CLIPood: Generalizing CLIP to Out-of-Distributions

Yang Shu [* 1] Xingzhuo Guo [* 1 2] Jialong Wu [1] Ximei Wang [3] Jianmin Wang [1] Mingsheng Long [1]

Abstract

Out-of-distribution (OOD) generalization, where the model needs to handle distribution shifts from training, is a major challenge of machine learning. Contrastive language-image pre-training (CLIP) models have shown impressive zero-shot ability, but the further adaptation of CLIP on downstream tasks undesirably degrades OOD performances. This paper aims at generalizing CLIP to out-of-distribution test data on downstream tasks. We propose CLIPood, a fine-tuning method that can adapt CLIP models to OOD situations where both domain shifts and open classes may occur on the unseen test data. To exploit the semantic relations between classes from the text modality, CLIPood introduces a new training objective, margin metric softmax (MMS), with class adaptive margins for fine-tuning. To incorporate both pre-trained zero-shot model and fine-tuned task-adaptive model, CLIPood leverages a new optimization strategy, Beta moving average (BMA), to maintain a temporal ensemble weighted by Beta distribution. Experiments on diverse datasets with different OOD scenarios show that CLIPood consistently outperforms existing generalization techniques.

1. Introduction

In a complex and changing open world, machine learning applications inevitably come across the problem of out-of-distribution (OOD) generalization (Bengio et al., 2021), which confronts new tasks with different distributions from the training situation. Even equipped with large-scale pre-trained models and carefully-designed transfer learning algorithms, OOD generalization still remains a significant challenge in the way of developing a reliable machine learn-

---
*Equal contribution [1]School of Software, BNRist, Tsinghua University. [2]Institute for Interdisciplinary Information Sciences, Tsinghua University. [3]Tencent Inc, China. E-mail: Yang Shu <shuy18@mails.tsinghua.edu.cn>. Correspondence to: Mingsheng Long <mingsheng@tsinghua.edu.cn>.
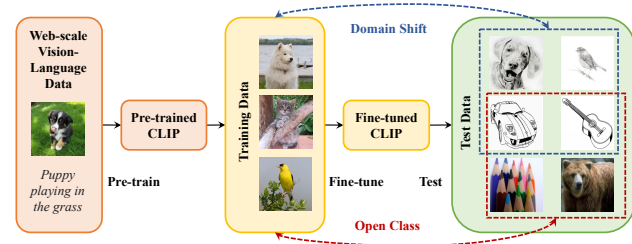
Figure 1: We adapt pre-trained CLIP models on downstream tasks with training data, while maintaining OOD generalization ability to overcome both *domain shift* and *open class*.

ing system for the open world (Taori et al., 2020; Gulrajani & Lopez-Paz, 2021; Miller et al., 2021). Instead of learning from human-labeled data, recent advances in vision-language pre-training seek to learn from naturally formed supervision of web-scale image-language pairs (Radford et al., 2021; Jia et al., 2021), which enables learning from diverse domains, and recognizing concepts from an open world. As a result, vision-language pre-trained models demonstrate impressive zero-shot learning performance and outperform models trained from only labeled images, which reveals a promising approach toward OOD generalization.

Despite the good zero-shot performance, vision-language models such as Contrastive Language-Image Pretraining (CLIP) (Radford et al., 2021) achieves OOD generalization in a task-agnostic way. In order for more satisfactory performance on downstream tasks of interest, the pre-trained models still need to utilize task-specific data to make adaptations such as fine-tuning (Agrawal et al., 2014; Girshick et al., 2014). Although fine-tuning achieves better performance than using fixed representations (Kornblith et al., 2019), for CLIP it comes at the cost of OOD generalization: the performance of fine-tuned models may be even worse than zero-shot models on related tasks with distribution shifts (Radford et al., 2021; Pham et al., 2021; Wortsman et al., 2022). These results leave OOD generalization an important yet unsolved problem for adapting CLIP models.

Motivated by the promising zero-shot performance, the stronger transfer learning performance against image-only models, and the under-explored challenge of generalization degradation during adaptation, in this paper, we explore the problem of generalizing CLIP models to out-of-distribution data in downstream tasks. As shown in Figure 1, a more

general and challenging setting is proposed, where both types of OOD situations, i.e., *domain shift* (where the training and test data come from different domains), and *open class* (where the test data contain new classes not appearing during training), may occur. We manage to solve this problem from the view of fine-tuning and seek to handle the dilemma during the adaptation of CLIP models. On the one hand, the pre-trained model should be given the flexibility to fine-tune with the downstream data thus mitigating the gap between upstream and downstream task distributions. On the other hand, since the downstream data are limited and the concrete relationship between the specific training task and the OOD task is unconstrained, the generalization property from large-scale vision-language pre-training should be exploited or maintained to enable safe model adaptation and finally boosts OOD generalization.

Based on this insight, we propose CLIPood, a simple and effective fine-tuning method to improve the OOD generalization ability of CLIP models on downstream tasks. Instead of training an additional classifier for the downstream task, we choose to conduct classification by comparing image embeddings with text embeddings generated from task prompts, which utilizes the knowledge from text modality and keeps the ability of open-class image-text alignment. From the point of the training objective, we propose Margin Metric Softmax (MMS). MMS adds an adaptive margin term for each negative class in the metric softmax loss, which is based on its distance from the positive class in the pre-trained text space. By adding such a margin, MMS explores semantic relationships from vision-language pre-training to boost the OOD generalization during fine-tuning. From the point of model optimization, we propose Beta Moving Average (BMA). Tailored to the fine-tuning trajectory of CLIP models, where the pre-trained model has good zero-shot performance and the adapted model has the knowledge of specific downstream tasks, BMA maintains a temporal ensemble for the intermediate models in the fine-tuning trajectory, and the contributions of models from different training steps are determined according to their corresponding probability in the Beta distribution.

The contributions of the paper can be summarized as:

- We aim at an under-explored problem of generalizing CLIP models to out-of-distributions, and propose a more general and challenging in-the-wild setting where both domain shift and open class occur on test data.

- We propose CLIPood, a simple and effective fine-tuning method of CLIP. Based on metric softmax fine-tuning, CLIPood proposes a new training objective Margin Metric Softmax, and a new model optimization strategy Beta Moving Average to boost OOD generalization performance on downstream tasks.

- We conduct experiments on various datasets with dif-

ferent OOD scenarios, including domain shift, open class and co-occurrence of both. Experimental results show that the proposed CLIPood method consistently outperforms existing generalization techniques.

2. Related Work

Vision-Language Pre-training. Vision-language models connect images and texts through a common embedding space to enable cross-modal learning (Frome et al., 2013; Socher et al., 2013; Elhoseiny et al., 2013). Recent advances employ architectures with better representation learning abilities such as Transformer (Vaswani et al., 2017) and web-scale training datasets and build stronger vision-language pre-trained models. One type of the approach learns the common embedding space by masked language modeling or masked region prediction (Lu et al., 2019; Tan & Bansal, 2019; Su et al., 2020; Kim et al., 2021). In this paper, we focus on another typical type of contrastive language-image pre-training such as CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021). Recent research also seeks to improve the pre-training paradigm, such as using additional supervision (Li et al., 2021; Mu et al., 2022), employing pre-trained image encoders (Zhai et al., 2022), and adding cross-modal and in-modal consistency constraints (Goel et al., 2022). In this paper, instead of designing better pre-training techniques, we aim at utilizing recent advances in vision-language pre-trained models such as CLIP and achieving better OOD performance.

Out-of-Distribution (OOD) Generalization. Research on OOD generalization aims to improve the performance of the model on new data with different distributions from the training data. One typical research topic domain generalization explores domain distribution shifts, which trains the model with source-domain data and aims at achieving high performance in unseen target domains (Khosla et al., 2012; Muandet et al., 2013). Most domain generalization methods focus on the training strategies on source domains, including cross-domain feature alignment (Li et al., 2018b), decomposing domain-specific and domain-common knowledge (Piratla et al., 2020; Chattopadhyay et al., 2020), meta-learning over domains (Li et al., 2018a; Balaji et al., 2018), designing data-augmentation tasks (Volpi et al., 2018; Carlucci et al., 2019), and weight ensemble (Cha et al., 2021). Besides domain distribution shift, recent research also explores open classes in heterogeneous domain generalization (Li et al., 2019) and open domain generalization (Shu et al., 2021). However, these settings still cannot ensure training-free generalization or can only perform open-class detection, which is limited by the closed-set property of the pre-trained models. Gulrajani & Lopez-Paz (2021) compare various methods fairly on the same benchmark and show that only focusing on algorithm design may not fully address
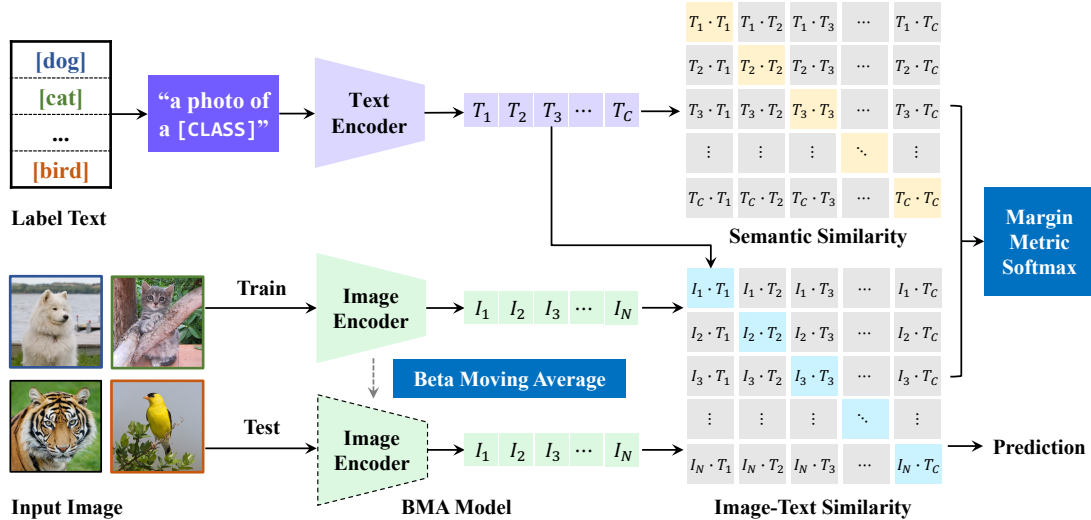
Figure 2: Overview of the proposed CLIPood method. CLIPood compares image embeddings with class text embeddings. Margin Metric Softmax is introduced to exploit semantic relationships between classes. Moreover, a Beta Moving Average model is maintained for prediction, which incorporates both the pre-trained zero-shot model and the fine-tuned model.

the OOD generalization problem.

Vision-language pre-trained models such as CLIP exhibit impressive zero-shot generalization ability to the open world, which opens a new path towards stronger OOD generalization. Despite the good zero-shot performance, research finds that further adapting CLIP models with task-specific data comes at the cost of OOD generalization ability (Radford et al., 2021; Wortsman et al., 2022). Recent advances explore improving the OOD generalization of CLIP models on the downstream tasks by adapter learning (Gao et al., 2021; Zhang et al., 2021), model ensemble (Wortsman et al., 2022), test-time adaptation (Shu et al., 2022), and prompt learning (Zhou et al., 2022b; Lu et al., 2022; Zhou et al., 2022a). Compared with most of the existing works on prompt learning or adapter learning, we focus on the aspect of fine-tuning CLIP models, which is a simple and common practice for transfer learning but an under-explored point for generalizing CLIP to out-of-distributions. Compared with some recent fine-tuning-based methods (Wortsman et al., 2022; Kumar et al., 2022; Goyal et al., 2023), we propose a novel design from the aspects of both training objectives and model optimization. We consider two types of OOD situations with domain shift and open class and propose to solve a more general and challenging problem where both these two OOD situations may appear in unseen test data.

3. Method

3.1. Generalizing CLIP to Out-of-Distributions

CLIP Models. In this paper, we focus on generalizing the vision-language pre-trained model CLIP (Radford et al.,

2021) to OOD distributions. Instead of supervision from human labels, CLIP learns directly from the raw texts about images. During the training time, CLIP jointly trains an image encoder $g_I(\cdot)$ and a text encoder $g_T(\cdot)$ in a contrastive learning way to predict the correct pairings of the image-text training samples. At the test time, the names of the target dataset's classes are embedded by the learned text encoder to synthesize a zero-shot classifier for the target task.

Problem Setup. We explore the problem of generalizing CLIP models to out-of-distributions. Given a pre-trained CLIP model, it is first adapted with some training data $\mathcal{S} = \{(\mathbf{x}, y)\}$ of the downstream task, with class label $y \in \mathcal{Y}$. The adapted model should achieve good generalization on related but *out-of-distribution* test data $\mathcal{T} = \{(\mathbf{x}', y')\}$, with the class label $y' \in \mathcal{Y}'$. We explore two OOD scenarios in this paper. In the *domain shift* scenario, we have $P(\mathbf{x}, y) \neq P(\mathbf{x}', y')$, which means the test data may come from domains with different distributions. In the *open class* scenario, we have $\mathcal{Y} \neq \mathcal{Y}'$, which means the test data may contain new classes not appearing in the training data. Instead of the controlled setting that considers two scenarios separately, we propose an even more challenging *in-the-wild* setting where both types of distribution shifts may occur simultaneously in the test data.

3.2. CLIP Fine-tuning

In this paper, we aim to design a method that further fine-tunes and adapts CLIP models to downstream tasks. In standard fine-tuning, a linear classifier $\mathbf{W} = \{\mathbf{w}_c\}_{c=1}^{C}$ is employed to fit for the new task, with each learnable parameter vector $\mathbf{w}_c$ representing a new class c. The classifier

makes predictions on the image feature $g_I(\mathbf{x})$, and outputs the probability that the sample $\mathbf{x}$ belongs to the class y as:

$$P(y|\mathbf{x}) = \frac{\exp(\mathbf{w}_y \cdot g_I(\mathbf{x}))}{\sum_{c=1}^{C} \exp(\mathbf{w}_c \cdot g_I(\mathbf{x}))}. \qquad (1)$$

Then training signals, e.g. cross-entropy losses, are used to simultaneously train the classifier $\mathbf{W}$ and fine-tune the image encoder $g_I(\cdot)$ on the downstream task.

For CLIP models, applying the standard fine-tuning strategy only transfers the knowledge in the image modality, which discards the knowledge in the text modality and breaks the connection between them. This may decrease the generalization ability benefiting from image-text alignment. Besides, the added classifier is tailored to the training dataset, making it hard to generalize to unseen classes. Therefore, we propose to perform a vision-language fine-tuning strategy on CLIP to enhance its OOD generalization ability.

Inspired by the zero-shot prediction protocol of CLIP, for each class c, we generate a text prompt $\mathbf{t}_c$ describing it, such as "a photo of a [CLASS]," where the [CLASS] token is replaced by the name of class c. We then get the text embedding of each class $\mathbf{T}_c = g_T(\mathbf{t}_c)$ extracted by the text encoder. For making predictions, we compare the image embedding $\mathbf{I_x} = g_I(\mathbf{x})$ of input image $\mathbf{x}$ with the class text embeddings. Concretely, the probability that a sample $\mathbf{x}$ belongs to class y is computed as a metric softmax:

$$P(y|\mathbf{x}) = \frac{\exp(S(\mathbf{I_x}, \mathbf{T}_y)/\tau)}{\sum_{c=1}^{C} \exp(S(\mathbf{I_x}, \mathbf{T}_c)/\tau)}, \qquad (2)$$

where $S(\cdot, \cdot)$ is a similarity metric between embeddings, and τ is a temperature. We follow the training protocol in CLIP to use the cosine similarity. We can fine-tune the CLIP model with such predictions, which utilizes knowledge in both image and text modalities and preserves the alignment between them. Different from the pre-training stage with abundant and diverse image-text pairs, in downstream scenarios with images and class text prompts, image patterns are still rich, but the diversity of text corpus is limited. Fine-tuning the text encoder would ruin the semantic relations of concepts pre-trained from a diverse world and overfit to sided knowledge of downstream training classes, which degrades the performance in open class scenarios. Thus, we fine-tune the image encoder to adapt to downstream tasks and freeze the text encoder to avoid representation collapse. To further exploit the inherent unequal relations of classes, we next introduce our proposed Margin Metric Softmax.

3.3. Margin Metric Softmax

When directly applying cross-entropy loss with the prediction in Equation (2) to update the model, it enforces that the similarity between the image embedding $\mathbf{I_x}$ of each sample
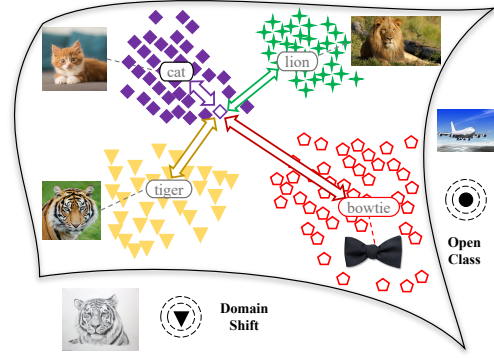

Figure 3: The illustration of Margin Metric Softmax (MMS), where the hollow diamond at the center represents the image embedding $\mathbf{I_x}$. Since $D(\mathbf{T}_y, \mathbf{T}_c)$ varies across classes, the *adaptive margin* is attained, preserving the inherent unequal relations of classes.

$\mathbf{x}$ and the text embedding $\mathbf{T}_y$ of the correct class y is higher than those $\mathbf{T}_c$ of other false classes c. This aligns the image embedding with the correct text embedding but treats all the false equally, which ignores the potential semantic relations between classes. On the other hand, the pre-trained text modality contains more detailed semantic knowledge, which quantifies the semantic relationships between texts in detail other than just discriminating between classes. Thus, we propose to explore such knowledge to enhance the generalization during vision-language fine-tuning.

For each training sample $(\mathbf{x}, y)$, we newly propose the Margin Metric Softmax (MMS) loss as:

$$\mathcal{L} = -\log \frac{\exp(S(\mathbf{I_x}, \mathbf{T}_y)/\tau)}{\sum_{c=1}^{C} \exp((S(\mathbf{I_x}, \mathbf{T}_c) + \lambda \cdot D(\mathbf{T}_y, \mathbf{T}_c))/\tau)}. \qquad (3)$$

Here, $D(\mathbf{T}_y, \mathbf{T}_c)$ represents the distance between the text embeddings of classes y and c, instantiated naturally as:

$$D(\mathbf{T}_y, \mathbf{T}_c) = 1 - S(\mathbf{T}_y, \mathbf{T}_c). \qquad (4)$$

The term $\lambda \cdot D(\mathbf{T}_y, \mathbf{T}_c)$ serves as an *adaptive margin* for each $S(\mathbf{I_x}, \mathbf{T}_c)$ in the loss. λ is a hyper-parameter that trades off the image-text similarity and the class-embedding distance. Note that $D(\mathbf{T}_y, \mathbf{T}_y) = 0$, thus these margin terms enforce that the similarity with the correct text label is higher than those with false text labels by an adaptive margin, which strengthens the image-class alignment. Different from a fixed margin for all classes, the adaptive term $D(\mathbf{T}_y, \mathbf{T}_c)$ implies that when the semantic distance between classes y and c is small, the margin term only makes a small difference, but when the semantic distance is larger (indicating that this false text label is much more different from the correct text label in semantic meanings), the margin term pushes the image embedding further way from such a false text label. In this way, MMS exploits the more

detailed knowledge of semantic relations in the pre-trained text modality to achieve a better image-text cross-modal alignment and enhance the generalization of the model during vision-language fine-tuning.

3.4. Beta Moving Average

Despite generally better performance on downstream tasks, fine-tuning pushes the model far away from the pre-trained one at the risk of catastrophic forgetting and representation collapse. This may hurt OOD generalization ability, especially considering that the CLIP pre-trained model itself is a good zero-shot learner. Additional regularization can be applied to preserve the pre-trained knowledge. However, it always needs task-specific designs and careful tuning of extra hyper-parameters, which hinders its flexibility in real-world applications. In this paper, from the perspective of model optimization, we newly propose Beta Moving Average (BMA) to maintain the benefits of both sides.

Consider a fine-tuning procedure of T training steps, we can get a trajectory of models $\{\theta_t\}_{t=0}^T$, where θ_0 is the pre-trained model and θ_t is the model at the t-th step. We aim to compute the temporal ensembling θ^{TE} of the training procedure as a weighted average of intermediate models:

$$\theta^{\text{TE}} = \sum_{t=0}^T \frac{\alpha_t}{\sum_{k=0}^T \alpha_k} \cdot \theta_t, \qquad (5)$$

where α_t determines the contribution of each model θ_t. During CLIP fine-tuning, the checkpoints near the pre-trained model (with a smaller step t) keep more knowledge from large-scale pre-training, which results in better generalization on various domains and classes in a task-agnostic manner but lacks task-specific knowledge. In contrast, the checkpoints near the fine-tuned model (with a larger step t) has been injected more task-related knowledge through training, but the generalization of such knowledge is not guaranteed due to the unknown relationship with the OOD test data. Since both sides contribute to the final OOD generalization performance, we want to strengthen the influence of the models near the two sides with a distribution prior on their weights. We propose Beta Temporal Ensemble which normalizes the training steps to $(0, 1)$ and samples from a Beta distribution $\text{Beta}(\beta, \beta)$ to determine the weight of each model as its corresponding probability in the distribution:

$$\alpha_t = \text{Beta}(\beta, \beta) \left(\frac{t + 0.5}{T + 1} \right). \qquad (6)$$

Here, β is a hyper-parameter and we choose $\beta < 1$ to focus more on pre-trained and fine-tuned models.

Directly performing the temporal model ensemble requires saving many snapshots of the model on the training trajectory, which greatly increases the storage cost. To mitigate
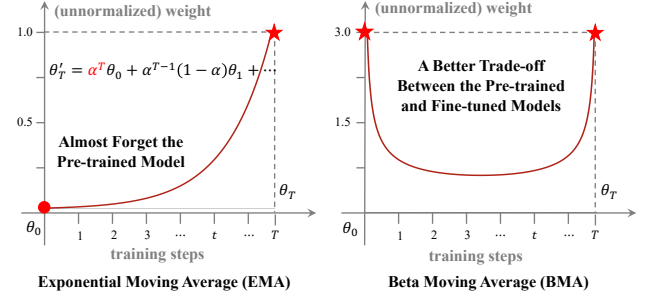

Figure 4: A comparison between Exponential Moving Average (EMA) and Beta Moving Average (BMA), in which the first term of EMA is $\alpha^T \theta_0$ over the T training step and θ_0 is the pre-trained model. Since $\alpha^T \to 0$ when $0 < \alpha < 1$, the fine-tuned model with EMA will almost forget the knowledge of the pre-trained model.

this problem, we further adjust Beta Temporal Ensemble into Beta Moving Average, which computes the average of current models on the fly. We maintain a moving average model θ^{BMA} and at each time step t, the current model θ_t is added into θ_t^{BMA} to update the moving average:

$$\theta_t^{\text{BMA}} = \frac{\sum_{k=0}^{t-1} \alpha_k}{\sum_{k=0}^t \alpha_k} \cdot \theta_{t-1}^{\text{BMA}} + \frac{\alpha_t}{\sum_{k=0}^t \alpha_k} \cdot \theta_t. \qquad (7)$$

We present a comparison between common-used Exponential Moving Average (EMA) and the proposed Beta Moving Average (BMA) in Figure 4.

The whole architecture of the CLIPood model is shown in Figure 2. In Algorithm 1, we show the overall training procedure of the proposed CLIPood method. The pre-trained model is fine-tuned with Margin Metric Softmax (MMS), and at each step of the fine-tuning, the Beta Moving Average (BMA) of the models is computed and updated on the fly. The final BMA model is stored to make OOD predictions.

Algorithm 1 Training Procedure of CLIPood

Input: Pre-trained CLIP model θ_0, learning rate η
    Initialize the BMA model $\theta_0^{\text{BMA}} \leftarrow \theta_0$
    for $t \in [1, T]$ **do**
        Sample data $\{(\mathbf{x}, y)\}$ from the training set $\mathcal{S}$
        Calculate MMS loss $\mathcal{L}$ as in Eq. (3)
        Update model parameters $\theta_t \leftarrow \theta_{t-1} - \eta \nabla_{\theta_{t-1}} \mathcal{L}$
        Calculate α_t of the current model as in Eq. (6)
        Update the BMA model θ_t^{BMA} as in Eq. (7)
    end for
Output: The final BMA model θ_T^{BMA}

---

4. Experiments

We explore two types of out-of-distributions in this paper. One is *domain shift*, where the test data follow different do-

Table 1: Accuracy on the DomainBed benchmark with domain shift.

METHOD	BACKBONE	PACS	VLCS	OFFICEHOME	TERRAINC	DOMAINNET	AVG.
ERM	RESNET	85.5	77.5	66.5	46.1	40.9	63.3
CORAL (2016)	RESNET	86.2	78.8	68.7	47.6	41.5	64.6
ZERO-SHOT	CLIP	96.2	81.7	82.0	33.4	57.5	70.2
ERM	CLIP	$96.1_{\pm 0.5}$	$83.0_{\pm 0.2}$	$83.3_{\pm 0.3}$	$\mathbf{60.9}_{\pm 0.2}$	$59.9_{\pm 0.1}$	$76.7_{\pm 0.2}$
MIRO (2022)	CLIP	95.6	82.2	82.5	54.3	54.0	73.7
DPL (2022)	CLIP	**97.3**	84.3	84.2	52.6	56.7	75.0
CLIPOOD	CLIP	$\mathbf{97.3}_{\pm 0.1}$	$\mathbf{85.0}_{\pm 0.4}$	$\mathbf{87.0}_{\pm 0.2}$	$60.4_{\pm 0.7}$	$\mathbf{63.5}_{\pm 0.1}$	$\mathbf{78.6}_{\pm 0.1}$

Table 2: Accuracy on ImageNet with various domain shifts.

METHOD	IN-DISTRIBUTION	OUT-OF-DISTRIBUTIONS				
	IMAGENET	IMAGENET-V2	IMAGENET-S	IMAGENET-A	IMAGENET-R	AVG.
ZERO-SHOT	66.7	60.8	46.1	47.8	74.0	57.2
FINE-TUNE	$68.2_{\pm 0.1}$	$61.9_{\pm 0.1}$	$46.8_{\pm 0.1}$	$46.4_{\pm 0.1}$	$75.1_{\pm 0.1}$	$57.6_{\pm 0.1}$
COOP (2022B)	71.5	64.2	48.0	49.7	75.2	59.3
COCOOP (2022A)	71.0	64.2	48.8	**50.6**	76.2	59.9
CLIPOOD	$\mathbf{71.6}_{\pm 0.1}$	$\mathbf{64.9}_{\pm 0.1}$	$\mathbf{49.3}_{\pm 0.1}$	$50.4_{\pm 0.1}$	$\mathbf{77.2}_{\pm 0.1}$	$\mathbf{60.4}_{\pm 0.1}$

main distributions from the training data. The other is *open class*, where the test data contain different classes unseen in the training data. We conduct experiments on three OOD scenarios. In the first two scenarios, we explore the two OOD types separately. In the third scenario, we newly propose to solve a more general and challenging OOD situation where both domain shift and open class appear in test data. Code is available at https://github.com/thuml/CLIPood.

Implementation Details. We use the CLIP pre-trained model with the ViT-B/16 (Dosovitskiy et al., 2021) image encoder and run experiments with half-precision (FP16) during training and inference. We keep the temperature of the softmax function the same as the pre-trained model as $\tau = 0.01$, and use the same hyper-parameter $\lambda = 0.3$ for all datasets to avoid over-tuning on specific tasks. We adopt a batch size of 36. We use the AdamW (Loshchilov & Hutter, 2019) optimizer with the cosine learning rate strategy for all datasets. By default, we set $\beta = 0.5$, use a learning rate of 5×10^{-6}, and train for 5000 iterations. For each result of CLIPood, we report the average result and the standard deviation of three runs with random seeds. More details can be found in supplementary materials.

4.1. Generalize CLIP to Domain Shift

Benchmarks. We evaluate generalization to domain shift with two benchmarks. On the first benchmark, we use five multi-domain datasets in DomainBed (Gulrajani & Lopez-Paz, 2021): PACS (Li et al., 2017), VLCS (Torralba & Efros, 2011), OfficeHome (Venkateswara et al., 2017), TerraIncognita (Beery et al., 2018) and DomainNet (Peng et al., 2019). We follow the train-validate-test split of each dataset as the DomainBed benchmark and the leave-one-out evaluation

protocol, where at each time, one domain is chosen as the test domain for evaluating OOD generalization, and other domains are chosen as the training domains.

On the second benchmark, we use ImageNet (Deng et al., 2009) as the training dataset and evaluate the performance on four variants of ImageNet with distribution shifts: ImageNet-V2 (Recht et al., 2019), ImageNet-Sketch (Wang et al., 2019), ImageNet-A (Hendrycks et al., 2021b) and ImageNet-R (Hendrycks et al., 2021a). We follow the protocol in (Zhou et al., 2022a) and randomly sample a 16-shot training set while using the original test set for evaluation.

Results. For each dataset in the DomainBed benchmark, we report the average accuracy on all test domains in Table 1. We consider the methods with the CLIP pre-trained model and the ResNet-50 (He et al., 2016) model pre-trained on ImageNet. We compare with the zero-shot performance and standard fine-tuning of the model using all training domains (ERM). For ERM with CLIP, we follow Wortsman et al. (2022) and initialize the classifier head with text embeddings to achieve competitive performance. Its results are reported from our own implementation, following the details mentioned in our paper. We also compare with the state-of-the-arts using the CLIP pre-trained model for domain generalization: MIRO (Cha et al., 2022), DPL (Zhang et al., 2022), and the best-performing method using ResNet-50 reported in DomainBed (Gulrajani & Lopez-Paz, 2021), CORAL (Sun & Saenko, 2016). We present the results reported in original papers of these methods for comparison in Table 1. For some methods such as MIRO and other methods where the original papers do not report results under our setting, we also re-implement them with our desings for a unified comparison, and these results are shown in

Table 3: Generalization performance on 11 downstream datasets with open classes.

(a) Average over 11 datasets

	BASE	NEW	H
CLIP	69.3	74.2	71.7
CoOp (2022b)	82.7	63.2	71.7
CoCoOp (2022a)	80.5	71.7	75.8
CLIPood	$83.9_{\pm 0.1}$	$74.5_{\pm 0.1}$	$78.9_{\pm 0.1}$

(b) ImageNet

	BASE	NEW	H
CLIP	72.4	68.1	70.2
CoOp (2022b)	76.5	67.9	71.9
CoCoOp (2022a)	76.0	**70.4**	73.1
CLIPood	$77.5_{\pm 0.1}$	$70.3_{\pm 0.1}$	$73.7_{\pm 0.1}$

Table 4: Accuracy on OfficeHome and DomainNet with both domain shift and open classes.

SPLIT	METHOD	OFFICEHOME				DOMAINNET					
		A	C	P	R	C	I	P	Q	R	S
BASE	CLIP	86.8	75.5	89.5	92.6	72.8	51.7	66.0	13.5	83.4	66.9
	CoOp	$87.0_{\pm 0.4}$	$78.3_{\pm 1.2}$	$92.4_{\pm 0.2}$	$91.4_{\pm 0.6}$	$75.7_{\pm 0.2}$	$58.8_{\pm 0.5}$	$68.5_{\pm 1.3}$	$13.1_{\pm 1.0}$	$84.0_{\pm 0.5}$	$70.0_{\pm 0.1}$
	CLIPood	$90.1_{\pm 0.2}$	$79.7_{\pm 0.2}$	$93.1_{\pm 0.1}$	$94.8_{\pm 0.1}$	$79.0_{\pm 0.2}$	$62.2_{\pm 0.1}$	$73.0_{\pm 0.2}$	$20.2_{\pm 0.2}$	$86.2_{\pm 0.1}$	$73.8_{\pm 0.1}$
NEW	CLIP	76.6	59.4	88.1	86.2	70.2	44.1	66.4	14.1	83.5	61.0
	CoOp	$76.5_{\pm 1.1}$	$56.6_{\pm 2.4}$	$88.0_{\pm 1.9}$	$86.8_{\pm 0.7}$	$71.5_{\pm 0.2}$	$47.2_{\pm 0.3}$	$67.3_{\pm 0.7}$	$14.8_{\pm 0.7}$	$83.7_{\pm 0.7}$	$63.1_{\pm 0.3}$
	CLIPood	$77.8_{\pm 0.2}$	$60.0_{\pm 0.2}$	$88.3_{\pm 0.1}$	$86.7_{\pm 0.1}$	$71.2_{\pm 0.1}$	$48.1_{\pm 0.1}$	$68.2_{\pm 0.2}$	$18.0_{\pm 0.4}$	$83.4_{\pm 0.1}$	$62.9_{\pm 0.1}$
TOTAL	CLIP	82.6	67.3	88.8	89.5	71.4	47.1	66.2	13.8	83.4	63.4
	CoOp	$82.7_{\pm 0.5}$	$67.2_{\pm 0.7}$	$90.2_{\pm 1.0}$	$89.2_{\pm 0.6}$	$73.4_{\pm 0.3}$	$51.8_{\pm 0.3}$	$67.9_{\pm 1.0}$	$13.7_{\pm 0.8}$	$83.9_{\pm 0.5}$	$66.0_{\pm 0.2}$
	CLIPood	$85.1_{\pm 0.1}$	$69.6_{\pm 0.2}$	$90.8_{\pm 0.1}$	$91.0_{\pm 0.1}$	$74.8_{\pm 0.1}$	$53.6_{\pm 0.1}$	$70.6_{\pm 0.1}$	$19.1_{\pm 0.3}$	$84.8_{\pm 0.1}$	$67.4_{\pm 0.1}$

Appendix B.1. CLIPood outperforms methods with ResNet pre-trained model by a large margin, which indicates that utilizing knowledge in vision-language models provides a promising way for improving OOD generalization. It also outperforms state-of-the-arts using CLIP models: MIRO and DPL, indicating CLIPood is a simple and effective method to generalize CLIP to out-of-distributions. We further report the OOD generalization results on different variants of ImageNet in Table 2. CLIPood achieves comparable performance on the in-distribution test data and outperforms the state-of-the-art methods CoOp (Zhou et al., 2022b) and CoCoOp (Zhou et al., 2022a) on the OOD datasets, which shows its generalizability on various domain shifts.

4.2. Generalize CLIP to Open Class

Benchmarks. We evaluate generalization to open classes on the benchmark covering a diverse set of recognition tasks, including general object classification: ImageNet (Deng et al., 2009) and Caltech101 (Fei-Fei et al., 2004); fine-grained classification: OxfordPets (Parkhi et al., 2012), StanfordCars (Krause et al., 2013), Flowers102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014) and FGVCAircraft (Maji et al., 2013); specific classification tasks: SUN397 (Xiao et al., 2010) for scene recognition, UCF101 (Soomro et al., 2012) for action recognition, DTD (Cimpoi et al., 2014) for texture classification and EuroSAT (Helber et al., 2019) for satellite image recognition. We follow the protocol in (Zhou et al., 2022a) and split the classes in each dataset equally into two parts, one as base classes and the other as new classes. We train the model

on base-class data and test on base classes and new classes separately to evaluate the generalization ability.

Results. The results of generalizing CLIP to open classes are shown in Table 3. We report the accuracies on base classes and new classes as well as their harmonic mean (H) to highlight the trade-off between downstream adaptation and open-class generalization. We compare CLIPood with the zero-shot prediction of CLIP and the state-of-the-art methods on this benchmark: CoOp (Zhou et al., 2022b) and CoCoOp (Zhou et al., 2022a). The detailed results on each dataset are presented in the appendix. As shown in the table, CoOp suffers from a large decrease in the accuracy of new classes after its adaptation to base classes. CoCoOp mitigates the decrease in the accuracy of new classes but sacrifices the adaptation performance on the base classes, and the gap with zero-shot performance on new classes is still large. On some datasets such as ImageNet, CoCoOp and CLIPood successfully improve the performance on unseen classes over zero-shot prediction by adapting the model with related training classes. Comparing the average results, CLIPood outperforms zero-shot prediction and existing methods by a large margin, showing that it simultaneously adapts the model to improve the performance on the downstream tasks and keeps the OOD generalization ability on open classes.

4.3. Generalize CLIP to Domain Shift and Open Class

Benchmarks. We further propose a more realistic in-the-wild setting where both domain shift and open class may appear in the test data. We choose OfficeHome and Domain-

(a) Ablation Study (b) Metric Softmax (c) Adaptive Margin (d) Beta Moving Average
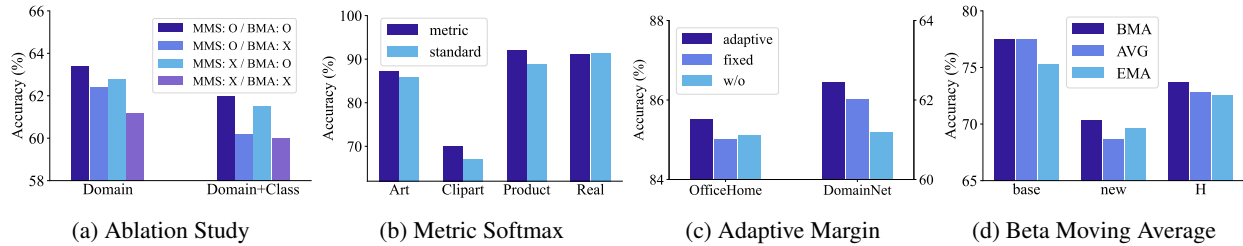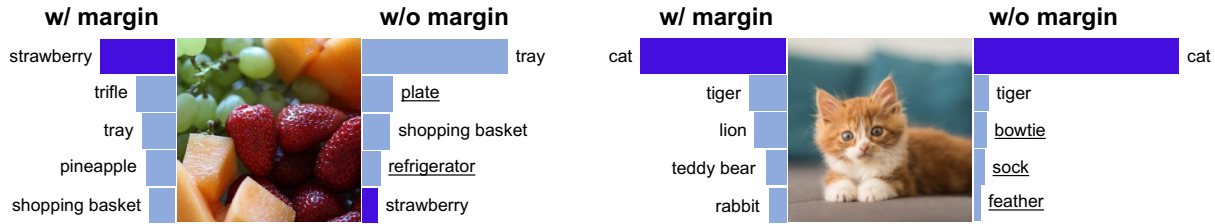
Figure 5: Analysis experiments for CLIPood.

Figure 6: Predictions from models trained with and without adaptive margin.

Net from DomainBed because they have sufficient numbers of classes for evaluating open class situations. We split the classes in each dataset into two parts, one as base classes and the other as new classes. We adopt the leave-one-domain-out protocol and train the model on base-class data in training domains and test on all test data with both base and new classes to evaluate the OOD generalization ability. We put the text embeddings of base and new classes together for the classification of all test data, which is a more realistic OOD testing protocol, since we cannot know whether the test data is from base or new classes in advance.

Results. The experimental results are shown in Table 4. We report the accuracy of each test domain. We compare with zero-shot performance (CLIP) and CoOp and do not reimplement CoCoOp because of its great memory and time cost on these datasets. CoOp improves upon zero-shot on some domains but also suffers from degradation in other domains. CLIPood consistently outperforms zero-shot and CoOp on all test domains, showing its effectiveness in more general and realistic OOD situations.

4.4. Analysis of CLIPood

Ablation Study. We explore the efficacy of each module in CLIPood, including Margin Metric Softmax (MMS) and Beta Moving Average (BMA). We compare CLIPood with its variants with or without MMS and BMA on the DomainNet dataset. We report the average results over all test domains with domain shift (Domain) and under the OOD setting with both domain shift and open classes (Domain+Class). From Figure 5a, we can observe that adding MMS and BMA improves the generalization performance on the domain shift and open class situations, which demonstrates the effectiveness of these two modules. CLIPood

incorporates both these two modules and achieves the best performance, which demonstrates that the designs from the aspects of training objective and model optimization work together towards better OOD generalization of CLIP. Note that the variant without MMS and BMA still outperforms standard fine-tuning because it utilizes metric softmax as the training objective, which we will also discuss below.

Analysis on CLIP Fine-tuning. In CLIPood, we choose to fine-tune the model by metric softmax, which compares the image and class text embeddings. Here we compare it with standard fine-tuning, which adds and trains a parametric linear classifier together with the pre-trained backbone. We show the performance on OfficeHome with domain shifts as in Section 4.1. As shown in Figure 5b, metric softmax fine-tuning outperforms standard fine-tuning on most of the test domains, which indicates that metric softmax is a better choice to improve OOD generalization for CLIP.

Analysis on Adaptive Margin. Margin Metric Softmax (MMS) adds an adaptive margin to preserve inherent unequal relations of classes in the language space during fine-tuning. We compare MMS to the variants with a fixed margin for all data or without margin on OfficeHome and DomainNet with both domain shift and open classes as in Section 4.3. As shown in Figure 5c, adding a margin in metric softmax may improve the performance. Still, a fixed margin does not give a stable improvement on the variant without margins (the vanilla metric softmax), and an adaptive margin consistently outperforms the variants with a fixed margin or no margins, especially achieving remarkable (1.2%) improvement on DomainNet. This demonstrates the efficacy of the proposed Margin Metric Softmax for OOD generalization of the CLIP model.

Table 5: Performance with different weight ensemble methods.

METHOD	DOMAINNET	IMAGENET			SUN397		
		BASE	NEW	H	BASE	NEW	H
CLIPOOD W/ SWAD	62.3	77.2	68.7	72.7	80.7	78.2	79.4
CLIPOOD W/ WISE-FT	62.7	77.3	69.9	73.4	79.2	78.8	79.0
CLIPOOD W/ BMA (OURS)	**63.5**	**77.5**	**70.3**	**73.7**	**81.0**	**79.3**	**80.2**

In Figure 6, we show the top-5 predictions on test images from the models trained with and without adaptive margins. The cat image comes from a domain with the distribution shift. The model without margins outputs unrelated classes (underlined) such as bowtie (probably because the ears of the cat may look like a bowtie), indicating that it may overfit the image modality and forget semantic relations. The model trained with margins outputs related classes of animals. The strawberry image comes from a new class unseen during training. The model without margins outputs unrelated classes and makes wrong predictions, while the model with margins makes related and right predictions. These results show that the adaptive margin keeps the semantic relationship of classes better and improves generalization under domain shift and open classes.

Analysis on BMA. To evaluate the efficacy of the proposed Beta Moving Average (BMA), we compare it with the commonly used Exponential Moving Average (EMA) and the uniform average (AVG) of the model checkpoints. We report results on ImageNet with open classes as in Section 4.2. As shown in Figure 5d, EMA focuses on only one side of the training trajectory, which may perform worse than average weighting. BMA performs the best balance on base and new classes, showing the importance of the Beta-distribution weighting to focus on both zero-shot and fine-tuned models.

We also compare BMA with other weight ensemble methods. We re-implement and compare with SWAD (Cha et al., 2021) and WiSE-FT (Wortsman et al., 2022). We investigate how CLIPood performs when BMA in our method is replaced with SWAD or WiSE-FT, which are denoted as CLIPood w/ SWAD or CLIPood w/ WiSE-FT. We conduct experiments on DomainNet for domain shift settings and on ImageNet and SUN397 for open class settings. As shown in Table 5, for both domain shift and open class, CLIPood with BMA outperforms CLIPood with SWAD and WiSE-FT in most cases, which also shows that BMA is a better model averaging choice for CLIP models.

5. Discussion

CLIPood consistently outperforms zero-shot pre-trained models and existing generalization techniques of pre-trained models, showing its efficacy for different OOD generalization situations. Here we also discuss some limitations of this method and some possible future directions regarding these limitations. In CLIPood, we propose Metric Margin Softmax and Beta Moving Average, which introduce negligible additional costs in storage or computation compared with standard fine-tuning. Computing BMA requires the storage of one more model during training. It would be imperceptible in common situations but may be constrained when the memory resource is extremely limited. One of the limitations is that we mainly consider better fine-tuning the image encoder for OOD generalization. A possible future work regarding this would be exploring adaptation on both text and image modalities for better OOD generalization. Another limitation is that the performance of our method may still be influenced by the zero-shot performance of the pre-trained model. For some domains or classes with extremely poor zero-shot performance, indicating that the pre-trained knowledge may not help certain cases, the improvements from exploiting knowledge in pre-trained models may be minor. A possible future work regarding this may need to consider the whole pre-training-fine-tuning pipeline, exploring pre-trained models with better generalization and designing corresponding adaptation methods for them.

6. Conclusion

In this paper, we propose to solve the problem of generalizing CLIP to out-of-distributions with both domain shift and open classes in downstream tasks. We propose CLIPood to fine-tune CLIP with a simple and effective design to improve its OOD generalization. CLIPood introduces Margin Metric Softmax, which adds class adaptive margins in metric softmax training to exploit semantic relations between classes from the text modality. It also introduces Beta Moving Average to maintain a temporal ensemble according to a Beta distribution, which incorporates both the zero-shot model and the adapted model. Experiments on various datasets with different OOD scenarios show that CLIPood consistently outperforms existing generalization techniques.

Acknowledgements

References

Agrawal, P., Girshick, R., and Malik, J. Analyzing the performance of multilayer neural networks for object recognition. In *ECCV*, 2014.

Balaji, Y., Sankaranarayanan, S., and Chellappa, R. Metareg: Towards domain generalization using meta-regularization. In *NeurIPS*, 2018.

Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *ECCV*, 2018.

Bengio, Y., Lecun, Y., and Hinton, G. Deep learning for ai. *Communications of the ACM*, 64(7):58–65, 2021.

Bossard, L., Guillaumin, M., and Van Gool, L. Food-101–mining discriminative components with random forests. In *ECCV*, 2014.

Carlucci, F. M., D'Innocente, A., Bucci, S., Caputo, B., and Tommasi, T. Domain generalization by solving jigsaw puzzles. In *CVPR*, 2019.

Cha, J., Chun, S., Lee, K., Cho, H.-C., Park, S., Lee, Y., and Park, S. Swad: Domain generalization by seeking flat minima. In *NeurIPS*, 2021.

Cha, J., Lee, K., Park, S., and Chun, S. Domain generalization by mutual-information regularization with pretrained models. *arXiv*, 2022.

Chattopadhyay, P., Balaji, Y., and Hoffman, J. Learning to balance specificity and invariance for in and out of domain generalization. In *ECCV*, 2020.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. Describing textures in the wild. In *CVPR*, 2014.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Elhoseiny, M., Saleh, B., and Elgammal, A. Write a classifier: Zero-shot learning using purely textual descriptions. In *ICCV*, 2013.

Fei-Fei, L., Fergus, R., and Perona, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR workshop*, 2004.

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., and Mikolov, T. Devise: A deep visual-semantic embedding model. In *NeurIPS*, 2013.

Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., and Qiao, Y. Clip-adapter: Better vision-language models with feature adapters. *arXiv*, 2021.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

Goel, S., Bansal, H., Bhatia, S., Rossi, R. A., Vinay, V., and Grover, A. Cyclip: Cyclic contrastive language-image pretraining. In *NeurIPS*, 2022.

Goyal, S., Kumar, A., Garg, S., Kolter, Z., and Raghunathan, A. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *CVPR*, 2023.

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. In *ICLR*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, 2016.

Helber, P., Bischke, B., Dengel, A., and Borth, D. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021a.

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., and Song, D. Natural adversarial examples. In *CVPR*, 2021b.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

Khosla, A., Zhou, T., Malisiewicz, T., Efros, A. A., and Torralba, A. Undoing the damage of dataset bias. In *ECCV*, 2012.

Kim, W., Son, B., and Kim, I. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, 2021.

Kornblith, S., Shlens, J., and Le, Q. V. Do better imagenet models transfer better? In *CVPR*, 2019.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. 3d object representations for fine-grained categorization. In *ICCV workshops*, 2013.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and under-perform out-of-distribution. In *ICLR*, 2022.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *ICCV*, 2017.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, 2018a.

Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., and Tao, D. Deep domain generalization via conditional invariant adversarial networks. In *ECCV*, 2018b.

Li, Y., Yang, Y., Zhou, W., and Hospedales, T. M. Feature-critic networks for heterogeneous domain generalization. In *ICML*, 2019.

Li, Y., Liang, F., Zhao, L., Cui, Y., Ouyang, W., Shao, J., Yu, F., and Yan, J. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv*, 2021.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *ICLR*, 2019.

Lu, J., Batra, D., Parikh, D., and Lee, S. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019.

Lu, Y., Liu, J., Zhang, Y., Liu, Y., and Tian, X. Prompt distribution learning. In *CVPR*, 2022.

Maji, S., Rahtu, E., Kannala, J., Blaschko, M., and Vedaldi, A. Fine-grained visual classification of aircraft. *arXiv*, 2013.

Miller, J. P., Taori, R., Raghunathan, A., Sagawa, S., Koh, P. W., Shankar, V., Liang, P., Carmon, Y., and Schmidt, L. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In *ICML*, 2021.

Mu, N., Kirillov, A., Wagner, D., and Xie, S. Slip: Self-supervision meets language-image pre-training. In *ECCV*, 2022.

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation. In *ICML*, 2013.

Nilsback, M.-E. and Zisserman, A. Automated flower classification over a large number of classes. In *ICVGIP*, 2008.

Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. Cats and dogs. In *CVPR*, 2012.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. Moment matching for multi-source domain adaptation. In *ICCV*, 2019.

Pham, H., Dai, Z., Ghiasi, G., Liu, H., Yu, A. W., Luong, M.-T., Tan, M., and Le, Q. V. Combined scaling for zero-shot transfer learning. *arXiv*, 2021.

Piratla, V., Netrapalli, P., and Sarawagi, S. Efficient domain generalization via common-specific low-rank decomposition. In *ICML*, 2020.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019.

Shu, M., Nie, W., Huang, D.-A., Yu, Z., Goldstein, T., Anandkumar, A., and Xiao, C. Test-time prompt tuning for zero-shot generalization in vision-language models. In *NeurIPS*, 2022.

Shu, Y., Cao, Z., Wang, C., Wang, J., and Long, M. Open domain generalization with domain-augmented meta-learning. In *CVPR*, 2021.

Socher, R., Ganjoo, M., Manning, C. D., and Ng, A. Zero-shot learning through cross-modal transfer. In *NeurIPS*, 2013.

Soomro, K., Zamir, A. R., and Shah, M. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv*, 2012.

Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., and Dai, J. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*, 2020.

Sun, B. and Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, 2016.

Tan, H. and Bansal, M. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP-IJCNLP*, 2019.

Taori, R., Dave, A., Shankar, V., Carlini, N., Recht, B., and Schmidt, L. Measuring robustness to natural distribution shifts in image classification. In *NeurIPS*, 2020.

Torralba, A. and Efros, A. A. Unbiased look at dataset bias. In *CVPR*, 2011.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *CVPR*, 2017.

Volpi, R., Namkoong, H., Sener, O., Duchi, J. C., Murino, V., and Savarese, S. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, 2018.

Wang, H., Ge, S., Lipton, Z., and Xing, E. P. Learning robust global representations by penalizing local predictive power. In *NeurIPS*, 2019.

Wortsman, M., Ilharco, G., Kim, J. W., Li, M., Kornblith, S., Roelofs, R., Lopes, R. G., Hajishirzi, H., Farhadi, A., Namkoong, H., et al. Robust fine-tuning of zero-shot models. In *CVPR*, 2022.

Xiao, J., Hays, J., Ehinger, K. A., Oliva, A., and Torralba, A. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

Zhai, X., Wang, X., Mustafa, B., Steiner, A., Keysers, D., Kolesnikov, A., and Beyer, L. Lit: Zero-shot transfer with locked-image text tuning. In *CVPR*, 2022.

Zhang, R., Fang, R., Zhang, W., Gao, P., Li, K., Dai, J., Qiao, Y., and Li, H. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv*, 2021.

Zhang, X., Gu, S. S., Matsuo, Y., and Iwasawa, Y. Domain prompt learning for efficiently adapting clip to unseen domains. *arXiv*, 2022.

Zhou, K., Yang, J., Loy, C. C., and Liu, Z. Conditional prompt learning for vision-language models. In *CVPR*, 2022a.

Zhou, K., Yang, J., Loy, C. C., and Liu, Z. Learning to prompt for vision-language models. *IJCV*, 130(9):2337–2348, 2022b.

A. Experimental Details

A.1. Implementation Details

We use the CLIP pre-trained model with ViT-B/16 (Dosovitskiy et al., 2021) as the image encoder. We keep the temperature of the softmax function the same as the pre-trained model as $\tau = 0.01$. All the default configs are shown in Table 6. We use the same hyper-parameter $\lambda = 0.3$ for all datasets to avoid over-tuning on specific tasks. We adopt a batch size of 36 except for DomainNet with 60, and all images are randomly resized and cropped to 224×224. We use the AdamW (Loshchilov & Hutter, 2019) optimizer with the cosine learning rate strategy for all datasets. By default, we set $\beta = 0.5$, and use a learning rate of 5×10^{-6} and train for 5000 iterations. Considering the small amount of data, we set β to 0.1 for all 16-shot benchmarks. For specific datasets, we adjust different numbers of iterations and learning rates. We train 10000 iterations for DomainNet, 2500 iterations for StanfordCars and SUN397, 1000 iterations for UCF101, and 500 iterations for other 16-shot datasets except for ImageNet and FGVCAirCraft. We adopt the learning rate of 1×10^{-5} for DomainNet and OfficeHome in the case of domain shifts. For each result of CLIPood, we report the average result and the standard deviation of three runs with random seeds.

Table 6: Default configs for the experiments.

Default Config	Value
optimizer	AdamW
base lr	5×10^{-6}
weight decay	0.1
lr scheduler	cosine decay
augmentation	RandomResizedCrop
batch size	36
# iterations	5000
temperature	0.01
λ for MMS	0.3
β for BMA	0.5

A.2. Prompt Templates for Each Dataset

By default, we use "a photo of [CLASS]." as the prompt template for class labels, where [CLASS] refers to the name of a class with the hyphens replaced by spaces. Following Zhou et al. (2022b), for fine-grained classification datasets such as FGVCAircraft, we add the name of the superclass (aircraft) or the description to the template. The full templates for all datasets are shown as follows.

- OxfordPets: "a photo of a [CLASS], a type of pet."

- FGVCAircraft: "a photo of a [CLASS], a type of aircraft."

- DTD: "[CLASS] texture."

- EuroSAT: "a centered satellite photo of [CLASS]."

- Food101: "a photo of a [CLASS], a type of food."

- UCF101: "a photo of a person doing [CLASS]."

- Other datasets: "a photo of a [CLASS]."

A.3. Computing Infrastructure

For the experiments, we use PyTorch 1.13.1, torchvision 0.14.1, and CUDA 11.6 libraries. We use a machine with 32 CPUs, 256 GB memory, and the NVIDIA TITAN X GPU.

A.4. Licenses of Datasets

OxfordPets is under CC BY-SA 4.0 license. Other datasets are publicly available online and under custom licenses for non-commercial usage.

B. More Experimental Results

B.1. Unified Comparison with More Methods on Domain Shifts

We compare our method with other state-of-the-art baselines, including WiSE-FT (Wortsman et al., 2022), FLYP (Goyal et al., 2023), LP-FT (Kumar et al., 2022), and MIRO (Cha et al., 2022) with SWAD (Cha et al., 2021). Since there are no available results in our setting, we re-implement them with our codebase for a unified comparison. We report the accuracy on each domain of DomainNet. Results are shown in Table 7. Under our setting, all the state-of-the-art methods give better performance than ERM. Among them, MIRO benefits from SWAD, and MIRO+SWAD achieves previous state-of-the-art results. Compared with these baselines, CLIPood can outperform them generally with a significant accuracy gain.

Table 7: Performance of more baselines on DomainNet.

METHOD	CLIPART	INFOGRAPH	PAINTING	QUICKDRAW	REAL	SKETCH	AVERAGE
ERM	76.3	47.8	68.1	19.7	80.9	66.5	59.9
WISE-FT	76.8	49.5	69.4	20.1	81.7	67.2	60.8
FLYP+WISE-FT	**78.1**	51.9	69.8	20.0	84.3	67.6	61.9
LP-FT	75.1	51.7	70.7	16.2	85.0	67.1	61.0
MIRO	76.6	51.0	70.9	18.8	82.3	68.5	61.3
MIRO+SWAD	75.9	51.6	71.3	20.0	82.5	68.4	61.6
CLIPOOD (OURS)	77.6	**54.7**	**72.5**	**20.7**	**85.7**	**69.9**	**63.5**

B.2. Detailed Results on Open Classes

We report the full results of generalizing to open classes as in Section 4.2. We report the results in each dataset and the average results over all 11 datasets in Table 8. On most of the datasets, CLIPood achieves better adaptation performance on base classes and still narrows the gap between zero-shot prediction on new classes or even performs better. Comparing the average results, CLIPood outperforms zero-shot prediction and existing methods by a large margin, showing that it simultaneously adapts the model to improve the performance on the downstream tasks and keeps the OOD generalization ability on open classes.
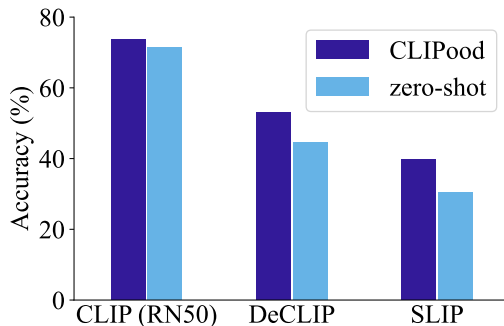
Figure 7: Results on other vision-language models.

B.3. Generalization on Other Vision-Language Models

In this paper, we mainly focus on the CLIP pre-trained model and use its open-source version with ViT-B/16 as the image encoder. Here we investigate whether CLIPood is general for other backbones or other variants of vision-language models. For other backbones, we use the CLIP pre-trained model with ResNet-50 as its image encoder (CLIP-RN50). For other variants of vision-language models, we use DeCLIP (Li et al., 2021) and Slip (Mu et al., 2022), and use their open-source

Table 8: Generalization performance on 11 downstream datasets with open classes.

(a) Average over 11 datasets

	BASE	NEW	H
CLIP	69.3	74.2	71.7
CoOp	82.7	63.2	71.7
CoCoOp	80.5	71.7	75.8
CLIPooD	$\textbf{83.9}_{\pm0.1}$	$\textbf{74.5}_{\pm0.1}$	$\textbf{78.9}_{\pm0.1}$

(b) ImageNet

	BASE	NEW	H
CLIP	72.4	68.1	70.2
CoOp	76.5	67.9	71.9
CoCoOp	76.0	$\textbf{70.4}$	73.1
CLIPooD	$\textbf{77.5}_{\pm0.1}$	$70.3_{\pm0.1}$	$\textbf{73.7}_{\pm0.1}$

(c) Caltech101

	BASE	NEW	H
CLIP	96.8	94.0	95.4
CoOp	98.0	89.8	93.7
CoCoOp	98.0	93.8	95.8
CLIPooD	$\textbf{98.7}_{\pm0.1}$	$\textbf{94.6}_{\pm0.1}$	$\textbf{96.6}_{\pm0.1}$

(d) OxfordPets

	BASE	NEW	H
CLIP	91.2	97.3	94.1
CoOp	93.7	95.3	94.5
CoCoOp	95.2	$\textbf{97.7}$	$\textbf{96.4}$
CLIPooD	$\textbf{95.7}_{\pm0.1}$	$96.4_{\pm0.2}$	$96.0_{\pm0.1}$

(e) StanfordCars

	BASE	NEW	H
CLIP	63.4	$\textbf{74.9}$	68.7
CoOp	78.1	60.4	68.1
CoCoOp	70.5	73.6	72.0
CLIPooD	$\textbf{78.6}_{\pm0.1}$	$73.5_{\pm0.3}$	$\textbf{75.9}_{\pm0.2}$

(f) Flowers102

	BASE	NEW	H
CLIP	72.1	$\textbf{77.8}$	74.8
CoOp	$\textbf{97.6}$	59.7	74.1
CoCoOp	94.9	71.8	81.7
CLIPooD	$93.5_{\pm0.2}$	$74.5_{\pm0.5}$	$\textbf{82.9}_{\pm0.3}$

(g) Food101

	BASE	NEW	H
CLIP	90.1	91.2	90.7
CoOp	88.3	82.3	85.2
CoCoOp	$\textbf{90.7}$	91.3	91.0
CLIPooD	$\textbf{90.7}_{\pm0.1}$	$\textbf{91.7}_{\pm0.1}$	$\textbf{91.2}_{\pm0.1}$

(h) FGVCAircraft

	BASE	NEW	H
CLIP	27.2	36.3	31.1
CoOp	40.4	22.3	28.8
CoCoOp	33.4	23.7	27.7
CLIPooD	$\textbf{43.3}_{\pm0.3}$	$\textbf{37.2}_{\pm0.5}$	$\textbf{40.0}_{\pm0.4}$

(i) SUN397

	BASE	NEW	H
CLIP	69.4	75.4	72.2
CoOp	80.6	65.9	72.5
CoCoOp	79.7	76.9	78.3
CLIPooD	$\textbf{81.0}_{\pm0.1}$	$\textbf{79.3}_{\pm0.1}$	$\textbf{80.2}_{\pm0.1}$

(j) DTD

	BASE	NEW	H
CLIP	53.2	$\textbf{59.9}$	56.4
CoOp	79.4	41.2	54.2
CoCoOp	77.0	56.0	64.9
CLIPooD	$\textbf{80.8}_{\pm0.6}$	$58.6_{\pm0.6}$	$\textbf{67.9}_{\pm0.3}$

(k) EuroSAT

	BASE	NEW	H
CLIP	56.5	$\textbf{64.1}$	60.0
CoOp	92.2	54.7	68.7
CoCoOp	87.5	60.0	71.2
CLIPooD	$\textbf{97.5}_{\pm0.2}$	$\textbf{64.1}_{\pm1.1}$	$\textbf{77.3}_{\pm0.8}$

(l) UCF101

	BASE	NEW	H
CLIP	70.5	77.5	73.9
CoOp	84.7	56.1	67.5
CoCoOp	82.3	73.5	77.6
CLIPooD	$\textbf{85.7}_{\pm0.1}$	$\textbf{79.3}_{\pm0.2}$	$\textbf{82.4}_{\pm0.1}$

pre-trained models with ViT-B/32 as the image encoders. We evaluate the performance of CLIPood on the OfficeHome dataset with both domain shift and open classes as the protocol in Section 4.3. As shown in Figure 7, CLIPood consistently outperforms the zero-shot prediction on these three models, demonstrating its generalization ability on different architectures and variants of vision-language models.

B.4. Parameter Sensitivity

To investigate the robustness of hyper-parameters, we conduct detailed analysis experiments with different values of λ and β. We experiment on the DomainNet dataset and the results are shown in Table 9. As shown, only if λ is set too large to let the margin term dominate the loss will it hurt the performance. In a wide range, CLIPood on various values of hyper-parameters performs robustly and still outperforms the baseline results (59.9% on DomainNet).

Table 9: Performance with different λ and β on DomainNet dataset.

λ	0.01	0.03	0.1	0.3	1.0
CLIPOOD	62.9	63.1	**63.3**	**63.5**	61.8
β	0.1	0.3	0.5	0.7	0.9
CLIPOOD	63.0	**63.7**	**63.5**	**63.5**	63.1

B.5. Analysis on Zero-shot Performance

To further analyze how our approach exploits CLIP's pre-trained knowledge, we conduct an analysis on zero-shot performance. We compare the improvement of ERM and our approach on classes and domains where CLIP's zero-shot performance is poor. We first select the two domains where CLIP has the worst zero-shot performance as in Table 10. On the worst domain Quickdraw, our CLIPood still slightly outperforms ERM, while on the second worst domain Infograph, CLIPood outperforms ERM significantly.

Table 10: Performance on the worst 2 domains.

METHOD	QUICKDRAW	INFOGRAPH
ZERO-SHOT	13.8	46.7
ERM	19.7	47.6
CLIPOOD	**20.7**	**54.7**

Then, we investigate class-wise performance on the domain Infograph. Since there exist over 300 classes, we can only show part of them. According to zero-shot performance, we show 3 groups of classes with 12 classes in total. As shown in Table 11, for the first group of classes where CLIP has poor zero-shot performance (class 1-4), CLIPood and ERM may lead to minor improvements upon zero-shot, or may outperform each other in different cases. This is because the pre-trained knowledge may not help certain classes. However, for the classes where the zero-shot performance is moderate (class 5-8), and where the zero-shot performance is near the average performance on the dataset (class 9-12), we find CLIPood, which benefits from better exploitation of pre-trained knowledge, generally provides better accuracies.

Table 11: Performance on 12 different classes.

INFOGRAPH	LOW ZERO-SHOT ACC				MEDIUM ZERO-SHOT ACC				HIGH ZERO-SHOT ACC			
	1	2	3	4	5	6	7	8	9	10	11	12
ZERO-SHOT	**0.0**	1.3	2.8	4.6	20.8	**21.2**	22.5	23.1	46.2	46.3	**46.8**	47.0
ERM	**0.0**	1.3	**11.1**	22.7	**37.5**	7.7	17.5	61.5	46.2	53.7	40.4	49.4
CLIPOOD	**0.0**	**3.9**	2.8	**27.3**	**37.5**	19.2	**30.0**	**65.4**	**61.5**	**56.1**	44.7	**53.0**