DOMAIN GENERALIZATION IN-THE-WILD: DISENTANGLING CLASSIFICATION FROM DOMAIN-AWARE REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Traditional domain generalization (DG) methods commonly encourage domain-invariant representations to improve out-of-distribution (OOD) performance. However, we find that this does not transfer directly to finetuning CLIP, a widely used foundation model. Applying invariance regularization during finetuning gives little or no benefit on OOD data compared to standard finetuning. We hypothesize this is because invariance and the preservation of pretrained features can conflict with each other. Domain invariance discourages reliance on domain-specific features, but in foundation models these features were learned during pretraining and are part of what makes the model useful. Retaining domain-specific knowledge would require domain awareness, which standard invariance methods do not provide. To address this, we propose CLIP-DCA (**D**isentangling **C**lassification from enhanced domain **A**ware representations), which encourages invariance only at the classification head through disentanglement, while domain awareness is encouraged in the encoder using a separate domain head supervised by synthetic domain images and style descriptions generated by a multimodal LLM (MLLM). Rather than removing all domain information from the classifier, CLIP-DCA balances invariance with awareness, as we find that pushing the two heads further apart lowers accuracy. Since recent work shows that standard DG benchmarks can be contaminated by CLIP's web-scale pretraining, we evaluate on both established out-of-pretraining (OOP) splits and a 33-dataset benchmark indexed by a continuous multi-modal OOD score. CLIP-DCA shows improvements on both. Our results suggest that, for CLIP, traditional DG principles remain useful when invariance is balanced with domain awareness.

1 INTRODUCTION

Traditional domain generalization (DG) suggests that learning representations that are domain invariant improves performance on OOD samples (Ganin et al., 2016; Zhou et al., 2022a). However, for CLIP (Radford et al., 2021), a widely used foundation model, we find this does not directly transfer. Applying standard invariance regularization to CLIP gives little or no benefit over regular finetuning, and some methods degrade OOD performance (Figure 1, Sec. 4). We hypothesize this is because CLIP's web-scale pretraining produces features that are domain-specific but are also what enables its zero-shot generalization.

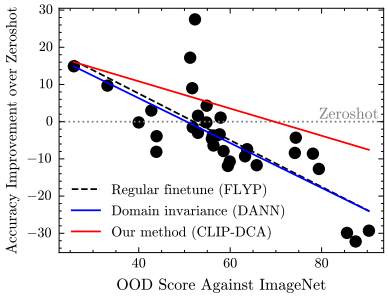


Figure 1: Improvement over zeroshot after finetuning on ImageNet (in %). Each dot represents one of the 33 target datasets. OOD scores are quantified relative to ImageNet (source dataset), illustrating the challenge of DG in-the-wild.

Two existing theoretical results offer possible explanations. Zhao et al. (2019) showed that strictly enforcing domain invariance forces a model to discard discriminative features whenever they correlate with the domain. Kumar et al. (2022) showed that standard finetuning distorts pretrained representations by moving weights off the robust pretrained manifold. This is not problematic for non-foundation models, which

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

likely have few domain-specific features worth keeping. For CLIP, however, many features that correlate with domains are fine-grained features learned during web-scale pretraining, and they are part of what makes CLIP useful for zero-shot performance and adaptation to highly specialized domains such as industrial (Moenck et al., 2024) and medical (Ghosh et al., 2024) fields. Invariance regularization encourages the model to remove them, consistent with the empirical degradation in Figure 1.

This suggests that for CLIP, domain invariance should not be applied to the entire model. Rather, we hypothesize that domain awareness in the encoder is a prerequisite for effective domain invariance at the classifier. The encoder must know what constitutes a domain to identify domain-specific features, and classification can then be disentangled from them so that decisions rely less on domain-specific signals. Without awareness, disentanglement may remove information indiscriminately and damage the pretrained knowledge we want to retain. Importantly, invariance at the classifier does not mean removing all domain information from the classification head, which would also discard useful pretrained features. Instead, the classification head is discouraged from aligning with the domain-specific signals that the encoder learns, and we find that separating the two more strongly lowers accuracy (Sec. 4.3). Awareness and invariance can therefore be complementary when balanced and applied to different parts of the model.

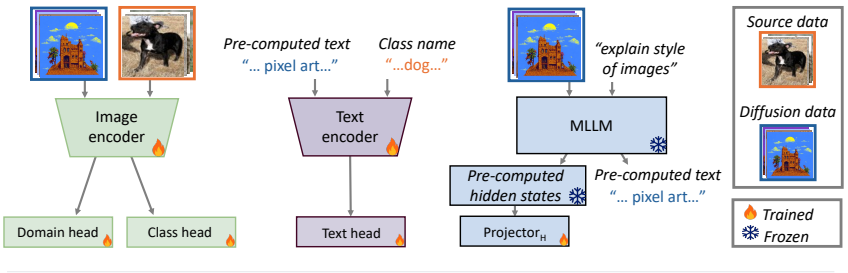
Existing work on adapting CLIP largely circumvents this problem by focusing on parameter-efficient finetuning (PEFT) (Zhou et al., 2022c;b; Gao et al., 2024; Zhang et al., 2022), regularization toward the original weights (Wortsman et al., 2022; Nam et al., 2024; Oh et al., 2024; Shu et al., 2023), or transductive adaptation (Wallingford et al., 2023; Martin et al., 2024), all of which rely on the assumption that CLIP's pretrained features are inherently OOD-robust. However, recent studies show that DG evaluations are often contaminated by data leakage from pretraining (Teterwak et al., 2024; Yu et al., 2024), and that retraining CLIP on non-contaminated data substantially degrades OOD performance (Mayilvahanan et al., 2024). Current robust finetuning methods may therefore preserve features that only appear robust because pretraining and test data overlap.

To address this, we propose that DG evaluation for foundation models should approximate "domain generalization in-the-wild," where CLIP encounters diverse and unseen data at deployment. We use two complementary approaches. First, we evaluate on the established out-of-pretraining (OOP) splits proposed by Teterwak et al. (2024), which separate samples using zero-shot embeddings to reduce the contamination effect. Second, we evaluate on 33 target datasets indexed by a continuous multi-modal OOD score (Sec. 3.2), which quantifies each dataset's distribution shift and predicts its performance after finetuning. After finetuning on ImageNet, CLIP's DG performance degrades on datasets with higher OOD scores (Figure 1).

CLIP-DCA combines domain awareness and domain invariance (Figure 2). We add a new head to the CLIP image encoder, the domain head, aligned to text descriptions of each domain, while the original classification head stays aligned to the class names. The classification head is then disentangled from both the domain head and the domain descriptions, encouraging domain awareness in the encoder and domain invariance at the classifier. The domain head is discarded after training, so inference is identical to standard CLIP. Since many datasets lack distinct domains or textual descriptions, and the definition of 'domain' is often vague in DG in-the-wild, we create artificial domains with diffusion models and describe them with Multimodal LLMs (MLLMs, Sec. 2.2). These domains mainly vary in visual style, which we use as a proxy for domain shift.

Our contributions are summarized as follows:

- We show that traditional invariance regularization (DANN, CDAN, DAN, CORAL, MDD) gives little or no benefit for CLIP finetuning, even with CLIP-DCA's synthetic domains.
- We propose that domain awareness in the encoder is a prerequisite for domain invariance at the classifier, and find that the two should be balanced (Sec. 4.3).
- We introduce CLIP-DCA, which encourages invariance at the classification layer through disentanglement, while building domain awareness in the encoder using a separate domain head supervised by synthetic domains and MLLM-generated descriptions. CLIP-DCA outperforms existing robust finetuning methods on both OOP splits and OOD data.
- Building on the OOP splits of Teterwak et al. (2024), we propose a more holistic evaluation framework for DG in-the-wild that adds a continuous multi-modal OOD score across 33 datasets.



Training objectives

↔ align (contrastive) ⊥ disentangle (orthogonalize)

	Source data	Diffusion data
↔ Classification	Class head ↔ Text head	—
⊥ Domain Invariance	Class head ⊥ Domain head	Class head ⊥ Domain head Class head ⊥ Text head
↔ Domain Awareness	—	Domain head ↔ Text head Projector$_H$ ↔ Text head

At inference, domain head is discarded. Standard CLIP is used: class head + text head only.

Figure 2: CLIP-DCA applies different losses to source and diffusion images. For source images, the class head is aligned to the class name embedding to preserve classification, and disentangled from the domain head to encourage domain invariance. For diffusion images, which span a broad range of visual styles, invariance is extended by additionally disentangling the class head from the style text embedding. Domain awareness is encouraged by aligning the domain head to the style description, and aligning the text encoder to the projected MLLM hidden state. At inference, the domain head is discarded and the standard CLIP classification pipeline is used unchanged.

Related Work. A comprehensive review is in Appendix A. Domain generalization (DG) has traditionally focused on learning domain-invariant representations (Ganin et al., 2016; Long et al., 2015; Sun & Saenko, 2016; Long et al., 2018; Zhang et al., 2019; Zhou et al., 2022a), but naively applying these methods to foundation models like CLIP can cause discarding of useful features. Consequently, most robust CLIP finetuning methods aim to preserve pretrained knowledge through parameter-efficient finetuning (PEFT) (Zhou et al., 2022c; Gao et al., 2024) or regularization towards the original weights (Wortsman et al., 2022; Shu et al., 2023). However, this reliance on pretrained knowledge is being questioned by recent findings of domain contamination in web-scale datasets (Teterwak et al., 2024; Yu et al., 2024; Mayilvahanan et al., 2024), which suggest current evaluations may overestimate true OOD robustness. Our work addresses this by proposing a more challenging evaluation framework and a method that learns targeted invariance without sacrificing pretrained knowledge. Separating domain-invariant from domain-specific features has also been explored with ground-truth domain labels, for example GD-FAS (Jung et al., 2025) for binary face anti-spoofing, whereas our setting is open-vocabulary without domain labels, so each head is driven toward its content by its own text target, and we balance the two heads rather than fully separating them.

2 CLIP-DCA: Disentangling Classification from Enhanced Domain-Aware Representations

We empirically find that invariance regularization alone does not help, and often harms, CLIP on OOD data (Sec. 4). To address this challenge, we introduce CLIP-DCA (**D**isentangling **C**lassification from enhanced domain **A**ware representations), a finetuning method designed to improve robustness by balancing domain invariance with domain awareness.

2.1 Encouraging domain awareness and invariance simultaneously

Our key hypothesis is that domain invariance at the decision-making stage is beneficial for generalizing to unseen domains. At the same time, domain awareness is required for retaining the vast pretrained knowledge of CLIP. We target both by encouraging domain awareness in the encoders, while encouraging domain invariance only in the classifier of CLIP through disentanglement. The intuition

```
# === 1. Source Data (Standard + Invariance) ===
img_src, txt_src = source_batch
feat, emb_cls, emb_txt = clip_model(img_src, txt_src)
emb_dom = domain_head(feat)

#          C1: Align Class-Text              C2: Disentangle Class-Domain
loss_src = contrastive_L(emb_cls, emb_txt) + disentangle_L(emb_cls, emb_dom)

# === 2. Diffusion Data (Domain + MLLM Align) ===
img_diff, hidden, txt_diff = diffusion_batch
feat, emb_cls, emb_sty = clip_model(img_diff, txt_diff)
emb_dom  = domain_head(feat)
emb_mllm = mllm_proj(hidden)

#           C3: Disentangle Class-Domain      C4: Disentangle Class-Style
loss_diff = disentangle_L(emb_cls, emb_dom) + disentangle_L(emb_cls, emb_sty)

#            C5: Align Domain-Style            C6: Align MLLM-Style
loss_diff += contrastive_L(emb_dom, emb_sty) + contrastive_L(emb_mllm, emb_sty)

batch_loss = loss_src + loss_diff
```

Algorithm 1: PyTorch-style pseudocode for CLIP-DCA losses. More details provided in Appendix B

is that **if a model can explicitly identify domain-specific features, it can more effectively learn to rely less on them during classification on unseen domains**.

Enforcing domain invariance in the encoder can be harmful. Our experiments show that invariance regularization gives little or no benefit compared to standard finetuning (Sec. 4), which we attribute to the forgetting of fine-grained features learned during pretraining. Conversely, existing CLIP robust finetuning methods discourage divergence from the original pretrained model, relying on the assumption that CLIP is inherently robust to OOD data, which has been challenged by evidence for domain contamination (Teterwak et al., 2024; Yu et al., 2024; Mayilvahanan et al., 2024).

Instead, we focus on encouraging domain invariance only at the final classification layer, while simultaneously encouraging the image encoder to become domain-aware. Our intuition is that a comprehensive understanding of various domains enables the model to more effectively disregard domain-specific influences during inference. The diverse set of generated diffusion images and their descriptions (detailed in Section 2.2) provides the necessary signals for enhancing this domain awareness.

To implement this, we introduce an architectural addition to the CLIP image encoder. We add an additional linear projection head, termed the image domain head (I_D), which has the same dimensionality as the original image projection head, referred to as the image class head (I_C), as shown in Figure 2. The text encoder's original projection head is denoted as the text head (T_H). We do not add a corresponding domain head to the text encoder, since most downstream classification datasets only provide class names as text inputs, and text inherently allows for easier separation of class and domain (e.g. "a sketch of a dog"). Note that for inference, the domain head is discarded and the standard pipeline is used (Figure 3).

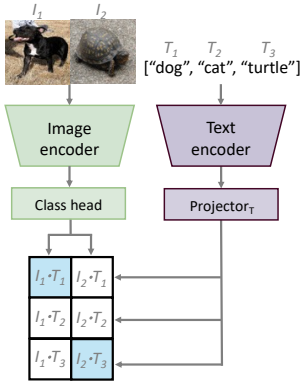


Figure 3: Standard CLIP inference pipeline using a dot product between image and text embeddings for classification.

During training, we use two distinct loss functions for the two types of data we use, the source dataset and the generated diffusion images. We use ℓ_a to refer to agreement loss (the standard CLIP contrastive loss (Radford et al., 2021) or finetuning (Goyal et al., 2023)). We use ℓ_d to refer to disentanglement, a per-sample orthogonality penalty between two sets of representations. We apply regularization per-sample rather than the batch-level cross-covariance penalties used in Barlow Twins and VICReg (Zbontar et al., 2021; Bardes et al., 2021). Batch-level decorrelation only enforces that the dimensions of the two representations are uncorrelated on average, which could allow a sample to share class and domain features as long as it is offset by another sample. Per-sample orthogonality is stricter, since for any individual image, the class embedding is penalized for aligning with the domain embedding. The loss minimizes the sum of the squared diagonal of the instance-wise similarity matrix between the standardized class embeddings and domain embeddings.

The role of the disentanglement loss is to encourage this separation. By minimizing the per-sample similarity between the class and domain embeddings, this loss encourages the class head to find representations that are predictive of the class label without relying on the directions used by the domain head, and encourages the domain head to focus on domain-specific information. Note that orthogonality is not equivalent to statistical independence, and we do not aim to guarantee it. Both heads are single linear layers and the loss is a soft penalty, a regularization, which was an intentional choice to bias the two heads apart without removing pretrained domain knowledge from the encoder. We measure how much domain information remains in each head in Sec. 4.3.

More formally, we simultaneously encourage accurate classification, domain awareness in both text and image encoders, and domain invariance at the classification stage using a weighted combination of loss terms (Algorithm 1, weights in Appendix D):

1. **For the source dataset images (e.g., ImageNet, with only class labels):**
   - A *classification loss* between the class head and the text embedding of the class name, preserving CLIP's classification capability: $C_1 := \ell_a(I_C, T_H)$.
   - A *disentanglement loss* between the class and domain heads, encouraging domain invariance in the class head: $C_2 := \ell_d(I_C, I_D)$.

   $\mathcal{L}_{\text{source}} = C_1 + C_2$
2. **For the diffusion images and their MLLM-generated style descriptions:**
   - A *disentanglement loss* between the class and domain heads, extending domain invariance across a broader distribution of visual styles than the source dataset provides: $C_3 := \ell_d(I_C, I_D)$.
   - A *disentanglement loss* between the class head and the style text embedding, discouraging the class head from encoding domain-specific text: $C_4 := \ell_d(T_H, I_C)$.
   - An *agreement loss* between the domain head and the style text embedding, providing the domain head with an explicit signal for each domain: $C_5 := \ell_a(T_H, I_D)$.
   - An *agreement loss* between the text encoder and the projected MLLM hidden state, enriching the text encoder's domain representations with vision-grounded style knowledge: $C_6 := \ell_a(T_H, P_H)$.

   $\mathcal{L}_{\text{diffusion}} = C_3 + C_4 + C_5 + C_6$

In practice, we tune only four grouped weights (Appendix D), and multiplying all awareness and invariance weights by $10\times$ changes average accuracy by only 0.1% (Table 3).

2.2 Generating diverse domains

Traditional DG benchmarks provide multi-domain datasets, enabling the learning of domain invariance. However, our evaluation setup, which involves finetuning on a single source dataset like ImageNet, lacks explicit multiple source domains, especially as the boundary for different domains becomes more vague for DG in-the-wild. Additionally, we hypothesize that to understand what constitutes domain-specific features, a diverse number of domains are required, which we verify in Sec. 4.3.

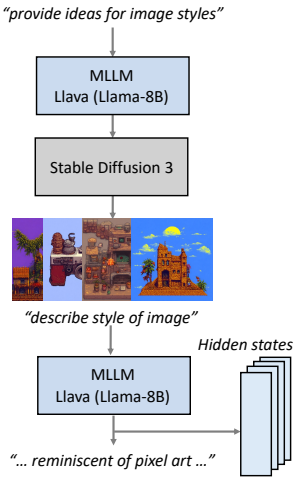


Figure 4: Pipeline for generating synthetic domain images and descriptions.

To address this, we construct a small dataset with a diverse number of domains. As illustrated in Figure 4, we prompt an MLLM, specifically LLaVA (Liu et al., 2023), to generate ideas of 512 distinct styles for images (e.g. "pixel art"). The complete list of styles is available in Table 19 in the Appendix. A text-to-image diffusion model (Stable Diffusion 3 (Esser et al., 2024)) then generates images from these stylistic prompts. We intentionally omit any class labels during image generation to ensure the styles are not biased towards specific classes. We generate 8 images per style, creating a dataset of 4096 images. Finally, the same MLLM generates textual domain descriptions (captions) for each style. We also store the last-layer MLLM hidden states for the style images and the prompt, averaged over all tokens, and project the result to the CLIP text embedding dimension. The exact prompts used for style and description generation are detailed in Appendix E.

3 EXPERIMENTAL SETUP FOR DG IN-THE-WILD

Our evaluation focuses on end-to-end finetuning given a labeled source dataset. Source-free adaptation methods are orthogonal to our setup, as they often require hard domain labels and do not utilize source images.

3.1 OUT-OF-PRETRAINING SPLITS (TETERWAK ET AL., 2024)

We evaluate our baselines on the established out-of-pretraining data. Teterwak et al. (2024) calculated the cosine similarity between the pre-trained image embedding and the ground truth text label embedding for all samples in 5 DG benchmarks. Following their exact protocol, thresholds of 0.15 and 0.21 were used to partition the data into in-pretraining and out-of-pretraining subsets. We evaluate performance on the full dataset as well as these specific splits for consistency.

3.2 EVALUATION ON MORE DATASETS (33 DATASETS)

We first analyze standard Domain Generalization (DG) benchmarks and find their domains are not well-separated. We treat each dataset as a training distribution to train a Spectral-normalized Neural Gaussian Process (SNGP) (Liu et al., 2020) model and calculate the uncertainty on other datasets. Using these pairwise OOD scores to compute a distance matrix for PCA, we observe strong intra-benchmark clustering, as visualized in Figure 5. This clustering, along with CLIP's high zero-shot accuracy and the success of transductive methods on these datasets (Wallingford et al., 2023; Martin et al., 2024), suggests that current DG evaluations are not sufficiently challenging for large-scale models, possibly due to pre-training data contamination.

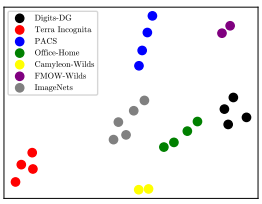


Figure 5: PCA visualization of domains from different domain generalization datasets

To address this, we finetune CLIP on ImageNet-1K (Deng et al., 2009) and evaluate its generalization capabilities across a more diverse benchmark of 33 target datasets spanning standard DG benchmarks and other challenging classification tasks (full list provided in Table 7). A cross-dataset evaluation is significantly more challenging compared to traditional DG setups, as it involves larger visual distribution shifts and also shifts in class labels. This evaluation also aligns with the methodologies of prior studies investigating robust CLIP finetuning (Zhou et al., 2022c;b; Gao et al., 2024; Shu et al., 2023), while adding a broader coverage of domains. We use the CLIP ViT-B/32 model for all main experiments. Further implementation details are in Appendix C and D.

Measuring OODness of the target datasets. Given that our DG in-the-wild evaluation includes many target datasets with varying degrees of OODness compared to ImageNet, establishing a quantitative OOD metric is beneficial for a more holistic assessment of OOD robustness. Since CLIP has a dual-encoder architecture, we use OOD measures for both the image and text modalities, normalizing and averaging them to produce a single scalar OOD score.

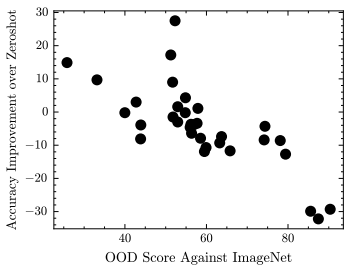


Figure 6: OOD score of 33 target datasets against ImageNet and classification accuracy improvement over zeroshot

For the image modality, we extract features using the frozen CLIP image encoder and train an SNGP (Liu et al., 2020) using the ImageNet-1K training set. We apply spectral normalization to a new linear layer and use Random Fourier Features to approximate the Gaussian Process, leaving the CLIP encoder frozen. We compute the image OOD score using the expected entropy of the predictive distribution, averaged across all samples within the target dataset. For the text modality, we measure the shift in label distribution (Fort et al., 2021) by forming a joint label space $C_{joint} = C_{source} \cup C_{target}$. For each target dataset image x, let $p(y|x)$ be the zero-shot softmax probability distribution over the joint label space C_{joint}. The sample-level text OOD score is the probability mass assigned to classes that are novel relative to the source: $\sum_{c \in C_{target} \setminus C_{source}} p(y = c|x)$. This probability mass assigned to target-specific classes is then averaged across all samples in the target dataset. Finally, the image and text OOD scores are each normalized across the 33 datasets and averaged to produce a single continuous OOD score.

Table 1: Performance on In-Pretraining (In, cosine similarity > 0.21) and Out-of-Pretraining (Out, < 0.15) splits following Teterwak et al. (2024). Results on the full datasets are in Appendix Table 8.

	PACS		TerraInc		VLCS		OfficeHome		DomainNet	
Method	In	**Out**	In	**Out**	In	**Out**	In	**Out**	In	**Out**
Zero-shot	95.1	82.6	16.4	12.9	**94.0**	74.6	80.9	45.7	66.6	23.2
DANN	92.6	82.3	43.8	17.8	89.3	75.7	81.2	52.5	65.8	24.5
FLYP	94.3	83.4	42.1	20.6	90.0	76.2	81.1	48.8	67.0	23.9
Adapter	95.7	84.6	31.9	8.2	93.1	74.4	81.3	47.0	68.1	24.2
CoOp	94.3	81.0	32.1	7.0	92.7	72.1	82.0	**57.1**	67.9	22.2
CLIP-OOD	94.7	84.3	**51.5**	16.7	92.1	80.5	**85.8**	57.0	69.6	26.3
Ours	**96.2**	**87.1**	50.3	**21.9**	90.4	**82.7**	82.2	57.0	**70.2**	**27.7**

We verify that our OOD score shows a strong negative correlation ($r = -0.756, p < 0.001$) with the classification accuracy improvement over zeroshot on target datasets after standard finetuning (FLYP), as shown in Figure 6. We believe this continuous metric complements the binary threshold of the OOP splits (Teterwak et al., 2024), since the OOP method identifies specific samples likely seen during pre-training, while our OOD score measures the distribution shift of an entire dataset relative to the source.

4 Results and Discussion

4.1 Performance on Out-of-Pretraining (OOP) Data

As shown in Table 1, CLIP-DCA outperforms baselines on OOP splits. Specifically, on four out of five benchmarks, our method performs the best on OOP splits. Interestingly, on Terra Incognita, baselines like CLIP-OOD perform better on In-Pretraining data, but their performance drops on OOP data. In contrast, CLIP-DCA maintains robustness on the OOP split, which supports our hypothesis that our method reduces reliance on contaminated pre-training knowledge.

4.2 Robustness Across 33 Datasets

We further evaluate CLIP-DCA across 33 datasets. As shown in Figure 7, CLIP-DCA consistently improves performance over standard finetuning across target datasets. Importantly, the best-fit line for CLIP-DCA shows a flatter slope, indicating that it is more robust to more severe OOD data compared to regular finetuning. This observation aligns with our hypothesis that encouraging domain invariance at the decision-making layer, while simultaneously encouraging domain awareness within the encoders, helps robust classification on unseen distributions.

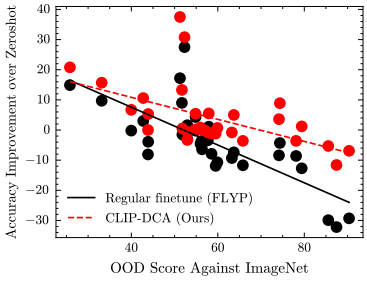


Figure 7: Comparison of CLIP-DCA against regular finetuning. Best-fit lines by linear regression.

Figure 8 and Table 2 provide a broader comparison against additional baselines. We observe that traditional domain invariance training (DANN (Ganin et al., 2016)), implemented by treating the synthetic diffusion data as the domain to discriminate against, is harmful for CLIP, showing poorer performance compared to regular finetuning. To check that this is not specific to adversarial training, we also evaluate other invariance regularization methods (CDAN (Long et al., 2018), DAN (Long et al., 2015), CORAL (Sun & Saenko, 2016) and MDD (Zhang et al., 2019)) using the same synthetic domains (Appendix Table 9). CLIP-DCA shows the best performance, while some baselines perform worse than regular finetuning. This suggests that for CLIP, invariance alone is not enough, possibly because enforcing it across the entire image encoder discards features learned during pretraining, and that it is more effective when balanced with domain awareness. These results support the importance of approaches such as our proposed learning of targeted invariance through disentanglement.

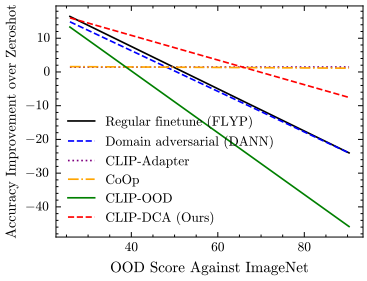


Figure 8: Comparison against more baselines

Interestingly, on the most extremely OOD datasets, parameter-efficient finetuning (PEFT) techniques like CoOp (Zhou et al., 2022c) and CLIP-Adapter (Gao et al., 2024) perform best. PEFT methods

minimally change the original CLIP weights, so their performance shows much lower variance across the datasets (around 1-2% improvements). It is important to note that on extreme OOD datasets, all end-to-end finetuning methods show lower performance than the zero-shot CLIP baseline. This drop is an expected consequence of finetuning on ImageNet (1.2M images), which covers far less of the original distribution than the 400M-image pretraining corpus. Because addressing data leakage directly by training CLIP from scratch is computationally prohibitive, we perform experiments using simulated unlearning (detailed in Appendix M.1) to act as a proxy for evaluating a model pretrained without specific domains. We find that even when specific domain knowledge is actively suppressed, CLIP-DCA maintains its performance advantages. The same trend also holds with the larger ViT-B/16 backbone (Appendix Table 10).

This strong zero-shot performance has often been attributed to CLIP's inherent OOD generalization capability. However, recent studies (Teterwak et al., 2024; Mayilvahanan et al., 2024) challenge this assumption and show that this generalization could be attributed to domain contamination. They show that when CLIP is retrained solely on natural images, its OOD performance drops to similar levels as models trained exclusively on ImageNet. This drop could offer a plausible explanation for observations like those motivating Wise-FT (Wortsman et al., 2022), where standard finetuning was found to degrade OOD performance.

Table 2: Accuracy on ImageNet variants.

Method	V1	V2	Sketch	A	R
Zero-shot	54.2	48.4	32.3	26.2	59.7
FLYP (Standard)	69.1	58.1	35.3	18.1	55.8
DANN	69.0	58.0	33.1	18.3	53.7
PEFT Methods					
CLIP Adapter	59.5	52.9	32.3	**28.5**	58.9
CoOp	59.9	52.7	32.8	27.9	58.8
LwEIB	60.4	53.2	34.1	<u>28.1</u>	60.1
Robust FT Methods					
MMA	71.5	60.0	36.5	18.5	56.0
Wise-FT	72.9	61.5	40.0	26.2	<u>61.0</u>
MIRO	<u>74.0</u>	<u>62.5</u>	36.0	24.5	59.0
CLIP-OOD	71.0	60.2	<u>40.5</u>	13.6	45.2
Ours (CLIP-DCA)	**75.0**	**64.1**	**42.9**	26.2	**65.0**

4.3 Ablations

Probing the heads. To measure how much domain information each head carries, we train linear domain classifiers on the class and domain heads using the four domains of OfficeHome (Table 3 and Appendix J). Interestingly, when we use domain descriptions without disentanglement, the domain information leaks into the class head. With disentanglement, the class head remains similar to zero-shot, while the domain head becomes more domain-aware. Scaling the regularization further separates the two heads, but the average accuracy drops. This supports our claim that a careful balance between awareness and invariance is needed.

Table 3: Linear probing of domain labels (OfficeHome, 4 domains) on the class and domain heads. $n\times$ scales all regularization terms. CKA is between the two heads.

Model	Class	Domain	CKA	Δ Avg.
Zero-shot	80.6	-	-	-
DANN	79.1	-	-	-
Ours w/o Disent.	99.0	79.5	-	-
Ours (1$\times$)	80.1	80.9	0.880	-
Ours (10$\times$)	79.7	80.6	0.871	-0.1
Ours (100$\times$)	78.4	82.1	0.805	-1.4
Ours (1000$\times$)	77.1	86.9	0.742	-1.7

Balancing awareness and invariance. We study the effect of the different components of CLIP-DCA, as shown in Table 4. The results test our central hypothesis that domain awareness is a prerequisite for domain invariance. When we apply the disentanglement loss without any domain descriptions, performance drops below standard finetuning. The domain head, lacking any signal for what constitutes a domain, likely learns noisy features, and disentangling the class head from this noise discards useful information. Conversely, using domain descriptions without disentanglement only marginally improves performance (52.1% vs 51.8%). Indeed, we find that only when using both components, awareness in the encoder and invariance at the classifier, does performance improve significantly (55.1%). These results support our hypothesis that domain awareness is genuinely a prerequisite for effective invariance, since each component is needed for the other to be useful.

Table 4: Ablation of CLIP-DCA components: Domain descriptions (Domain), Disentanglement (Disent.), MLLM Hidden States (MLLM HS), and Avg. accuracy on 33 datasets.

Method / Config.	Domain	Disent.	MLLM HS	Avg.
MLLM (LLaVA)	-	-	-	24.2
FLYP	X	X	X	51.8
Ours	X	O	X	50.7
	O	X	X	52.1
	O	O	X	53.0
	O	X	O	51.9
Ours (full)	**O**	**O**	**O**	**55.1**

MLLM size and synthetic domains. To study the effect of the capacity of the MLLM, we replaced LLaVA-8B with Gemini-2.5-Pro and observed a marginal performance difference (see Table 12 in the Appendix), suggesting our method's efficacy is not primarily dependent on the MLLM's size. In contrast, we find that the diversity of the domains and the accuracy of the descriptions are important (Appendix K). Performance improves with the number of synthetic domains, and permuting or corrupting the descriptions removes the improvement over FLYP. These results are consistent with our central hypothesis that the **balance between domain awareness and disentangled invariance is the critical factor** for robust generalization, rather than the additional data.

Including GCC data. When finetuning CLIP-DCA, we also use the GCC dataset (unless otherwise noted), a dataset with 595,000 image-caption pairs, as a manageable proxy for the data CLIP was originally pretrained on. The image-caption pairs provide valuable supervision particularly for training the text encoder and possibly preventing the discarding of useful features during finetuning on classification datasets like ImageNet.

Table 5: Ablations on GCC inclusion. Accuracy on ImageNet variants and Avg. accuracy on 33 datasets.

Setting	V1	V2	Sketch	A	R	Avg.
Zeroshot	54.2	48.4	32.3	26.2	59.7	51.1
ImageNet only						
FLYP	69.1	58.1	35.3	18.1	55.8	47.0
DANN	69.0	58.0	33.1	18.3	53.7	46.2
CLIP-DCA	**75.2**	**64.0**	**40.5**	**25.5**	**63.1**	**51.5**
ImageNet+GCC						
FLYP	69.9	59.2	38.8	20.5	59.8	51.8
DANN	69.8	59.0	38.5	20.2	59.2	50.5
CLIP-DCA	**75.0**	**64.1**	**42.9**	**26.2**	**65.0**	**55.1**

We study the contribution of the GCC data as shown in Table 5. To ensure a fair comparison, all methods in the ImageNet+GCC setting receive identical access to the auxiliary data, and DANN receives the exact same diffusion images and MLLM descriptions used by CLIP-DCA. A key observation is that the inclusion of GCC provides a benefit even for standard finetuning (FLYP) (Goyal et al., 2023). This shows the general benefit of including diverse, captioned data during finetuning. Despite the general improvements, our method consistently shows higher performance even when the GCC dataset was not included.

Limitations. One concern might be the reliance on synthetically generated diffusion images and MLLM-extracted features for domain awareness. Another limitation is our focus on ImageNet-1K as the source dataset, and extending our evaluation to other source datasets is a valuable direction for future research. However, the reliance on synthetic data is mitigated by: (1) the small size of the diffusion dataset (4096 samples), (2) images synthesized using generic, class-agnostic style prompts, and (3) the MLLM processing multiple style-consistent images, which focuses it on style over objects. Furthermore, DANN (Ganin et al., 2016) and our ablations without disentanglement (Table 4), even with such data, fail to improve CLIP's OOD performance (Table 5).

The role of the MLLM may also be questionable, as LLaVA internally uses a CLIP-L encoder. However, LLaVA's poor zero-shot image classification performance (Table 4), a known issue attributed to MLLMs' improper alignment for classification (Zhang et al., 2024), justifies not using it as a direct classifier. Instead, we use an MLLM because CLIP captures global information from images, which prioritizes overall style (Tong et al., 2024), and the MLLM is then able to explain the perceived domain styles into textual descriptions.

5 Conclusion

In this work, we address a challenge in the robust finetuning of CLIP, where traditional domain invariance regularization can harm the pretrained features that enable generalization. We argue that this does not mean the principle of invariance does not apply to CLIP, but that the place where it is enforced requires change. We introduce CLIP-DCA, built on the hypothesis that domain awareness in the encoder is a prerequisite for domain invariance at the classifier. By injecting domain awareness via synthetic images and MLLM-generated descriptions, and encouraging invariance only at the classification head through disentanglement, CLIP-DCA preserves pretrained knowledge while encouraging classification to rely less on domain-specific signals. We validate our approach by simulating "DG in-the-wild," using both established OOP splits and a 33-dataset multi-modal OOD benchmark. Our ablations show that applying invariance or awareness alone is ineffective, but balancing the two together improves performance on OOD data, supporting our central claim, and representation probing shows that this balance does not remove domain information from the classifier. Overall, our results suggest that grounding robust finetuning in traditional DG principles is a viable direction for foundation models.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets: an MLLM (LLaVA-8B (Liu et al., 2023)) proposes the image styles and describes each style, a text-to-image diffusion model (Stable Diffusion 3 (Esser et al., 2024)) generates the style images used for training (Sec. 2.2), and Gemini-2.5-Pro writes the style descriptions in the MLLM ablation (Appendix L). We have not used generative AI tools to develop theoretical models or conceptual frameworks, propose or refine hypotheses, implement methods, assist with translation, or clean and reformat datasets, and formulating or proving mathematical claims and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to correct grammar and improve the wording of the manuscript. We have reviewed all AI-assisted work. The synthetic data are used only for training, reported results comes from experiments we ran ourselves, and the authors checked all AI-suggested text and code changes before using them. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Sravanti Addepalli, Ashish Ramayee Asokan, Lakshay Sharma, and R Venkatesh Babu. Leveraging vision-language models for improving domain generalization in image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23922–23932, 2024.

Shuanghao Bai, Yuedi Zhang, Wanqi Zhou, Zhirong Luan, and Badong Chen. Soft prompt generation for domain generalization. In *European Conference on Computer Vision*, pp. 434–450. Springer, 2024.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. *Advances in neural information processing systems*, 24, 2011.

Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *European conference on computer vision*, pp. 440–457. Springer, 2022.

De Cheng, Zhipeng Xu, Xinyang Jiang, Nannan Wang, Dongsheng Li, and Xinbo Gao. Disentangled prompt representation for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23595–23604, 2024.

Zhixiang Chi, Li Gu, Tao Zhong, Huan Liu, Yuanhao Yu, Konstantinos N Plataniotis, and Yang Wang. Adapting to distribution shift by visual domain prompt generation. *arXiv preprint arXiv:2405.02797*, 2024.

Junhyeong Cho, Gilhyun Nam, Sungyeon Kim, Hunmin Yang, and Suha Kwak. Promptstyler: Prompt-driven style generation for source-free domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15702–15712, 2023.

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

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in neural information processing systems*, 34:7068–7081, 2021.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595, 2024.

Shantanu Ghosh, Clare B Poynton, Shyam Visweswaran, and Kayhan Batmanghelich. Mammo-clip: A vision language foundation model to enhance data efficiency and robustness in mammography. In *International conference on medical image computing and computer-assisted intervention*, pp. 632–642. Springer, 2024.

Sachin Goyal, Ananya Kumar, Sankalp Garg, Zico Kolter, and Aditi Raghunathan. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19338–19347, 2023.

Seungjin Jung, Kanghee Lee, Yonghyun Jeong, Haeun Noh, Jungmin Lee, and Jongwon Choi. Group-wise scaling and orthogonal decomposition for domain-invariant feature extraction in face anti-spoofing. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13372–13381. IEEE, 2025.

Muhammad Uzair Khattak, Muhammad Ferjad Naeem, Muzammal Naseer, Luc Van Gool, and Federico Tombari. Learning to prompt with text only supervision for vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4230–4238, 2025.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

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

Gyuseong Lee, Wooseok Jang, Jinhyeon Kim, Jaewoo Jung, and Seungryong Kim. Domain generalization using large pretrained models with mixture-of-adapters. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 8259–8269. IEEE, 2025.

Aodi Li, Liansheng Zhuang, Shuo Fan, and Shafei Wang. Learning common and specific visual prompts for domain generalization. In *Proceedings of the Asian conference on computer vision*, pp. 4260–4275, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Jeremiah Liu, Zi Lin, Shreyas Padhy, Dustin Tran, Tania Bedrax Weiss, and Balaji Lakshminarayanan. Simple and principled uncertainty estimation with deterministic deep learning via distance awareness. *Advances in neural information processing systems*, 33:7498–7512, 2020.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pp. 97–105. PMLR, 2015.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Ségolène Martin, Yunshi Huang, Fereshteh Shakeri, Jean-Christophe Pesquet, and Ismail Ben Ayed. Transductive zero-shot and few-shot clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28816–28826, 2024.

Prasanna Mayilvahanan, Roland S Zimmermann, Thaddäus Wiedemer, Evgenia Rusak, Attila Juhos, Matthias Bethge, and Wieland Brendel. In search of forgotten domain generalization. *arXiv preprint arXiv:2410.08258*, 2024.

Keno Moenck, Duc Trung Thieu, Julian Koch, and Thorsten Schüppstuhl. Industrial language-image dataset (ilid): Adapting vision foundation models for industrial settings. *Procedia CIRP*, 130: 250–263, 2024.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International conference on machine learning*, pp. 10–18. PMLR, 2013.

Giung Nam, Byeongho Heo, and Juho Lee. Lipsum-ft: Robust fine-tuning of zero-shot models using random text guidance. *arXiv preprint arXiv:2404.00860*, 2024.

Changdae Oh, Hyesu Lim, Mijoo Kim, Dongyoon Han, Sangdoo Yun, Jaegul Choo, Alexander Hauptmann, Zhi-Qi Cheng, and Kyungwoo Song. Towards calibrated robust fine-tuning of vision-language models. *Advances in Neural Information Processing Systems*, 37:12677–12707, 2024.

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

Nazanin Mohammadi Sepahvand, Eleni Triantafillou, Hugo Larochelle, Doina Precup, James J Clark, Daniel M Roy, and Gintare Karolina Dziugaite. Selective unlearning via representation erasure using domain adversarial training. In *The Thirteenth International Conference on Learning Representations*, 2025.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, 2018.

Yang Shu, Xingzhuo Guo, Jialong Wu, Ximei Wang, Jianmin Wang, and Mingsheng Long. Clipood: Generalizing clip to out-of-distributions. In *International Conference on Machine Learning*, pp. 31716–31731. PMLR, 2023.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pp. 443–450. Springer, 2016.

Piotr Teterwak, Kuniaki Saito, Theodoros Tsiligkaridis, Bryan A Plummer, and Kate Saenko. Is large-scale pretraining the secret to good domain generalization? *arXiv preprint arXiv:2412.02856*, 2024.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9568–9578, 2024.

Matthew Wallingford, Vivek Ramanujan, Alex Fang, Aditya Kusupati, Roozbeh Mottaghi, Aniruddha Kembhavi, Ludwig Schmidt, and Ali Farhadi. Neural priming for sample-efficient adaptation. *Advances in Neural Information Processing Systems*, 36:65566–65584, 2023.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7959–7971, 2022.

Lingxiao Yang, Ru-Yuan Zhang, Yanchen Wang, and Xiaohua Xie. Mma: Multi-modal adapter for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23826–23837, 2024.

Lingxiao Yang, Ru-Yuan Zhang, Qi Chen, and Xiaohua Xie. Learning with enriched inductive biases for vision-language models. *International Journal of Computer Vision*, 133(6):3746–3761, 2025.

Han Yu, Xingxuan Zhang, Renzhe Xu, Jiashuo Liu, Yue He, and Peng Cui. Rethinking the evaluation protocol of domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21897–21908, 2024.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pp. 12310–12320. PMLR, 2021.

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

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *International conference on machine learning*, pp. 7404–7413. PMLR, 2019.

Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and Serena Yeung-Levy. Why are visually-grounded language models bad at image classification? *arXiv preprint arXiv:2405.18415*, 2024.

Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *International conference on machine learning*, pp. 7523–7532. PMLR, 2019.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4396–4415, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16816–16825, 2022b.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022c.

A Extended Related Work

Domain Generalization. Learning domain-invariant representations has historically been a central idea in domain generalization (Blanchard et al., 2011; Zhou et al., 2022a). The intuition is that when classifying images from entirely new distributions, learning abstract features common across source domains should provide better robustness for classification in new domains (Blanchard et al., 2011; Muandet et al., 2013). Among these, domain-adversarial learning methods have become a relatively standard approach within the DG field due to its conceptual simplicity and effectiveness (Zhou et al., 2022a). For instance, Domain Adversarial Neural Networks (DANN) (Ganin et al., 2016) uses an auxiliary domain classifier trained adversarially against the encoder, encouraging the encoder to produce features indistinguishable across source domains. Given the focus of DANN on the central idea of domain invariance, we use DANN as our main invariance baseline, and additionally evaluate other invariance regularization methods (DAN (Long et al., 2015), CORAL (Sun & Saenko, 2016), CDAN (Long et al., 2018), MDD (Zhang et al., 2019)) in Appendix H. Notably, despite the prevalence of such DG methods, the direct application for CLIP is not well-established and remains underexplored. Naively enforcing domain invariance on foundation models like CLIP, with large pretrained knowledge, risks discarding of useful information.

Feature decomposition for DG. A related line of work separates domain-invariant from domain-specific features. GD-FAS (Jung et al., 2025) decomposes CLIP features into two orthogonal subspaces via Gram-Schmidt orthogonalization for face anti-spoofing. Their design fits their setting, which is binary classification with ground-truth domain labels over a fixed number of domains, so there is a single decision boundary per domain to align. Our setting is different. We assume open-vocabulary classification with an arbitrary number of classes, and no domain labels, since the boundaries between domains are blurred in the wild. We therefore generate synthetic domains, supervise a separate domain head with MLLM descriptions, and use a soft disentanglement regularizer at the classification head rather than a hard decomposition, with the goal of retaining pretrained domain awareness in the encoder.

Robust Finetuning of CLIP. The introduction of CLIP marked a significant shift in DG research. The original study (Radford et al., 2021) demonstrated impressive zero-shot classification performance across diverse benchmarks, including OOD datasets. The authors attributed this capability to CLIP learning representations that are less reliant on spurious correlations specific to downstream target datasets, as CLIP was not trained on these specific datasets during its initial pretraining.

The assumption of the inherent OOD robustness in CLIP motivated numerous methods aimed at finetuning CLIP for downstream tasks while enhancing its perceived robustness. A common approach is parameter-efficient finetuning (PEFT) strategies. An early influential study, CoOp (Zhou et al., 2022c), introduced learnable textual prompts, motivated by observations that manually crafted prompt ensembles improved CLIP's zero-shot accuracy. Building on this, CoCoOp (Zhou et al., 2022b) made these prompts dynamic by conditioning them on individual image features through a cross-attention mechanism. Similarly, CLIP-Adapter (Gao et al., 2024) proposed adding lightweight, learnable MLP layers (adapters) to the CLIP encoders, finetuning only these small adapters instead of the entire network. Many more subsequent PEFT methods have also been explored (Cho et al., 2023; Chi et al., 2024; Lee et al., 2025; Addepalli et al., 2024; Bai et al., 2024; Li et al., 2022; Khattak et al., 2025; Cheng et al., 2024; Lafon et al., 2024).

End-to-end finetuning methods have also been explored, yet many still depend on the original pretrained CLIP weights for regularization or guidance. Wise-FT (Wortsman et al., 2022), motivated by observing that standard finetuning often degraded zero-shot OOD performance, ensembles the weights of the finetuned model with the original CLIP weights. CLIP-OOD (Shu et al., 2023) used a beta-moving average of the weights during finetuning alongside a regularization term to enhance semantic relationships learned during pretraining. MIRO (Cha et al., 2022) used mutual information regularization between the finetuning model and the frozen pretrained CLIP model to retain pretrained features.

While many other methods show strong performance on OOD benchmarks, this overview highlights representative approaches, their trends, and assumptions in robust CLIP finetuning. Our work, however, *questions whether current evaluation protocols are sufficiently challenging, and suggests the reliance on the pretrained weights may be suboptimal for true OOD generalization*, a concern supported by evidence of domain contamination during pretraining (Mayilvahanan et al., 2024). Consequently, we explore more challenging evaluations and alternative strategies for training CLIP grounded in DG principles.

B Pseudocode

B.1 Main Training Loop (CLIP-DCA)

```
# Note: logit scales (temperature term) were omitted for simplicity
def classification_loss(image, text):
    logits_per_image = image @ text.T
    logits_per_text = text @ image.T
    labels = torch.arange(len(image), device=device)
    return (F.cross_entropy(logits_per_image, labels) +
            F.cross_entropy(logits_per_text, labels)) / 2

def disentangle_loss(x, y):
    # Standardize using batch statistics for numerical stability
    x = (x - x.mean(0)) / (x.std(0) + 1e-4)
    y = (y - y.mean(0)) / (y.std(0) + 1e-4)
    # Per-sample squared similarity, with batch-size scaling
    return ((x * y).sum(dim=-1) / len(x)).pow(2).sum()

# Architectural additions (training only)
domain_head = nn.Linear(...)
mllm_projector = nn.Linear(...)

# 1. Diffusion data batch
d_images, d_hidden, d_text = d_batch
penultimate_img_emb, class_img_emb, text_emb = clip_model(d_images,
    d_text)
domain_img_emb = domain_head(penultimate_img_emb.clone())
domain_mllm_emb = mllm_projector(d_hidden)

C5 = classification_loss(domain_img_emb, text_emb)   # Domain
    Agreement
C6 = classification_loss(domain_mllm_emb, text_emb) # MLLM Agreement
```

```
C3 = disentangle_loss(class_img_emb, domain_img_emb)
C4 = disentangle_loss(class_img_emb, text_emb)

loss_diffusion = C3 + C4 + C5 + C6

# 2. Source data batch
s_images, s_text = s_batch
penultimate_img_emb, class_img_emb, text_emb = clip_model(s_images, s_text)
domain_img_emb = domain_head(penultimate_img_emb.clone())

C1 = classification_loss(class_img_emb, text_emb)   # Classification Loss
C2 = disentangle_loss(class_img_emb, domain_img_emb)

loss_source = C1 + C2

# 3. Final loss
loss = loss_diffusion + loss_source
```

Algorithm 2: Main training loop for CLIP-DCA.

B.2 Unlearning Loop

```
# Note: logit scales (temperature term) were omitted for simplicity
discriminator = nn.Linear(...)

# Following Ganin et al. (2016)
p = float(batch_idx + start_steps) / total_steps
alpha = 2. / (1. + np.exp(-10 * p)) - 1

def classification_loss(image, text):
    # ... (same as in Algorithm 1)

# 1. Retention on GCC dataset
gcc_images, gcc_captions = gcc_batch
_, image_emb, text_emb = clip_model(gcc_images, gcc_captions)
retention_loss = classification_loss(image_emb, text_emb)

# 2. Unlearning on DomainNet
dn_images, _ = dn_batch
noise = torch.randn_like(dn_images)

# Pass both through encoder; apply gradient reversal to discriminator
features_dn, _ = clip_model.encode_image(dn_images)
features_noise, _ = clip_model.encode_image(noise)

combined_features = torch.cat((features_dn, features_noise), dim=0)
reversed_features = GradientReversalFunction.apply(combined_features, alpha)

domain_logits = discriminator(reversed_features)
domain_targets = torch.cat((torch.ones(len(dn_images)),
                            torch.zeros(len(noise))), dim=0).long()
unlearning_loss = F.cross_entropy(domain_logits, domain_targets)

# 3. Final loss
loss = retention_loss + unlearning_loss
```

Algorithm 3: Unlearning loop with gradient reversal.

C TRAINING DETAILS

For all experiments, we used the CLIP ViT-B/32 model. Models were finetuned on the ImageNet-1k training set for 5 epochs. The official ImageNet validation set was used. We used the AdamW optimizer with a learning rate of 1e-5, with a cosine learning rate scheduler. Due to computational constraints, a consistent batch size of 128 was maintained across all methods. For baseline methods, any additional method-specific hyperparameters were adopted from the default configurations provided in their publicly available codebases. All experiments were conducted with a single NVIDIA RTX A5000 GPU and an AMD EPYC 7763 CPU.

D HYPERPARAMETER TUNING

This section provides details for hyperparameter tuning. In the main manuscript, we report 6 different losses for the distinction between source data and diffusion generated data (C_1 - C_6). For hyperparameter tuning, we group these losses into four terms. C_2 and C_3 are grouped together as the *image disentangle* term, while C_1 and C_4 are grouped together as the *text disentangle* term. We report our hyperparameter tuning in Table 6. Due to computation limitation, the tuning was limited to single term increments.

Table 6: Average accuracy across 33 datasets for specific hyperparameter combinations.

MLLM hidden state	**MLLM description**	**Text disentangle**	**Image disentangle**	**Avg. Acc. (%)**
1×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-4}	50.8
1×10^{-3}	1×10^{-4}	1×10^{-4}	1×10^{-4}	49.6
1×10^{-2}	1×10^{-4}	1×10^{-4}	1×10^{-4}	49.0
1×10^{-1}	1×10^{-4}	1×10^{-4}	1×10^{-4}	50.2
1×10^{-4}	1×10^{-3}	1×10^{-4}	1×10^{-4}	51.3
1×10^{-4}	1×10^{-2}	1×10^{-4}	1×10^{-4}	50.4
1×10^{-4}	1×10^{-1}	1×10^{-4}	1×10^{-4}	49.6
1×10^{-4}	1×10^{-4}	1×10^{-3}	1×10^{-4}	52.1
1×10^{-4}	1×10^{-4}	1×10^{-2}	1×10^{-4}	51.4
1×10^{-4}	1×10^{-4}	1×10^{-1}	1×10^{-4}	51.0
1×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-3}	50.6
1×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-2}	49.6
1×10^{-4}	1×10^{-4}	1×10^{-4}	1×10^{-1}	49.1
1×10^{-4}	1×10^{-4}	1×10^{-3}	1×10^{-4}	52.1 (**Best**)

E PROMPTS USED FOR MULTI-MODAL LANGUAGE MODELS (MLLMS)

This section details the specific prompts provided to Multi-Modal Language Models (MLLMs) for the generation tasks.

E.1 PROMPT TO MLLM TO GENERATE IDEAS FOR DIFFERENT STYLES OF IMAGES

The following prompt was used to instruct the MLLM to generate a diverse list of image style ideas:

```
Give me ideas of 512 different styles of images.
Each style should be less than 5 words. Do not overlap styles.
Make the styles diverse.
Be brief.
```

E.2 Prompt to MLLM to generate descriptions and hidden states

The following prompt was used to instruct the MLLM to generate detailed descriptions of image styles (independent of object category) and to also extract corresponding hidden states. The following prompt was input together with images in each style:

```
Attached are multiple images in the same style.
Describe the aspects of the style that applies regardless of category.
Provide a description.
Do not describe the object in the image, but the style of image.
Be as detailed, complete, and comprehensive as possible.
Explain every minute detail.
```

F Datasets

We report all target datasets used for our experiments. Note that for training, the ImageNet-1K training set was used.

Table 7: List of datasets used in experiments, including number of classes and images. DG benchmarks are listed first. Each domain within a DG benchmark is treated as a distinct dataset. The number of images represent the validation/test set.

Dataset Name	Brief Description	# Classes	# Images
Domain Generalization (DG) Benchmarks			
Digits DG	Collection of digit recognition datasets.	–	–
MNIST	Grayscale handwritten digits (28x28).	10	6,000
MNIST-M	MNIST digits with color patches blended from BSDS500.	10	6,000
SVHN	Colored house numbers from Google Street View images (32x32).	10	6,000
SYN	Synthetically generated digit images (32x32).	10	6,000
Terra Incognita	Camera trap images of wild animals from different locations.	–	–
Location 100	Animal images from Location 100.	10	4,741
Location 38	Animal images from Location 38.	10	9,736
Location 43	Animal images from Location 43.	10	3,970
Location 46	Animal images from Location 46.	10	5,883
PACS	Object recognition with domain shifts.	–	–
Art Painting	Artistic paintings of objects.	7	2,048
Cartoon	Cartoon images of objects.	7	2,344
Photo	Photographic images of objects.	7	1,670
Sketch	Sketch drawings of objects.	7	3,929
Office-Home	Object recognition in different settings.	–	–
Art	Artistic depictions of everyday objects.	65	1,972
Clipart	Clipart images of everyday objects.	65	3,910
Product	Product images of everyday objects (typically clean backgrounds).	65	3,984
Real	Real-world photographic images of everyday objects.	65	3,902
Individual Benchmark Datasets			
Caltech-101	101 object categories (+1 background).	101	8,677
Oxford-IIIT Pets	Images of pet breeds.	37	3,669

Continued on next page

Table 7 – continued from previous page

Dataset Name	Brief Description	# Classes	# Images
Oxford Flowers 102	Images of flower categories.	102	6,149
Stanford Cars	Images of car makes, models, and years.	196	8,041
Food-101	Images of food categories.	101	25,250
FGVC Aircraft	Images of aircraft variants.	100	3,333
SUN397	Scene understanding dataset with scene categories.	397	108,754
Describable Textures Dataset (DTD)	Textures in the wild, organized by 47 human-perceivable attributes.	47	1,880
EuroSAT	Satellite imagery of land use and land cover classes.	10	27,000
UCF101	Action recognition dataset of human action categories from videos.	101	13,320
ImageNet-1K	1.28M natural images in 1000 classes (ILSVRC 2012).	1000	50,000
ImageNet-V2	New test set for ImageNet-1K.	1000	50,889
ImageNet-Sketch	Sketch images corresponding to ImageNet-1K.	1000	50,000
ImageNet-A	"Natural adversarial examples" of 200 classes.	200	7,500
ImageNet-R	"Renditions" (art, cartoons, etc.) of 200 ImageNet classes.	200	30,000
WILDS Benchmark Datasets (Treated as Individual)			
Camelyon17-Wilds	Histopathological images for tumor detection with hospital-based shifts.	2	85,054
FMOW-Wilds	Satellite imagery for land use classification with temporal/regional shifts.	62	53,473

G Full Out-of-Pretraining Results

Table 8 reports the Full, In-Pretraining and Out-of-Pretraining results for all five benchmarks.

Table 8: Performance on Out-of-Pretraining (OOP) splits. Following Teterwak et al. (2024), In-Pretraining is defined as cosine similarity > 0.21 and Out-of-Pretraining as < 0.15. The Full set additionally includes all data in the dataset, not limited to the splits, so Full accuracy is not bounded by the In and Out splits.

Dataset	Split	Zero-shot	DANN	FLYP	Adapter	CoOp	CLIP-OOD	Ours
PACS	Full	93.9	92.6	93.6	94.6	93.9	93.7	**95.3**
	In-Pretraining	95.1	92.6	94.3	95.7	94.3	94.7	**96.2**
	Out-of-Pretraining	82.6	82.3	83.4	84.6	81.0	84.3	**87.1**
TerraInc	Full	16.2	40.7	39.8	29.2	30.3	**47.7**	47.0
	In-Pretraining	16.4	43.8	42.1	31.9	32.1	**51.5**	50.3
	Out-of-Pretraining	12.9	17.8	20.6	8.2	7.0	16.7	**21.9**
VLCS	Full	82.4	80.5	80.6	81.9	82.3	**84.2**	81.2
	In-Pretraining	**94.0**	89.3	90.0	93.1	92.7	92.1	90.4
	Out-of-Pretraining	74.6	75.7	76.2	74.4	72.1	80.5	**82.7**
OfficeHome	Full	78.2	78.8	78.5	78.6	77.3	**83.5**	81.0
	In-Pretraining	80.9	81.2	81.1	81.3	82.0	**85.8**	82.2
	Out-of-Pretraining	45.7	52.5	48.8	47.0	**57.1**	57.0	57.0
DomainNet	Full	55.1	56.3	56.1	57.2	56.6	59.2	**60.7**
	In-Pretraining	66.6	65.8	67.0	68.1	67.9	69.6	**70.2**
	Out-of-Pretraining	23.2	24.5	23.9	24.2	22.2	26.3	**27.7**

H Additional Invariance Regularization Baselines

To check that our findings are not specific to DANN, we evaluate four additional invariance regularization methods. All methods were implemented following TLlib, which is maintained by the authors of CDAN and MDD, and use the same synthetic diffusion images as the domains to align. λ is the trade-off between the task loss and the alignment loss (default 1.0).

Table 9: Invariance regularization methods. Change in average accuracy on 33 datasets relative to FLYP.

Method	**Type**	**Δ vs FLYP**
DAN ($\lambda = 1.0$)	MMD alignment	$+1.1$
CDAN ($\lambda = 1.0$)	Conditional adversarial	$+0.5$
CORAL ($\lambda = 10^6$)	Covariance alignment	$+0.5$
DANN ($\lambda = 1.0$)	Adversarial	-0.3
CORAL ($\lambda = 10^7$)	Covariance alignment	-0.8
MDD ($\lambda = 1.0$)	Margin discrepancy	-3.6
CLIP-DCA	Awareness + head-level invariance	$\mathbf{+3.3}$

I Larger Backbone (ViT-B/16)

Table 10: Average accuracy on 33 datasets with CLIP ViT-B/16.

Method (ViT-B/16)	**Avg. on 33**
Zero-shot	58.2
FLYP	55.3
DANN	52.0
CLIP-DCA	**60.9**

J Representation Probing Details

For Table 3, we freeze each finetuned model, add a linear layer on top of the class head or the domain head, and train it to predict the domain label on OfficeHome (4 domains). We also scale all awareness and invariance weights (every term except the classification loss C_1) by a common multiplier. CKA between the class and domain head embeddings is computed on our 4096 synthetic images.

K Sensitivity to the Synthetic Pipeline

Table 11: Left: effect of the number of synthetic domains (8 images per domain). Right: controls on the style descriptions, reported as the change in average accuracy on 33 datasets relative to FLYP.

Number of domains	**Avg. on 33**
32 (256 images)	48.5
64 (512 images)	49.0
128 (1024 images)	49.9
256 (2048 images)	51.8
512 (4096 images)	52.1

Style descriptions	**Δ vs FLYP**
Correct (full CLIP-DCA)	$+3.3$
Permuted across styles	$+0.4$
Random token sequences (same length)	-0.1
40% of words replaced with filler	-0.5

L Ablation Study on MLLM Scale

To investigate the impact of the Multimodal Large Language Model's (MLLM) scale on our method's performance, we conducted an ablation study where we replaced the LLaVA-8B model used in our main experiments with the significantly larger Gemini-2.5-Pro. As shown in Table 12, the performance difference is marginal. The results from Gemini-2.5-Pro show a slight improvement on less OOD datasets but are nearly identical on more challenging ones. This suggests that the effectiveness of CLIP-DCA is not primarily dependent on the scale or capacity of the MLLM used for generating domain-aware signals. All experiments were run with the default hyperparameters reported in the main paper.

Table 12: Ablation on MLLM Scale: Comparison between LLaVA-8B and Gemini-2.5-Pro using unlearned weights. Performance is reported on ImageNet variants and the average across all 33 target datasets. The results show only a marginal difference, highlighting that our framework is not primarily dependent on the MLLM's scale.

Method / Setting	**INet V1**	**-V2**	**-Sketch**	**-A**	**-R**	**Avg. on 33**
Zeroshot	46.0	40.4	27.4	15.1	51.6	45.5
Regular Finetune	69.8	58.4	34.7	15.0	52.6	43.6
CLIP-DCA (LLaVA-8B)	**75.1**	**63.9**	**42.2**	**22.9**	**62.2**	**52.1**
CLIP-DCA (Gemini-2.5-Pro)	76.1	64.9	42.7	22.9	61.9	52.5

M Unlearning

M.1 Simulating Unseen Domains via Unlearning

Retraining a foundation model like CLIP from scratch while omitting specific domains is computationally prohibitive. To overcome this, we use an unlearning method as a proxy to approximate a model that is not contaminated with domains relevant to our evaluation. This controlled experiment allows us to study the following question: *How do robust fine-tuning methods perform when prior exposure to target-like domains is reduced?* Our results point to weaknesses in existing approaches that rely on pretrained weights.

Table 13: Unlearning effectiveness. ZS: Original zero-shot performance. FT: Baseline fine-tuning on the GCC retention set. Unlearn: Full unlearning combining retention on GCC with adversarial unlearning on DomainNet.

Metric/Data	**ZS**	**FT**	**Unlearn**
Imagenet			
IN 1	54.2	52.0	48.8
IN 2	48.4	45.5	41.8
IN Sketch	32.3	31.5	30.7
IN A	26.2	19.0	18.2
IN R	59.7	56.8	52.7
DomainNet			
Clipart	64.3	67.0	53.0
Infograph	41.6	41.0	34.0
Painting	54.4	53.9	47.0
Real	80.5	80.7	73.3
Sketch	57.9	57.2	45.5
Quickdraw	12.1	8.2	0.3
Avg. on 33	51.1	49.7	45.5

Specifically, we adapt the adversarial learning-based unlearning method (Sepahvand et al., 2025) for domain forgetting. We finetune CLIP (Goyal et al., 2023) using a dual objective. First, to retain general knowledge, we train on a 595,000-image subset of the CC3M dataset (Sharma et al., 2018), referred to as GCC, previously used in LLaVA pretraining (Liu et al., 2023), serving as a manageable proxy for CLIP's original training data. Second, to approximate a scenario where domains similar to DomainNet are removed, we apply domain adversarial training (Ganin et al., 2016) on the DomainNet dataset, which we exclude from our target datasets. We attach a binary classifier to the penultimate layer of the image encoder. During training batches, this classifier is fed representations of random noise (assigned label 0) and images from DomainNet (assigned label 1). The gradient reversal layer (Ganin et al., 2016) forces the image encoder to learn representations that confuse this classifier, making embeddings of DomainNet images and random noise indistinguishable, thereby encouraging the model to unlearn domain-specific features from DomainNet. The unlearning occurs concurrently with standard training on the GCC dataset to preserve CLIP's core capabilities. A pseudocode of the unlearning process is in Appendix B.

Table 14: Accuracy on ImageNet variants (Unlearned weights)

Method	V1	V2	Sketch	A	R
Zeroshot (unlearned)	48.8	41.8	30.7	18.2	52.7
Regular Finetune	69.8	58.4	34.7	15.0	52.6
DANN	70.0	58.2	33.2	16.5	52.0
CLIP Adapter	52.9	45.7	28.4	15.0	51.6
CoOp	53.3	46.2	29.1	16.1	52.8
MMA	71.7	60.0	36.1	7.8	37.0
LwEIB	53.8	46.7	30.4	16.3	54.1
Wise-FT	72.9	61.3	40.0	9.4	43.0
MIRO	74.1	62.7	35.7	7.3	33.2
CLIP-OOD	69.0	58.2	35.3	15.0	45.8
CLIP-DCA (Ours)	**75.1**	**63.9**	**42.2**	**22.9**	**62.2**

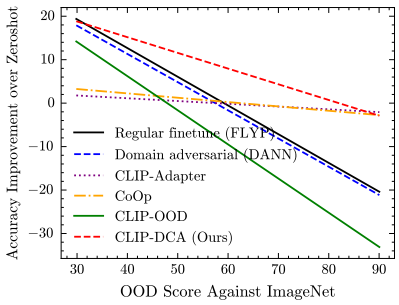


Figure 9: Comparison against baselines after unlearning.

We deliberately unlearn on DomainNet, a dataset we do not use for final evaluation. Unlearning our target evaluation datasets directly would unfairly penalize baseline methods. Many methods are designed to regularize against large deviations from the original pretrained weights. By using DomainNet as a proxy for domain contamination, we ensure a fairer comparison. While we observe a performance drop on DomainNet, Table 13 also shows some degradation in generalized zero-shot performance (e.g., ImageNet-A), likely due to the aggressive nature of mapping features to noise. We interpret these results as a lower-bound test for recovering performance from a checkpoint.

This experimental setup is distinct from other recent proposals. While Teterwak et al. (2024) separate samples based on whether they were learned during pre-training, their focus is on generalization from well-learned to seen-but-unlearned concepts. In contrast, Yu et al. (2024) evaluate domain generalization by training models from scratch without web-scale pre-training. Our approach is a practical middle ground. We measure the performance of a model that benefits from web-scale pre-training but has had specific domain knowledge removed. This allows us to more directly isolate the effect of domain contamination on robust fine-tuning.

M.2 Finetuning after unlearning

To further investigate the impact of potential domain contamination and to establish a more rigorously "unseen" evaluation, we applied the unlearning procedure detailed in Section M.1 to the pretrained CLIP model. We then finetuned this "unlearned" model on ImageNet-1K and evaluated its performance. Table 14 shows the accuracies on the ImageNet variant datasets. Full per-dataset accuracy details for all methods, both before and after unlearning, are provided in Appendix N. For this analysis, we also include several end-to-end robust finetuning methods that add a linear classifier to CLIP. Due to their architecture, these specific baselines are evaluated only on the ImageNet variants as they cannot be adapted to datasets with different class labels.

Our results show that robust end-to-end finetuning methods remain effective for datasets that are less OOD even after unlearning. For instance, MIRO (Cha et al., 2022) and Wise-FT (Wortsman et al., 2022) outperform regular finetuning on ImageNet-V1, ImageNet-V2, and ImageNet-Sketch. To broaden our comparison, we also include other recent CLIP-based DG methods such as MMA (Yang et al., 2024) and LwEIB (Yang et al., 2025), which similarly demonstrate that performance does not consistently generalize to more OOD datasets. However, consistent with the trends seen with the non-unlearned model, performance significantly drops on datasets with larger OOD scores, such as ImageNet-A and ImageNet-R. Similarly, PEFT methods show slight improvements over zero-shot on ImageNet-V1, V2, and Sketch, but their performance drops on ImageNet-A and R.

Figure 9 shows that the performance of all methods, even PEFT methods, further drops as OODness increases across target datasets when finetuning the unlearned model. If the unlearning process successfully reduced the knowledge of target-like domains, existing robust finetuning methods, which rely on the pretrained weights, would struggle on genuinely OOD data. These results suggest that our unlearning approach was effective in simulating a less contaminated starting point.

With the unlearned model, CLIP-DCA shows high performance. For datasets with moderate OOD scores relative to ImageNet, CLIP-DCA achieves larger performance improvements compared to other methods. More importantly, on the extremely OOD datasets, the performance of our method remains close to the zero-shot model, without significant performance drops. This suggests that our mechanism of encouraging domain awareness while selectively enforcing invariance at the decision layer is particularly beneficial when starting from a model with reduced prior exposure to target-like domains. This suggests that the robustness gains of our disentanglement strategy do not rely only on pre-existing knowledge.

N ACCURACY DETAILS

This section provides accuracy details for all target datasets. We report accuracies for both the original weights (before unlearning) and the unlearned weights. The average value across all datasets were reported in the main manuscript.

N.1 USING ORIGINAL WEIGHTS (BEFORE UNLEARNING)

Table 15: Model performance on each dataset for all baselines using the original weights

Dataset Name	Zeroshot	FLYP	DANN	Adapter	CoOp	OOD	Ours
Digits DG							
MNIST	22.4	26.7	27.6	24.2	23.6	15.4	27.8
MNIST-M	16.8	18.4	22.7	11.9	14.9	17.3	16.2
SVHN	16.1	13.1	15.1	11.6	13.1	12.8	12.8
SYN	24.5	21.1	28.0	15.4	16.0	18.3	23.7
Terra Incognita							
Location 100	4.7	21.9	12.3	18.6	18.9	15.2	42.2
Location 38	4.8	32.3	34.7	20.6	20.4	11.4	35.6
Location 43	31.9	30.4	26.1	27.7	27.0	12.5	32.4
Location 46	23.1	32.1	24.3	23.0	22.4	7.2	36.4
PACS							
Art Painting	95.2	91.4	89.3	96.2	96.0	74.5	95.1
Cartoon	96.7	87.4	90.4	96.8	96.5	72.4	95.9
Photo	99.5	99.3	99.3	99.6	99.7	86.7	99.7
Sketch	83.3	75.9	58.7	84.1	84.0	68.8	88.3
Office-Home							
Art	77.5	73.8	72.2	78.3	77.4	63.5	77.6
Clipart	61.4	56.8	56.3	64.1	63.9	58.7	62.2
Product	85.9	78.0	77.2	87.3	86.8	70.0	84.8
Real	86.7	80.3	79.4	88.3	87.6	70.6	86.9
Caltech-101	83.4	84.5	83.0	83.2	83.4	63.1	88.9
Oxford-IIIT Pets	83.9	73.2	74.6	85.9	83.8	64.3	84.6
Oxford Flowers 102	60.1	30.8	31.4	64.6	64.9	9.2	53.2
Stanford Cars	52.2	20.0	21.2	56.4	55.7	1.6	40.6
Food-101	80.2	50.3	51.5	83.6	83.1	19.0	74.9
FGVC Aircraft	16.1	4.4	4.6	17.6	17.5	2.4	12.5
SUN397	60.2	51.8	51.0	57.8	58.3	30.6	63.8
Describable Textures Dataset	40.7	28.8	28.7	40.1	39.6	11.7	39.5
EuroSAT	30.3	26.0	23.9	38.1	38.2	16.2	39.2
UCF101	61.1	48.4	48.8	63.6	63.1	29.4	62.3
ImageNet-1K	54.2	69.1	69.0	59.5	59.9	71.0	75.0
ImageNet-V2	48.4	58.1	58.0	52.9	52.7	60.2	64.1
ImageNet-Sketch	32.3	35.3	33.1	32.3	32.8	40.5	42.9
ImageNet-A	26.2	18.1	18.3	28.5	27.9	13.6	26.2

Continued on next page

Table 15 – continued from previous page

Dataset Name	Zeroshot	FLYP	DANN	Adapter	CoOp	OOD	Ours
ImageNet-R	59.7	55.8	53.7	58.9	58.8	45.2	65.0
Camelyon-Wilds	50.2	50.0	51.0	50.1	50.1	50.0	56.9
FMOW-V2 Wilds	16.5	7.9	9.8	13.2	12.8	11.3	12.9

N.2 USING UNLEARNED WEIGHTS (AFTER UNLEARNING)

Table 16: Model performance on each dataset for all baselines using the unlearned weights

Dataset Name	Zeroshot	FLYP	DANN	Adapter	CoOp	OOD	Ours
Digits DG							
MNIST	33.5	22.4	28.4	29.3	28.9	18.9	40.6
MNIST-M	25.4	16.7	17.1	23.4	24.8	15.8	24.7
SVHN	13.8	13.5	12.0	15.7	16.7	12.0	16.5
SYN	24.6	22.8	18.1	17.4	19.7	13.0	29.0
Terra Incognita	–	–	–				
Location 100	27.8	13.6	22.9	23.2	22.5	9.1	21.5
Location 38	5.8	31.8	27.0	4.7	6.1	2.5	40.4
Location 43	26.5	27.6	25.9	22.2	23.4	9.2	28.1
Location 46	28.4	25.8	30.9	30.0	32.4	5.1	32.0
PACS	–	–	–				
Art Painting	93.8	87.0	86.3	93.9	92.9	65.2	92.1
Cartoon	94.6	82.8	87.8	92.1	94.1	63.4	92.8
Photo	99.5	99.5	99.0	99.0	98.0	80.0	99.6
Sketch	30.0	71.1	32.5	31.2	32.2	58.5	79.7
Office-Home	–	–	–				
Art	68.2	68.1	67.1	68.7	68.7	62.7	76.8
Clipart	50.7	53.7	53.3	47.0	46.7	50.9	61.3
Product	77.2	73.7	73.6	73.2	73.9	61.7	81.3
Real	80.4	76.9	76.9	78.9	78.1	66.6	83.4
Caltech-101	83.1	81.4	80.2	83.2	82.3	69.9	86.5
Oxford-IIIT Pets	74.9	71.0	69.3	74.2	76.4	67.5	81.0
Oxford Flowers 102	43.5	19.3	16.7	43.5	42.6	8.3	45.7
Stanford Cars	30.6	11.1	10.8	30.6	31.8	5.6	39.6
Food-101	66.3	34.3	33.6	65.9	63.4	17.4	62.4
FGVC Aircraft	8.0	2.7	2.2	8.0	7.1	1.1	9.4
SUN397	56.7	44.7	44.9	56.1	54.3	30.8	59.9
Describable Textures Dataset	30.7	25.6	24.9	31.1	30.2	9.8	35.0
EuroSAT	30.5	27.7	28.8	30.0	31.5	14.3	29.0
UCF101	55.5	42.1	41.8	55.1	56.6	36.7	56.9
ImageNet-1K	46.0	69.8	70.0	52.9	53.3	69.0	75.1
ImageNet-V2	40.4	58.4	58.2	45.7	46.2	58.2	63.9
ImageNet-Sketch	27.4	34.7	33.2	28.4	29.1	35.3	42.2
ImageNet-A	15.1	15.0	16.5	15.0	16.1	15.0	22.9
ImageNet-R	51.6	52.6	52.0	51.6	52.8	45.8	62.2
Camelyon-Wilds	50.2	53.0	51.9	60.0	59.7	50.5	55.0
FMOW-V2 Wilds	10.1	8.2	8.7	10.3	9.7	4.7	12.4

O OOD SCORES

This section provides the OOD scores for all target datasets. We report the OOD scores for both the original weights (before unlearning) and the unlearned weights. The average value was used to create the graphs in the main manuscript.

O.1 USING ORIGINAL WEIGHTS (BEFORE UNLEARNING)

Table 17: Out-of-Distribution (OOD) detection scores using original weights.

Dataset Name	SNGP	Label	Average
Digits DG	–	–	–
MNIST	12.4	97.3	54.9
MNIST-M	8.1	97.8	52.9
SVHN	8.8	97.1	52.9
SYN	20.1	95.3	57.7
Terra Incognita	–	–	–
Location 100	10.0	92.5	51.2
Location 38	8.9	95.7	52.3
Location 43	9.5	94.1	51.8
Location 46	7.9	95.5	51.7
PACS	–	–	–
Art Painting	20.4	93.4	56.9
Cartoon	33.6	92.9	63.2
Photo	29.6	80.0	54.8
Sketch	35.1	92.3	63.7
Office-Home	–	–	–
Art	34.8	77.6	56.2
Clipart	28.8	83.2	56.0
Product	45.1	72.0	58.5
Real	43.1	69.6	56.3
Caltech-101	39.3	76.5	57.9
Oxford-IIIT Pets	57.2	62.6	59.9
Oxford Flowers 102	96.8	83.9	90.3
Stanford Cars	98.7	76.2	87.5
Food-101	93.4	77.7	85.6
FGVC Aircraft	97.9	33.7	65.8
SUN397	71.6	76.7	74.2
Describable Textures Dataset	32.3	86.7	59.5
EuroSAT	50.5	98.2	74.3
UCF101	74.4	84.4	79.4
ImageNet-1K	51.6	0.0	25.8
ImageNet-V2	66.3	0.0	33.1
ImageNet-Sketch	85.5	0.0	42.8
ImageNet-A	87.7	0.0	43.9
ImageNet-R	87.8	0.0	43.9
WILDS Benchmark Datasets			
Camelyon-Wilds	0.8	79.2	40.0
FMOW-V2 Wilds	60.5	95.7	78.1

O.2 USING UNLEARNED WEIGHTS (AFTER UNLEARNING)

Table 18: Out-of-Distribution (OOD) detection scores for unlearned model.

Dataset Name	SNGP	Label	Average
Digits DG	–	–	–
MNIST	93.2	40.2	66.7
MNIST-M	97.4	26.4	61.9
SVHN	97.5	2.5	50.0
SYN	98.8	16.4	57.6
Terra Incognita	–	–	–
Location 100	95.1	6.9	51.0
Location 38	95.7	4.6	50.1
Location 43	94.5	15.8	55.1
Location 46	96.7	12.3	54.5
PACS	–	–	–
Art Painting	93.5	32.7	63.1
Cartoon	93.2	39.0	66.1
Photo	79.4	48.0	63.7
Sketch	97.3	19.3	58.3
Office-Home	–	–	–
Art	81.6	43.6	62.6
Clipart	87.0	40.9	64.0
Product	77.9	52.0	65.0
Real	94.7	55.0	74.8
Caltech-101	76.6	55.8	66.2
Oxford-IIIT Pets	68.0	46.4	57.2
Oxford Flowers 102	81.7	98.4	90.1
Stanford Cars	74.7	95.2	85.0
Food-101	78.5	89.6	84.0
FGVC Aircraft	38.5	92.8	65.7
SUN397	77.6	83.4	80.5
Describable Textures Dataset	87.9	53.2	70.6
EuroSAT	98.1	26.3	62.2
UCF101	86.1	81.0	83.5
ImageNet-1K	59.7	0.0	29.9
ImageNet-V2	71.8	0.0	35.9
ImageNet-Sketch	89.6	0.0	44.8
ImageNet-A	88.8	0.0	44.4
ImageNet-R	89.4	0.0	44.7
WILDS Benchmark Datasets			
Camelyon-Wilds	92.9	24.3	58.6
FMOW-V2 Wilds	95.8	63.0	79.4

P LIST OF ALL SYNTHETIC STYLES

This section provides the list of all synthetic style ideas that were generated by the LLM.

Table 19 shows the 512 distinct style prompts used for generating synthetic data. The styles are listed alphabetically across the columns.

Table 19: List of 512 synthetic data generation styles (alphabetical order).

3D rendered image	3D rendering, virtual objects	ASCII art text	ASCII art, text characters

Continued on next page

Table 19 – continued from previous page

Aboriginal dot painting	Abstract expressionism	Abstract expressionism art	Abstract symbolic representation
Abstract, non-representational	Abstract, non-representational form	Achromatic grayscale image	Achromatic, no color
Acrylic paint vibrant	Action painting, dynamic	Aerial drone footage	Afrofuturism cultural sci-fi
Algorithmic art, code-based	Ambient light, natural tones	American scene painting	Analog film, imperfections
Analogous colors harmony	Anamorphic distorted perspective	Ancient Egyptian hieroglyphs	Animal at rest
Animal drinking, water source	Animal eye contact	Animal fighting, intense conflict	Animal grooming, self-care
Animal hiding, partially obscured	Animal hunting, focused gaze	Animal looking away	Animal marking territory
Animal mid-stride	Animal playing, energetic	Animal sleeping, peaceful	Animal tracks, foreground focus
Animal vocalizing, mouth open	Anime Japanese animation	Anime, Japanese animation	Architectural, building structures
Art Deco geometry	Art Nouveau curves	Artificial light controlled	Arts and Crafts
Assemblage found objects	Assemblage, 3D collage	Astrophotography star trails	Asymmetrical dynamic balance
Asymmetry, unbalanced design	Augmented reality, overlaid	Autumn leaves, warm palette	Available light natural
Avant-garde, experimental	Backlit silhouette lighting	Backlit subject, glowing outline	Baroque dramatic lighting
Bio art, living organisms	Biopunk organic technology	Biopunk, genetic engineering	Bird's-eye view elevated
Bird's-eye view, distant	Black and white film	Blacklight fluorescent colors	Blooming flowers, vibrant colors
Blue hour twilight	Blueprint architectural plan	Body art, human canvas	Bokeh light effect
Bold geometric patterns	Boomerang, looping video	Botanical art, plant subjects	Bright cheerful aesthetic
Broad lighting face	Butterfly lighting beauty	Byzantine mosaic icons	Calligraphy elegant lettering
Calligraphy, elegant handwriting	Camera flash, harsh light	Camouflaged animal, hidden	Candid street photography
Candid, unposed moment	Caricature exaggerated features	Cartoon simplified drawing	Cartoon, exaggerated features
Cave entrance, dark frame	Cave painting prehistoric	Charcoal sketch drawing	Charcoal sketch, rough lines
Chibi cute style	Chromatic aberration, color fringing	Cinemagraph, subtle movement	Claymation stop-motion animation
Clear sky, bright blue	Close up macro	Close-up, animal portrait	Close-up, feather detail
Close-up, fur texture	Close-up, scale pattern	Cloudscape art, sky scenes	Collage mixed media
Collage, mixed media	Color contrast, complementary	Color field painting	Color grading cinematic
Color splash accent	Color temperature cool	Color temperature warm	Comic book style
Comic book, panel style	Complementary colors contrast	Conceptual, idea-driven	Cross polarization, vibrant colors
Cross-hatching line work	Cross-processed film	Cross-processed, altered colors	Cubism, geometric shapes
Cubist geometric forms	Cyanotype process print	Cyanotype, blue print	Cyberpunk cityscape night
Cyberpunk style, dystopian future	Daguerreotype antique look	Dark ominous undertones	Data bending corrupted
Data visualization, information art	Decorative, ornamental style	Decoupage, glued paper cutouts	Deep focus sharp

Continued on next page

Table 19 – continued from previous page

Delicate fine details	Dense jungle, lush green	Depth of field	Depth of field, blurred/sharp
Desaturated muted colors	Desaturated, almost monochrome	Dieselpunk retro-futuristic	Different species together
Diffuse reflection matte	Digital art, computer-generated	Digital glitchy aesthetic	Digital noise, grain effect
Digital painting software	Digital print, inkjet/laser	Distortion, warped perspective	Documentary style, realistic
Doodle art casual	Double exposure overlay	Double exposure, ghost image	Dramatic low-key lighting
Dramatic sky, storm clouds	Dramatic spotlight, single source	Dreamy ethereal soft focus	Duotone color scheme
Duotone, two-color palette	Dusty trail, arid environment	Dutch angle tilted	Dynamic energetic composition
Earth art, natural materials	Embroidery thread texture	Engraving detailed metal	Engraving, incised lines
Environmental art, nature-focused	Environmental portrait, surroundings	Establishing shot context	Etching fine lines
Etching, acid-etched lines	Expressionist bold colors	Extreme close-up detail	Extreme close-up, eye detail
Fantasy art, mythical creatures	Fashion illustration stylish	Fashion, clothing focus	Fast motion, sped-up action
Fauvist wild beasts	Feeding animals, close action	Film noir style	Fine art, aesthetic focus
Fish-eye lens view	Fish-eye lens, distorted	Flat lay top-down	Flowing river, blurred water
Focus stacking, all sharp	Foggy morning, atmospheric haze	Folk art naive	Folk art, traditional craft
Forced perspective trick	Forced perspective, size illusion	Formal, posed shot	Found object art
Fractal art mathematical	Fractal art, mathematical	Frontlit subject, clear view	Frozen lake, icy surface
Full shot composition	Futuristic sci-fi vision	Futuristic style, advanced	Generative art algorithmic
Generative art, algorithms	Geometric abstract pattern	Glamour, idealized beauty	Glassblowing molten glass
Glitch art digital	Glitch art distortion	Glitch art, corrupted data	Glitch art, digital errors
Golden hour sunlight	Golden ratio composition	Golden ratio, proportions	Gothic art, dark, romantic
Gothic dark shadows	Gouache opaque matte	Graffiti art, street tagging	Graffiti wildstyle lettering
Graffiti, tagged look	Grainy film texture	Grainy film, retro style	Graphic design, visual communication
Gritty black and white	Gritty urban decay	Group of animals, social	Group portraiture, multiple people
HDR photo rendering	HDR, high dynamic range	Halation, glowing highlights	Hand-drawn sketchy feel
Hard light defined	Hard light shadows	Heavily textured impasto	High contrast, dramatic lighting
High saturation, vivid colors	High-angle shot looking	High-key bright lighting	High-speed photography
Holographic iridescent effect	Horror art, scary imagery	Hudson River School	Hyperlapse, moving time-lapse
Hyperrealism, beyond realism	Illuminated manuscript gold	Illustrative, narrative imagery	Impressionism, loose brushstrokes
Impressionist brushstrokes	Industrial mechanical elements	Infographic data visualization	Infrared luminescence, glowing foliage
Infrared photography	Infrared, false color	Infrared, heat signature	Ink wash fluid

Continued on next page

Table 19 – continued from previous page

Installation art, three-dimensional	Interactive art, participation	Isometric projection view	Jewelry intricate design
Kinetic art movement	Kinetic art, movement	Land art earthworks	Landscape art, natural scenery
Leading lines perspective	Leading lines, guide eye	Lens flare sunlight	Lens flare, bright streaks
Light art, illumination	Light leak, color streaks	Light painting trails	Line art contour
Line drawing, simple outline	Linocut bold lines	Lithograph stone print	Lithography, planographic print
Lomography film look	Long exposure shot	Long exposure, light trails	Long shot distance
Loop lighting portrait	Low angle, animal towering	Low poly geometric	Low saturation, muted tones
Low-angle shot upwards	Low-key dark lighting	Lowbrow art, underground	Lowbrow pop surrealism
Luminism glowing light	Macro lens close-up	Macro shot, tiny details	Mandala circular symmetry
Manga graphic novel	Map cartographic representation	Maximalist busy composition	Maximalist, elaborate design
Medium shot framing	Metalwork shaped metal	Migrating herd, vast landscape	Miniature diorama world
Miniature effect, tilt-shift	Minimalism, essential elements	Minimalist simple lines	Minimalist, simplified design
Mixed lighting combined	Monochrome single color	Monochrome, single color	Monotype, unique print
Moody atmospheric lighting	Moonlit night, stark shadows	Mosaic tile pieces	Mosaic, small piece patterns
Motion blur capture	Motion blur, animal running	Motion blur, speed lines	Mountain range, panoramic view
Multiple exposure, layered images	Mural art, large-scale painting	Mural large-scale painting	Naive art, childlike simplicity
Natural organic forms	Negative space drawing	Negative space framing	Negative space, empty area
Neoclassical refined style	Neon light glowing	Nesting birds, detailed feathers	Night photography cityscape
Night vision, green tint	Night vision, red tint	Oil painting texture	Oil painting, thick texture
Op Art optical	Op art, visual illusions	Optical illusion, trickery	Origami, paper folding
Ornate intricate design	Orthochromatic film effect	Outrun style neon	Outsider art raw
Outsider art, untrained	Over-the-shoulder perspective shot	Overcast sky diffusion	Overexposed, bright white
Overgrown, vegetation focus	Panning motion blur	Panning, blurred background	Panoramic stitched view
Panoramic, wide environment	Paper cut layered	Paper marbling, swirling patterns	Parent and offspring
Pastel drawing, blended colors	Pastel soft blending	Pattern repetition, visual rhythm	Pen and ink
Pencil shading detailed	Performance art, live action	Photojournalism, storytelling	Photorealism hyperdetailed
Photorealism, lifelike detail	Pinhole camera image	Pixel art retro	Pixel art, retro game
Pixelated low resolution	Pixelated, low resolution	Point-of-view subjective shot	Pointillism, tiny dots
Pointillist dot technique	Polaroid transfer, image manipulation	Polychrome, many colors	Pop art bright
Pop art, bold colors	Portraiture, individual likeness	Positive space, subject focus	Pottery ceramic art

Continued on next page

Table 19 – continued from previous page

Pre-Raphaelite detailed beauty	Predator-prey interaction	Psychedelic art, mind-altering	Quilling, paper filigree
Quilting patchwork design	Rack focus shift	Radial balance, circular focus	Rainy day, blurred drops
Realist everyday life	Rembrandt lighting portrait	Renaissance classical style	Retro style, vintage look
Rocky terrain, jagged edges	Rococo ornate details	Romantic emotional landscape	Rule of thirds
Rule of thirds, composition	Rustic textured surface	Sandy desert, dunes stretch	Saturated vibrant colors
Schematic diagram layout	Sci-fi art, space, technology	Screen printing bold	Screen printing, stencil print
Sculpture three-dimensional form	Seascape art, ocean views	Selective color isolation	Selective focus, sharp animal
Self-portraiture, artist's image	Sepia tone, vintage look	Sepia toned photograph	Serene calming atmosphere
Shallow focus blur	Sharp contrasting lines	Sharp focus, crisp details	Short exposure, frozen motion
Short lighting slimming	Sidelit subject, defined features	Silhouette backlit subject	Silhouette black shape
Silhouette, dark shape	Single-point lighting setup	Sleek modern minimalist	Slow motion, extended time
Smooth airbrushed finish	Smooth digital, clean look	Snowy scene, whiteout effect	Social realism commentary
Soft focus, dreamy effect	Soft light diffused	Soft pastel hues	Softbox diffused light
Solar punk green	Solitary animal, minimalist	Sound art, auditory	Sparse woodland, bare trees
Specular highlights reflections	Split lighting dramatic	Split toning effect	Split toning, colored highlights/shadows
Spring growth, fresh shoots	Square color scheme	Staged photography	Stained glass colorful
Steampunk Victorian sci-fi	Steampunk style, Victorian sci-fi	Stencil art spray	Stencil art, cut-out shapes
Stippling dot shading	Stop motion, frame-by-frame	Street art graffiti	Street art, urban style
Studio portrait lighting	Sumi-e ink painting	Summer heat, shimmering air	Sunlit glade, dappled light
Sunrise glow, warm tones	Sunset silhouette, golden hour	Surreal, bizarre, dreamlike	Surrealism, dreamlike imagery
Surrealist dreamlike scene	Symmetrical balanced framing	Symmetry, balanced image	Technical drawing precise
Telephoto, compressed perspective	Tessellated repeating design	Tetradic color rectangle	Textile art fabric
Texture contrast, rough/smooth	Thermal imaging, body heat	Three-point lighting classic	Tilt-shift effect
Time-lapse sequence frame	Time-lapse, motion sequence	Time-lapse, star trails	Tintype vintage photo
Tonal contrast, light/dark	Tonalism muted colors	Triadic color scheme	Tritone, three-color scheme
Trompe-l'oeil illusionistic	Two animals, interaction	Two-point lighting setup	Two-shot composition framing
Typography, letterforms art	Ukiyo-e Japanese woodblock	Ultraviolet, unseen spectrum	Underexposed, deep shadows
Underwater photography scene	Underwater, murky view	Urban art, cityscapes	Vanishing point perspective
Vaporwave aesthetic photo	Vector graphic, stylized	Vector graphics scalable	Vibrant color explosion
Vignette, darkened edges	Vignetting, dark corners	Vintage Polaroid picture	Vintage retro charm

Continued on next page

Table 19 – continued from previous page

Virtual reality, immersive	Visionary art, spiritual	Water reflection, mirrored image	Watercolor painting, soft edges
Watercolor wash effect	Waterfall cascade, misty spray	Wet collodion, antique photography	Wet plate collodion
Wheatpaste poster art	Whimsical playful style	Wide shot, animal small	Wide-angle perspective
Wide-angle, forest scene	Wildlife art, animal subjects	Winter frost, intricate patterns	Wood carving relief
Woodcut print rustic	Woodcut, relief print	Worm's-eye view low	X-ray skeletal view
X-ray vision, skeletal structure	Zentangle intricate patterns	Zoom burst effect	Zoom burst, radial blur