microCLIP: Unsupervised CLIP Adaptation via Coarse-Fine Token Fusion for Fine-Grained Image Classification

Anonymous ACL submission

Abstract

Unsupervised adaptation of CLIP-based vision-language models (VLMs) for fine-grained image classification is challenging because pseudo-labels must be inferred from tiny local cues. CLIP's contrastive pretraining yields strong zero-shot transfer, but its coarse-grained [CLS] visual token often misses fine spatial details. Existing UA methods mainly enrich text prompts with large language model (LLM) descriptions while still relying on [CLS], which limits spatial precision. We propose **microCLIP**, a label-free self-training framework that jointly adapts visual features and LLM-derived text prototypes using fine-grained cues. First, a Saliency-Oriented Attention Pooling (SOAP) mechanism inside a lightweight TokenFusion module constructs a saliency-guided [FG] token from patch features and fuses it with the global [CLS] representation for coarse–fine alignment. Second, a two-headed LLM-derived classifier combines a frozen text head, used with multi-view CLIP features as a stable prior for pseudo-labels, with a learnable head initialized from the same LLM-derived descriptions. Finally, Dynamic Knowledge Aggregation convexly combines fixed CLIP/LLM priors with TokenFusion logits during self-training. Across 13 diverse classification benchmarks including fine-grained ones, microCLIP yields a mean improvement of $+\mathbf{2.90}\%$ points over the strongest prior UA baseline while fine-tuning only layer norms and a tiny head.

1 Introduction

CLIP's Global Objective: Foundation vision-language models (VLMs) (Jia et al., 2021; Li et al., 2021, 2022a; Singh et al., 2022; Li et al., 2022b; Xu et al., 2024) have reshaped zero-shot learning by coupling visual encoders with powerful language supervision. CLIP (Radford et al., 2021) is pretrained with a contrastive objective on image-caption pairs, aligning global image representations, typically the [CLS] token, with sentence-level text embeddings in a shared space. This training produces strong coarse-grained semantics and enables impressive zero-shot transfer for many classification tasks. Building on this, prior work improves performance either with handcrafted or learned prompts (Zhang et al., 2022; Pratt et al., 2023; Li et al., 2024; Zhou et al., 2021; Khattak et al., 2023; Lafon et al., 2024; Lin et al., 2024) or by adapting CLIP to a target domain using unlabeled data in an *Unsupervised Adaptation* (UA) setting (Huang et al., 2022; Tanwisuth et al., 2023; Mirza et al., 2023; Hu et al., 2024; Ali et al., 2025), where only class names and optional class-specific natural-language descriptions are available.

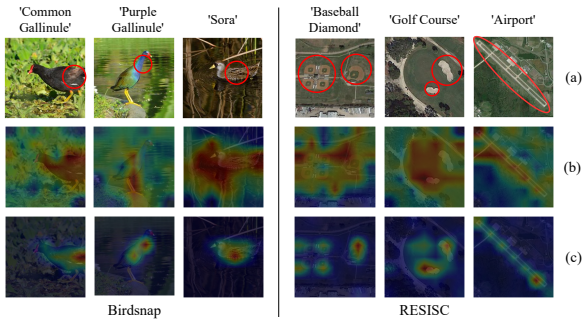


Figure 1: Attention maps on two fine-grained datasets. Row (a): input images; (b): global attention from DPA (Ali et al., 2025); (c): local attention from **microCLIP** (ours). By guiding the [FG] token with SOAP queries, microCLIP focuses on semantically critical regions, yielding sharper, more discriminative attention. Red circles highlight referenced regions in the text.

Gaps in UA Literature: Fine-grained image classification (Krause et al., 2013; Nilsback and Zisserman, 2008; Parkhi et al., 2012; Berg et al., 2014) requires distinguishing closely related categories based on subtle, localized visual cues. In the UA setting, this is particularly challenging: the model must discover such cues from unlabeled images while being guided only by language (e.g., class names or descriptions). However, existing UA

methods often underutilize this multimodal structure. For instance, while LaFTer (Mirza et al., 2023) enriches text prompts with LLM descriptions, it still relies on CLIP's coarse [CLS] token during adaptation, failing to fully align fine-grained visual and textual cues. Similarly, DPA (Ali et al., 2025) builds dual prototypes in the image and text spaces, yet also operates on the global visual representation. Consequently, both LaFTer and DPA inherit the limitation that CLIP's pretrained [CLS] token aligns well with high-level category names but poorly with fine-grained details. This often causes them to miss *local semantics*, spatially localized patterns that differentiate fine-grained categories, and results in attention maps that highlight irrelevant or diffuse regions, as illustrated in fig. 1. The training-free method WCA (Li et al., 2024) addresses this limitation by densely aligning LLM descriptions with many random crops. We adopt a similar idea as a strong text-based prior for pseudo-labeling and show that a much smaller number of crops suffices ($\times 8$ less). Moreover, we introduce explicit modeling of fine-grained visual cues to improve alignment with fine textual details.

Our Contributions: We argue that exploiting fine-grained cues in only one modality (usually the text prompts) is inherently limited for UA. Motivated by these observations and recent attention-pooling methods (Xiao et al., 2025; Zheng et al., 2024), we introduce **microCLIP**, a self-training framework that coordinates the fine-grained adaptation of CLIP using only unlabeled images, class names, and LLM-generated descriptions. The coarse [CLS] token still carries valuable global knowledge from CLIP pretraining, so rather than discarding it, we treat it as a global prior and complement it with a dedicated fine-grained visual token. At the same time, we refine a text-based classifier derived from LLM descriptions so that both modalities are adapted jointly. To our knowledge, microCLIP is the first UA method to introduce a dedicated fine-grained visual token and is jointly grounded in LLM-derived descriptions. Our contributions are threefold:

- We propose a *Saliency-Oriented Attention Pooling (SOAP)* mechanism within a lightweight *TokenFusion* module. SOAP builds a saliency query over CLIP's patch tokens to pool a compact fine-grained [FG] token; TokenFusion then fuses [FG] with CLIP's global [CLS] token for coarse-fine alignment in the shared image-text space.

- We design a *two-headed LLM-derived classifier*: a frozen classifier W_{LLM}, obtained from LLM-generated class descriptions, which provides a stable text-based prior for initializing pseudo-labels from multi-view CLIP features, and a learnable classifier W^*_{LLM}, initialized from the same descriptions and refined jointly with our TokenFusion module.

- We introduce *Dynamic Knowledge Aggregation (DKA)*, an iterative pseudo-labeling scheme that convexly combines fixed CLIP/LLM priors from multi-view alignment with TokenFusion's evolving logits, enabling stable yet progressively adaptive learning, crucial for capturing fine-grained distinctions.

We empirically show that these components reveal CLIP's latent fine-grained signals, achieving an average gain of $+\mathbf{2.90}\%$ across 13 different classification datasets with only lightweight adaptation. Moreover, our saliency-based localized attention consistently highlights class-defining local semantics (see fig. 1, bottom), for example, capturing the reddish-brown body of the 'Common Gallinule', the purple neck of the 'Purple Gallinule', and the dark feathers of the 'Sora' in Birdsnap (Berg et al., 2014), or the infield layout of the 'Baseball Diamond', sandy areas of the 'Golf Courses', and runways of the 'Airport' in RESISC (Cheng et al., 2017).

2 Related Works

Unsupervised Adaptation of CLIP: While CLIP (Radford et al., 2021) provides strong zero-shot baselines by aligning images and text in a shared space, adapting it to a new domain without labels remains challenging. Recent methods have approached this through various strategies: UPL (Huang et al., 2022) performs unsupervised prompt learning using top-K pseudo-labels; POUF (Tanwisuth et al., 2023) aligns prototypes with target data through transport-based distribution matching; LaFTer (Mirza et al., 2023) fine-tunes visual prompts with LLM-generated descriptions and unlabeled images; ReCLIP (Hu et al., 2024) jointly fine-tunes both encoders with label propagation; and DPA (Ali et al., 2025) constructs and refines dual prototypes in image and text spaces. These UA methods all rely on CLIP's

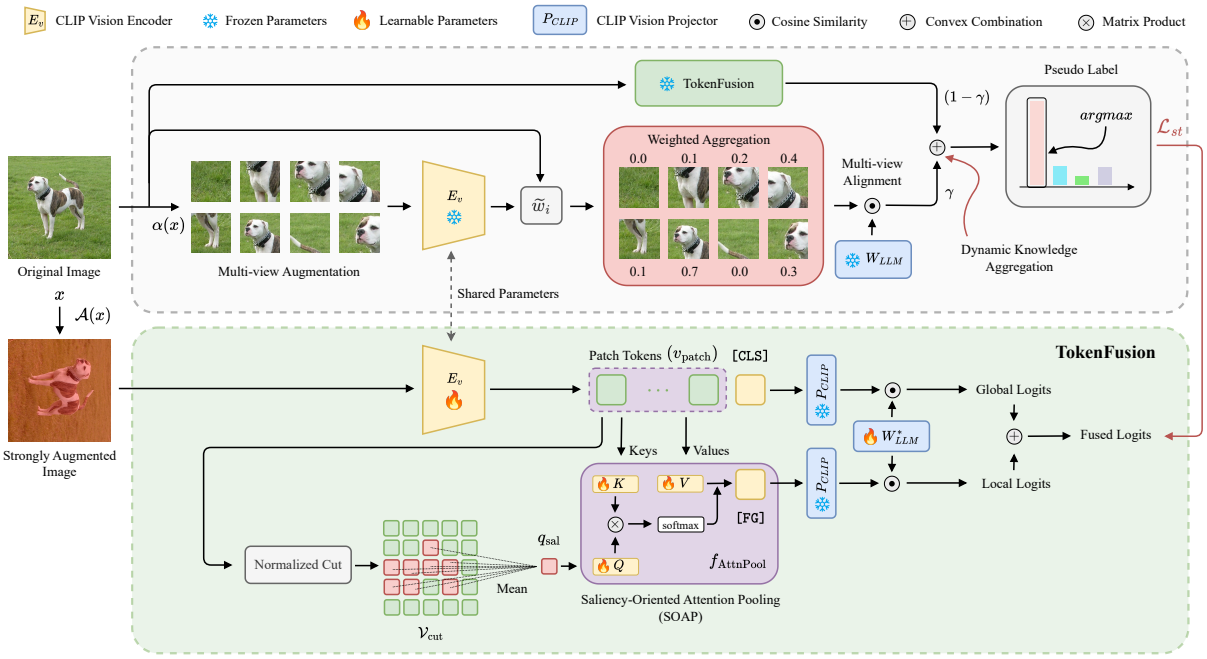


Figure 2: **Overall architecture of microCLIP.** The top shows our pseudo-labeling pipeline, where fixed knowledge from CLIP via the alignment between multi-view augmented representations and fine-grained LLM-generated descriptions is combined with dynamic knowledge learned in TokenFusion. The bottom illustrates our TokenFusion module.

coarse global [CLS] token, which lacks the granularity needed for fine-grained distinctions. In contrast, microCLIP directly addresses this by introducing a saliency-guided, fine-grained visual token and a two-headed LLM-derived classifier, fusing them through DKA for effective UA adaptation on fine-grained benchmarks.

Multi-view Representations: Multi-view features are widely used to strengthen image-text alignment. DINO-MC (Wanyan et al., 2023) uses multi-scale crops, and VCR (Lu et al., 2024) selects confident regions to build robust representations. For training-free adaptation, WCA (Li et al., 2024) aggregates many random crops with similarity-based weights. We instead view multi-view sampling as weak augmentation, showing that a small, consistent set of crops can form a stable prior. We align this prior with a frozen LLM-derived classifier and integrate it with TokenFusion logits through our DKA pseudo-labeler.

Salient Regions and Patch Tokens: Unsupervised salient region discovery localizes informative areas without pixel-level supervision. SelfMask (Shin et al., 2022) and FOUND (Siméoni et al., 2023) use self-supervised features, and TokenCut (Wang et al., 2023) applies Normalized Cut graph algorithm (NCut) (Shi and Malik, 2000) to ViT (Dosovitskiy, 2020) tokens. Prior work such as MaskCLIP (Zhou et al., 2022) and SCLIP (Wang et al., 2024) mainly targets segmentation/localization. We instead use NCut to select salient patch tokens and pool them with SOAP into a fine-grained [FG] token, designed to complement [CLS] for fine-grained unsupervised adaptation and alignment with detailed descriptions.

3 Methodology

3.1 Overall Architecture

We adapt a pretrained CLIP model, consisting of a visual encoder E_v and a text encoder E_t, to a target fine-grained classification task in a UA setting: the target domain provides only unlabeled images $\{x\}$ and a label set $\mathcal{Y}$ of class names for closed-set classification. As illustrated in fig. 2, microCLIP consists of two key components: (1) the *TokenFusion* module with *Saliency-Oriented Attention Pooling* (SOAP) to extract fine-grained visual cues and (2) an iteratively improving pseudo-labeler based on *Dynamic Knowledge Aggregation*. To induce CLIP to capture fine-grained cues, we initialize a two-headed LLM-derived classifier: a frozen classifier W_{LLM}, obtained by encoding and averaging GPT-3 descriptions (Pratt et al., 2023) for each class using the CLIP text encoder (see Supp. A), and

a learnable classifier W^*_{LLM}, initialized from the same embeddings, which is fine-tuned with TokenFusion. The text encoder E_t is used only for this initialization and is discarded afterward. During training, Dynamic Knowledge Aggregation combines predictions from a fixed CLIP/LLM branch (multi-view features with W_{LLM}) and the evolving TokenFusion branch (single-view features with W^*_{LLM}) to generate pseudo-labels. At inference, only the adapted visual encoder, SOAP, and TokenFusion with W^*_{LLM} are used on a single view.

3.2 TokenFusion Module

SOAP: While most UA methods operate solely on CLIP's global `[CLS]` token, recent work shows that the intermediate patch tokens, particularly from the penultimate layer, retain rich spatial information valuable for fine-grained reasoning (Lin et al., 2024; Xiao et al., 2025; Zheng et al., 2024). However, simply averaging these tokens dilutes class-specific evidence and degrades UA performance (Table 3). To extract a focused and informative visual representation, we introduce *Saliency-Oriented Attention Pooling* (SOAP). SOAP first applies Normalized Cut (NCut) on the patch-token similarity graph to identify a foreground set of salient tokens. This set is averaged to form a saliency query, which then drives a single-head attention pooling layer over all patch tokens. The result is a compact fine-grained `[FG]` token that emphasizes class-defining regions while remaining in CLIP's embedding space.

Formally, for an input image x, the CLIP visual encoder E_v outputs n patch tokens $v_{\text{patch}} \in \mathbb{R}^{n\times d}$ and a global token $v^{\texttt{CLS}} \in \mathbb{R}^d$:

$$[v_{\text{patch}}, v^{\texttt{CLS}}] = E_v(x),$$

Following TagCLIP (Lin et al., 2024), we take patch tokens from the penultimate transformer block by bypassing the final self-attention layer to better preserve spatial detail (see Supp. D). We represent the patch tokens as nodes in a similarity graph and apply the NCut algorithm (Shi and Malik, 2000) to select the subset $\mathcal{V}_{\text{cut}}$ corresponding to the most salient regions of the image:

$$\mathcal{V}_{\text{cut}} = \text{NCut}(v_{\text{patch}}), \tag{1}$$

Implementation details for NCut are provided in Supp. D. Since v_{patch} already encodes spatial layout via positional embeddings, we simply average the tokens in $\mathcal{V}_{\text{cut}}$ to form a saliency-aware query:

$$q_{\text{sal}} = \frac{1}{|\mathcal{V}_{\text{cut}}|} \sum_{v\in\mathcal{V}_{\text{cut}}} v. \tag{2}$$

The query q_{sal} guides a single-head attention module f_{AttnPool} to produce the fine-grained token $v^{\texttt{FG}} \in \mathbb{R}^d$:

$$v^{\texttt{FG}} = f_{\text{AttnPool}}(q_{\text{sal}}, v_{\text{patch}}). \tag{3}$$

Here, f_{AttnPool} is implemented with learnable projections W_Q, W_K, W_V; the explicit formula is standard and omitted for brevity (details in Supp. F). We append an empty token to v_{patch} allowing q_{sal} to attend to it when the saliency query and patch tokens are not well aligned (Xiao et al., 2025).

TokenFusion: To leverage both coarse and fine-grained cues, we fuse $v^{\texttt{FG}}$ from SOAP with the global token $v^{\texttt{CLS}}$ to form class predictions. Both tokens reside in CLIP's visual space, so we project them via the original CLIP projection head P_{CLIP} into the shared image–text space and compute cosine similarity $s(\cdot,\cdot)$ with the learnable classifier W^*_{LLM}.

Unlike prior UA methods (Mirza et al., 2023; Hu et al., 2024; Ali et al., 2025), which align only the global representation, TokenFusion assumes that fine-grained classification benefits from combining local and global evidence. Accordingly, it computes separate logits from $v^{\texttt{FG}}$ and $v^{\texttt{CLS}}$ before fusing them. The local and global logits are computed as shown in eq. (4) and eq. (5). The final prediction is then obtained by symmetrically fusing the two, as given in eq. (6).

$$\text{Logits}_{\text{local}} = s\big(P_{\text{CLIP}}(v^{\texttt{FG}}), W^*_{LLM}\big) \tag{4}$$

$$\text{Logits}_{\text{global}} = s\big(P_{\text{CLIP}}(v^{\texttt{CLS}}), W^*_{LLM}\big) \tag{5}$$

$$\text{TokenFusion}(x) = \tfrac{1}{2}\big(\text{Logits}_{\text{local}} + \text{Logits}_{\text{global}}\big) \tag{6}$$

We use this symmetric fusion during both training and inference so that coarse and fine-grained representations receive the same supervision and are encouraged to agree on the predicted class.

3.3 Dynamic Knowledge Aggregation:

We adopt the core insight from WCA (Li et al., 2024) that diverse local views improve image-text alignment, but we reinterpret it as a lightweight multi-view augmentation that provides *static* pre-trained knowledge. Instead of using many iterative "visual prompts" ($N \approx 60$), we treat a small

set of random crops as a weak augmentation $\alpha(x)$ and aggregate them into a single representation aligned with the frozen LLM classifier W_{LLM}. Dynamic Knowledge Aggregation (DKA) then fuses this static CLIP/LLM prior with the *dynamic* logits from our learnable TokenFusion branch.

For an unlabeled image $x \in \mathbb{R}^{H\times W\times 3}$, we sample N random local views:

$$\alpha(x) = \{x_i\}_{i=1}^{N}, \tag{7}$$

where each x_i is a random crop of x with a scale $\lambda_i \sim \mathcal{U}(a, b)$ (crop details in Supp. B). We encode each crop using the CLIP visual encoder to obtain features $f(x_i)$, which are then compared to the global image feature $f(x) = P_{\text{CLIP}}(v^{\texttt{CLS}})$. Finally, we compute a relevance weight (Li et al., 2024) for each view, as given in eq. (8).

$$w_i = \frac{\exp\big(s(f(x), f(x_i))\big)}{\sum_{l=1}^{N} \exp\big(s(f(x), f(x_l))\big)} \tag{8}$$

The multi-view feature is then a weighted average of the crop features:

$$f^{\text{agg}}(x) = \sum_{i=1}^{N} w_i \, f(x_i \mid \alpha) \tag{9}$$

Here, $f(x_i \mid \alpha)$ emphasizes that features come from the multi-crop augmentation α. This aggregated feature is aligned with the frozen LLM classifier to produce a static prior:

$$\text{Pseudo-logits}_{\text{CLIP}} = s\big(f^{\text{agg}}(x), W_{\text{LLM}} \mid \alpha\big) \tag{10}$$

Unlike WCA, which aggregates multi-crop predictions at test time, our multi-view branch is used only to provide a stable static prior during training, where both the CLIP encoder and W_{LLM} remain frozen and do not receive gradients. To progressively refine predictions, DKA blends the static logits from the multi-view branch with the dynamic logits from the TokenFusion branch, which uses the learnable classifier W^*_{LLM}. For each image x we define:

$$z^{\text{CLIP}}(y) = \text{Pseudo-logits}_{\text{CLIP}}(y), \tag{11}$$

$$z^{\text{TF}}(y) = \text{TokenFusion}(x)_y, \tag{12}$$

and obtain the pseudo-label by a convex combination, as expressed in eq. (13).

$$\hat{y} = \arg\max_{y\in\mathcal{Y}} \big[\gamma \, z^{\text{CLIP}}(y) + (1-\gamma)\, z^{\text{TF}}(y)\big] \tag{13}$$

Here, $\gamma \in [0, 1]$ controls the balance between pre-trained and newly learned knowledge. We then apply a strong augmentation $\mathcal{A}(x)$ and train the model to predict $\hat{y}$ from this view using a cross-entropy loss:

$$\mathcal{L}_{\text{st}} = -\mathbb{E}_{x\in\mathcal{X}_t} \log p^{\text{TF}}_{\hat{y}}\big(\mathcal{A}(x)\big). \tag{14}$$

In eq. (14), $p^{\text{TF}}(\mathcal{A}(x))$ is the softmax over $z^{\text{TF}}(\mathcal{A}(x))$. We further add a fairness regularization term (Li et al., 2023) to mitigate confirmation bias and class imbalance in pseudo-labels:

$$\mathcal{L}_{\text{reg}} = -\frac{1}{C}\sum_{k=1}^{C} \log \bar{p}^{(k)}_{\mathcal{A}(x)}, \tag{15}$$

where $\bar{p}^{(k)}_{\mathcal{A}(x)}$ is the average predicted probability for class k over a mini-batch. This encourages a more uniform prediction distribution across classes. The overall training objective is defined as $\mathcal{L} = \mathcal{L}_{\text{st}} + \mathcal{L}_{\text{reg}}$.

4 Experiments and Analyses

Datasets and Training Setup: We evaluate micro-CLIP on 13 diverse classification datasets covering birds, generic objects, textures, food, scenes, and actions: Birdsnap (Berg et al., 2014), Caltech (Fei-Fei et al., 2004), Cars (Krause et al., 2013), CIFAR100 (Krizhevsky and Hinton, 2009), DTD (Cimpoi et al., 2014), FGVC (Maji et al., 2013), Flowers (Nilsback and Zisserman, 2008), Food101 (Bossard et al., 2014), ImageNet (Deng et al., 2009), Pets (Parkhi et al., 2012), RESISC (Cheng et al., 2017), SUN397 (Xiao et al., 2010), and UCF101 (Soomro et al., 2012). We compare against eight strong baselines: zero-shot CLIP (Radford et al., 2021), CuPL (Pratt et al., 2023), WCA (Li et al., 2024), and five UA methods: UPL (Huang et al., 2022), POUF (Tanwisuth et al., 2023), LaFTer (Mirza et al., 2023), ReCLIP (Hu et al., 2024), and DPA (Ali et al., 2025). All methods use the ViT-B/32 CLIP backbone (Radford et al., 2021). Following standard practice in UA (Hu et al., 2024; Ali et al., 2025), we fine-tune only the layer-normalization weights of the CLIP image encoder (Ba et al., 2016) and the learnable text-classifier embeddings W^*_{LLM}, keeping the other parameters frozen. We adopt a consistent learning rate selection procedure across all methods: a rate of 1e-4 for most datasets and 1e-6 for Food101, SUN397, and ImageNet, as motivated by the analysis in Supp. B.1.

Method	Venue	Birdsnap	Caltech	Cars	CIFAR100	DTD	FGVC	Flowers	Food101	Imagenet	Pets	RESISC	SUN397	UCF101	Avg
Zero-shot / Training-free Methods															
CLIP (Radford et al., 2021)	ICML'21	37.45	90.69	58.70	64.47	44.63	19.50	66.42	83.95	63.30	87.50	57.59	61.32	61.86	61.34
CuPL (Pratt et al., 2023)	ICCV'23	37.02	94.62	60.79	65.22	50.11	20.94	69.51	84.05	64.26	87.16	61.14	65.57	66.90	63.64
WCA* (Li et al., 2024)	ICML'24	37.63	94.02	61.95	51.78	51.60	21.15	68.70	83.97	**65.01**	86.32	62.56	64.93	65.82	62.73
UA Methods															
UPL (Huang et al., 2022)	-	32.80	92.36	49.41	67.41	45.37	17.07	67.40	84.25	58.22	83.84	57.63	62.12	62.04	59.99
POUF (Tanwisuth et al., 2023)	ICML'23	38.40	94.10	57.70	62.00	46.10	18.20	67.80	82.10	52.20	87.80	66.40	60.00	61.20	61.08
LaFTer (Mirza et al., 2023)	NeurIPS'23	21.14	94.39	57.44	69.79	50.32	19.86	72.43	82.45	61.63	84.93	61.60	65.87	65.08	62.07
ReCLIP† (Hu et al., 2024)	WACV'24	37.38	93.84	58.84	71.43	53.88	18.87	72.63	84.22	63.95	85.27	73.05	65.23	67.06	64.69
DPA‡ (Ali et al., 2025)	WACV'25	31.54	**95.54**	56.83	74.22	55.96	20.10	75.48	84.76	64.64	90.11	71.11	68.13	66.69	65.78
microCLIP (Ours)	-	**38.59**	94.93	**65.81**	**77.41**	**60.00**	**22.74**	**75.84**	**85.58**	64.45	**90.24**	**77.25**	**68.98**	**70.98**	**68.68**

Table 1: Top-1 accuracy (%) comparison for 13 datasets of state-of-the-art methods using the ViT-B/32 backbone. $*$ WCA is evaluated under a matched crop budget ($N = 8$) equivalent to microCLIP. † ReCLIP (Hu et al., 2024) is reproduced under inductive settings. ‡ DPA reproduced under the same learning rate selection protocol as microCLIP.

4.1 Main Results

We report the overall accuracy across 13 different datasets in Table 1. microCLIP outperforms on average both zero-shot and UA baselines, which rely on CLIP's coarse-grained representations, using the same ViT-B/32 backbone. Compared to the strongest existing UA method, DPA, microCLIP achieves an overall accuracy of 68.68%, setting a new state-of-the-art with a 2.90% gain. Notably, our method yields substantial improvements on challenging datasets where fine-grained discrimination is crucial: FGVC (+2.64%), Cars (+8.98%), RESISC (+6.14%), UCF101 (+4.29%), CIFAR100 (+3.19%), and DTD (+4.04%). Notably, UA methods have historically struggled with the Cars dataset due to its high inter-class similarity and subtle intra-class variations; yet microCLIP outperforms the best-performing UA method on Cars (ReCLIP) by +6.97%. See Supp. B.2 for 1-2 shot comparisons and Supp. B.3 for comparisons on additional VLMs.

4.2 Ablation Studies

Naive Coarse-feature Fine-tuning Baselines: Table 2 demonstrates the importance of incorporating fine-grained cues during fine-tuning. We compare two baselines that differ only in the visual representation used for pseudo-label (PL) generation, namely single-view versus multi-view features. In both baselines, only the `[CLS]` token is aligned with a learnable classifier (W^*_{LLM}) during training. Our approach achieves a 2.40% improvement over the best-performing baseline. To ensure a fair comparison, we evaluate two pseudo-labeling setups: (1) using fixed classifier embeddings (W_{LLM}) for PL generation, and (2) a shared-classifier setting, where $W_{\text{LLM}} = W^*_{\text{LLM}}$. In both cases, the results confirm that relying solely on the `[CLS]` token leads to suboptimal performance, underscoring the necessity of the proposed `[FG]` token.

Ablation	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Fixed Classifier Embeddings for PL							
Single-view Alignment PL	61.95	53.72	21.96	72.51	89.18	68.86	61.36
Multi-view Alignment PL	63.28	55.96	21.72	72.35	88.69	69.23	61.87
Shared Learnable Classifier Embeddings for PL							
Single-view Alignment PL	56.81	59.10	16.26	72.67	89.78	70.16	60.80
Multi-view Alignment PL	56.01	**61.76**	11.31	72.31	90.24	**71.95**	60.60
microCLIP (Ours)	**65.81**	60.00	**22.74**	**75.84**	**90.24**	70.98	**64.27**

Table 2: Ablation on coarse-feature fine-tuning baselines.

Saliency-Oriented Attention Pooling: We assess SOAP's impact in Table 3. Replacing it with naive token averaging for `[FG]` leads to a 1.97% drop in average accuracy. Using the average of NCut selection only results in 60.71%, likely because averaging disregards the relative importance and saliency of the selected tokens, thereby diluting the focus on discriminative features. Since SOAP relies on a saliency-aware query, we test two weaker alternatives: (i) naive token averaging and (ii) random token selection, resulting in 1.71% and 1.39% drops, respectively. These queries fail to empha-

size semantically relevant regions, unlike NCut, which selects the most coherent and salient tokens for more discriminative attention. Figure 1 (bottom row) and fig. 3 in the supplementary materials further provide qualitative evidence supporting SOAP's effectiveness.

Ablation	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
No Attention Pooling							
Naive Token Average as `[FG]`	63.23	57.61	18.72	74.30	89.13	70.82	62.30
NCut Token Average as `[FG]`	59.21	56.54	17.49	73.24	88.36	69.42	60.71
Attention Pooling Query							
Naive Token Average	62.83	58.56	21.15	73.45	89.23	70.13	62.56
Random Token Selection	63.89	58.03	19.86	**76.17**	89.83	69.52	62.88
SOAP (Ours)	**65.81**	**60.00**	**22.74**	75.84	**90.24**	**70.98**	**64.27**

Table 3: Ablation on Attention Pooling.

Pseudo Labeler: In Table 4, we ablate Dynamic Knowledge Aggregation (DKA). When the fine-tuned classifier shares parameters with the PL head, performance drops by 3.61 and 3.11 points (single- vs. multi-view alignment), whereas keeping the PL head fixed yields smaller drops of 1.57 and 1.48 points. Removing the CLIP/LLM prior and using only TokenFusion logits ('TokenFusion logits Only', $\gamma = 0$) further reduces average accuracy to 58.03%, showing that pretrained and newly learned knowledge are both necessary. Figure 3 plots pseudo-label accuracy on Cars: the 'Multi-view Alignment Only' curve ($\gamma = 1$, CLIP/LLM branch only) and the 'TokenFusion Only' curve both saturate at lower accuracies, while DKA with $\gamma = 0.5$ attains the highest PL accuracy, illustrating the benefit of fusing static and dynamic supervision.

Ablation	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Fixed Classifier Embeddings for PL							
Single-view Alignment PL	64.81	55.80	22.47	73.69	89.32	70.08	62.69
Multi-view Alignment PL	65.10	56.76	**23.01**	73.57	88.55	69.76	62.79
Shared Learnable Classifier Embeddings for PL							
Single-view Alignment PL	60.27	57.71	15.48	72.59	88.91	68.97	60.66
Multi-view Alignment PL	59.91	59.95	16.56	72.84	88.39	69.28	61.16
TokenFusion Logits Only	55.34	54.36	10.14	70.40	89.34	68.62	58.03
DKA (Ours)	**65.81**	**60.00**	22.74	**75.84**	**90.24**	**70.98**	**64.27**

Table 4: Ablation on the pseudo-labeler.

Two-headed Classifier: To evaluate the impact of text prompt initialization for our two classifiers,

Ablation	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Handcrafted W^*_{LLM}	65.08	58.98	19.95	69.50	89.97	69.97	62.24
Both handcrafted	64.32	57.07	19.05	74.26	90.11	68.86	62.28
Both LLM-derived (Ours)	**65.81**	**60.00**	**22.74**	**75.84**	**90.24**	**70.98**	**64.27**

Table 5: Ablation on the Two-headed Classifier.

W_{LLM} and W^*_{LLM}, we conduct an ablation study on various strategies, as detailed in Table 5. We employ the same prompt ensembling technique as CLIP (Radford et al., 2021) for class-specific handcrafted prompts. Consistent with WCA's design choices (Li et al., 2024), we exclude the ablation where handcrafted prompts are used for a fixed W_{LLM}. The results in Table 5 demonstrate that our method achieves superior performance across the ablation datasets, with overall accuracy gains of 2.03% and 1.99% compared to the two ablation settings.

Ablation on Token Fusion: We conduct an ablation by removing the fusion in eq. (6), using only one of the two components. microCLIP normally averages the global `[CLS]` token logits and local patch token logits to balance coarse and fine-grained cues. We test two variants: (i) global-only and (ii) local-only. As shown in Table 6, the global-only model performs poorly (17.26%), while the local-only variant does better (57.84%), highlighting the importance of fine-grained features. Still, both fall short of our full method, confirming that combining global and local cues is crucial for robust pseudo-labeling.

Ablation	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Fixed Classifier Embeddings for PL							
Global logits only	5.71	34.41	2.19	24.20	30.93	6.13	17.26
Local logits only	60.05	52.29	21.72	60.63	86.35	65.98	57.84
Symmetric Fusion (Ours)	**65.81**	**60.00**	**22.74**	**75.84**	**90.24**	**70.98**	**64.27**

Table 6: Ablation on TokenFusion Symmetry.

Saliency-based Region Extraction with NCut: We visualize the bipartition mask produced by our NCut-based saliency mechanism in fig. 4. For visualization, we upsample and interpolate the NCut output and apply a Conditional Random Field (CRF). The NCut over patch tokens consistently highlights object-centric regions across diverse bird images, providing the saliency prior used by SOAP.

Sensitivity to γ: We ablate the knowledge-

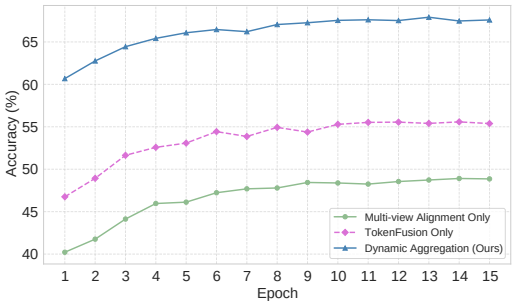


Figure 3: Pseudo-labeling accuracy of each component and Dynamic Knowledge Aggregation over time on the Stanford Cars train split.

Figure 4: NCut-based saliency masks on bird images from Birdsnap (Berg et al., 2014). Top: input images; bottom: salient regions after CRF refinement.

weighting coefficient γ of Dynamic Knowledge Aggregation on DTD (fig. 5). Accuracy is stable for moderate values and peaks at 60.00% when $\gamma = 0.5$, so we adopt this setting for all datasets.

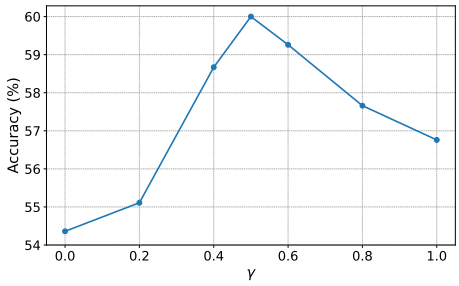
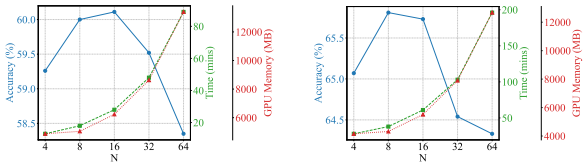


Figure 5: γ sensitivity analysis on the DTD dataset.

Numbers of Crops: We study the effect of the number of crops N on DTD and Cars (figs. 6a and 6b). On DTD, accuracy saturates around $N = 8$ (60.00% with 8 crops and 60.11% with 16), while training time and GPU usage keep increasing. On Cars, accuracy similarly peaks at 65.81% with $N = 8$. We therefore fix $N = 8$ as the default trade-off between accuracy and computational efficiency.



(a) DTD dataset. (b) Cars dataset.

Figure 6: Analysis of accuracy, training time, and GPU memory usage across varying sampled crop sizes (N).

5 Limitations and Future Directions

microCLIP provides consistent gains over strong UA baselines on most benchmarks, but is less advantageous when categories are defined by spatially diffuse cues rather than localized parts. In such cases (e.g., textures and some flower datasets), the class signal is distributed across the full image, so emphasizing a small set of salient regions can be less helpful. Concretely, our pseudo-labeler fuses coarse `[CLS]` and fine-grained `[FG]` predictions with a fixed 1:1 weight, which can introduce an unnecessary locality bias when global evidence is sufficient. This effect is most visible on DTD and Flowers, where purely global variants are already strong (see Supp. E for qualitative examples). These observations motivate future work on adaptive, instance-dependent coarse-fine fusion and saliency mechanisms that better distinguish part-centric from globally distributed concepts.

6 Conclusion

We address the unsupervised adaptation of CLIP-based VLMs to fine-grained image classification, where pseudo-labels must be inferred from subtle local cues. microCLIP augments CLIP with a saliency-guided `[FG]` token, obtained via SOAP within a lightweight TokenFusion module, and jointly adapts visual features and LLM-derived text prototypes through a two-headed classifier and Dynamic Knowledge Aggregation. This label-free framework improves the alignment between fine-grained visual regions and rich textual descriptions while updating only layer normalization parameters and a compact classification head. Empirically, microCLIP uncovers latent fine-grained cues in CLIP and achieves an average gain of $+2.90\%$ over the strongest UA baseline across 13 diverse benchmarks, providing a simple and effective recipe for adapting VLMs in the absence of target labels.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Eman Ali, Sathira Silva, and Muhammad Haris Khan. 2025. Dpa: Dual prototypes alignment for unsupervised adaptation of vision-language models. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6083–6093. IEEE.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Thomas Berg, Jiongxin Liu, Seung Woo Lee, Michelle L Alexander, David W Jacobs, and Peter N Belhumeur. 2014. Birdsnap: Large-scale fine-grained visual categorization of birds. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2011–2018.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer.

Gong Cheng, Junwei Han, and Xiaoqiang Lu. 2017. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. 2020. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. IEEE.

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Li Fei-Fei, Rob Fergus, and Pietro Perona. 2004. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE.

Priya Goyal, Quentin Duval, Jeremy Reizenstein, Matthew Leavitt, Min Xu, Benjamin Lefaudeux, Mannat Singh, Vinicius Reis, Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Ishan Misra. 2021. Vissl. https://github.com/facebookresearch/vissl.

Xuefeng Hu, Ke Zhang, Lu Xia, Albert Chen, Jiajia Luo, Yuyin Sun, Ken Wang, Nan Qiao, Xiao Zeng, Min Sun, and 1 others. 2024. Reclip: Refine contrastive language image pre-training with source free domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2994–3003.

Tony Huang, Jack Chu, and Fangyun Wei. 2022. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19113–19122.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561.

Daniel Kressner, Yuxin Ma, and Meiyue Shao. 2023. A mixed precision lobpcg algorithm. *Numerical Algorithms*, 94(4):1653–1671.

Alex Krizhevsky and Geoffrey Hinton. 2009. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario.

Marc Lafon, Elias Ramzi, Clément Rambour, Nicolas Audebert, and Nicolas Thome. 2024. Gallop: Learning global and local prompts for vision-language models. In *European Conference on Computer Vision*, pages 264–282. Springer.

Jinhao Li, Haopeng Li, Sarah Erfani, Lei Feng, James Bailey, and Feng Liu. 2024. Visual-text cross alignment: Refining the similarity score in vision-language models. In *International Conference on Machine Learning*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022a. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR.

Junnan Li, Silvio Savarese, and Steven CH Hoi. 2023. Masked unsupervised self-training for label-free image classification. *International Conference on Learning Representations*.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705.

Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. 2022b. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *International Conference on Learning Representations*.

Yuqi Lin, Minghao Chen, Kaipeng Zhang, Hengjia Li, Mingming Li, Zheng Yang, Dongqin Lv, Binbin Lin, Haifeng Liu, and Deng Cai. 2024. Tagclip: A local-to-global framework to enhance open-vocabulary multi-label classification of clip without training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 3513–3521.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Jinda Lu, Shuo Wang, Yanbin Hao, Haifeng Liu, Xiang Wang, and Meng Wang. 2024. Rethinking visual content refinement in low-shot clip adaptation. *arXiv preprint arXiv:2407.14117*.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Muhammad Jehanzeb Mirza, Leonid Karlinsky, Wei Lin, Horst Possegger, Mateusz Kozinski, Rogerio Feris, and Horst Bischof. 2023. Lafter: Label-free tuning of zero-shot classifier using language and unlabeled image collections. *Advances in Neural Information Processing Systems*, 36:5765–5777.

Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. 2012. Cats and dogs. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505.

Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. 2023. What does a platypus look like? generating customized prompts for zero-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15691–15701.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Jianbo Shi and Jitendra Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Gyungin Shin, Samuel Albanie, and Weidi Xie. 2022. Unsupervised salient object detection with spectral cluster voting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3971–3980.

Oriane Siméoni, Chloé Sekkat, Gilles Puy, Antonín Vobeckỳ, Éloi Zablocki, and Patrick Pérez. 2023. Unsupervised object localization: Observing the background to discover objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3176–3186.

Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. 2022. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15638–15650.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild. *Center for Research in Computer Vision*.

Korawat Tanwisuth, Shujian Zhang, Huangjie Zheng, Pengcheng He, and Mingyuan Zhou. 2023. Pouf: Prompt-oriented unsupervised fine-tuning for large pre-trained models. In *International Conference on Machine Learning*, pages 33816–33832. PMLR.

Feng Wang, Jieru Mei, and Alan Yuille. 2024. Sclip: Rethinking self-attention for dense vision-language inference. In *European Conference on Computer Vision*, pages 315–332. Springer.

Yangtao Wang, Xi Shen, Yuan Yuan, Yuming Du, Maomao Li, Shell Xu Hu, James L Crowley, and Dominique Vaufreydaz. 2023. Tokencut: Segmenting objects in images and videos with self-supervised transformer and normalized cut. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(12):15790–15801.

Xinye Wanyan, Sachith Seneviratne, Shuchang Shen, and Michael Kirley. 2023. Dino-mc: Self-supervised contrastive learning for remote sensing imagery with multi-sized local crops. *arXiv preprint arXiv:2303.06670*, 2(6):26.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE.

Rui Xiao, Sanghwan Kim, Mariana-Iuliana Georgescu, Zeynep Akata, and Stephan Alaniz. 2025. Flair: Vlm with fine-grained language-informed image representations. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24884–24894.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. 2023. Demystifying clip data. *arXiv preprint arXiv:2309.16671*.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. 2024. Demystifying clip data. *International Conference on Learning Representations*.

Maxime Zanella and Ismail Ben Ayed. 2024. Low-rank few-shot adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1593–1603.

Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. 2022. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European conference on computer vision*, pages 493–510. Springer.

Kecheng Zheng, Yifei Zhang, Wei Wu, Fan Lu, Shuailei Ma, Xin Jin, Wei Chen, and Yujun Shen. 2024. Dreamlip: Language-image pre-training with long captions. In *European Conference on Computer Vision*, pages 73–90. Springer.

Chong Zhou, Chen Change Loy, and Bo Dai. 2022. Extract free dense labels from clip. In *European Conference on Computer Vision*, pages 696–712. Springer.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2021. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337 – 2348.

Supplementary Material for microCLIP

This supplementary complements the main paper with implementation details, additional experiments, qualitative visualizations, and derivations. Appendix A details implementation choices and the construction of the two-headed LLM-derived classifier. Appendix B reports additional experiments and specifies the multi-crop procedure used in Dynamic Knowledge Aggregation (DKA). Appendix C provides qualitative attention visualizations. Appendix D derives the Normalized Cut operator used in SOAP. Appendix E discusses limitations and future directions. Appendix F provides the SOAP attention pooling formula, pseudocode, and a consolidated notation table.

A Additional Implementation and Technical Details

A.1 Two-headed LLM-derived Classifier

CuPL descriptions and initialization. For all experiments that use LLM-derived classifiers, the class-specific descriptions used to construct W_{LLM} and W^*_{LLM} are sourced from CuPL (Pratt et al., 2023). CuPL generates descriptions using two configurations: *base* (three generic handcrafted templates) and *full* (dataset-specific prompts). Following CuPL, we adopt the *full* configuration since it produces stronger descriptions and improves downstream zero-shot performance.

Given a set of class names $\mathcal{Y}$, we initialize each class prototype by averaging CLIP text-encoder embeddings over the M CuPL descriptions for that class. We form two classifier heads: (i) a **frozen** head W_{LLM} used by the multi-view alignment component in the pseudo-labeler, and (ii) a **learnable** head W^*_{LLM} used inside TokenFusion. Figure 7 illustrates this initialization. After this step, the CLIP text encoder E_t is discarded and not used during training or inference.

In Table 7, we additionally report results for DPA using GPT-3-generated descriptions as text prototypes, to compare fairly against microCLIP and other related methods under the same description source.

Regenerating descriptions with GPT-4o. CuPL uses GPT-3 to generate descriptions. As an ablation, we regenerate descriptions following CuPL's *full* configuration using GPT-4o (Achiam et al., 2023). In addition to CuPL's original prompt templates, we use the following system prompt: "You are a helpful assistant. Give 10 numbered sentences answering the prompt as visually identifiable descriptions." We further append "Include '{CLASS}' in each sentence." to each prompt to ensure that the target class name appears in every description.

Figure 7: Initialization of the LLM-derived classifiers W_{LLM} (frozen) and W^*_{LLM} (learnable).

Table 8 shows that richer descriptions can improve performance for both microCLIP and related methods. Under GPT-4o descriptions, microCLIP improves its average accuracy compared to GPT-3 descriptions, with the largest gains appearing on fine-grained and appearance-sensitive datasets. DPA remains slightly higher on DTD and Flowers in this setting, consistent with the observation that globally distributed cues can be especially helpful for categories that span most of the image. We discuss this behavior further in appendix E.

Method	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Zero-shot	60.79	50.11	20.94	69.51	61.14	66.90	54.90
WCA	61.95	51.60	21.15	68.70	86.32	65.82	59.26
LaFTer	57.44	50.32	19.86	72.43	84.93	65.08	58.34
DPA	57.32	58.60	22.08	**77.71**	90.06	68.38	62.36
microCLIP	**65.81**	**60.00**	**22.74**	75.84	**90.24**	**70.98**	**64.27**

Table 7: Top-1 accuracy (%) using GPT-3 descriptions (CuPL).

Method	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Zero-shot	58.33	52.39	21.66	72.63	88.55	65.42	59.83
WCA	60.97	55.37	22.80	72.60	89.38	64.73	60.98
LaFTer	49.59	50.90	19.05	72.72	85.17	65.90	57.22
DPA	57.64	**59.26**	22.23	**84.57**	90.11	67.78	63.60
microCLIP	**64.30**	59.20	**24.03**	83.56	**90.68**	**70.00**	**65.30**

Table 8: Top-1 accuracy (%) using GPT-4o descriptions under the CuPL full prompting setup.

A.2 Other Implementation Details

Backbone choice. Unless otherwise specified, we use CLIP (Radford et al., 2021) with a ViT-B/32 image encoder. Our method relies on transformer patch tokens and thus targets ViT-based CLIP backbones. Extending SOAP to ResNet-based CLIP variants would require a different tokenization and pooling design.

Augmentations. All images are resized and normalized following CLIP preprocessing, with a final resolution of 224×224. For self-training, we use a strong augmentation $\mathcal{A}(\cdot)$ that includes RandomResizedCrop, HorizontalFlip, and RandAugment (Cubuk et al., 2020). For the single-view TokenFusion branch, we use CenterCrop as the weak view. For DKA's multi-view alignment branch, we use N random crops $\alpha(\cdot)$ as detailed in appendix B.

Optimization and training. We use AdamW (Loshchilov and Hutter, 2017) with a cosine learning-rate schedule, batch size 64, and 15 epochs. The default learning rate is 10^{-4}, except for Food101, SUN397, and ImageNet where we use 10^{-6} for stability. All experiments are conducted on a single NVIDIA A100-SXM4-40GB GPU.

Hyperparameter selection protocol. Some prior work tunes separate learning rates per dataset. To reduce dataset-specific over-optimization, we adopt the ReCLIP-style protocol and tune hyperparameters on a single dataset, then reuse them across benchmarks. appendix B.1 reports the learning-rate sensitivity used for this selection.

Baselines, splits, and dataset subset for ablations. We reproduce prior methods using their official codebases when available. We adopt the dataset splits defined by VISSL (Goyal et al., 2021) to standardize evaluation. For extensive ablations, we follow ReCLIP (Hu et al., 2024) and use 6 out of the 13 datasets to balance diversity and computational feasibility. Table 9 summarizes dataset statistics and the number of LLM descriptions used per class.

Dataset	Desc/Class	Classes	Train	Test
Birdsnap	30	500	31,900	7,977
Caltech101	30	100	4,403	6,645
Stanford Cars	90	196	8,144	8,041
CIFAR100	40	100	50,000	10,000
DTD	60	47	3,760	1,880
FGVC	20	102	3,334	3,333
Flowers102	20	102	4,093	2,463
Food101	30	101	75,750	25,250
ImageNet-1K	50	1000	50,000	50,000
Oxford Pets	20	37	3,680	3,669
RESISC45	50	45	25,200	6,300
SUN397	30	397	76,129	21,758
UCF101	50	101	9,537	3,783

Table 9: Dataset statistics and the number of LLM-generated descriptions per class used to form text prototypes.

B Additional Experiments

Multi-crop augmentation used in DKA. The main paper defines $\alpha(x)$ as a set of N random crops and samples the crop scale via $\lambda_i \sim \mathcal{U}(a, b)$. Concretely, for an image $x \in \mathbb{R}^{H\times W\times 3}$, we generate N square crops. For crop i, we sample $\lambda_i \sim \mathcal{U}(a, b)$ and take a square window of side length $\lambda_i \cdot \min(H, W)$, with the crop location sampled uniformly among valid windows. Each crop is then resized to 224×224 and passed through the CLIP image encoder. In all experiments, we use $N = 8$ crops per image. The values of a and b should be set consistently across methods; in our implementation we keep them fixed for all benchmarks.

This section reports additional experiments that further characterize microCLIP. appendix B.1 analyzes sensitivity to the learning rate. appendix B.2 compares microCLIP against 1-shot and 2-shot adaptation methods. appendix B.3 evaluates microCLIP with alternative pretrained VLM backbones.

B.1 Sensitivity to Learning Rate Selection

Following ReCLIP (Hu et al., 2024), we tune the learning rate on a single dataset, DTD (Cimpoi et al., 2014), and then apply the selected value across all 13 benchmarks to avoid dataset-specific overfitting. As shown in fig. 8, a learning rate of 10^{-4} achieves the best accuracy on DTD and is used as the default. For datasets with many classes and higher visual diversity, namely Food101, SUN397, and ImageNet, we use 10^{-6} to improve stability. We apply the same tuning protocol and search space to all compared methods.

B.2 Comparison with Few-Shot Methods

Table 10 compares microCLIP, which operates without labeled target samples, against recent few-shot adaptation methods CoOp (Zhou et al., 2021), MaPLe (Khattak et al., 2023), and CLIP-

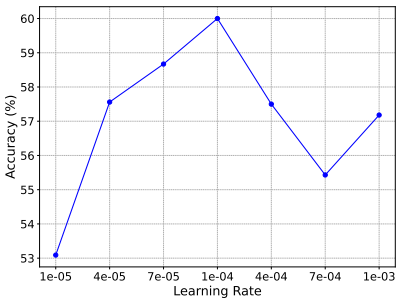


Figure 8: Learning-rate selection on DTD.

LoRA (Zanella and Ben Ayed, 2024) under 1-shot and 2-shot settings. Despite using no target labels, microCLIP outperforms these few-shot baselines on most datasets and achieves the best average accuracy.

Method	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
1-shot							
CoOp	57.70	44.40	19.60	67.10	86.90	68.00	57.28
MaPLe	57.50	28.60	13.30	64.10	89.40	65.50	53.07
CLIP-LoRA	51.51	19.17	24.09	77.75	32.25	17.54	37.05
2-shot							
CoOp	62.80	48.40	22.40	75.40	88.60	**71.40**	61.50
MaPLe	61.30	48.10	21.20	66.80	83.70	65.80	57.82
CLIP-LoRA	55.12	30.61	**24.69**	**84.94**	49.86	34.43	46.61
microCLIP	**65.73**	**59.31**	22.74	75.07	**89.56**	70.82	**63.17**

Table 10: Few-shot comparison using VISSL (Goyal et al., 2021) splits.

B.3 Comparison with other VLMs

ViT-B/16. Using ViT-B/16 as the CLIP backbone, microCLIP improves over prior approaches in Table 11. We attribute these gains to the smaller patch size of ViT-B/16, which provides more detailed patch tokens for SOAP.

Method	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
Zero-shot Methods							
CLIP (Radford et al., 2021)	64.70	44.70	23.97	70.89	89.00	69.10	60.39
CuPL (Pratt et al., 2023)	64.92	53.46	27.72	73.37	90.71	69.42	63.27
Unsupervised Adaptation Methods							
UPL (Huang et al., 2022)	60.33	45.90	22.53	73.93	87.98	67.43	59.68
POUF (Tanwisuth et al., 2023)	63.50	48.60	24.40	72.10	91.80	71.50	61.98
LaFTer (Mirza et al., 2023)	64.72	54.79	22.38	75.15	85.28	67.20	61.59
DPA (Ali et al., 2025)	63.97	50.32	20.10	78.64	93.35	74.44	63.47
microCLIP	**72.50**	**60.74**	**31.29**	**79.86**	**93.43**	**75.18**	**68.83**

Table 11: Top-1 accuracy (%) using ViT-B/16.

MetaCLIP. Table 12 reports results when applying microCLIP to MetaCLIP (Xu et al., 2023) ViT-B/32 models trained on 400M and 2.5B image-text pairs. We compare against the corresponding zero-shot baseline and DPA (Ali et al., 2025). Across most datasets, microCLIP improves over zero-shot and remains competitive with DPA, highlighting that SOAP and TokenFusion transfer beyond CLIP.

Method	Cars	DTD	FGVC	Flowers	Pets	UCF101	Avg
MetaCLIP (ViT-B/32) 400M							
Zero-shot	68.23	60.69	28.20	69.91	87.90	64.10	63.17
DPA	69.40	56.90	**30.87**	**76.86**	89.80	**72.10**	65.99
microCLIP	**74.93**	**66.60**	30.12	75.52	**90.62**	71.56	**68.23**
MetaCLIP (ViT-B/32) 2.5B							
Zero-shot	69.60	60.96	29.79	69.47	88.50	65.40	63.95
DPA	76.00	61.86	30.48	75.56	**91.50**	**76.90**	68.72
microCLIP	**80.66**	**65.37**	**31.71**	**76.82**	90.73	72.54	**69.64**

Table 12: Top-1 accuracy (%) using MetaCLIP.

C Qualitative Analysis

Figure 9 visualizes how patch tokens are attended by the pretrained [CLS] token (global attention) and by the [FG] token produced by SOAP (local attention). For [CLS], we visualize its attention weights to patch tokens from the last transformer block. For [FG], we visualize the SOAP attention pooling weights over patch tokens. These examples highlight that [FG] complements [CLS] by emphasizing fine-grained, class-discriminative regions.

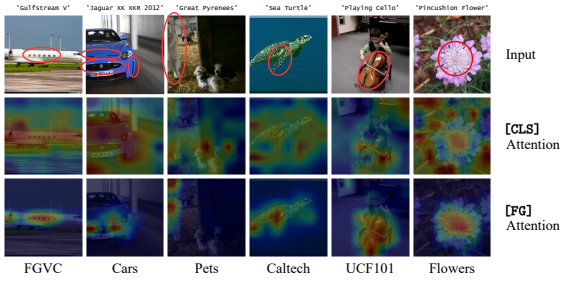


Figure 9: Attention visualizations in microCLIP. Best viewed zoomed in.

FGVC (Gulfstream V). [CLS] focuses on the fuselage and wing, capturing overall shape but missing subtle cues. [FG] concentrates on the row of circular windows, which is a fine-grained cue that can separate visually similar aircraft variants. **Cars (Jaguar XK XKR 2012).** [CLS] attends broadly to the front of the car. [FG] emphasizes

localized parts such as the grille badge, headlight shape, and hood vent, which are discriminative for trim-level recognition.

Pets (Great Pyrenees). `[CLS]` spreads attention across the scene, including background clutter. `[FG]` isolates the dog despite occlusion and background entanglement, emphasizing coat texture and local contours.

Caltech (Sea Turtle). `[CLS]` covers the turtle and surrounding water. `[FG]` focuses on the textured shell and flipper region, grounding the prediction in discriminative parts.

UCF101 (Playing Cello). `[CLS]` attends to the person and scene context. `[FG]` emphasizes the cello body and bow, where the key action-object interaction occurs.

Flowers (Pincushion Flower). `[CLS]` highlights the full flower region. `[FG]` concentrates on the central cluster of florets, which is important for species-level distinctions.

SOAP also behaves sensibly when multiple objects or complex backgrounds are present. For example, in the UCF101 case, `[FG]` suppresses background clutter while highlighting the instrument. On scene-centric datasets such as SUN397 and RESISC45, NCut often assigns high saliency to dominant class-relevant structures rather than a single isolated instance. This qualitative behavior aligns with the consistent gains over global-only baselines reported in the main results.

D Normalized Cut Algorithm

SOAP operates on attention-bypassed patch tokens v_{patch} returned by the CLIP vision encoder and applies a Normalized Cut (NCut) operator on their similarity graph. Following TagCLIP (Lin et al., 2024), we bypass the last self-attention layer and keep the value pathway of the final transformer block, yielding patch tokens that preserve spatial detail while remaining in CLIP's embedding space. In the main paper this step is written compactly as $\text{NCut}(v_{\text{patch}})$; here we expand its definition and summarize the derivation from TokenCut (Wang et al., 2023).

D.1 Derivation

Consider a weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where each node corresponds to a patch token and edge weights encode pairwise affinities. A partition of $\mathcal{V}$ into two disjoint sets $\mathcal{A}$ and $\mathcal{B}$ satisfies $\mathcal{A} \cup \mathcal{B} = \mathcal{V}$ and $\mathcal{A} \cap \mathcal{B} = \varnothing$. The cut between $\mathcal{A}$ and $\mathcal{B}$ is:

$$\text{Cut}(\mathcal{A}, \mathcal{B}) = \sum_{u \in \mathcal{A}, v \in \mathcal{B}} w(u, v). \tag{16}$$

Let $\mathbf{A}$ be the affinity matrix, where $\mathbf{A}_{i,j}$ is the edge weight between nodes i and j. Let $\mathbf{D}$ be the diagonal degree matrix with $\mathbf{D}_{i,i} = \sum_j \mathbf{A}_{i,j}$. Normalized Cut (Shi and Malik, 2000) minimizes:

$$\frac{\text{Cut}(\mathcal{A}, \mathcal{B})}{\text{assoc}(\mathcal{A}, \mathcal{V})} + \frac{\text{Cut}(\mathcal{A}, \mathcal{B})}{\text{assoc}(\mathcal{B}, \mathcal{V})}, \tag{17}$$

where $\text{assoc}(\mathcal{A}, \mathcal{V})$ is the total affinity between nodes in $\mathcal{A}$ and all nodes in the graph.

With a standard relaxation, this becomes the Rayleigh quotient:

$$\min_{\mathbf{y}} \frac{\mathbf{y}^T(\mathbf{D} - \mathbf{A})\mathbf{y}}{\mathbf{y}^T\mathbf{D}\mathbf{y}} \quad \text{s.t.} \quad \mathbf{y}^T\mathbf{D}\mathbf{1} = 0. \tag{18}$$

By substituting $\mathbf{z} = \mathbf{D}^{\frac{1}{2}}\mathbf{y}$, we obtain the eigenvalue problem for the normalized Laplacian:

$$\min_{\mathbf{z}} \frac{\mathbf{z}^T\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-\frac{1}{2}}\mathbf{z}}{\mathbf{z}^T\mathbf{z}}. \tag{19}$$

The minimizer is the second smallest eigenvector (the Fiedler vector). TokenCut (Wang et al., 2023) thresholds this vector to obtain the most salient region, which we use to form the saliency-oriented query q_{sal}.

D.2 Computational Cost

Let n be the number of patch tokens (for ViT-B/16, $n = 14 \times 14 = 196$). Forming a dense affinity matrix costs $O(n^2)$ time and memory. We compute the Fiedler vector using an iterative eigensolver (for example LOBPCG (Kressner et al., 2023)); for dense matrices, each iteration is $O(n^2)$. Since n is small for CLIP ViT backbones, NCut introduces negligible overhead compared to a CLIP forward pass in practice. If desired, the affinity can be sparsified (for example via a kNN graph) to reduce per-iteration cost.

E Limitations and Future Directions

While microCLIP performs strongly across diverse targets, it can be limited when the target dataset requires a careful balance between local and global evidence during fine-tuning. Our pseudo-labeler depends on the current model to produce reliable self-labels. When categories are defined by coarse,

(a) DTD (b) Flowers

Figure 10: Examples from DTD and Flowers. In these datasets, categories often span most of the image and exhibit spatially diffuse features. Such cases can favor global representations such as the pretrained [CLS] token.

spatially diffuse cues (for example DTD or Flowers; see fig. 10), symmetric fusion between fine-grained and coarse-grained logits can introduce localized bias that is less aligned with the task.

This effect is reflected by the fact that DPA becomes competitive on DTD and Flowers when equipped with an LLM-derived classifier, even though microCLIP improves over the standard DPA setting in the main results. These observations motivate future work on an adaptive fusion strategy, for example learning a per-dataset or per-sample weighting between coarse and fine-grained predictions.

F Pseudocode and Notation

F.1 SOAP Attention Pooling Formula

Given patch tokens $v_{\text{patch}} \in \mathbb{R}^{n \times d}$ and a saliency-oriented query $q_{\text{sal}} \in \mathbb{R}^{1 \times d}$, SOAP produces a fine-grained token via single-query attention pooling:

$$v^{\texttt{FG}} = \text{softmax}\left(\frac{(q_{\text{sal}} W_Q)(v^{+}_{\text{patch}} W_K)^\top}{\sqrt{d}}\right)(v^{+}_{\text{patch}} W_V), \tag{20}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d \times d}$ are learnable projections and v^{+}_{patch} denotes the patch-token sequence with one appended empty token (a zero vector) to allow q_{sal} to attend to a null option when saliency and class knowledge are weakly aligned.

F.2 Training Procedure

Please refer to algorithm 1.

F.3 Notation Summary

Please refer to appendix F.3.

Symbol	Description
Abbreviations	
VLM	Vision-language model
LLM	Large language model
UA	Unsupervised adaptation
SOAP	Saliency-oriented attention pooling
Symbols	
E_v	CLIP vision encoder
E_t	CLIP text encoder (used only for initialization)
$\mathcal{X}_t$	Unlabeled target image set
$\mathcal{Y}, C$	Class-name set and number of classes
$v_{\text{patch}} \in \mathbb{R}^{n \times d}$	Patch tokens (attention-bypassed tokens for SOAP input)
n	Number of patch tokens
N	Number of random crops used in $\alpha(\cdot)$
v^{CLS}	CLS token (global representation)
v^{FG}	FG token (SOAP pooled fine-grained representation)
q_{sal}	Saliency-oriented query formed from NCut-selected tokens
W_{LLM}	Frozen LLM-derived classifier used in multi-view alignment
W^*_{LLM}	Learnable LLM-derived classifier used in TokenFusion
W_Q, W_K, W_V	SOAP attention projections
P_{CLIP}	Frozen vision-to-text projection
γ	Weight for pseudo-logits vs TokenFusion logits in DKA
$\alpha(\cdot)$	Multi-crop augmentation operator for DKA
$\mathcal{A}(\cdot)$	Strong augmentation operator for self-training
$\mathcal{L}_{st}$	Self-training loss
$\mathcal{L}_{reg}$	Prediction-entropy regularizer (uniform prior)

Algorithm 1 microCLIP self-training

Require: CLIP vision encoder E_v^{Θ} where Θ are the LayerNorm affine parameters to be optimized;
Frozen vision-to-text projection P_{CLIP};
Learnable SOAP parameters W_Q, W_K, W_V;
Unlabeled target images $\mathcal{X}_t = \{x_i\}_{i=1}^{T}$;
LLM $h(\cdot)$; class names $\mathcal{Y}$ with $C = |\mathcal{Y}|$;
Multi-crop augmentation $\alpha(\cdot)$; strong augmentation $\mathcal{A}(\cdot)$;
Similarity $s(\cdot,\cdot)$; knowledge weight γ;
Epochs `MaxEpochs`; batch size `B`.

function INITCLASSIFIERS($E_t, \mathcal{Y}, h$)
  Initialize $W \in \mathbb{R}^{C\times d}$
  for each $y \in \mathcal{Y}$ **do**
    $\{t_{y,1}, \ldots, t_{y,M}\} \leftarrow h(y)$
    $W_y \leftarrow \frac{1}{M}\sum_{m=1}^{M} E_t(t_{y,m})$
  $W_{\text{LLM}} \leftarrow \text{StopGrad}(W)$
  $W^*_{\text{LLM}} \leftarrow \text{ParamInit}(W)$
  return $W_{\text{LLM}}, W^*_{\text{LLM}}$
$W_{\text{LLM}}, W^*_{\text{LLM}} \leftarrow$ INITCLASSIFIERS($E_t, \mathcal{Y}, h$)
Discard E_t
function ATTNPOOL($q_{\text{sal}}, v_{\text{patch}}$)
  $v^+_{\text{patch}} \leftarrow [v_{\text{patch}}; \mathbf{0}]$
  $A \leftarrow \text{softmax}\left(\frac{(q_{\text{sal}}W_Q)(v^+_{\text{patch}}W_K)^\top}{\sqrt{d}}\right)$
  return $A(v^+_{\text{patch}}W_V)$
function TOKENFUSION(x, W^*_{LLM})
  $[v_{\text{patch}}, v^{\texttt{CLS}}] \leftarrow E_v(x)$
  $\mathcal{V}_{\text{cut}} \leftarrow \text{NCut}(v_{\text{patch}})$
  $q_{\text{sal}} \leftarrow \frac{1}{|\mathcal{V}_{\text{cut}}|}\sum_{v\in\mathcal{V}_{\text{cut}}} v$
  $v^{\texttt{FG}} \leftarrow$ ATTNPOOL($q_{\text{sal}}, v_{\text{patch}}$)
  $\text{logits}_{\text{local}} \leftarrow s(P_{\text{CLIP}}(v^{\texttt{FG}}), W^*_{\text{LLM}})$
  $\text{logits}_{\text{global}} \leftarrow s(P_{\text{CLIP}}(v^{\texttt{CLS}}), W^*_{\text{LLM}})$
  return $\frac{\text{logits}_{\text{local}}+\text{logits}_{\text{global}}}{2}$
function MULTIVIEWALIGNMENT(x, W_{LLM})
  $[\_, v^{\texttt{CLS}}] \leftarrow E_v(x)$; $f(x) \leftarrow P_{\text{CLIP}}(v^{\texttt{CLS}})$
  $\{x_1, \ldots, x_N\} \leftarrow \alpha(x)$
  for $i = 1$ to N **do**
    $[\_, v_i^{\texttt{CLS}}] \leftarrow E_v(x_i)$; $f(x_i) \leftarrow P_{\text{CLIP}}(v_i^{\texttt{CLS}})$
  $w_i \leftarrow \frac{\exp(s(f(x),f(x_i)))}{\sum_{\ell=1}^{N}\exp(s(f(x),f(x_\ell)))}$
  $f^{\text{agg}}(x) \leftarrow \sum_{i=1}^{N} w_i f(x_i)$
  return $s(f^{\text{agg}}(x), W_{\text{LLM}})$
for `epoch` $= 1$ to `MaxEpochs` **do**
  Sample minibatch $\mathbf{x} \subset \mathcal{X}_t$ of size `B`
  No gradient:
  plogits $\leftarrow$ MULTIVIEWALIGNMENT($\mathbf{x}, W_{\text{LLM}}$)
  tflogits $\leftarrow$ TOKENFUSION($\mathbf{x}, W^*_{\text{LLM}}$)
  $\hat{y} \leftarrow \arg\max_{y\in\mathcal{Y}} (\gamma \cdot \text{plogits} + (1-\gamma)\cdot \text{tflogits})$
  $\text{logits}_{\mathcal{A}} \leftarrow$ TOKENFUSION($\mathcal{A}(\mathbf{x}), W^*_{\text{LLM}}$)
  $\mathcal{L}_{st} \leftarrow \text{CrossEntropy}(\text{logits}_{\mathcal{A}}, \hat{y})$
  $\bar{p} \leftarrow \text{MeanBatch}(\text{softmax}(\text{logits}_{\mathcal{A}}))$
  $\mathcal{L}_{reg} \leftarrow -\frac{1}{C}\sum_{j=1}^{C}\log(\bar{p}_j)$
  $\mathcal{L} \leftarrow \mathcal{L}_{st} + \mathcal{L}_{reg}$
  Update Θ, W^*_{LLM}, W_Q, W_K, W_V by backprop on $\mathcal{L}$