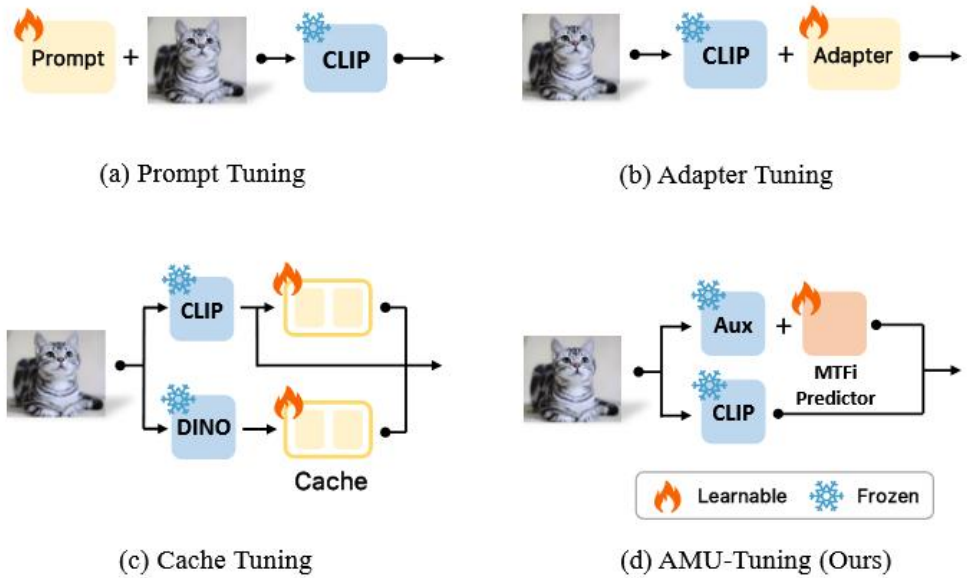
南京航空航天大學
NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

AMU-Tuning: Effective Logit Bias for CLIP-based Few-shot Learning

Yuwei Tang*, Zhenyi Lin*, Qilong Wang†, Pengfei Zhu, Qinghua Hu
Tianjin Key Lab of Machine Learning, College of Intelligence and Computing, Tianjin University, China
{tangyuwei, linzhenyi, qlwang,

CVPR 2024

Background



(a) Prompt Tuning (b) Adapter Tuning

(c) Cache Tuning (d) AMU-Tuning (Ours)

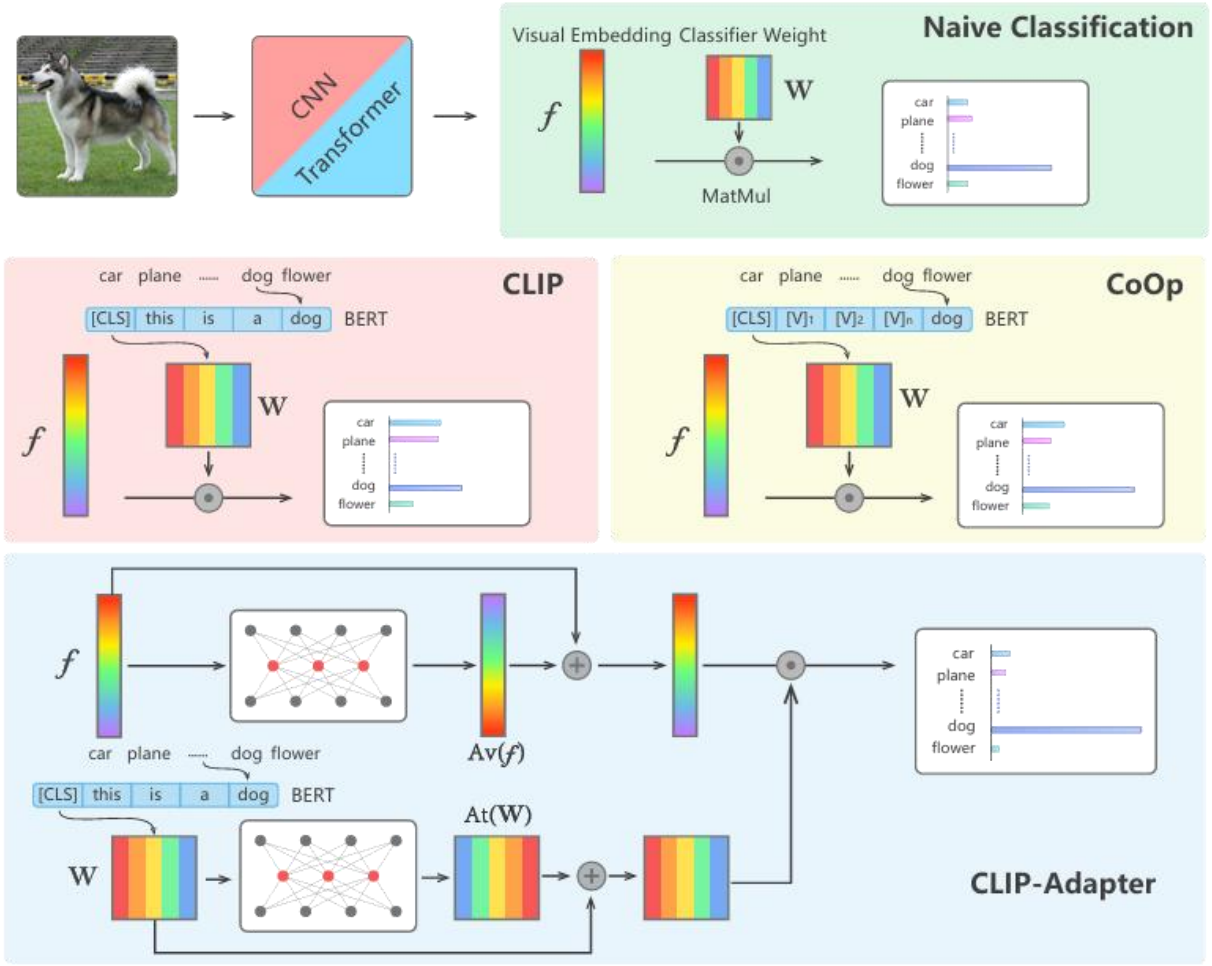
Figure 1. Comparison of the existing CLIP-based few-shot learning methods in terms of architecture design.

(a) prompt-tuning methods improve the few-shot learning ability of CLIP by introducing learnable text prompt for the text encoder of CLIP.

(b) For adapter-based tuning, some lightweight modules, multi-layer perceptron (MLP) , are built at the end of text and visual encoders to adjust text and visual features for downstream tasks.

(c) cache-based tuning methods present “soft” K-nearest neighbor classifiers storing visual features and labels of training samples, which are combined with zero-shot CLIP for final classification.

Background



$$\bar{f}_T(T_{\text{bias}})\mathbf{f}_0^C$$

$$f_T^{\text{Ada}}(\mathbf{W}_0)\mathbf{f}_0^C + \mathbf{W}_0 f_V^{\text{Ada}}(\mathbf{f}_0^C) + f_T^{\text{Ada}}(\mathbf{W}_0) f_V^{\text{Ada}}(\mathbf{f}_0^C), \quad (2)$$

where $f_T^{\text{Ada}}(\cdot)$ and $f_V^{\text{Ada}}(\cdot)$ are adapters for text and visual features, which are achieved by two MLP. Cache-based Tip-

Figure 1: Comparison of different visual classification architectures. The image in the top row with a green region shows the naive pipeline for image classification (Krizhevsky et al., 2012), where f and $\mathbf{W}$ represents the feature and classifier weight respectively. The following pink, yellow and blue regions represent the pipeline of CLIP (Radford et al., 2021), CoOp (Zhou et al., 2021), and our proposed CLIP-Adapter respectively.

Background

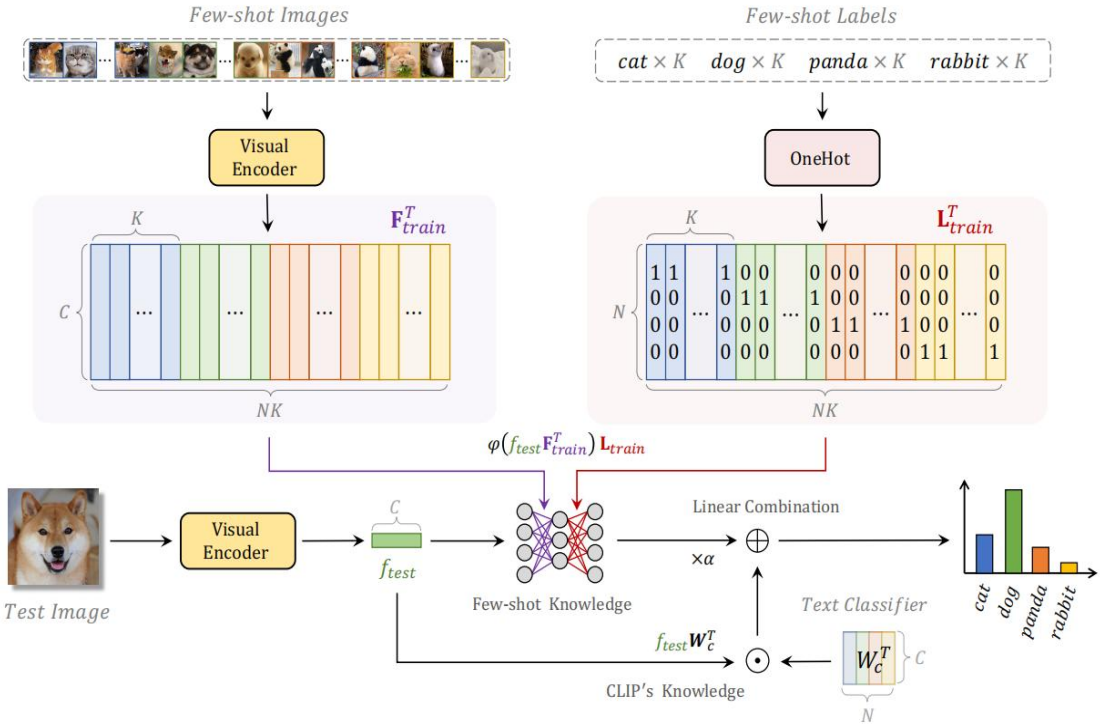


Figure 2. **The Pipeline of Tip-Adapter.** Given a K-shot N-class training set, we construct the weights of the two-layer adapter by creating a cache model from the few-shot training set. It contains few-shot visual features F_{train} encoded by CLIP's visual encoder and few-shot ground-truth labels L_{train}. F_{train} and L_{train} can be used as the weights for the first and second layers in the adapter.CSDN @栗栗子kury

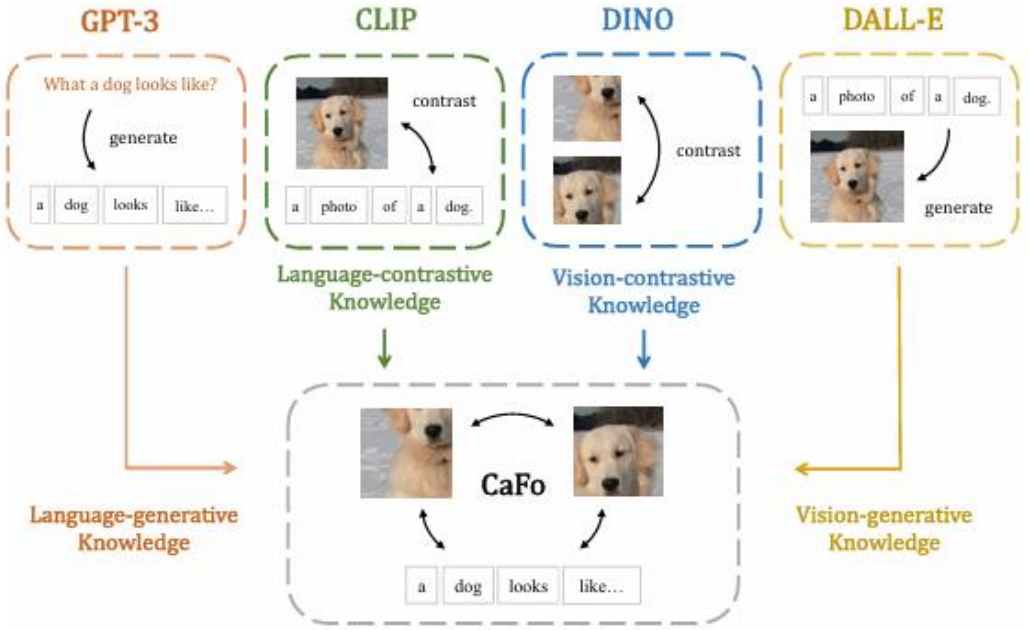


Figure 1. **The Cascade Paradigm of CaFo.** We adaptively incorporate the knowledge from four types of pre-training methods and achieve a strong few-shot learner.

neighbor classifier on a trainable cache of visual CLIP features ($\mathbf{F}_{\text{TrC}}$) to generate $\mathbf{s}_{\text{bias}}$, i.e.

$$\phi\left(\mathbf{F}_{\text{TrC}}^{\mathsf{T}}\mathbf{f}_0^C\right)\mathbf{V}, \tag{3}$$

ploits an extra trainable cache of visual DINO features [7] ($\mathbf{F}_{\text{TrD}}$) to compute $\mathbf{s}_{\text{bias}}$ as

$$\alpha\phi\left(\mathbf{F}_{\text{TrC}}^{\mathsf{T}}\mathbf{f}_0^C\right)\mathbf{V} + (1-\alpha)\phi\left(\mathbf{F}_{\text{TrD}}^{\mathsf{T}}\mathbf{f}_0^D\right)\mathbf{V}, \tag{4}$$

where α is a trade-off parameter computed based on the

Background

Model	Bias	Feature	Predictor	Fusion	16-shot Acc (%)
Zero-shot CLIP [43]	-	-	-	-	60.33
CoOp [67]	$\simeq f_T(T_{\text{bias}})\mathbf{f}_0^C$	T_{bias}	f_T	-	62.95
CLIP-Adapter [17]	$f_T^{\text{Ada}}(\mathbf{W}_0)\mathbf{f}_0^C + \mathbf{W}_0 f_V^{\text{Ada}}(\mathbf{f}_0^C) + f_T^{\text{Ada}}(\mathbf{W}_0) f_V^{\text{Ada}}(\mathbf{f}_0^C)$	CLIP	MLP	Manual Tuning	63.59
Tip-Adapter-F [63]	$\phi\left(\mathbf{F}_{\text{TrC}}^{\text{T}}\mathbf{f}_0^C\right)\mathbf{V}$	CLIP	Cache	Manual Tuning	65.51
CaFo [64]	$\alpha\phi\left(\mathbf{F}_{\text{TrC}}^{\text{T}}\mathbf{f}_0^C\right)\mathbf{V} + (1-\alpha)\phi\left(\mathbf{F}_{\text{TrD}}^{\text{T}}\mathbf{f}_0^D\right)\mathbf{V}$	CLIP+DINO	Cache	Similarity-based	68.79
AMU-Tuning (Ours)	$\widehat{\mathbf{W}}\mathbf{f}^{\text{Aux}}$	Aux	MTFi LP	Uncertainty-based	**70.02**

Table 1. Comparison of existing CLIP-based few-shot learning methods from the perspective of logit bias. Different from previous works, our AMU-Tuning learns logit bias by exploiting the appropriate auxiliary (Aux) features with multi-branch training feature-initialized (MTFi) LP followed by an uncertainty-based fusion, while achieving higher accuracy (Acc) on ImageNet-1K with 16-shot training samples.

we disassemble the computation of logit bias into three key components, logit feature, logit predictor, logit fusion.

Features for Computation of Logit Bias

To compute the logit bias, we train a simple LP for all auxiliary features. Then, the logit bias is combined with prediction of zero-shot CLIP by summation for few-shot classification. Particularly, we train an individual LP for all auxiliary features within 50 epochs, whose results represent the superiority of different auxiliary features (indicated by $\mathrm{SUP}_{\mathrm{Aux}}$). For measuring the complementarity ($\mathrm{CMY}_{\mathrm{Aux}}$) of different auxiliary features, we define $\mathrm{CMY}_{\mathrm{Aux}}$ by inverse of similarity between LP prediction of auxiliary features ($\mathbf{s}_{\mathrm{Aux}}$) and prediction of zero-shot CLIP ($\mathbf{s}_0$):

$$\begin{aligned} \mathrm{CMY}_{\mathrm{Aux}} &= 1 - \mathrm{SIM}(\mathbf{s}_0, \mathbf{s}_{\mathrm{Aux}}), \\ \mathrm{SIM}(\mathbf{s}_0, \mathbf{s}_{\mathrm{Aux}}) &= \frac{\mathbf{s}_0 \cdot \mathbf{s}_{\mathrm{Aux}}}{\|\mathbf{s}_0\|_2 \cdot \|\mathbf{s}_{\mathrm{Aux}}\|_2}, \end{aligned} \tag{5}$$

where SIM computes the cosine similarity between $\mathbf{s}_{\mathrm{Aux}}$ and $\mathbf{s}_0$. Clearly, smaller similarity means less correlation between $\mathbf{s}_{\mathrm{Aux}}$ and $\mathbf{s}_0$, indicating the auxiliary features may be more complementary to zero-shot CLIP.

Motivation

南京航空航天大學 NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

Features for Computation of Logit Bias

Model	SUP_{Aux} (%)	CMY_{Aux}	Fusion (%)
ZS-CLIP [43]	N/A	N/A	60.33
CLIP [43]	56.93	0.438	65.34
DINO [7]	55.65	<u>0.816</u>	68.32
MoCov3 [10]	<u>57.68</u>	**0.837**	**69.35**
MAE [21]	38.98	0.722	65.49
SparK [52]	28.31	0.770	63.56
MILAN [25]	**66.36**	0.718	<u>69.24</u>

Table 2. Comparison of different auxiliary features in terms of complementary (CMY_{Aux}), superiority (SUP_{Aux}) and fused results on ImageNet-1K with 16-shot samples. The best and second-best results are highlighted in **bold** and <u>underline</u>, respectively.

1、complementarity (CMY_{Aux}) is more important than superiority (SUP_{Aux}) for auxiliary features。 (CLIP,DINO)

2、the auxiliary features with higher SUP_{Aux} achieve better fusion results, when they have similar $CMY_{Aux.}$ (MoCov3，DINO)

Motivation

南京航空航天大學 NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

Logit Predictor

To evaluate the effect of logit predictor, we empirically compare with several predictors, including **MLP**, **Cache, Cache with random initialization** (Cache-Random), and a simple **linear probing** (LP) as baseline.

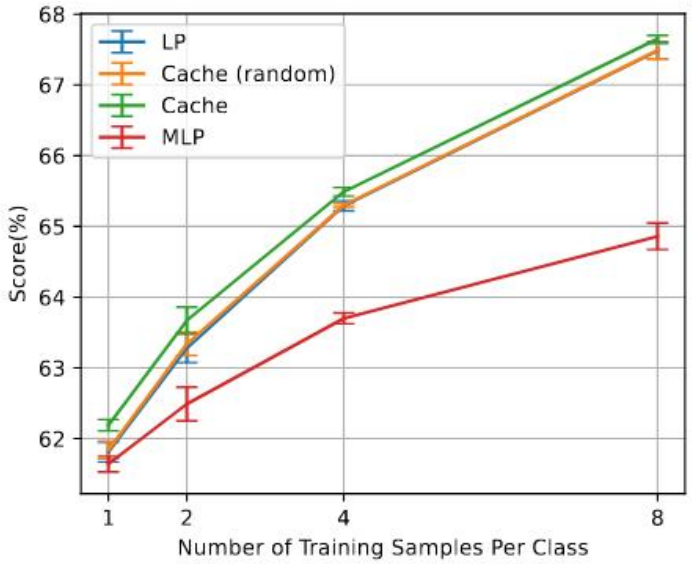


Figure 3. Results of different logit predictors on ImageNet-1K.

conclusion

1、LP achieves similar performance with Cache-Random, and both of them are clearly superior to MLP.
2、that feature initialization is helpful for the logit predictor.

Motivation

南京航空航天大學 NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

Logit Predictor

individual training of bias branch and joint training of bias branch with zero-shot CLIP

Auxiliary Features	Individual	Joint	Joint+ZO
CLIP [43]	56.93	11.23	65.34
DINO [7]	55.65	36.24	68.32
MoCov3 [10]	57.68	42.82	69.35

Table 3. Comparison (%) of two training strategies (i.e., Individual and Joint) for the bias branch on ImageNet-1K. Joint+ZO indicates the fused results of joint training bias branch with zero-shot CLIP.

the joint training strategy makes logit bias as a pure supplement to zero-shot CLIP by considering the complementarity of auxiliary features, but it cannot fully explore the superiority of auxiliary features.

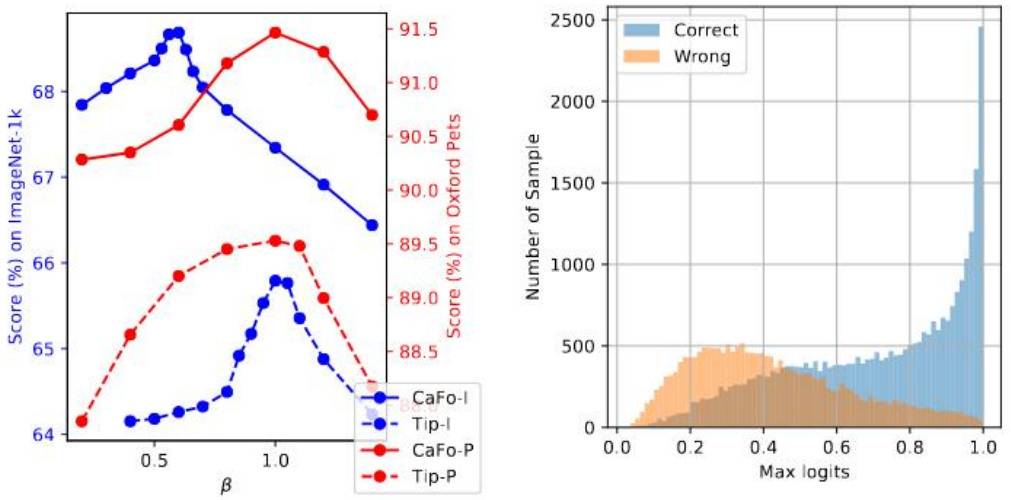
conclusion

existing logit predictors do not fully explore the superiority of auxiliary features

Motivation

Logit Fusion

To fuse logit bias with zero-shot CLIP, a manually tuned parameter β is used to control the effect of logit bias



(a) Results with various β (b) Visualization of max logits

Figure 4. (a) Results of Tip-Adapter-F and CaFo with various β on ImageNet-1K and OxfordPets. (b) Visualization of the distribution of max logits for zero-shot CLIP on ImageNet-1K.

conclusion

trade-off parameter greatly affects performance of fusion, while prediction confidence of zero-shot CLIP can be regarded as an indicator of logit fusion

Auxiliary Features f^{Aux}

Auxiliary Features $\mathbf{f}^{\mathbf{Aux}}$ According to the conclusion in Sec. 3.2.1, we can seek the optimal auxiliary features $\mathbf{f}^{\mathbf{Aux}}$ from a group of feature candidates based on the metrics of superiority and complementarity (Eq. (5)). Specifically, we employ a certain of features lying in $\Omega_S^{\text{Top-K}} \cap \Omega_C^{\text{Top-M}}$ for various downstream tasks, where $\Omega_S^{\text{Top-K}}$ and $\Omega_C^{\text{Top-M}}$ indicate the sets of features with Top-K superiority and Top-M complementarity, respectively. For efficiency, we adopt MoCov3 model with the backbone of RN50 to obtain the auxiliary features $\mathbf{f}^{\mathbf{Aux}}$ with no special declaration.

Methods

Multi-branch Training of Feature-initialized (MTFi) Logit Predictor

ically, under C-way-N-shot setting with C classes and N samples of each class, we initialize the weights $\widehat{\mathbf{W}}$ of LP by using the mean of auxiliary features from different classes:

$$\widehat{\mathbf{W}}_0 = [\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_C]^{\mathsf{T}},$$
$$\mathbf{m}_i = \frac{1}{N}\sum_{j=1}^{N} \mathbf{f}_{ij}^{\text{Aux}}, \quad i = \{1, 2, \cdots, C\}, \tag{6}$$

where $\widehat{\mathbf{W}}_0$ is the initialization of $\widehat{\mathbf{W}}$, and $\mathbf{f}_{ij}^{\text{Aux}}$ is the j-th feature of i-th class. As such, our feature-initialized LP predicts logit bias $\mathbf{s}_{\text{bias}}$ for j-th training sample as

$$\mathbf{s}_{\text{bias}}^{j} = \widehat{\mathbf{W}}\mathbf{f}_{j}^{\text{Aux}}. \tag{7}$$

Furthermore, we propose a multi-branch training strategy to fully explore the superiority of auxiliary features. Specifically, besides the original classification loss (i.e., ℓ_{Fusion}) based on the fused logit $\mathbf{s}$, we introduce an extra training branch to minimize the cross-entropy loss between logit bias $\mathbf{s}_{\text{bias}}$ and the ground-truth label of $\mathbf{y}$ as

$$\ell_{\text{Aux}} = -\sum_{j=1}^{C\times N} \mathbf{y}_j \cdot \log(g(\mathbf{s}_{\text{bias}}^{j})), \tag{8}$$

where $g(\cdot)$ is a softmax function. As such, the total loss of our multi-branch training can be formulated as:

$$\ell_{\text{total}} = (1-\lambda)\ell_{\text{Aux}} + \lambda\ell_{\text{Fusion}}, \tag{9}$$

where λ is a hyper-parameter to balance effect of ℓ_{Fusion} and ℓ_{Aux}. From Eq. (6) and Eq. (8), we can see than our pro-

1、在 C-way N-shot 设定下（即 C 类，每类 N 个样本），使用 每类辅助特征的均值 来初始化 LP 的权重矩阵
2、Logit bias：计算辅助特征贡献的 logit 偏置 s bias，s bias ， 并与 zero-shot CLIP 预测融合
3、Multi-branch Training　l Fusion负责优化最终分类目标；l Aux 让辅助特征的 logit bias 也具备分类能力

Methods

Uncertainty-based Fusion

Uncertainty-based Fusion Based on the analysis in Sec. 3.2.3, the hyper-parameter β of bias fusion is very sensitive to models and datasets. Meanwhile, such hyper-parameter is related to prediction confidence of zero-shot CLIP. Therefore, we present an uncertainty-based fusion to adaptively combine zero-shot CLIP with logit bias based on prediction confidence of zero-shot CLIP. Specifically, we introduce an uncertainty (κ) based on Kurtosis (i.e., the fourth moment) [51] to represent prediction confidence as

$$\kappa = \mathbb{E}\left[\left(\frac{\mathbf{s}_0 - \mu}{\sigma}\right)^4\right]^{\rho}, \tag{10}$$

where μ and σ are the mean and the standard deviation of $\mathbf{s}_0$, respectively. ρ is a parameter to control the power of uncertainty. As such, we can adopt κ to balance effect of logit bias. Specifically, we increase effect of logit bias for small κ; otherwise, effect of logit bias is decreased. In conclusion, our AMU-Tuning method can be formulated as

$$\mathbf{s} = \mathbf{s}_0 + \frac{\beta}{\kappa}\widehat{\mathbf{W}}\mathbf{f}^{\text{Aux}}, \tag{11}$$

where only a lightweight LP with the parameters of $\widehat{\mathbf{W}}$ is optimized by the loss ℓ_{total} (Eq. (9)).

1、引入峰度（Kurtosis）作为不确定性度量

如果峰度高，说明预测值分布集中，CLIP 预测置信度高，k大。
如果峰度低，说明预测值分布分散，CLIP 预测较不确定，k小。

2、通过不确定性调整 logit bias 的影响

当 κ 大（CLIP 预测自信），logit bias 贡献减少。
当 κ 小（CLIP 预测不确定），logit bias 贡献增加。

Experiments

Method	Score				
	1-shot	2-shot	4-shot	8-shot	16-shot
LP-CLIP [43]	22.17	31.90	41.20	49.52	56.13
CoOp [67]	57.15	57.81	59.99	61.56	62.95
CLIP-Adapter [17]	61.20	61.52	61.84	62.68	63.59
VT-CLIP [42]	60.53	61.29	62.02	62.81	63.92
Tip-Adapter-F [63]	61.32	61.69	62.52	64.00	65.51
CaFo [64]	**63.80**	**64.34**	65.64	66.86	68.79
CaFo* [64]	61.58	62.76	64.31	66.25	68.05
AMU-Tuning (Ours)	62.60	64.25	**65.92**	**68.25**	**70.02**

Table 4. Comparison (in %) of different SOTA methods on ImageNet-1K under various few-shot settings.

Dataset	Source	Target			
	IN-1K	v2	-S	-A	-R
ZS-CLIP [43]	60.33	53.27	35.44	21.65	56.00
CoOp [67]	62.95	55.40	34.67	23.06	56.60
CLIP-Adapter [17]	63.59	55.69	35.68	-	-
Tip-Adapter-F [63]	65.51	57.11	36.00	-	-
CaFo [64]	68.79	57.99	39.43	-	-
AMU-Tuning (RN50)	**70.02**	**58.64**	**40.04**	**25.65**	**57.10**
CoCoOp [66]	71.02	64.20	47.99	49.71	75.21
MaPLe [28]	70.72	64.07	49.15	50.90	76.98
AMU-Tuning (ViT)	**74.98**	**65.42**	**50.37**	**52.05**	**78.09**

Table 5. Comparison (%) of different methods under OOD setting.

a CaFo variant (namely CaFo*) by excluding use of the extra DALL-E and GPT-3.

Experiments

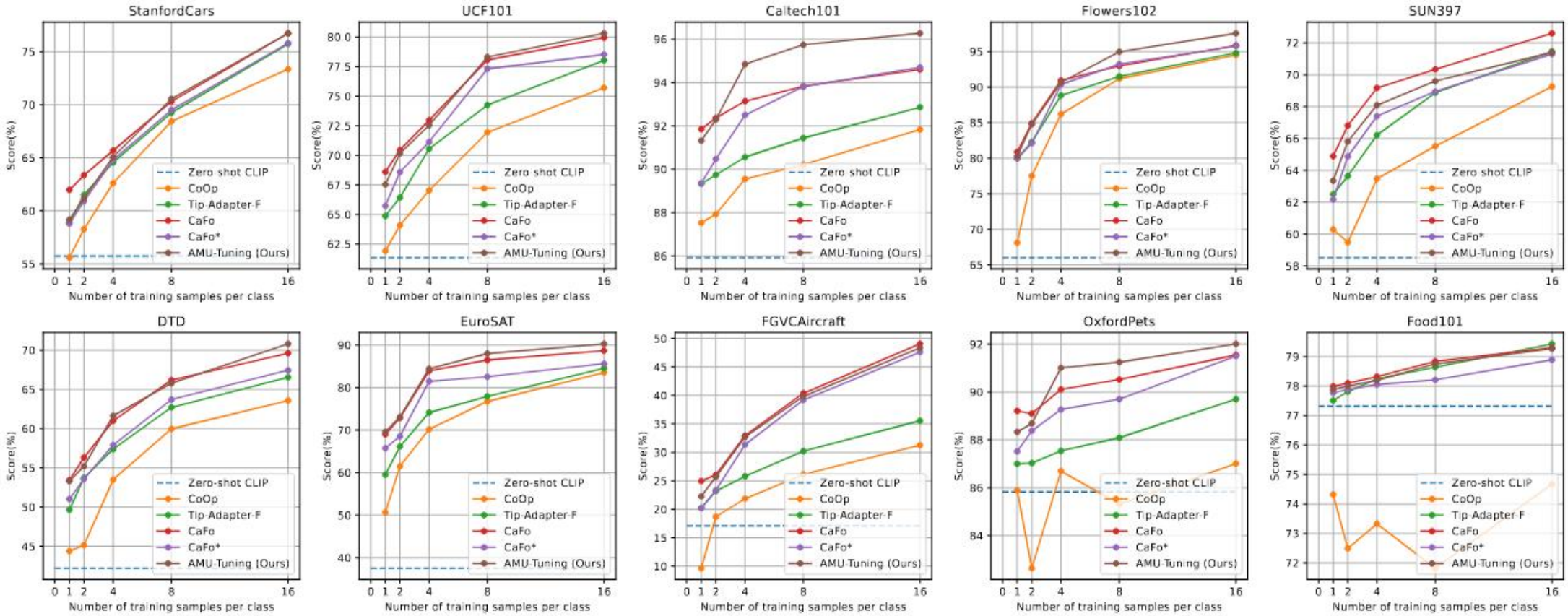


Figure 5. Comparison (in %) of different SOTA methods under various few-shot settings on ten downstream tasks.

Experiments

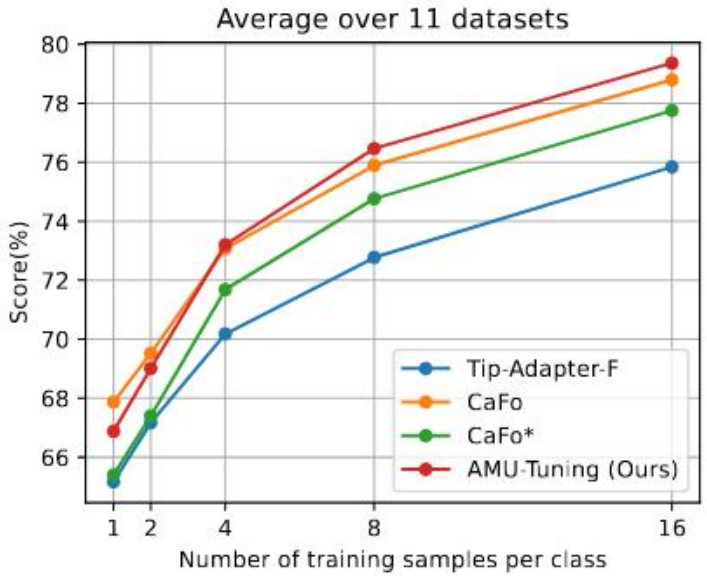


Figure 6. Results on eleven downstream tasks by average.

Models	Backbone			
	RN50	RN101	ViT-B/32	ViT-B/16
ZS-CLIP [43]	60.33	65.53	63.80	68.73
CoOp [67]	62.95	66.60	66.85	71.92
CLIP-Adapter [17]	63.59	65.39	66.19	71.13
Tip-Adapter-F [63]	65.51	68.56	68.65	73.69
CaFo [64]	68.79	70.82	70.82	74.48
CaFo* [64]	68.03	70.21	70.44	74.11
AMU-Tuning (Ours)	**70.02**	**71.58**	**71.65**	**74.98**

Table 7. Comparison (%) of SOTA methods with different visual encoders of CLIP on IN-1K with 16-shot training samples.

Ablation study

Component			Score (%)		
AUX	MTFi	UF	1-shot	4-shot	16-shot
Baseline			61.16	62.33	65.34
✓			62.15	65.31	69.35
	✓		61.83	63.16	66.17
		✓	61.70	63.08	65.90
✓	✓		62.35	65.61	69.72
✓	✓	✓	**62.60**	**65.92**	**70.02**

Table 6. Results of AMU-Tuning with various modules on IN-1K.

Thanks