10	25.70	67.27	60.70	73.83	70.93	70.28
CLAP		60.73	90.62	86.51	65.50	87.66	75.92	25.65	65.99	58.85	73.15	69.88	69.13
DAT (Ours)		**63.70**	**92.05**	**88.72**	67.08	**91.84**	**78.24**	**29.49**	**68.68**	**62.77**	**81.51**	**73.33**	**72.49**
Linear-probe CLIP		49.55	87.78	65.94	60.82	92.00	63.82	29.55	62.17	56.56	76.93	69.64	64.98
CoOp		61.56	90.21	85.32	68.43	91.18	71.82	26.13	65.52	59.97	76.73	71.94	69.89
CLIP-Adapter		62.68	91.40	87.65	67.89	91.72	78.04	26.25	67.50	61.00	77.93	73.30	71.40
Tip-Adapter-F	8-shot	64.00	91.44	88.09	69.25	91.51	78.64	30.21	68.87	62.71	77.93	74.25	72.45
TaskRes		64.67	92.40	87.17	71.83	94.73	76.40	31.50	68.73	64.77	79.33	75.33	73.35
CLAP		62.98	91.45	87.75	70.35	92.06	77.42	28.97	68.61	63.24	76.66	73.34	72.08
DAT (Ours)		**65.11**	**92.49**	**90.05**	**73.35**	**95.78**	**79.03**	**36.54**	**71.06**	**66.37**	**83.90**	**77.58**	**75.57**
Linear-probe CLIP		55.87	90.63	76.42	70.08	94.95	70.17	36.39	67.15	63.97	82.76	73.72	71.10
CoOp		62.95	91.83	87.01	73.36	94.51	74.67	31.26	69.26	63.58	83.53	75.71	73.42
CLIP-Adapter		63.59	92.49	87.84	74.01	93.90	78.25	32.10	69.55	65.96	84.43	76.76	74.44
Tip-Adapter-F	16-shot	65.51	92.86	89.70	75.74	94.80	79.43	35.55	71.47	66.55	84.54	78.03	75.83
TaskRes		65.73	93.43	87.83	76.83	96.03	77.60	36.30	70.67	67.13	84.03	77.97	75.78
CLAP		65.02	91.93	88.51	75.12	94.21	78.55	33.59	70.78	66.41	80.07	76.29	74.57
DAT (Ours)		**67.01**	**94.04**	**90.90**	**79.84**	**97.93**	**80.14**	**46.53**	**73.16**	**70.51**	**88.65**	**81.79**	**79.14**

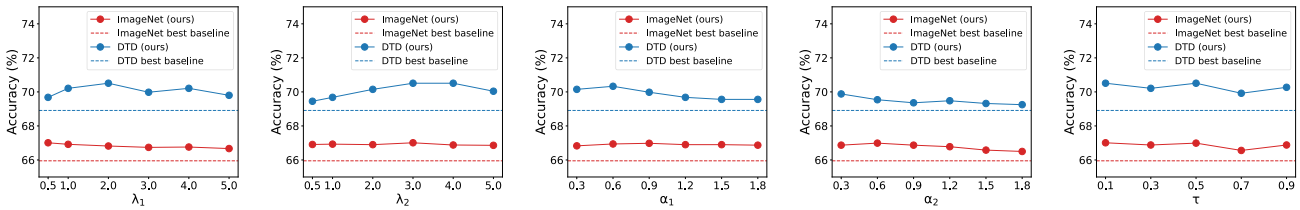

Fig. 4. Sensitivity study on the Imagenet and DTD datasets. Our method (solid curves) consistently outperforms the best baseline (dashed lines) across different hyper-parameter settings.

we use dual-branch adapters to ensure sufficient adaptation of the multi-modality model as a whole and provide greater flexibility in aligning visual and language representations, meanwhile using MLP module to generate a new logit vector for better image classification. From the perspective of loss functions, we use logit consistency loss $\mathcal{L}_{G,1}$ and $\mathcal{L}_{G,2}$ to achieve the knowledge distillation to alleviate the over-fitting issue, meanwhile using feature consistency loss $\mathcal{L}_F$ to well align the image branch and the text branch during the training process. The design of the model structure and the loss functions together make our method achieve the best overall performance.

E. Additional Ablation Study

In addition to the ablation experiments for each of the important components, we show more ablation experiments in Table X to demonstrate the effectiveness of our proposed framework. Additional ablation study is also performed on three representative datasets, including ImageNet, Caltech101 and OxfordPets. The experimental setup includes:

No skip connection in textual adapter: We add skip connection to the textual adapter (T.A.) module to mitigate the possibility of over-fitting on the text branch. The results without skip connection and not utilizing prior knowledge from the original pre-trained text encoder are shown in the first row of Table X.

No skip connection in adapted visual encoder: We add skip connection to the adapted visual encoder so that the original feature maps and the feature maps are fine-tuned by the visual adapter (V.A.) through the addition function to better mitigate the over-fitting in few-shot scenarios. The results without skip connection and utilizing only the fine-tuned feature maps are shown in the second row of Table X.

TABLE X
ADDITIONAL ABLATION STUDY OF OUR METHOD ON THREE REPRESENTATIVE DATASETS UNDER THE 16-SHOT SETTING

Ablation Setup	Datasets		
	ImageNet	Caltech101	OxfordPets
No skip connection in textual adapter	66.75	92.98	89.86
No skip connection in adapted visual encoder	65.79	92.74	89.34
Visual adapter after pooling layer	65.83	92.58	90.02
Removing descriptive prompts	66.85	93.51	90.11
MLP reduced to one linear layer	66.74	93.06	90.16
Class prediction without Zero-shot logits	66.96	93.71	90.79
Directly applying LoRA to the CLIP encoder	65.72	93.06	89.18
Original DAT framework	**67.01**	**94.04**	**90.90**

Visual adapter after pooling layer: We consider that more visual information exists in the feature map than in the feature vector, and in order to better utilize such visual information, we use the visual adapter (V.A.) to fine-tune the feature map rather than the feature vector. We place the visual adapter after the self-attention pooling layer and replace the 1×1 convolutional layer with the fully connected layer in the third row of Table X.

Removing descriptive prompts: We use descriptive prompts with more detailed class information to encode more prior knowledge for each class, enriching the properties and characteristics of each visual class. The results of using only vanilla prompts without descriptive prompts are shown in the fourth row of Table X.

MLP reduced to one linear layer: We use two fully connected layers in the MLP to provide better learning ability. The results of transforming the dimension from D (the dimensionality of the feature vector space) to C (the number of classes) using only one linear layer are shown in the fifth row of Table X.

Class prediction without zero-shot logits: In order to demonstrate that the DAT framework plays a dominant role through the categorized Adapted logit vector $\mathbf{z}^a$ and MLP logit vector $\mathbf{z}^m$ obtained by training process, the results without zero-shot logit vector $\mathbf{z}^o$ during the class prediction stage (Equation 6 without zero-shot logit vector $\mathbf{z}^o$) are placed in the sixth row of Table X, and competitive results are obtained to show the effectiveness of our framework.

Directly applying LoRA to the CLIP encoder: To demonstrate the effectiveness of our designed adapters, the results of applying LoRA directly to CLIP encoder without using adapters are shown in the seventh row of Table X. We found that using LoRA does not give us as good results as the adapters we have devised. We think that this is because the token vectors are fine-tuned using LoRA in Self-attention Pooling Layer, whereas the visual adapter we designed is used to fine-tune the feature maps, considering that more visual information exists in the feature maps than in the token feature vectors, and therefore potentially discriminative features specifically for the few-shot learning task can be more likely preserved with the help of the visual adapter.

F. Sensitivity Study

The sensitivity of our method to the hyper-parameters, i.e., λ_1 and λ_2 (Equation 5), α_1 and α_2 (Equation 6), and τ (Equation 2), is evaluated on two representative datasets Imagenet and DTD. As Figure 4 shows, by varying each hyper-parameter in certain range, the performance of our method varies a bit but it always outperforms the best baseline. This suggests that our method is largely insensitive to the choice of hyper-parameter values.

V. CONCLUSION

In this study, we propose a novel dual-branch adapter-tuning few-shot learning framework, where both the textual adapter and the visual adapter can be effectively optimized with the help of feature consistency and logit consistency constraints. Extensive evaluations on multiple image classification datasets and under various few-shot settings consistently suggest that our method outperforms current state-of-the-art methods for few-shot learning. Our learning framework provides a solution when pre-trained encoders of multiple modalities all need to be adapted for downstream tasks.

REFERENCES

[1] S. Deng, D. Liao, X. Gao, J. Zhao, and K. Ye, "A survey on cross-domain few-shot image classification," in *Proc. Big Data*, 2023, pp. 3–17.
[2] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. ICML*, Jan. 2017, pp. 1–10.
[3] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 113–123.
[4] J. Xie, F. Long, J. Lv, Q. Wang, and P. Li, "Joint distribution matters: Deep Brownian distance covariance for few-shot classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7962–7971.
[5] Z. Guo et al., "CALIP: Zero-shot enhancement of CLIP with parameter-free attention," in *Proc. AAAI*, 2023, pp. 1–9.
[6] P. Gao et al., "CLIP-adapter: Better vision-language models with feature adapters," *Int. J. Comput. Vis.*, vol. 132, pp. 581–595, Sep. 2023.
[7] J. Xu, B. Liu, and Y. Xiao, "A variational inference method for few-shot learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 269–282, Jan. 2023.
[8] Z. Zheng, G. Huang, X. Yuan, C.-M. Pun, H. Liu, and W.-K. Ling, "Quaternion-valued correlation learning for few-shot semantic segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 5, pp. 2102–2115, May 2023.
[9] S. Shao, L. Xing, Y. Wang, B. Liu, W. Liu, and Y. Zhou, "Attention-based multi-view feature collaboration for decoupled few-shot learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 5, pp. 2357–2369, May 2023.
[10] J. Zhang, J. Huang, S. Jin, and S. Lu, "Vision-language models for vision tasks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 8, pp. 5625–5644, Aug. 2024.
[11] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. ICML*, Jan. 2021, pp. 1–16.
[12] Y. Xing, J. Kang, A. Xiao, J. Nie, L. Shao, and S. Lu, "Rewrite caption semantics: Bridging semantic gaps for language-supervised semantic segmentation," in *Proc. NeurIPS*, Feb. 2023, pp. 1–12.
[13] J. He, C. Zhou, X. Ma, T. Berg-Kirkpatrick, and G. Neubig, "Towards a unified view of parameter-efficient transfer learning," in *Proc. ICLR*, 2022, pp. 1–12.
[14] H. Jia, Y. Xu, L. Zhu, G. Chen, Y. Wang, and Y. Yang, "MoS^2: Mixture of scale and shift experts for text-only video captioning," in *Proc. ACM MM*, 2024, pp. 8498–8507.
[15] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," *Int. J. Comput. Vis.*, vol. 130, no. 9, pp. 2337–2348, Sep. 2022.
[16] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Conditional prompt learning for vision-language models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16795–16804.

[17] R. Zhang et al., "Tip-adapter: Training-free adaption of CLIP for few-shot classification," in *Proc. ECCV*, Oct. 2022, pp. 493–510.

[18] R. Quan, X. Yu, Y. Liang, and Y. Yang, "Removing raindrops and rain streaks in one go," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9143–9152.

[19] T. Yu, Z. Lu, X. Jin, Z. Chen, and X. Wang, "Task residual for tuning vision-language models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 10899–10909.

[20] C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision," in *Proc. ICML*, 2021, pp. 1–34.

[21] L. Yao et al., "FILIP: Fine-grained interactive language-image pre-training," 2021, *arXiv:2111.07783*.

[22] H. Gharoun, F. Momenifar, F. Chen, and A. H. Gandomi, "Meta-learning approaches for few-shot learning: A survey of recent advances," *ACM Comput. Surveys*, vol. 56, no. 12, pp. 1–41, Dec. 2024.

[23] Y. Shao, W. Wu, X. You, C. Gao, and N. Sang, "Improving the generalization of MAML in few-shot classification via bi-level constraint," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 7, pp. 3284–3295, Jul. 2023.

[24] Z. Hu, L. Shen, S. Lai, and C. Yuan, "Task-adaptive feature disentanglement and hallucination for few-shot classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 3638–3648, Aug. 2023.

[25] F. Ji, Y. Chen, L. Liu, and X.-T. Yuan, "Cross-domain few-shot classification via dense-sparse-dense regularization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 3, pp. 1352–1363, Mar. 2024.

[26] X. Wang et al., "Task-aware dual-representation network for few-shot action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 10, pp. 5932–5946, Oct. 2023.

[27] X. Li, Q. Song, J. Wu, R. Zhu, Z. Ma, and J.-H. Xue, "Locally-enriched cross-reconstruction for few-shot fine-grained image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7530–7540, Dec. 2023.

[28] H. Zhang, M. Cissé, Y. Dauphin, and D. López-Paz, "Mixup: Beyond empirical risk minimization," in *Proc. ICLR*, 2017, pp. 1–13.

[29] G. Jiang, P. Zhu, Y. Wang, and Q. Hu, "OpenMix+: Revisiting data augmentation for open set recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 11, pp. 6777–6787, Apr. 2023.

[30] J. Wang and Y. Zhai, "Prototypical Siamese networks for few-shot learning," in *Proc. IEEE 10th Int. Conf. Electron. Inf. Emergency Commun. (ICEIEC)*, Jul. 2020, pp. 178–181.

[31] P. Li, H. Xie, Y. Jiang, J. Ge, and Y. Zhang, "Neighborhood-adaptive multi-cluster ranking for deep metric learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 4, pp. 1952–1965, Apr. 2023.

[32] Z. Lin, S. Yu, Z. Kuang, D. Pathak, and D. Ramanan, "Multimodality helps unimodality: Cross-modal few-shot learning with multimodal models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 19325–19337.

[33] J. Silva-Rodríguez, S. Hajimiri, I. B. Ayed, and J. Dolz, "A closer look at the few-shot adaptation of large vision-language models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 23681–23690.

[34] S. Zhao, R. Quan, L. Zhu, and Y. Yang, "CLIP4STR: A simple baseline for scene text recognition with pre-trained vision-language model," 2023, *arXiv:2305.14014*.

[35] C. Liao, T. Tsiligkaridis, and B. Kulis, "Descriptor and word soups: Overcoming the parameter efficiency accuracy tradeoff for out-of-distribution few-shot learning," in *Proc. CVPR*, Jan. 2023, pp. 27015–27025.

[36] F. Peng, X. Yang, L. Xiao, Y. Wang, and C. Xu, "SgVA-CLIP: Semantic-guided visual adapting of vision-language models for few-shot image classification," *IEEE Trans. Multimedia*, vol. 26, pp. 3469–3480, 2024.

[37] S. Pratt, I. Covert, R. Liu, and A. Farhadi, "What does a platypus look like? Generating customized prompts for zero-shot image classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 15645–15655.

[38] T. B. Brown et al., "Language models are few-shot learners," in *Proc. NeurIPS*, Jan. 2020, pp. 1–14.

[39] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, Jun. 2019, pp. 4171–4186.

[40] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. AISTATS*, Apr. 2011, pp. 1–14.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[42] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, Jan. 2015, pp. 1–23.

[43] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. ICLR*, Jan. 2020, pp. 1–22.

[44] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[45] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Comput. Vis. Image Understand.*, vol. 106, no. 1, pp. 59–70, Apr. 2007.

[46] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar, "Cats and dogs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3498–3505.

[47] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. ICCV*, 2013, pp. 1–23.

[48] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. 6th Indian Conf. Comput. Vis., Graph. Image Process.*, Dec. 2008, pp. 722–729.

[49] L. Bossard, M. Guillaumin, and L. V. Gool, "Food-101—Mining discriminative components with random forests," in *Proc. ECCV*, Jan. 2014, pp. 446–461.

[50] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," 2013, *arXiv:1306.5151*.

[51] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "SUN database: Large-scale scene recognition from abbey to zoo," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3485–3492.

[52] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3606–3613.

[53] P. Helber, B. Bischke, A. Dengel, and D. Borth, "EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 7, pp. 2217–2226, Jul. 2019.

[54] K. Soomro, A. Roshan Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*.

[55] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do ImageNet classifiers generalize to ImageNet," in *Proc. ICML*, May 2019, pp. 5389–5400.

[56] H. Wang, S. Ge, E. P. Xing, and Z. C. Lipton, "Learning robust global representations by penalizing local predictive power," in *Proc. NeurIPS*, Jan. 2019, pp. 1–23.

[57] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, "Natural adversarial examples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15257–15266.

[58] D. Hendrycks et al., "The many faces of robustness: A critical analysis of out-of-distribution generalization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8320–8329.

[59] A. Vaswani et al., "Attention is all you need," in *Proc. NeurIPS*, vol. 30, Jun. 2017, pp. 5998–6008.

[60] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. ICLR*, Jan. 2017, pp. 1–23.

[61] M. Wortsman et al., "Robust fine-tuning of zero-shot models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7949–7961.

[62] B. Zhu, Y. Niu, Y. Han, Y. Wu, and H. Zhang, "Prompt-aligned gradient for prompt tuning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 15613–15623.

[63] G. Chen, W. Yao, X. Song, X. Li, Y. Rao, and K. Zhang, "PLOT: Prompt learning with optimal transport for vision-language models," in *Proc. ICLR*, Jan. 2022, pp. 1–22.

[64] N. Mu, A. M. Kirillov, D. Wagner, and S. Xie, "SLIP: Self-supervision meets language-image pre-training," in *Proc. ECCV*, Jan. 2021, pp. 529–544.

Junxi Chen received the bachelor's degree in Internet of Things from Sichuan Agricultural University in 2022. He is currently pursuing the master's degree with the School of Computer Science and Engineering, Sun Yat-sen University. His research interests include computer vision, machine learning, and few-shot learning. He has already published two related articles.

Guangxing Wu received the bachelor's degree in computer science and technology from Shanghai University in 2022. He is currently pursuing the master's degree with the School of Computer Science and Engineering, Sun Yat-sen University. His research focuses on computer vision and few-shot learning. He has already published two related articles.

Hongxiang Li is currently pursuing the master's degree with Peking University. He is majoring in computer application technology. His research interests lie in video understanding, video generation, and visual language learning.

Jiankang Chen received the Graduate degree in computer science and technology from the School of Computer Science and Technology, Jilin University. He is currently pursuing the master's degree with the School of Computer Science, Sun Yat-sen University. His research directions are out-of-distribution detection in the computer vision, participate in the EEG image based on the classification of patients with depression hospital project, and participate in the AI quality detection project of China Unicom; during his study period, he has published two AAAI articles as first and second authors and one ACM MM article as first author.

Wentao Zhang received the B.S. and M.S. degrees in 2017 and 2020, respectively. He is currently pursuing the Ph.D. degree with Sun Yat-sen University. His research interests include computer vision, medical image analysis, and deep learning.

Wei-Shi Zheng is currently a Full Professor with Sun Yat-sen University. His research interests include person/object association and activity understanding in visual surveillance and the related large-scale machine learning algorithm. He has ever served as the Area Chair for ICCV, CVPR, ECCV, BMVC, and IJCAI. He is also an Associate Editor of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and *Pattern Recognition*. He has ever joined Microsoft Research Asia Young Faculty Visiting Programme. He is a Cheung Kong Scholar Distinguished Professor, a recipient of the Excellent Young Scientists Fund of the National Natural Science Foundation of China, and a recipient of the Royal Society-Newton Advanced Fellowship of the U.K.

Ruixuan Wang received the Ph.D. degree from the National University of Singapore in 2008. He was a Post-Doctoral Researcher with the University of Dundee, U.K. He is currently an Associate Professor with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. His research interests include computer vision, medical image analysis, and machine learning.