# IS LARGE-SCALE PRETRAINING THE SECRET TO GOOD DOMAIN GENERALIZATION?

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Multi-Source Domain Generalization (DG) is the task of training on multiple source domains and achieving high classification performance on unseen target domains. Recent methods combine robust features from web-scale pretrained backbones with new features learned from source data, and this has dramatically improved benchmark results. However, it remains unclear if DG finetuning methods are becoming better over time, or if improved benchmark performance is simply an artifact of stronger pre-training. Prior studies have shown that perceptual similarity to pre-training data correlates with zero-shot performance, but we find the effect limited in the DG setting. Instead, we posit that having perceptually similar data in pretraining is not enough; and that it is how well these data were learned that determines performance. This leads us to introduce the Alignment Hypothesis, which states that the final DG performance will be high if and only if alignment of image and class label text embeddings is high. Our experiments confirm the Alignment Hypothesis is true, and we use it as an analysis tool of existing DG methods evaluated on DomainBed datasets by splitting evaluation data into In-pretraining (IP) and Out-of-pretraining (OOP). We show that all evaluated DG methods struggle on DomainBed-OOP, while recent methods excel on DomainBed-IP. Put together, our findings highlight the need for DG methods which can generalize beyond pretraining alignment.

## 1 INTRODUCTION

Domain Generalization (DG) addresses the challenge of enabling AI models to generalize from known domains to unseen ones, a critical task given the inevitable distribution shifts between training and real-world deployment (Saenko et al., 2010). DG pipelines typically consist of three stages: pretraining a model on a large, general dataset; finetuning the model with one or more source domains; and finally evaluating the model on target domains that are distinct from source domains. More and more, DG methods rely on huge-scale foundation models for initialization. Simultaneously, finetuning has increasingly incorporated regularization to prevent catastrophic forgetting. As a result, it remains unclear whether DG adaptation methods are genuinely improving or if enhanced benchmark performance is simply due to stronger pre-training combined with regularization, or even the presence of target domains within the hundred million-scale pre-training data.

In this work, we examine the reliance of recent DG methods on pre-trained features. We focus on CLIP-based models, as they are used in all state-of-the-art DG methods (Addepalli et al., 2024; Cho et al., 2023; Shu et al., 2023; Mao et al., 2024); we believe extensions to pure vision models represent interesting future work. While prior studies (Mayilvahanan et al., 2024) have shown that perceptual similarity to pre-training data explains zero-shot performance—referred to as the Image Similarity Hypothesis—we find this relationship to be limited in the DG setting. Despite evidence of target domains being present in pre-training (Figure 4), perceptual similarity alone does not fully explain accuracy in the DG context. We propose that it is not just the presence of similar data in pre-training that matters, but also how well this data was learned. To this end, we introduce the Alignment Hypothesis, which states that pre-trained alignment between image and class embeddings is still predictive of DG performance even after source finetuning. We find that, performance for low alignment samples can be almost 0, while performance for high alignment samples is close to perfect. These results confirm the Alignment Hypothesis. We note that we do not make assumptions of how or why alignment arose. As illustrated in Figure 1, these findings suggest that current DG

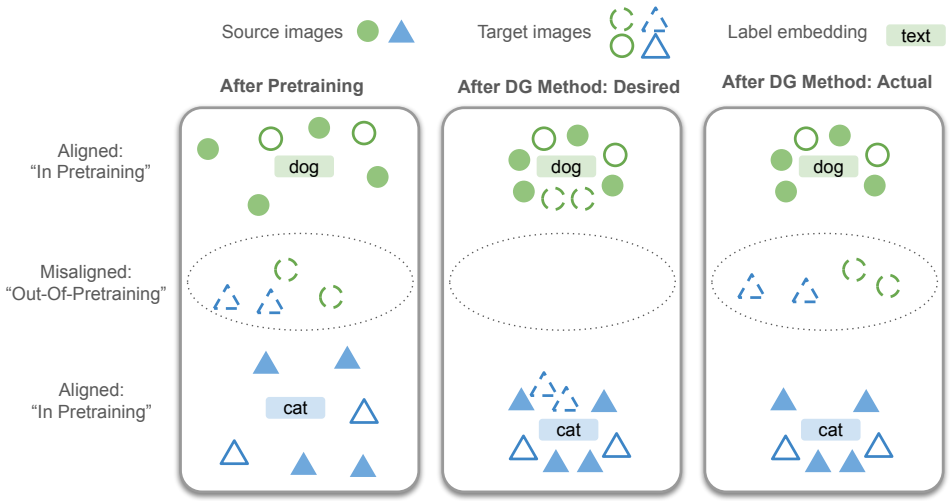


Figure 1: **An overview of desired and actual behaviour of DG methods. 1)** DG methods are initialized with foundation models like CLIP. Pre-trained embeddings are relatively well aligned with ground truth labels on both source and target data for most samples (In-Pretraining, IP), but some samples are not well aligned (Out-of-pretraining, OOP). **2)** An ideal DG method would strengthen alignment for both OOP and IP data with ground truth labels. **3)** Our analysis shows that DG methods only result in strong alignment for IP data, leaving OOP data misaligned (Figure 2).

methods largely fail to learn new, general features from the source data when the pretraining does not already provide a strong alignment.

The confirmation of the Alignment Hypothesis gives us a tool to separate aligned and well learned in-pretraining (IP) data from misaligned and poorly learned out-of-pretraining (OOP) data for a particular backbone, and we do so for five DG datasets with OpenCLIP-ViT/B-16. We call the resulting splits DomainBed-IP and DomainBed-OOP. Evaluating on DomainBed-IP/OOP offers a view of where current DG methods fail and where they succeed. We find that all methods, including those considered state-of-the-art, perform poorly on OOP data, i.e. data that the pretrained backbone hadn't already aligned well. Furthermore, recent state-of-the-art methods do not outperform older methods on OOP data. For example, CLIPood (Shu et al., 2023) slightly under performs a combination of older methods (MIRO (Cha et al., 2022) + MPA (Arpit et al., 2022)) on DomainBed-OOP. At the same time, existing DG methods show exceptional performance on DomainBed-IP, sometimes even outperforming an oracle model *trained on the target domain*. These results suggest that future research should aim to enhance DG methods on low-alignment data while preserving the already strong performance on high-alignment data. In summary, we make the following contributions:

- **Introduce the Alignment Hypothesis**: We demonstrate that pre-training alignment between image and class text embeddings is a stronger predictor of Domain Generalization (DG) success than the previously proposed Image Similarity Hypothesis (Mayilvahanan et al., 2024). Based on this, we define In-Pretraining (IP) as data well-aligned with pre-trained embeddings, and Out-of-Pretraining (OOP) as data with weaker alignment.

- **Propose a new IP/OOP evaluation framework**: We demonstrate that splitting target data by its alignment with the pre-trained backbone can effectively test Out-of-Pretraining (OOP) generalization. We will release IP/OOP splits for the DomainBed dataset to support future research.

- **Expose strengths and limitations of state-of-the-art DG methods**: Using DomainBed-IP/OOP we find that leading DG methods perform well on data well-aligned by pre-training but struggle on misaligned samples, emphasizing the need for methods that move beyond reliance on pre-training.

## 2 RELATED WORK

**Multi-Source Domain Generalization:** Domain Generalization aims to mitigate the impacts of domain shifts between source (training) and target (deployment) domains. These can include subpopulation shifts where all testing subpopulations are present in the training data but in different proportions(Dehdashtian et al., 2024), or it could be the case we consider in this work where the testing subpopluation is not at all present in the training subpopulation. One standard approach is **domain-invariant feature learning**, which leverages domain labels to learn domain-invariant features. CORAL (Sun & Saenko, 2016) aligns second-order statistics, while DANN (Ganin et al., 2016) uses an adversarial loss. Gulrajani & Lopez-Paz (2020) show that ERM, which does not align features between domains, can outperform most prior work while being easier to tune. Another common approach is **domain-aware data augmentation** to expand the training domain to become closer to or even overlap the target domain. Inter-domain mixup (Yan et al., 2020) blends images from different domains. Similarly, style transfer can diversify training images (Zhong et al., 2022). **Deep ensembles** are effective for domain generalization (Arpit et al., 2022). Since they are computationally inefficient for inference, many recent works average model weights from either multiple finetuning runs or from a single training trajectory (Cha et al., 2021; Arpit et al., 2022; Rame et al., 2022; Jain et al., 2023; Li et al., 2023; Shu et al., 2023). More recently, several methods perform **regularized finetuning** towards the initialization of a pretrained model. This works under the assumption that pretrained features are useful for target data, and should not be unlearned. The general idea can be applied to weight space (L2SP (Xuhong et al., 2018)), feature space (MIRO (Cha et al., 2022)), or output space (CAR-FT (Mao et al., 2024),CLIPood (Shu et al., 2023)).

**Large-Scale Pretraining for DG:** Recent DG literature (Cho et al., 2023; Cha et al., 2022; Addepalli et al., 2024; Mao et al., 2024; Arpit et al., 2022) leverages large-scale pretrained initializations stronger than ImageNet (Russakovsky et al., 2015), and CLIP (Radford et al., 2021) is the most common choice. CLIP leverages a cross-domain contrastive loss to align images and captions. Due to the large scale of training data (typically at least 400 million samples) and the free-form nature of the text, CLIP enables effective zero-shot classification and learns features that generalize very well. Other choices for very strong pretraining include SWAG (Singh et al., 2022) and DinoV2 (Oquab et al., 2023). SWAG uses supervision from Instagram hashtags, while DinoV2 is trained without text supervision and instead relies on augmentation-based alignment. While our analysis focuses on image-text models like CLIP due to its popularity, the concept of alignment can extend to other types of pretraining models. We leave the exploration of this extension to future work.

**Impact of Data on Model Performance:** Several recent studies have explored the influence of pretraining data on model performance. Mayilvahanan et al. (2024) investigated how the presence of perceptually similar images in CLIP (Radford et al., 2021) pretraining affects performance, introducing the Similarity Hypothesis, which posits that nearest neighbor similarity is strongly correlated with zero-shot accuracy. Udandarao et al. (2024) demonstrated that concept frequency in pretraining is correlated with zero-shot performance and introduced a dataset focusing on infrequent concepts. Fang et al. (2022) found that diversity in pretraining data is critical for improving performance on benchmarks such as ImageNetV2 (Recht et al., 2019), ImageNet-R (Hendrycks et al., 2021), ImageNet-Sketch (Wang et al., 2019), and ObjectNet (Barbu et al., 2019). However, these studies focus on the zero-shot setting, where models are evaluated without further training. In contrast, we examine the domain generalization setting, where pre-trained models are fine-tuned on source domains and tested on held-out target domains. Yu et al. (2024) recommend using self-supervised pre-training to avoid data leakage. In contrast we study DG model behaviour in the more realistic setting of CLIP-pretraining. Our findings suggest that comparing target images to pre-trained images, as proposed by Mayilvahanan et al. (2024), is less predictive of final DG performance than directly measuring the alignment between the image and its class embedding.

## 3 ANALYZING THE ROLE OF PRETRAINING IN DOMAIN GENERALIZATION

This work explores Multi-Source Domain Generalization for classification, where samples from multiple source domains (e.g., sketches, product photos) and a held-out target domain (e.g., wildlife camera images) are annotated with both domain and class labels. We construct a training dataset by aggregating all sample-label pairs from all training domains $d \in \{d_1, \ldots, d_n\}$, denoted as

$$D = \{(X^{d_1}, Y^{d_1}), \ldots, (X^{d_n}, Y^{d_n})\}.$$

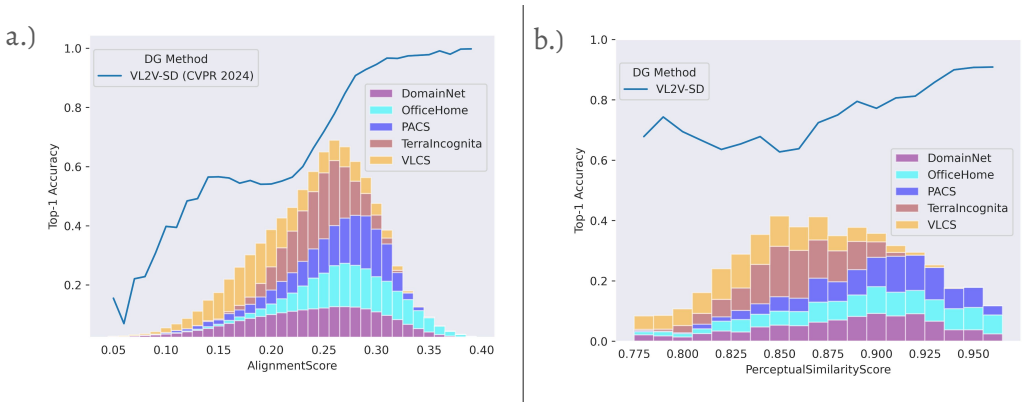


Figure 2: **Comparing the Predictive Power of the Alignment and Image Similarity Hypotheses for Domain Generalization (DG). a.) Alignment Hypothesis**: The cosine similarity between image and ground truth text-label embedding after pre-training (Alignment Score) is highly predictive of model accuracy after fine-tuning on five DG datasets, with Alignment Score distributions shown in the colored histograms. This suggests that image-text pairs well-aligned during pre-training result in better performance on target tasks. **b.) Image Similarity Hypothesis:** In contrast, the cosine similarity between a test image and its closest match from the pre-training set (Perceptual Similarity Score) shows weaker predictive power for accuracy, implying that visual resemblance alone is less indicative of downstream performance.

We initialize a classifier $f$ with a contrastively pre-trained vision-language model (e.g., CLIP) and finetune it on $D$. The scale of pre-training datasets is many orders of magnitude larger than that of source datasets. Most methods fully fine-tune $f$, though LP-FT (Kumar et al., 2021) fine-tunes the linear probe before the main network and Attention Tuning (Teterwak et al., 2023; Touvron et al., 2022) only tunes attention layers. The performance is then evaluated on a held-out testing domain $d_{\text{test}}$. The key assumption is that $d_{test}$ has a different distribution from the source domains.

We aim to analyze how reliant existing DG methods are on pre-training. A recent analysis of CLIP proposed the Image Similarity Hypothesis (Mayilvahanan et al., 2024), which supposes that high CLIP performance on a given test sample is a result of highly similar nearest-neighbor images in pre-training, and tested it on zero-shot classification tasks. They found a strong correlation between nearest-neighbor similarity and zero-shot classification performance, but did not analyze OOD performance after fine-tuning. Therefore, we apply an equivalent testing setup for the DG setting, where a pre-trained model is fine-tuned on a source distribution and tested on a different target distribution. We find only a limited influence of image similarity in Section 3.1. To better understand the role of pretraining in domain generalization, we introduce the Alignment Hypothesis, which we explore in detail in Section 3.2. We later use the Alignment Hypothesis to split DG datasets and analyze existing DG methods (Section 4).

### 3.1 Testing the Image Similarity Hypothesis

The **Image Similarity Hypothesis** (Mayilvahanan et al., 2024) posits that test performance improves when there are perceptually similar images in the pre-training dataset. The PerceptualSimilarityScore measures perceptual similarity and is defined as the cosine similarity between a target image $I$ and its nearest neighbor $I_k$ in pre-training:

$$\text{PerceptualSimilarityScore}(I, I_k) = \frac{\langle f_I(I), f_I(I_k)\rangle}{\|f_I(I)\| \cdot \|f_I(I_k)\|} \qquad \text{(Eq. 1)}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product, and$\| \cdot \|$ denotes the Euclidean norm (magnitude). To evaluate the Image Similarity Hypothesis, we group held-out target domain samples from five DomainBed datasets based on their PerceptualSimilarityScore and compute the accuracy of a Domain Generalization (DG) adaptation method. Specifically, we build a histogram using Perceptual Similarity Score as described Algorithm 1, and visualize it in Figure 2. The PerceptualSimilarityScore is com-

**Algorithm 1** Evaluating the Image Similarity Hypothesis

**Require:** Target domain samples $D_{\text{target}}$, trained DG model $M$, image encoder $f_I$,
1: **for** each sample $I \in D_{\text{target}}$ **do**
2: Retrieve nearest neighbor of $I$ in Laion-400M, assign to $I_k$
3: Compute PerceptualSimilarityScore$(I, I_k)$ using Equation Eq. 1,
4: Record correctness of $M(I)$
5: **end for**
6: Bin samples based on PerceptualSimilarityScore
7: Compute DG accuracy within each bin
8: **return** Accuracy for each bin

puted using approximate nearest neighbors over the LAION-400M dataset (Schuhmann et al., 2021) with the CLIP-retrieval library (Beaumont, 2022).

We evaluate the recent, high-performing *VL2V-SD* (Addepalli et al., 2024) method. The results are shown in Figure 2 (b). While the Image Similarity Hypothesis is somewhat predictive of DG performance, its influence is not very strong. This suggests that perceptually similar pretraining data alone may not guarantee high DG performance; additional factors, such as how effectively the pretraining data were learned, may also be significant.

### 3.2 Introducing the Alignment Hypothesis

To find a stronger predictor of DG accuracy than perceptual similarity, we focus on how effectively pre-training captures the relationship between an image and its label. This leads us to propose the **Alignment Hypothesis**, which states that if an input image and its corresponding text label (e.g., 'A photo of a {cls}') are well-aligned in the embedding space, final DG performance will be high. Crucially, alignment is measured before source fine-tuning while DG performance is measured after adaptation. This allows us to isolate the contribution of fine-tuning. Since models like CLIP optimize image-text pairs using a contrastive loss, cosine similarity between image and text embeddings is an alignment measure well coupled to their training objective. Therefore, we use it as our metric of pre-training generalization. More formally:

$$\text{AlignmentScore}(I, T) = \frac{\langle f_I(I), f_T(T) \rangle}{\|f_I(I)\| \cdot \|f_T(T)\|} \qquad \text{(Eq. 2)}$$

where $f_I(I)$ is the embedding of the image before finetuning on source, and $f_T(T)$ is the embedding of the text.

We verify the Alignment Hypothesis similarly to the Image Similarity Hypothesis, by binning samples using the AlignmentScore and computing accuracy for each bin using VL2V-SD. We provide the same analysis for many more DG methods in the Appendix Figure 10. In Figure 2 a.), we can see that the Alignment Hypothesis explains DG performance after source finetuning, significantly more strongly than for the Image Similarity Hypothesis in Figure 2 b.) This finding suggests that source fine-tuning in DG, which aims to achieve high performance across all target samples, only succeeds on those with high initial alignment.

## 4 Re-thinking Domain Generalization Benchmarking using the Alignment Hypothesis

Knowing that the Alignment Hypothesis holds for contrastively trained image-text models (Section 3), we can now use it as a tool to probe the performance of DG methods across different levels of pre-training alignment. We apply this approach to five widely-used DomainBed (Gulrajani & Lopez-Paz, 2020) DG datasets: VLCS (Fang et al., 2013), PACS (Li et al., 2017), OfficeHome (Ganin et al., 2016), TerraIncognita (Beery et al., 2018), and DomainNet (Peng et al., 2019). This section discusses how we create new splits for existing DG datasets using AlignmentScore. We start by computing AlignmentScore for all samples in 5 DG datasets (Figure 3). Based on our observation that some samples are mislabeled, we perform dataset cleaning (4.1). Finally, we find an AlignmentScore

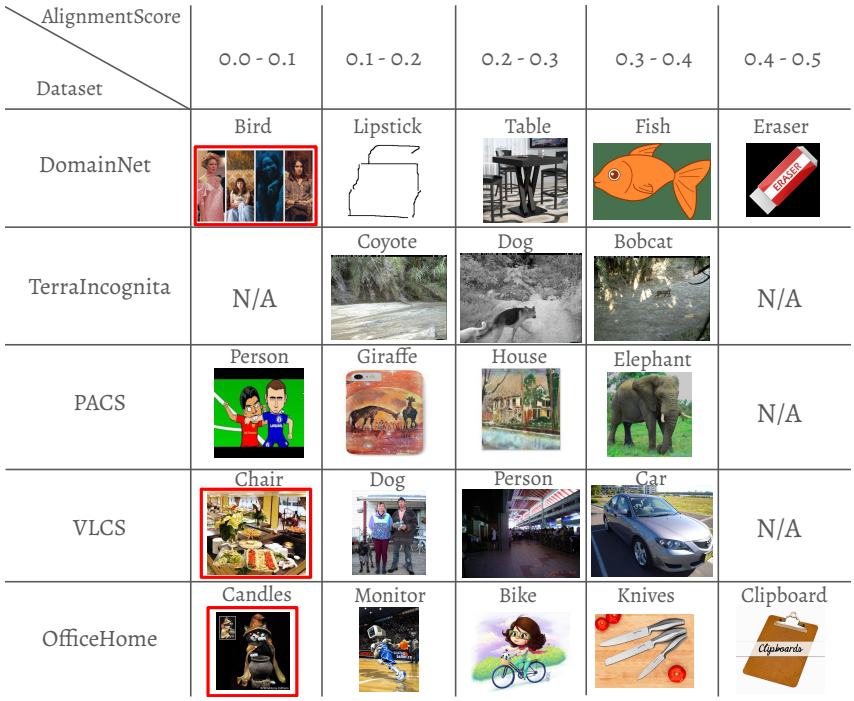


Figure 3: Representative DomainBed dataset samples and their labels at various AlignmentScore values. At very low AlignmentScores, most labels (red boxes) are incorrect. At very high AlignmentScores, text present in the image corresponds to the label.

threshold to split DG datasets into In-Pretraining (IP) and Out-of-Pretraining (OOP) evaluation subsets (Section 4.2), which we later use for evaluating DG methods (Section 5). In order to connect the AlignmentScore with the DG method, we use the same backbone both for splitting the datasets into IP and OOP subsets and for training DG methods. To facilitate further research, we make the DomainBed-IP/OOP dataset splits available to the research community.

### 4.1 Data Exploration and Cleaning

We start by visualizing the data of all the datasets at various AlignmentScore values. We show some representative samples in Figure 3. At very low scores, we find that a large fraction of labels are incorrect (red boxes in Figure 3).

As a result, we divide the data into AlignmentScore intervals (e.g., 0.00-0.05, 0.05-0.10, and so on, up to 0.2) and randomly sample 100 instances from each interval for every dataset. This allows us to systematically analyze the relationship between AlignmentScore and label accuracy across different score ranges. For each interval, we then count the fraction of mislabelled samples to better understand how low AlignmentScores are associated with labeling errors. We find that below an AlignmentScore of 0.15, label noise is unacceptably high, with all datasets suffering the most from mislabelling (Table 6 in Appendix). Therefore, we discard all samples with AlignmentScore less than 0.15 in DomainBed-IP/OOP. As shown in Table 1, we observe that the percentage of discarded samples due to mislabeling varies across datasets, with VLCS and DomainNet having the highest rates at 12.41% and 7.64%, respectively.

Furthermore, on the right side of Figure 3, we observe that at very high AlignmentScores (greater than 0.4), the images often contain text directly related to the label. Since our goal is to evaluate visual recognition rather than text recognition (OCR), and CLIP is known to have strong OCR abilities (Fort, 2021), we exclude all samples above AlignmentScore of 0.4 from DomainBed-IP/OOP. As shown in Table 1, although only a small portion of data is removed due to OCR filtering (0.00-0.15% across datasets), this issue may become more significant in future studies.

Table 1: Percentage of discarded samples due to mislabeling or label text in the image (OCR).

| Dataset | Dropped - Mislabelled | Dropped - OCR |
|---|---|---|
| OfficeHome | 0.92% | 0.15% |
| PACS | 3.05% | 0.00% |
| TerraIncognita | 0.22% | 0.00% |
| DomainNet | 7.64% | 0.03% |
| VLCS | 12.41% | 0.00% |

### 4.2 Data Splitting

After filtering, we focus on determining a threshold to split the dataset into In-Pretraining (IP) and Out-of-Pretraining (OOP) subsets. We select 0.21 as the threshold, based on the trends observed in Figure 2 (a.)), as this is the point where performance begins to improve significantly, indicating that existing methods become more effective. While this threshold represents a somewhat subjective choice informed by observed patterns, we provide AlignmentScores in the released data, allowing researchers the flexibility to experiment with their own thresholds.

Figure 16 in the Appendix illustrates how this split impacts the size and composition of each dataset. Certain domains naturally fall into the IP subset. For example, in DomainNet, the clipart domain is predominantly categorized as IP, likely due to its frequent presence on the internet and therefore the web-scraped pre-training data. On the other hand, TerraIncognita-OOP has relatively more balanced domains, but exhibits substantial class shift between IP and OOP splits (Figure 17), meaning some classes are better aligned than others during pre-training. More dataset statistics are in the Appendix.

## 5 Experiments

### 5.1 Training and Evaluation Protocol

We aim to evaluate how well Domain Generalization (DG) methods perform across both pretraining-aligned(IP) and pretraining-misaligned(OOP) data. We adhere to the DomainBed evaluation methodology, where one domain is chosen as the target, and the remaining domains act as source domains. To maintain a sufficient amount of training data, we train all DG methods on the original, unsplit datasets. We use hyper-parameter values recommended by the original implementation authors for each method.

After training, models are evaluated separately on the IP and OOP subsets, as well as the original, unsplit test domain. This approach allows us to measure how well each method generalizes to both pretraining-aligned and pretraining-misaligned data.

We follow the literature's standard practice of computing performance per target domain and averaging the results across all domains. This method ensures that any domain imbalances do not disproportionately influence the final performance metrics.

### 5.2 Algorithms

**Explicit Regularization towards Pretraining**: Several recent DG methods leverage explicit regularization towards the initialization. These methods generally operate either in weight space (by regularizing or freezing model parameters) or in feature space (by aligning internal feature representations with those of the pretrained model).

- **MIRO** (Cha et al., 2022): Minimizes the Mutual Information between DG model intermediate features and CLIP intermediate features.
- **Attention Tuning** (Teterwak et al., 2023; Touvron et al., 2022): Freezes all parameters except those in the Multiheaded-Attention Layers.
- **VL2V-SD** (Addepalli et al., 2024): Self-distills a linear combination of CLIP vision and text outputs into a model.

Table 2: Benchmarking DG methods on DB-IP/OOP

| **DomainBed-IP** | DomainNet | OfficeHome | PACS | TI | VLCS | Average |
|---|---|---|---|---|---|---|
| OpenCLIP ZS | 74.9 | 89.0 | 98.5 | 36.8 | 95.9 | 79.0 |
| CORAL (Sun & Saenko, 2016) | 63.3 | 76.1 | 84.3 | 42.9 | 86.5 | 70.6 |
| SAGM (Wang et al., 2023) | 64.3 | 79.5 | 90.1 | 44.0 | 88.0 | 73.2 |
| ERM* (Gulrajani & Lopez-Paz, 2020) | 63.1 | 78.1 | 87.1 | 42.0 | 85.3 | 71.1 |
| LP-FT (Kumar et al., 2021) | 64.4 | 78.5 | 90.3 | 40.9 | 86.0 | 72.0 |
| SWAD (Cha et al., 2021) | 72.3 | 84.8 | 94.6 | 52.7 | 88.5 | 78.6 |
| MIRO (Cha et al., 2022) | 72.4 | 88.8 | 97.6 | 58.9 | 91.0 | 81.7 |
| VL2V-SD (Addepalli et al., 2024) | 78.1 | **91.4** | 98.0 | 48.1 | 92.4 | 81.6 |
| Attn Tune (Teterwak et al., 2023) | 69.2 | 84.8 | 96.4 | 53.0 | 88.7 | 78.4 |
| MPA (Arpit et al., 2022) | 73.6 | 85.1 | 95.4 | 54.4 | 90.7 | 79.8 |
| CLIPOOD (Shu et al., 2023) | **78.9** | 90.9 | 97.7 | **63.5** | **92.5** | **84.7** |
| MIRO + SWAD | 77.0 | 90.5 | 97.6 | 62.1 | 91.1 | 83.6 |
| MIRO + MPA | 78.2 | 90.7 | **98.1** | 62.6 | 91.0 | 84.1 |
| Upper Bound (Target Finetune) | 81.6 | 88.5 | 97.8 | 93.4 | 93.8 | 91.0 |

(a) Samples with high AlignmentScore values, indicating good pretraining alignment.

| **DomainBed-OOP** | DomainNet | OfficeHome | PACS | TI | VLCS | Average |
|---|---|---|---|---|---|---|
| OpenCLIP ZS | 26.8 | 48.1 | 81.4 | 4.5 | 80.1 | 48.2 |
| CORAL (Sun & Saenko, 2016) | 22.3 | 42.6 | 74.1 | 16.0 | 74.0 | 45.8 |
| SAGM (Wang et al., 2023) | 23.0 | 44.5 | 74.2 | 19.3 | 73.3 | 46.9 |
| ERM* (Gulrajani & Lopez-Paz, 2020) | 22.3 | 42.9 | 76.9 | 16.5 | 76.4 | 47.0 |
| LP-FT (Kumar et al., 2021) | 22.7 | 43.4 | 78.6 | 23.1 | 70.7 | 47.7 |
| SWAD (Cha et al., 2021) | 28.6 | 49.9 | 79.1 | 21.0 | 77.0 | 51.1 |
| MIRO (Cha et al., 2022) | 28.4 | 56.6 | 84.7 | 18.5 | 73.7 | 52.4 |
| VL2V-SD (Addepalli et al., 2024) | 31.8 | 56.6 | 85.0 | 15.9 | 79.1 | 53.7 |
| Attn Tune (Teterwak et al., 2023) | 26.8 | 51.4 | 84.2 | 20.3 | 76.1 | 51.8 |
| MPA (Arpit et al., 2022) | 29.6 | 51.0 | 82.7 | 22.2 | 79.5 | 53.0 |
| CLIPOOD (Shu et al., 2023) | **33.9** | **63.9** | 87.2 | 19.9 | **80.7** | 57.1 |
| MIRO + SWAD | 32.0 | 59.0 | 85.4 | 21.1 | 78.9 | 55.3 |
| MIRO + MPA | 33.1 | 60.0 | **87.8** | **24.9** | 80.3 | **57.2** |
| Upper Bound (Target Finetune) | 48.8 | 61.9 | 92.9 | 83.2 | 92.4 | 75.8 |

(b) Samples with lower AlignmentScore values, representing cases where pretraining alignment is weak.

| **DomainBed-All** | DomainNet | OfficeHome | PACS | TI | VLCS | Average |
|---|---|---|---|---|---|---|
| OpenCLIP ZS | 59.6 | 85.4 | 97.0 | 33.2 | 82.4 | 71.5 |
| CORAL (Sun & Saenko, 2016) | 50.6 | 73.2 | 83.2 | 39.6 | 78.5 | 65.0 |
| SAGM (Wang et al., 2019) | 51.5 | 76.4 | 87.5 | 41.0 | 80.4 | 67.3 |
| ERM* (Gulrajani & Lopez-Paz, 2020) | 50.5 | 75.0 | 85.2 | 39.0 | 77.9 | 65.5 |
| LP-FT (Kumar et al., 2021) | 51.3 | 75.5 | 88.4 | 38.5 | 78.0 | 66.3 |
| SWAD (Cha et al., 2021) | 57.9 | 81.8 | 92.4 | 49.0 | 80.1 | 72.2 |
| MIRO (Cha et al., 2022) | 57.5 | 85.8 | 96.4 | 54.3 | 81.1 | 75.0 |
| VL2V-SD (Addepalli et al., 2024) | 62.0 | **88.3** | 96.9 | 44.4 | 82.7 | 74.9 |
| Attn Tune (Teterwak et al., 2023) | 55.4 | 81.9 | 95.4 | 49.1 | 81.8 | 72.7 |
| MPA (Arpit et al., 2022) | 58.9 | 82.0 | 94.3 | 50.7 | 82.3 | 73.6 |
| CLIPOOD (Shu et al., 2023) | **63.6** | **88.3** | 96.8 | **58.5** | **83.4** | **78.1** |
| MIRO + SWAD | 61.4 | 87.6 | 96.6 | 57.4 | 82.0 | 77.0 |
| MIRO + MPA | 62.4 | 87.9 | **97.2** | 58.2 | 82.8 | 77.7 |
| Upper Bound (Target Finetune) | 70.4 | 86.2 | 97.2 | 92.4 | 87.9 | 86.8 |

(c) Performance of DG methods on unsplit DomainBed

- **CLIPood** (Shu et al., 2023): Regularizes both weights and output features. Weights are averaged between the pre-trained CLIP model and the fine-tuned model, and outputs are regularized using a loss that incorporates information from the pre-trained text encoder.
- **Linear Probe - Fine Tuning (LP-FT)** (Kumar et al., 2021): Freezes the backbone, trains a linear probe, and then performs full finetuning. An untrained linear probe can cause finetuning to update the frozen backbone needlessly, potentially unlearning discriminative features. This biases the model towards pre-trained weights with smaller gradient updates.

**Domain Invariance**: A classic idea for Domain Generalization is Domain Invariance in the feature space, where the model learns only class-discriminative features shared among all training/source domains. **CORAL** (Sun & Saenko, 2016) matches the second moments of features across different domains and has been shown to be highly effective (Gulrajani & Lopez-Paz, 2020).

**Flat Optima**: Studies by Izmailov et al. (2018) and Cha et al. (2021) have shown that flat minima generalize better than sharp minima, as they make loss values less sensitive to perturbations in the loss surface, resulting in smaller increases in loss during domain shifts.

- **SWAD** (Cha et al., 2021): Averages model parameters during training, determining the interval over which to average using validation loss over source domain.
- **Model Parameter Averaging (MPA)** (Arpit et al., 2022): Starts averaging model parameters after a number of burn-in steps to find flat minima.
- **SAGM** (Wang et al., 2023): An optimizer that explicitly optimizes for flat minima.

**Baseline and Oracle**: We also train a baseline and oracle model for lower-bound and upper-bound reference. The baseline model is an Empirical Risk Minimization (ERM) model that is finetuned on source domains and evaluated on target domains, and has been found to be effective for the DG task (Gulrajani & Lopez-Paz, 2020). The oracle model is trained on an 80% training split of all domains and evaluated on a 20% test split. The oracle model removes the OOD aspect of generalization and provides a reasonable upper bound for DG methods.

### 5.3 Results

We show results in Table 2 for DomainBed-OOP, DomainBed-IP, and standard DomainBed datasets. Underlined results represent the best performance of any single method (excluding method combinations), while the bold numbers show the highest performance overall, excluding the upper bound.

**DG methods perform well on DomainBed-IP:** In most datasets within DomainBed-IP, domain generalization approaches achieve excellent performance.On three out of five datasets (DomainNet, OfficeHome, and PACS), the best DG method even outperforms the oracle! A notable exception is TerraIncognita, where CLIPood scores only 63.5%, far below the oracle's 93.4%, highlighting this dataset as a challenging outlier. Interestingly, all three datasets where performance exceeds the oracle have an average IP AlignmentScore of 0.28, while the others have a lower average AlignmentScore of around 0.26. Therefore, the underperformance of TerraIncognita may be partially explained by its lower IP AlignmentScore, suggesting that alignment plays a significant role in DG performance, even within the IP case. Another interesting observation is that on the IP subset, the zero-shot model can achieve performances greater than the finetuned models (for PACS and VLCS). This means that, for IP-data, DG finetuning sometimes causes more catastrophic forgetting than learning of new features from source!

**DG Methods leave much to be desired on DomainBed-OOP, but are still stronger than ERM:** In DomainBed-OOP, we observe that even the top-performing DG methods struggle with low-alignment data. For example, CLIPood achieves 57.1% accuracy, which is a significant drop compared to its performance on DomainBed-IP (84.7%) and DomainBed-All (78.1%). Despite this, DG methods still outperform Empirical Risk Minimization (ERM), which scores only 47.0% on DomainBed-OOP. Therefore DG methods are better equipped to handle domain shifts than ERM, possibly due to weak transfer of knowlegdge from pre-trained features. Nevertheless, there is still substantial room for improvement on low-alignment samples.

**State-of-the-Art methods do not consistently outperform older methods on OOP data:** While CLIPood (Shu et al., 2023) clearly outperforms other methods on DomainBed-All with 78.1%

accuracy, it performs comparably to older methods on DomainBed-OOP. For instance, MIRO + MPA (Cha et al., 2022; Arpit et al., 2022) achieves 57.2% on DomainBed-OOP, which is nearly the same as CLIPood's 57.1%. This suggests that CLIPood's primary advantage comes from well-aligned samples in DomainBed-IP, where it reaches 84.7% accuracy, half a percent better than the next-best method.

**Model Parameter Averaging (MPA) boosts performance on OOP data**: MPA performs significantly better than ERM on DomainBed-OOP, achieving 53.0% compared to ERM's 47.0%. This is half a percent better than MIRO on OOP data, despite being 2% worse than MIRO on IP data. When combined with MIRO, MPA delivers the best performance on DomainBed-OOP, scoring 57.2%, slightly surpassing CLIPood. This suggests that MPA can complement other regularization-based methods like MIRO. On DomainBed-IP, MIRO + MPA scores 84.1%, close to CLIPood's 84.7%, demonstrating versatility across both high- and low-alignment data. Interestingly, SWAD underperforms MPA on DomainBed-OOP by 2%, despite being conceptually similar. We attribute this to selecting the averaging interval on the source data, which introduces overfitting to source domains.

## 6 DICUSSION

As an increasing number of works in the Domain Generalization sub-field leverage pre-trained CLIP models for Domain Generalization benchmarks, it is important to better characterize the impacts of pre-training on DG. We leave the reader with the following takeaways:

**Pre-training Alignment Predicts DG Performance:** Our study demonstrates that pre-training alignment, measured as the cosine similarity between image and text embeddings, is a robust predictor of DG performance. This holds true even after source fine-tuning, highlighting that the quality of alignment achieved during pre-training has a significant impact on the generalization capability of models.

**Current DG Methods Exploit Pre-training Rather Than Learning New Features:** Our findings reveal a large difference in the performance of DG methods between pretraining-aligned (IP) and pretraining-misaligned (OOP) data. While state-of-the-art methods achieve near-oracle performance on IP data, they struggle significantly on OOP data. This indicates that current methods primarily leverage on pre-trained features rather than learning new, generalizable features from source data. Consequently, their success is heavily tied to the quality of pre-training, rather than the efficacy of the fine-tuning process itself.

**Benchmarks Should Reflect Pre-training Reliance:** The reliance on pre-trained alignment calls for a reevaluation of DG benchmarks. Existing benchmarks often aggregate results across all target data, masking the limitations of DG methods on low-alignment samples. To address this, we propose splitting evaluation datasets into In-Pretraining (IP) and Out-of-Pretraining (OOP) subsets. This provides a clearer picture of where DG methods succeed and where they fail. We hope that our proposed DomainBed-IP/OOP splits will guide the development of future methods that are better equipped to handle low-alignment data while maintaining performance on high-alignment samples.

## 7 CONCLUSION

In this paper, we systematically explore how Domain Generalization (DG) methods rely on pre-trained feature alignment from models like CLIP. We hypothesize that the alignment between image and text embeddings during pre-training strongly predicts DG performance. Our experiments confirm this, showing that methods perform well on high-alignment samples (DomainBed-IP) but struggle on low-alignment data (DomainBed-OOP). While we focus on multi-source DG, we expect similar results in single-source finetuning due to the foundational role of pre-trained alignment. Notably, state-of-the-art methods like CLIPood perform near oracle-level on aligned data but see significant drops on misaligned samples. This suggests current DG methods rely on pre-trained features and fail to learn new, generalizable features from source domains. Moving forward, two paths emerge: developing DG methods that better learn generalizable features, or focusing on improving pre-trained backbones. While foundation models will continue to advance, there will always be specialized distributions where they fail. We hope these findings inspire further research into improving generalization on low-alignment data, pushing DG beyond reliance on pre-trained alignment.

## REFERENCES


Sravanti Addepalli, Ashish Ramayee Asokan, Lakshay Sharma, and R Venkatesh Babu. Leveraging vision-language models for improving domain generalization in image classification. *CVPR*, 2024.

Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. Ensemble of averages: Improving model selection and boosting performance in domain generalization. *Advances in Neural Information Processing Systems*, 35:8265–8277, 2022.

Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.

Romain Beaumont. Clip retrieval: Easily compute clip embeddings and build a clip retrieval system with them. `https://github.com/rom1504/clip-retrieval`, 2022.

Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473, 2018.

Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. *European Conference on Computer Vision (ECCV)*, 2022.

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 (ICCV)*, 2023.

Sepehr Dehdashtian, Lan Wang, and Vishnu Naresh Boddeti. Fairerclip: Debiasing zero-shot predictions of clip in rkhss. In *International Conference on Learning Representations*, 2024.

Alex Fang, Gabriel Ilharco, Mitchell Wortsman, Yuhao Wan, Vaishaal Shankar, Achal Dave, and Ludwig Schmidt. Data determines distributional robustness in contrastive language image pre-training (clip). In *International Conference on Machine Learning*, pp. 6216–6234. PMLR, 2022.

Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1657–1664, 2013.

Stanislav Fort. Pixels still beat text: Attacking the openai clip model with text patches and adversarial pixel perturbations, March 2021. URL `https://stanislavfort.github.io/2021/03/05/OpenAI_CLIP_stickers_and_adversarial_examples.html`.

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.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2020.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. URL `https://doi.org/10.5281/zenodo.5143773`. If you use this software, please cite it as below.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

Samyak Jain, Sravanti Addepalli, Pawan Kumar Sahu, Priyam Dey, and R Venkatesh Babu. Dart: Diversify-aggregate-repeat training improves generalization of neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16048–16059, 2023.

Ananya Kumar, Aditi Raghunathan, Robbie Matthew Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2021.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5542–5550, 2017.

Ziyue Li, Kan Ren, XINYANG JIANG, Yifei Shen, Haipeng Zhang, and Dongsheng Li. SIMPLE: Specialized model-sample matching for domain generalization. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=BqrPeZ_e5P`.

Xiaofeng Mao, Yufeng Chen, Xiaojun Jia, Rong Zhang, Hui Xue, and Zhao Li. Context-aware robust fine-tuning. *International Journal of Computer Vision*, 132(5):1685–1700, 2024.

Prasanna Mayilvahanan, Thaddäus Wiedemer, Evgenia Rusak, Matthias Bethge, and Wieland Brendel. Does CLIP's generalization performance mainly stem from high train-test similarity? In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=tnBaiidobu`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1406–1415, 2019.

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.

Alexandre Rame, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, Patrick Gallinari, and Matthieu Cord. Diverse weight averaging for out-of-distribution generalization. In *NeurIPS*, 2022.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pp. 5389–5400. PMLR, 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pp. 213–226. Springer, 2010.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *ArXiv*, abs/2111.02114, 2021. URL https://api.semanticscholar.org/CorpusID:241033103.

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.

Mannat Singh, Laura Gustafson, Aaron Adcock, Vinicius de Freitas Reis, Bugra Gedik, Raj Prateek Kosaraju, Dhruv Mahajan, Ross Girshick, Piotr Dollár, and Laurens Van Der Maaten. Revisiting weakly supervised pre-training of visual perception models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 804–814, 2022.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, pp. 443–450. Springer, 2016.

Piotr Teterwak, Kuniaki Saito, Theodoros Tsiligkaridis, Kate Saenko, and Bryan A Plummer. Erm++: An improved baseline for domain generalization. *arXiv preprint arXiv:2304.01973*, 2023.

Hugo Touvron, Matthieu Cord, Alaaeldin El-Nouby, Jakob Verbeek, and Hervé Jégou. Three things everyone should know about vision transformers. In *European Conference on Computer Vision*, pp. 497–515. Springer, 2022.

Vishaal Udandarao, Ameya Prabhu, Adhiraj Ghosh, Yash Sharma, Philip HS Torr, Adel Bibi, Samuel Albanie, and Matthias Bethge. No" zero-shot" without exponential data: Pretraining concept frequency determines multimodal model performance. *arXiv preprint arXiv:2404.04125*, 2024.

Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Pengfei Wang, Zhaoxiang Zhang, Zhen Lei, and Lei Zhang. Sharpness-aware gradient matching for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3769–3778, 2023.

LI Xuhong, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International Conference on Machine Learning*, pp. 2825–2834. PMLR, 2018.

Shen Yan, Huan Song, Nanxiang Li, Lincan Zou, and Liu Ren. Improve unsupervised domain adaptation with mixup training. *arXiv preprint arXiv:2001.00677*, 2020.

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.

Zhun Zhong, Yuyang Zhao, Gim Hee Lee, and Nicu Sebe. Adversarial style augmentation for domain generalized urban-scene segmentation. *Advances in Neural Information Processing Systems*, 35:338–350, 2022.

# A Appendix

## A.1 Training and Evaluation Details

Code is zipped in the supplementary.

We use a slightly modified MIRO Cha et al. (2022) codebase for training and evaluation. We use **leave-one-out** evaluation, where a model is trained on all domains except the evaluation domain.

We emphasize that we use DomainBed-IP and DomainBed-OOP as evaluation data only, models are trained on full datasets.

For training, we use an OpenCLIP-ViT-B/16 Ilharco et al. (2021) trained on LAION-400M Schuhmann et al. (2021). We use default hyper-parameters as defined by Cha et al. (2022). This includes a learning rate of 5e-5, weight decay of 0.0, a batch size of 32 per-domain, an Adam Optimizer, and no dropout for all methods.

For evaluation, unlike DomainBed, we consider the entire test domain instead of an 80% random split. Following standard practice, we first compute accuracy for each domain, then average those accuracies to get dataset level statistics, and finally compute overall averages averaging across datasets.

For benchmarked methods, we also use hyper-parameters found to be best in respective papers. For **SWAD**, we use an optimum patience parameter value of 3, overfit patience parameter value of 6, and tolerance ratio of 6. For **MIRO**, we use use regularizer loss weight of 1.0. For **CORAL**, we use a CORAL regularizer weight of 1.0, following Cha et al. (2021). For **LP-FT**, we train the linear probe for 600 steps before unlocking the full backbone. For **Model Parameter Averaging**, we burn in the training for 600 steps before averaging iterates. For **VL2V-SD** and **CLIPood**, we directly use the author's implementation and hyper-parameters, except initializing with OpenCLIP Ilharco et al. (2021).

### A.2 Training Compute

Each run uses an A6000 48GB GPU, trained for up to 12 hours per domain-dataset combination.

## B Additional Results

### B.1 AlignmentScore vs Accuracy

In Figure 10, we plot all benchmarked methods from the main paper, with x-axis corresponding to AlignmentScore, and the y-axis corresponding to the Top-1 Accuracy. We normalize for dataset size, so that no dataset dominates the count. In Figures 11 through 15, we plot these statistics independently per dataset, and find the trends consistent across datasets.

### B.2 Per-Dataset Benchmarking Results

We expand Table 2 in the main paper into per-dataset results in Table 7 through 21.

### B.3 Similarity of Target to Pre-training

To evaluate the Image Similarity Hypothesis, we retrieve the nearest neighbors from the Laion-400M dataset Schuhmann et al. (2021). This raises the question of how similar the target domains are to the pre-training data and whether the source domains might be even more similar. To investigate this, we compute Maximum Mean Discrepancy (MMD) distances between PACS domains and their nearest neighbors from Laion-400M, as shown in Figure 5. Our results indicate that target domains are, in fact, more similar to the pre-training data than source domains. We inspect nearest neighbors manually, and find even exact duplicates (Figure 4). Interestingly, while we found not only domain-level duplicates but also exact matches in the pre-training data, the Image Similarity Hypothesis is ultimately less predictive than the Alignment Hypothesis.

### B.4 Other backbones

We benchmark 2 additional backbones(DINOv2 (Oquab et al., 2023) and OpenAI CLIP) using the MIRO + MPA Domain Generalization method, which we found to be the strongest in our paper, on two datasets (OfficeHome and PACS). This consistency is likely due to the similar nature of the pre-training datasets, both sourced from web scraping and of comparable scale. These findings reinforce that the usefulness of the DomainBed-IP/OOP split is not confined to a specific backbone.

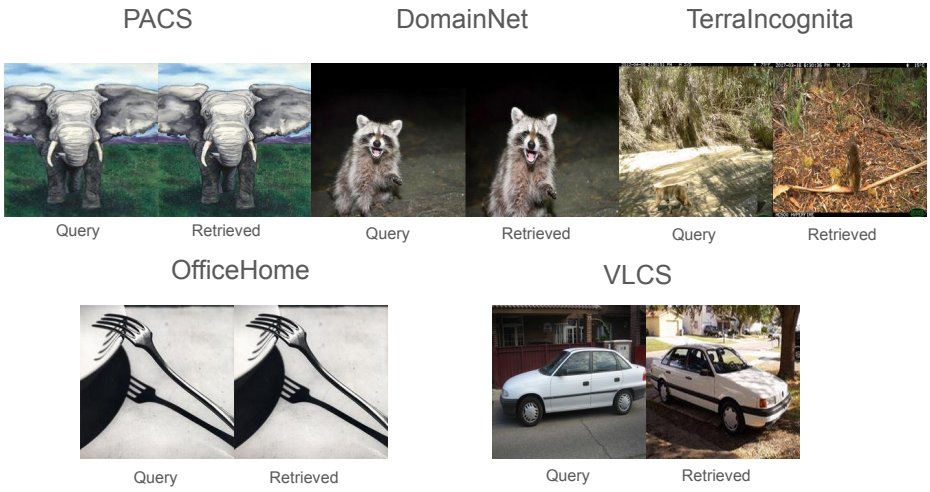


Figure 4: Nearest neighbors of target images in pre-training LAION data.

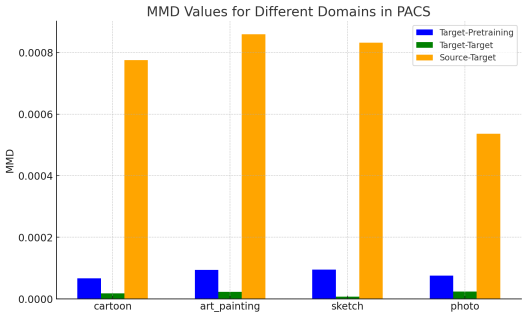


Figure 5: MMD between pre-training and target, source and target, and target and target for PACS. Target is more similar to pre-training than source. Despite this, Alignment is a better predictor of DG performance than perceptual similarity.

| Backbone | OfficeHome - IP | PACS - IP | OfficeHome - OOP | PACS - OOP |
|---|---|---|---|---|
| OpenCLIP-ViT/B 16 | 90.7 | 98.1 | 60.0 | 87.8 |
| CLIP-ViT/B 16 | 88.1 | 97.6 | 57.0 | 86.9 |
| DinoV2 | 87.4 | 97.1 | 58.9 | 85.0 |

Table 3: Benchmark results of different backbones on OfficeHome and PACS datasets.

### B.5 SPLITTING DOMAINBED USING PERCEPTUALSIMILARITYSCORE

In Figure 2 b.), we show that the slope of the relationship of Top-1 Accuracy vs Perceptual Similarity Score is positive but shallow. This suggests that using PerceptualSimilarityScore as an alternative to AlignmentScore for splitting DomainBed would not be very effective. To further prove this point, we split at a PerceptualSimilarityScore value of 0.86 in Table 4. We can see the differences are not very large between OOP and IP, indicating that AlignmentScore is a better thresholding metric.

### B.6 COMPARISNG ALIGNMENTSCORE WITH ZERO-SHOT CLASSIFICATION CONFIDENCE SCORE

Table 4: Splitting DomainBed by PerceptualSimilarityScore. The differences between IP and OOP with this split are much lower than with Alignment Score.

| **Dataset** | PACS | VLCS | TerraIncognita | OfficeHome | DomainNet |
|---|---|---|---|---|---|
| Perceptual IP | 97.2 | 74.8 | 60.6 | 86.8 | 61.4 |
| Perceptual OOP | 95.6 | 77.4 | 42.9 | 78.3 | 55.3 |

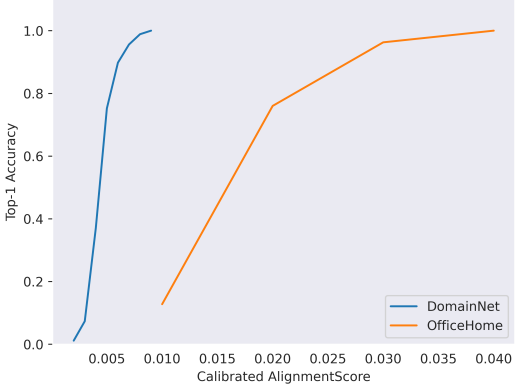


Figure 6: Top-1 DG Accuracy vs calibrated alignment: We use the confidence of the zero-shot classifier formed by the pre-trained CLIP models an alignment measure. Although the score predicts generalization for both OfficeHome and DomainNet, the scores have different scales for different datasets.

We also consider an alignment score which takes into account uncertainties, and compute a score using the confidence of the zero-shot classifier formed by the pre-trained CLIP model for each sample. Specifically, for a sample with ground truth class c, we calculate the softmax over the logits output by the zero-shot classifier, and use the resulting probability p(c) as the score. We refer to this as the Calibrated AlignmentScore and show the results in Figure 6. Although the score predicts generalization for both OfficeHome and DomainNet, the scores have different scales for different datasets. In contrast out AlignmentScore does align across datasets to a greater degree (Figure 7)

### B.7 Combining PerceptualSimilarity Score and AlignmentScore

We explore the effect of averaging PerceptualSimilarityScore and AlignmentScore in Figure 8. We can see that there is not much of a compositional effect, so therefore we stick with AlignmentScore as our generalization predictor.

### B.8 Image Similarity Hypothesis for source data

The main drawback of the Image Similarity hypothesis is that it does not consider how well the nearest perceptual neighbor is learned during pre-training. One reason for a sample being poorly learned during pre-training is that the pre-training caption is not very relevant to the DG task. Source data is unlikely to have this issue, since source and target domains share labels. Therefore it is interesting to ask how strongly correlated the PerceptualSimilarityScore is with DG accuracy when measured between source and target. Indeed , as seen in Figure 9, there is a strong correlation. However, simply using source-data to compute the PerceptualSimilarity results in an incomplete understanding of the relationship between target data and the training procedure, due to the lack of consideration of the pre-training. In fact, zero-shot models with NO learning from source are very performant (Table 2)

### B.9 Training from scratch

We focus on analyzing the generalization capabilities of CLIP-based DG methods, as these have been demonstrated to be the strongest in recent work. However,from-scratch experiments are valu-

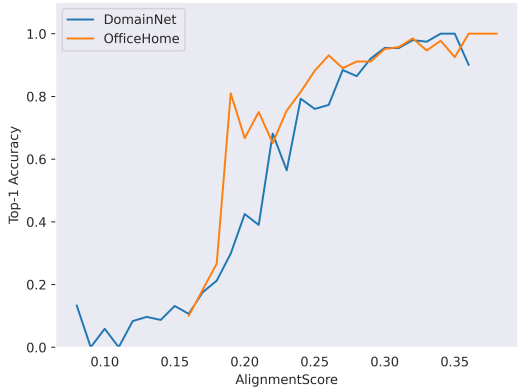


Figure 7: Top-1 DG Accuracy vs AlignmentScore: The AlignmentScore introduced in our work have scales which are comparable across different datasets.

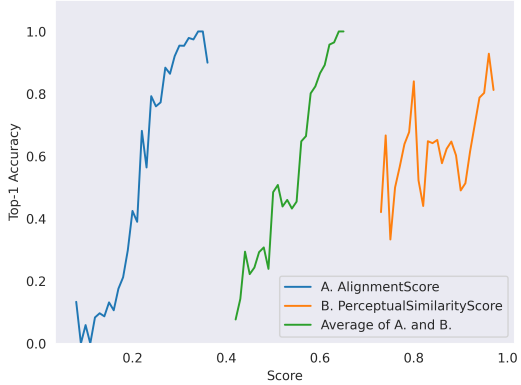


Figure 8: Combining PerceptualSimilarity Score and AlignmentScore: We explore the effect of averaging PerceptualSimilarityScore and AlignmentScore. There is no visible additional signal from averaging. We therefore stick with AlignmentScore as our generalization predictor.

able to measure how much finetuning on source data can learn, and so we also trained ResNet-50 models from scratch, using the MPA (Model Parameter Averaging) method and present the results in Table 5. We can see that performance is very low even on ALL DomainBed data (unsplit), further confirming that existing DG algorithms are poor at learning new knowledge from source data alone, and they rely on strong pre-training for good performance.

# C Additional DomainBed-(IP/OOP) Statistics and Analysis

## C.1 Constituent Datasets

We split 5 datasets in DomainBed-(IP/OOP),( VLCS Fang et al. (2013), DomainNet Peng et al. (2019), OfficeHome Ganin et al. (2016), PACS Li et al. (2017), and TerraIncognita Beery et al. (2018)). Here we provide basic statistics of each.

**VLCS** has 5 classes and 4 domains: Caltech101, LabelME, SUN09, and VOC2007, with 10729 samples. The domain shift is dataset source.

**DomainNet** contrains 345 classes and 6 domains: clipart, infograph, quickdraw, real, and sketch. It has a total of 586,575 samples. The dataset shift is style.

**OfficeHome** has 65 classes and 4 domains: art, clipart, product, and real. The dataset shift is style.

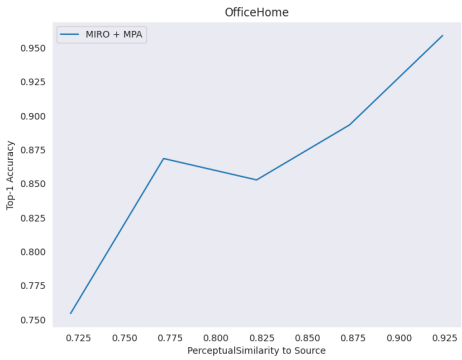


Figure 9: PerceptualSimilarity to Source for OfficeHome. DG performance is correlated with similarity to source images.

| | DomainNet | OfficeHome | PACS | TI | VLCS | Avg |
|---|---|---|---|---|---|---|
| ResNet-50 from Scratch | 7.6 | 17.2 | 36.3 | 23.8 | 55.2 | 28.0 |
| Clip ViT-B/16 | 58.9 | 82.0 | 94.3 | 50.7 | 82.3 | 73.6 |

Table 5: Training ResNet-50 from scratch. DG performance is very poor, showing DG methods rely on strong pre-training

**TerraIncognita** has 10 classes of wildlife cameras. There are 4 domains of different cameras and 24788 samples. The dataset shift is camera location.

**PACS** has 9991 sampes and 4 domains: arts, cartoon, photo, and sketch. There are 7 classes. The dataset shift is style.

### C.2 Mislabeling Rates

In Table 6 we present the mislabelling rates at various AlingmentScore values.

### C.3 Class Distribution of DomainBed-(IP/OOP):

In Figures 18 through 32, we provide class distribution statistics of different datasets before splitting and in our IP and OOP splits. We find some interesting patterns. For example, in Office-Home, the OOP class is dominated by **marker** and **toys**, while the IP split has a much more uniform distribution. Similarly, both PACS (DomainBed-OOP) and VLCS (DomainBed-OOP) are dominated by **person**.

Table 6: Mislabeling rates across different AlignmentScore ranges.

| **Dataset** | 0.0-0.05 | 0.05-0.10 | 0.10-0.15 | 0.15-0.20 |
|---|---|---|---|---|
| OfficeHome | 100% | 45% | 28% | 12% |
| PACS | - | 46% | 5% | 0% |
| TerraIncognita | - | 55% | 34% | 23% |
| DomainNet | 65% | 35% | 33% | 11% |
| VLCS | 33% | 14% | 9% | 3% |

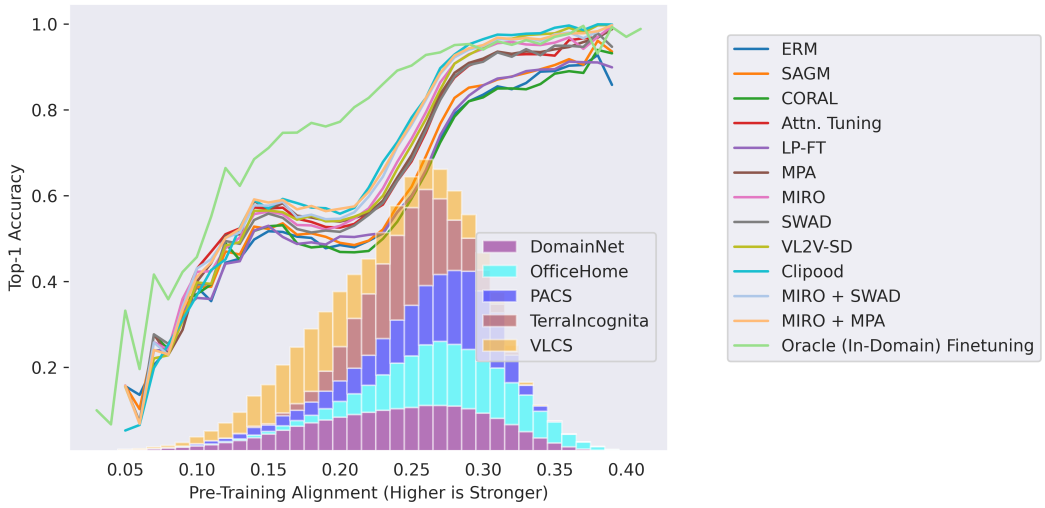


Figure 10: Plotting AlignmentScore vs Top-1 Accuracy for all benchmarked methods, for all datasets together. Although some methods are stronger than others, all follow the same trend of increasing accuracy with increased AlignmentScore.

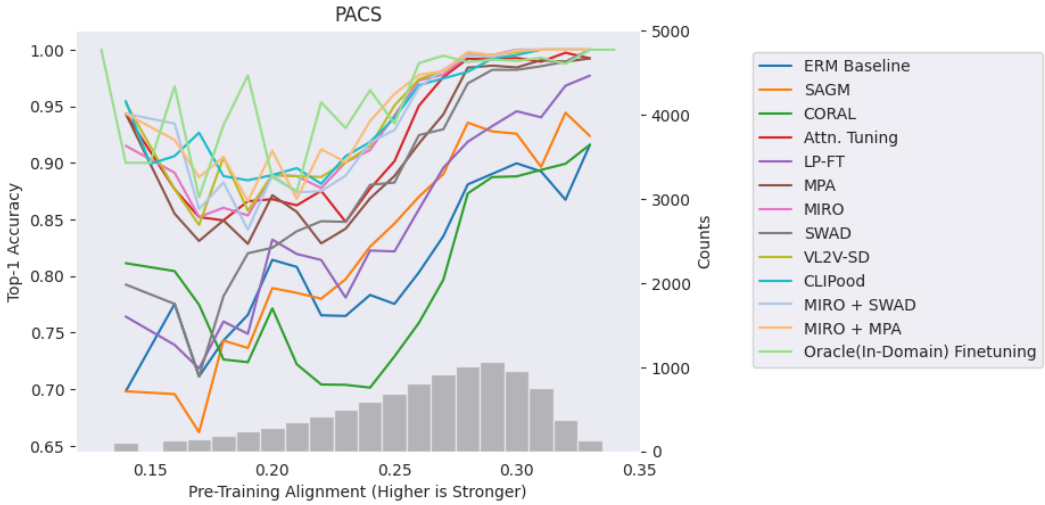


Figure 11: Plotting AlignmentScore vs Top-1 Accuracy on the PACS dataset.

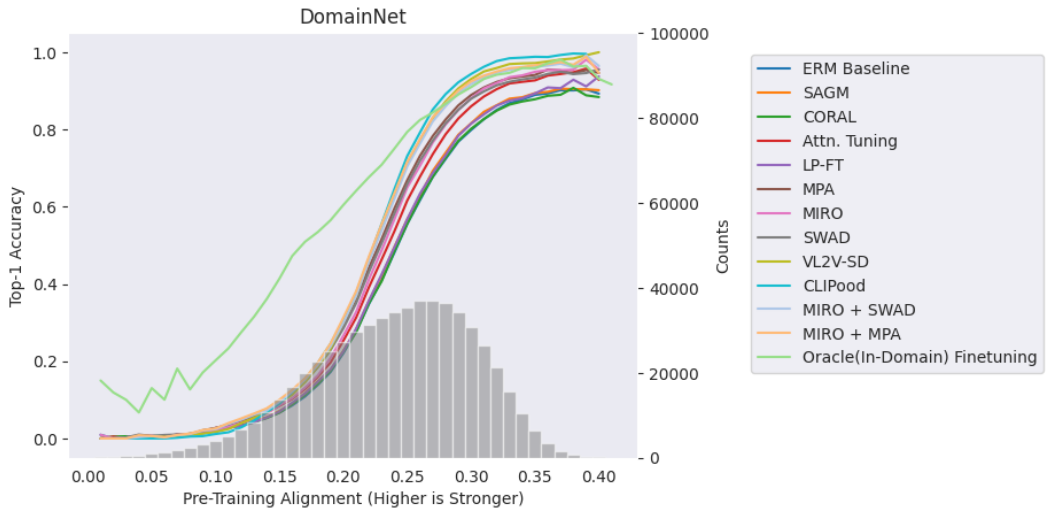


Figure 12: Plotting AlignmentScore vs Top-1 Accuracy on the DomainNet dataset.

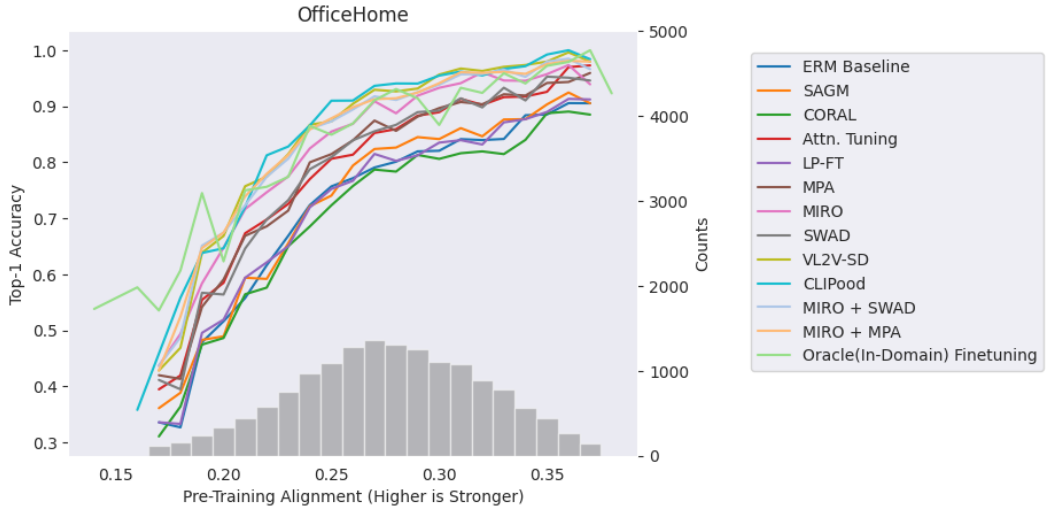


Figure 13: Plotting AlignmentScore vs Top-1 Accuracy on the OfficeHome dataset.

Table 7: Per-domain breakdown for OfficeHome (DomainBed-OOP)

| **Method** | Art | Clipart | Product | Real | Avg |
|---|---|---|---|---|---|
| CORALSun & Saenko (2016) | 33.0 | 29.4 | 50.0 | 58.0 | 42.6 |
| SAGMWang et al. (2023) | 36.8 | 30.7 | 51.7 | 58.7 | 44.5 |
| ERM*Gulrajani & Lopez-Paz (2020) | 36.0 | 28.1 | 51.3 | 56.0 | 42.9 |
| LP-FTKumar et al. (2021) | 37.2 | 29.7 | 52.0 | 54.7 | 43.4 |
| SWADCha et al. (2021) | 43.9 | 37.1 | 56.0 | 62.7 | 49.9 |
| MIRO Cha et al. (2022) | 49.4 | 39.6 | 62.7 | 74.7 | 56.6 |
| VL2V-SD Addepalli et al. (2024) | 56.4 | 39.3 | 65.3 | 65.3 | 56.6 |
| Attn. Tune Teterwak et al. (2023); Touvron et al. (2022) | 46.0 | 36.1 | 59.7 | 64.0 | 51.4 |
| Model Parameter Averaging (MPA) Arpit et al. (2022) | 46.0 | 36.4 | 55.7 | 66.0 | 51.0 |
| CLIPOOD (Shu et al., 2023) | 60.3 | 44.4 | 73.3 | 78.0 | 63.9 |
| MIRO + SWAD | 55.2 | 44.7 | 63.3 | 72.7 | 59.0 |
| MIRO + MPA | 53.5 | 44.1 | 65.7 | 76.7 | 60.0 |

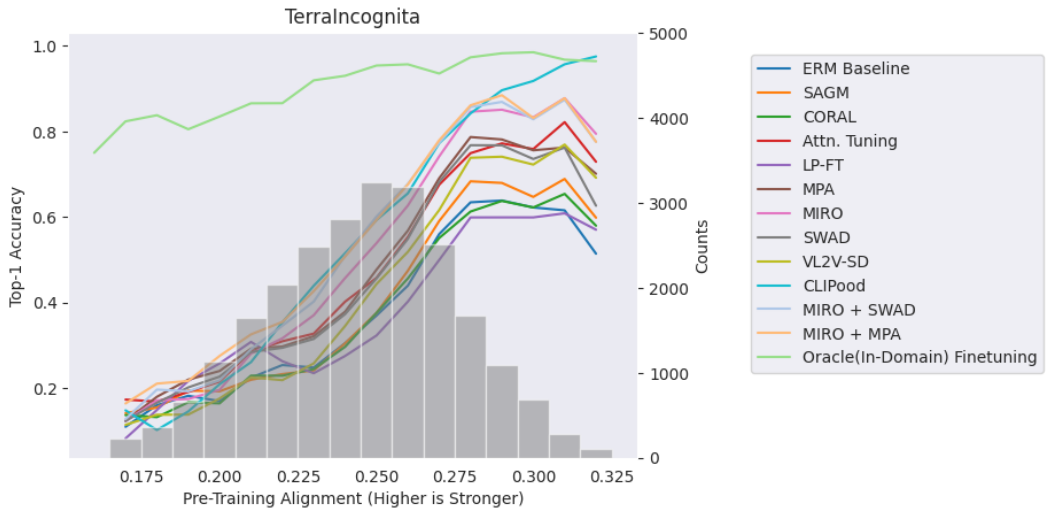


Figure 14: Plotting AlignmentScore vs Top-1 Accuracy on the TerraIncognita dataset.

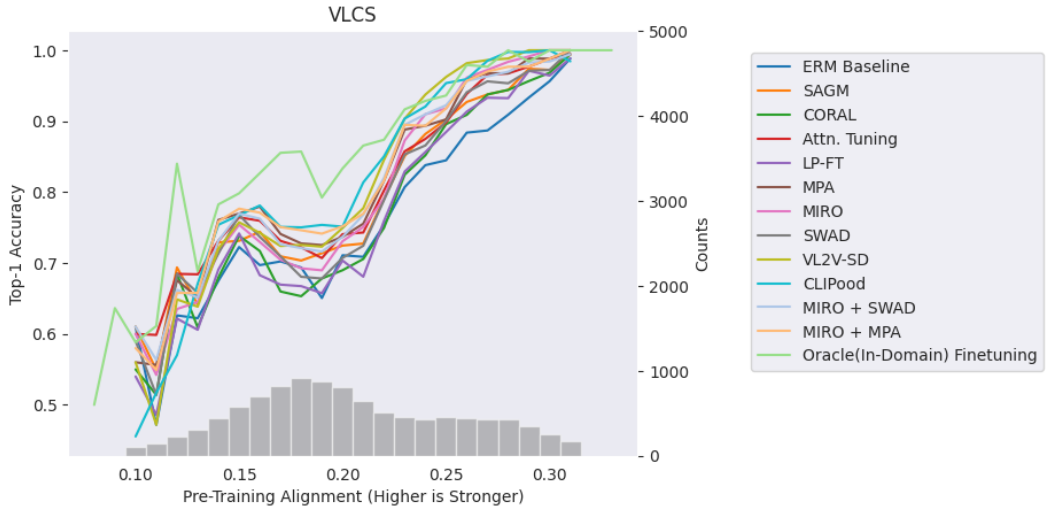


Figure 15: Plotting AlignmentScore vs Top-1 Accuracy on the VLCS dataset.

Table 8: Per-domain breakdown for OfficeHome (DomainBed-IP)

| **Method** | Art | Clipart | Product | Real | Avg |
|---|---|---|---|---|---|
| CORAL | 71.6 | 62.9 | 85.8 | 84.0 | 76.1 |
| SAGM | 75.7 | 70.8 | 87.0 | 84.5 | 79.5 |
| ERM* | 74.1 | 67.4 | 86.7 | 84.1 | 78.1 |
| LP-FT | 75.2 | 66.9 | 88.9 | 83.0 | 78.5 |
| SWAD | 81.4 | 77.8 | 91.8 | 88.2 | 84.8 |
| MIRO | 88.3 | 81.1 | 93.7 | 92.1 | 88.8 |
| VL2V-SD | 91.0 | 83.8 | 96.7 | 94.2 | 91.4 |
| Attn. Tune | 86.0 | 75.1 | 89.5 | 88.7 | 84.8 |
| Model Parameter Averaging (MPA) | 83.6 | 77.1 | 91.1 | 88.5 | 85.1 |
| CLIPOOD (Shu et al., 2023) | 92.3 | 81.3 | 96.0 | 94.2 | 90.9 |
| MIRO + SWAD | 90.4 | 83.2 | 95.5 | 92.8 | 90.5 |
| MIRO + MPA | 90.6 | 83.9 | 95.4 | 92.9 | 90.7 |

Table 9: Per-domain breakdown for OfficeHome (DomainBed-All)

| **Method** | Art | Clipart | Product | Real | Avg |
|---|---|---|---|---|---|
| CORAL | 66.6 | 60.1 | 83.0 | 83.0 | 73.2 |
| SAGM | 70.7 | 67.1 | 84.3 | 83.6 | 76.4 |
| ERM* | 69.2 | 63.9 | 83.9 | 83.1 | 75.0 |
| LP-FT | 70.3 | 63.6 | 86.0 | 82.0 | 75.5 |
| SWAD | 76.6 | 74.3 | 89.0 | 87.2 | 81.8 |
| MIRO | 83.2 | 77.5 | 91.2 | 91.4 | 85.8 |
| VL2V-SD | 86.4 | 79.7 | 94.1 | 93.1 | 88.3 |
| Attn. Tune | 80.8 | 71.7 | 87.2 | 87.8 | 81.9 |
| Model Parameter Averaging (MPA) | 78.7 | 73.4 | 88.4 | 87.6 | 82.0 |
| CLIPOOD (Shu et al., 2023) | 87.9 | 77.8 | 94.1 | 93.6 | 88.3 |
| MIRO + SWAD | 85.6 | 79.9 | 93.0 | 92.1 | 87.6 |
| MIRO + MPA | 85.6 | 80.5 | 93.0 | 92.3 | 87.9 |

Table 10: Per-domain breakdown for TerraIncognita (DomainBed-OOP)

| **Method** | L100 | L38 | L43 | L46 | **Avg** |
|---|---|---|---|---|---|
| CORAL | 13.9 | 8.1 | 21.1 | 21.1 | 16.0 |
| SAGM | 17.2 | 7.8 | 20.2 | 32.2 | 19.3 |
| ERM* | 13.1 | 9.6 | 22.9 | 20.5 | 16.5 |
| LP-FT | 16.7 | 5.1 | 25.6 | 44.9 | 23.1 |
| SWAD | 20.7 | 8.2 | 26.9 | 28.2 | 21.0 |
| MIRO | 19.4 | 6.6 | 30.0 | 17.9 | 18.5 |
| VL2V-SD | 9.3 | 5.6 | 20.2 | 28.4 | 15.9 |
| Attn. Tune | 20.5 | 10.7 | 24.8 | 25.3 | 20.3 |
| Model Parameter Averaging (MPA) | 21.7 | 8.6 | 30.9 | 27.6 | 22.2 |
| CLIPOOD (Shu et al., 2023) | 37.9 | 11.3 | 21.6 | 8.9 | 19.9 |
| MIRO + SWAD | 22.5 | 5.9 | 30.3 | 25.9 | 21.1 |
| MIRO + MPA | 24.8 | 9.6 | 32.5 | 32.8 | 24.9 |

Table 11: Per-domain breakdown for TerraIncognita (DomainBed-IP)

| **Method** | L100 | L38 | L43 | L46 | Avg |
|---|---|---|---|---|---|
| CORAL | 46.4 | 35.3 | 56.9 | 32.8 | 42.9 |
| SAGM | 55.1 | 38.7 | 52.5 | 29.5 | 44.0 |
| ERM* | 37.4 | 38.0 | 55.2 | 37.5 | 42.0 |
| LP-FT | 50.7 | 36.5 | 58.1 | 18.2 | 40.9 |
| SWAD | 60.1 | 41.7 | 65.7 | 43.5 | 52.7 |
| MIRO | 69.5 | 50.2 | 69.7 | 46.1 | 58.9 |
| VL2V-SD | 52.8 | 38.4 | 57.6 | 43.8 | 48.1 |
| Attn. Tune | 52.4 | 44.0 | 68.0 | 47.5 | 53.0 |
| Model Parameter Averaging (MPA) | 63.8 | 41.4 | 67.9 | 44.7 | 54.4 |
| CLIPOOD (Shu et al., 2023) | 77.7 | 56.4 | 69.0 | 51.0 | 63.5 |
| MIRO + SWAD | 68.9 | 54.8 | 73.4 | 51.2 | 62.1 |
| MIRO + MPA | 69.4 | 55.9 | 73.6 | 51.4 | 62.6 |

Table 12: Per-domain breakdown for TerraIncognita (DomainBed-All)

| **Method** | L100 | L38 | L43 | L46 | Avg |
|---|---|---|---|---|---|
| CORAL | 43.6 | 32.7 | 50.4 | 31.8 | 39.6 |
| SAGM | 51.9 | 35.8 | 46.6 | 29.7 | 41.0 |
| ERM* | 35.3 | 35.3 | 49.3 | 36.0 | 39.0 |
| LP-FT | 47.8 | 33.5 | 52.1 | 20.5 | 38.5 |
| SWAD | 56.7 | 38.5 | 58.6 | 42.2 | 49.0 |
| MIRO | 65.2 | 46.0 | 62.4 | 43.7 | 54.3 |
| VL2V-SD | 49.1 | 35.3 | 50.8 | 42.5 | 44.4 |
| Attn. Tune | 49.7 | 40.8 | 60.1 | 45.6 | 49.1 |
| Model Parameter Averaging (MPA) | 60.1 | 38.3 | 61.1 | 43.2 | 50.7 |
| CLIPOOD (Shu et al., 2023) | 74.3 | 52.1 | 60.4 | 47.4 | 58.5 |
| MIRO + SWAD | 64.9 | 50.1 | 65.5 | 49.0 | 57.4 |
| MIRO + MPA | 65.6 | 51.5 | 66.0 | 49.8 | 58.2 |

Table 13: Comparison of Methods for DomainNet Dataset (DomainBed-OOP)

| **Method** | Clp | Inf | Pnt | Qkdr | Real | Skt | Avg |
|---|---|---|---|---|---|---|---|
| CORAL | 34.9 | 19.2 | 26.9 | 5.7 | 24.6 | 22.8 | 22.3 |
| SAGM | 36.4 | 19.4 | 25.1 | 6.1 | 26.8 | 24.2 | 23.0 |
| ERM* | 36.2 | 19.0 | 23.4 | 5.4 | 26.6 | 23.4 | 22.3 |
| LP-FT | 34.7 | 20.9 | 25.2 | 6.1 | 25.9 | 23.2 | 22.7 |
| SWAD | 42.5 | 28.3 | 33.4 | 7.6 | 30.2 | 29.5 | 28.6 |
| MIRO | 42.5 | 37.0 | 27.9 | 3.5 | 33.5 | 25.8 | 28.4 |
| VL2V-SD | 43.0 | 43.2 | 34.8 | 3.0 | 37.1 | 29.5 | 31.8 |
| Attn. Tune | 40.4 | 29.1 | 29.2 | 4.3 | 31.3 | 26.6 | 26.8 |
| Model Parameter Averaging (MPA) | 42.3 | 30.9 | 34.9 | 7.4 | 31.6 | 30.7 | 29.6 |
| CLIPOOD (Shu et al., 2023) | 39.2 | 50.2 | 39.5 | 2.1 | 40.9 | 31.3 | 33.9 |
| MIRO + SWAD | 46.1 | 40.4 | 34.0 | 4.4 | 35.2 | 31.7 | 32.0 |
| MIRO + MPA | 46.2 | 42.6 | 36.0 | 4.4 | 36.0 | 33.3 | 33.1 |

Table 14: Comparison of Methods for DomainNet Dataset (DomainBed-IP)

| **Method** | Clp | Inf | Pnt | Qkdr | Real | Skt | Avg |
|---|---|---|---|---|---|---|---|
| CORAL | 80.3 | 52.3 | 71.1 | 33.2 | 71.7 | 71.2 | 63.3 |
| SAGM | 81.7 | 52.8 | 70.2 | 34.5 | 73.4 | 73.1 | 64.3 |
| ERM* | 80.6 | 53.4 | 69.0 | 31.3 | 73.0 | 71.6 | 63.1 |
| LP-FT | 80.0 | 56.5 | 70.5 | 34.0 | 73.9 | 71.5 | 64.4 |
| SWAD | 85.5 | 67.1 | 80.2 | 41.3 | 80.1 | 79.8 | 72.3 |
| MIRO | 85.4 | 77.7 | 79.1 | 34.6 | 83.2 | 74.2 | 72.4 |
| VL2V-SD | 88.3 | 85.6 | 84.6 | 38.0 | 88.3 | 84.0 | 78.1 |
| Attn. Tune | 83.5 | 66.1 | 77.3 | 32.6 | 80.3 | 75.5 | 69.2 |
| Model Parameter Averaging (MPA) | 85.8 | 69.7 | 81.6 | 41.4 | 81.4 | 81.4 | 73.6 |
| CLIPOOD (Shu et al., 2023) | 86.6 | 85.4 | 87.6 | 38.0 | 91.0 | 84.7 | 78.9 |
| MIRO + SWAD | 88.2 | 81.7 | 83.9 | 40.8 | 85.8 | 81.9 | 77.0 |
| MIRO + MPA | 88.1 | 83.7 | 85.5 | 41.3 | 86.5 | 84.0 | 78.2 |

Table 15: Comparison of Methods for DomainNet Dataset (DomainBed-All)

| **Method** | Clp | Inf | Pnt | Qkdr | Real | Skt | Avg |
|---|---|---|---|---|---|---|---|
| CORAL | 72.7 | 27.1 | 58.0 | 20.1 | 67.1 | 58.8 | 50.6 |
| SAGM | 74.1 | 27.4 | 56.9 | 20.9 | 68.9 | 60.6 | 51.5 |
| ERM* | 73.1 | 27.4 | 55.7 | 19.0 | 68.5 | 59.3 | 50.5 |
| LP-FT | 72.4 | 29.3 | 57.1 | 20.7 | 69.2 | 59.2 | 51.3 |
| SWAD | 78.0 | 36.0 | 66.1 | 25.2 | 75.2 | 66.7 | 57.9 |
| MIRO | 78.0 | 42.9 | 64.0 | 20.0 | 78.3 | 61.7 | 57.5 |
| VL2V-SD | 80.5 | 47.8 | 69.5 | 21.6 | 83.1 | 69.8 | 62.0 |
| Attn. Tune | 76.1 | 36.0 | 63.0 | 19.2 | 75.4 | 62.9 | 55.4 |
| Model Parameter Averaging (MPA) | 78.3 | 37.9 | 67.4 | 25.2 | 76.5 | 68.2 | 58.9 |
| CLIPOOD (Shu et al., 2023) | 78.4 | 52.9 | 72.6 | 21.2 | 85.9 | 70.7 | 63.6 |
| MIRO + SWAD | 80.8 | 45.6 | 68.8 | 23.7 | 80.7 | 68.7 | 61.4 |
| MIRO + MPA | 80.7 | 47.2 | 70.4 | 23.9 | 81.4 | 70.6 | 62.4 |

Table 16: Per-domain breakdown for VLCS (DomainBed-OOP)

| **Method** | Caltech101 | LabelMe | SUN09 | VOC2007 | Avg |
|---|---|---|---|---|---|
| CORAL | 93.3 | 54.6 | 72.9 | 75.3 | 74.0 |
| SAGM | 80.0 | 58.8 | 76.5 | 78.0 | 73.3 |
| ERM* | 100.0 | 57.1 | 73.3 | 75.3 | 76.4 |
| LP-FT | 80.0 | 55.4 | 71.8 | 75.7 | 70.7 |
| SWAD | 100.0 | 51.2 | 79.1 | 77.7 | 77.0 |
| MIRO | 86.7 | 52.6 | 82.9 | 72.7 | 73.7 |
| VL2V-SD | 100.0 | 55.0 | 80.4 | 81.1 | 79.1 |
| Attn. Tune | 86.7 | 61.0 | 77.9 | 79.0 | 76.1 |
| Model Parameter Averaging (MPA) | 100.0 | 53.6 | 83.6 | 80.9 | 79.5 |
| CLIPOOD (Shu et al., 2023) | 100.0 | 55.0 | 84.9 | 82.8 | 80.7 |
| MIRO + SWAD | 100.0 | 56.7 | 84.7 | 79.9 | 80.3 |
| MIRO + MPA | 100.0 | 54.6 | 83.4 | 77.4 | 78.9 |

Table 17: Per-domain breakdown for VLCS (DomainBed-IP)

| **Method** | Caltech101 | LabelMe | SUN09 | VOC2007 | Avg |
|---|---|---|---|---|---|
| CORAL | 98.2 | 83.5 | 84.5 | 79.8 | 86.5 |
| SAGM | 95.9 | 84.3 | 86.3 | 85.5 | 88.0 |
| ERM* | 96.9 | 83.7 | 86.1 | 74.6 | 85.3 |
| LP-FT | 97.1 | 83.2 | 82.8 | 80.7 | 86.0 |
| SWAD | 97.9 | 82.9 | 88.4 | 84.8 | 88.5 |
| MIRO | 98.9 | 83.3 | 93.5 | 88.2 | 91.0 |
| VL2V-SD | 99.5 | 83.9 | 91.8 | 94.6 | 92.4 |
| Attn. Tune | 98.0 | 84.7 | 85.7 | 86.5 | 88.7 |
| Model Parameter Averaging (MPA) | 98.4 | 83.0 | 92.6 | 88.8 | 90.7 |
| CLIPOOD (Shu et al., 2023) | 98.9 | 83.6 | 93.5 | 94.0 | 92.5 |
| MIRO + SWAD | 97.6 | 83.4 | 92.6 | 90.6 | 91.1 |
| MIRO + MPA | 98.0 | 83.2 | 92.9 | 90.1 | 91.0 |

Table 18: Per-domain breakdown for VLCS (DomainBed-All)

| **Method** | Caltech101 | LabelMe | SUN09 | VOC2007 | Avg |
|---|---|---|---|---|---|
| CORAL | 98.2 | 66.2 | 72.7 | 77.0 | 78.5 |
| SAGM | 95.8 | 68.8 | 76.2 | 80.8 | 80.4 |
| ERM* | 96.9 | 67.8 | 73.0 | 73.9 | 77.9 |
| LP-FT | 97.0 | 66.4 | 71.4 | 77.4 | 78.0 |
| SWAD | 98.0 | 64.5 | 77.7 | 80.3 | 80.1 |
| MIRO | 98.7 | 65.3 | 81.2 | 79.3 | 81.1 |
| VL2V-SD | 99.5 | 66.6 | 78.8 | 86.0 | 82.7 |
| Attn. Tune | 97.9 | 70.3 | 77.2 | 81.9 | 81.8 |
| Model Parameter Averaging (MPA) | 98.4 | 65.7 | 81.3 | 83.7 | 82.3 |
| CLIPOOD (Shu et al., 2023) | 98.9 | 66.7 | 81.6 | 86.6 | 83.4 |
| MIRO + SWAD | 97.6 | 66.4 | 81.6 | 82.6 | 82.0 |
| MIRO + MPA | 98.0 | 67.4 | 82.2 | 83.6 | 82.8 |

Table 19: Per-domain breakdown for PACS (DomainBed-OOP)

| **Method** | Art | Cartoon | Photo | Sketch | Avg |
|---|---|---|---|---|---|
| CORAL | 80.4 | 84.3 | 89.5 | 42.2 | 74.1 |
| SAGM | 74.4 | 78.1 | 89.5 | 54.7 | 74.2 |
| ERM* | 75.9 | 86.8 | 91.6 | 53.2 | 76.9 |
| LP-FT | 74.2 | 85.9 | 97.9 | 56.3 | 78.6 |
| SWAD | 84.6 | 81.6 | 95.8 | 54.3 | 79.1 |
| MIRO | 94.2 | 94.1 | 97.9 | 52.8 | 84.7 |
| VL2V-SD | 93.5 | 96.5 | 97.9 | 52.2 | 85.0 |
| Attn. Tune | 94.0 | 91.4 | 95.8 | 55.7 | 84.2 |
| Model Parameter Averaging (MPA) | 94.0 | 88.9 | 97.9 | 50.2 | 82.7 |
| CLIPOOD (Shu et al., 2023) | 96.0 | 96.8 | 97.9 | 58.3 | 87.2 |
| MIRO + SWAD | 96.0 | 94.9 | 100.0 | 50.7 | 85.4 |
| MIRO + MPA | 95.3 | 95.4 | 100.0 | 60.3 | 87.8 |

Table 20: Per-domain breakdown for PACS (DomainBed-IP)

| **Method** | Art | Cartoon | Photo | Sketch | Avg |
|---|---|---|---|---|---|
| CORAL | 83.3 | 89.8 | 91.8 | 72.1 | 84.3 |
| SAGM | 89.7 | 93.0 | 94.7 | 82.9 | 90.1 |
| ERM* | 88.7 | 93.0 | 91.5 | 75.2 | 87.1 |
| LP-FT | 85.3 | 93.7 | 97.9 | 84.2 | 90.3 |
| SWAD | 96.4 | 93.0 | 97.8 | 91.0 | 94.6 |
| MIRO | 99.2 | 96.7 | 99.9 | 94.4 | 97.6 |
| VL2V-SD | 99.6 | 98.9 | 99.9 | 93.6 | 98.0 |
| Attn. Tune | 97.3 | 97.6 | 98.9 | 91.7 | 96.4 |
| Model Parameter Averaging (MPA) | 98.6 | 95.8 | 98.2 | 89.2 | 95.4 |
| CLIPOOD (Shu et al., 2023) | 99.4 | 99.3 | 99.9 | 92.0 | 97.7 |
| MIRO + SWAD | 99.1 | 97.9 | 99.9 | 93.3 | 97.6 |
| MIRO + MPA | 99.0 | 98.5 | 100.0 | 94.7 | 98.1 |

Table 21: Per-domain breakdown for PACS (DomainBed-All)

| **Method** | Art | Cartoon | Photo | Sketch | Avg |
|---|---|---|---|---|---|
| CORAL | 81.4 | 89.2 | 91.7 | 70.5 | 83.2 |
| SAGM | 83.4 | 90.4 | 94.6 | 81.4 | 87.5 |
| ERM* | 83.0 | 92.1 | 91.5 | 74.0 | 85.2 |
| LP-FT | 80.5 | 92.7 | 97.9 | 82.7 | 88.4 |
| SWAD | 91.5 | 91.2 | 97.8 | 89.0 | 92.4 |
| MIRO | 97.1 | 96.4 | 99.8 | 92.2 | 96.4 |
| VL2V-SD | 97.7 | 98.5 | 99.9 | 91.3 | 96.9 |
| Attn. Tune | 96.4 | 96.5 | 98.8 | 89.7 | 95.4 |
| Model Parameter Averaging (MPA) | 97.2 | 94.8 | 98.2 | 87.1 | 94.3 |
| CLIPOOD (Shu et al., 2023) | 98.2 | 98.8 | 99.8 | 90.2 | 96.8 |
| MIRO + SWAD | 98.0 | 97.4 | 99.9 | 91.0 | 96.6 |
| MIRO + MPA | 97.8 | 98.0 | 99.9 | 92.9 | 97.2 |

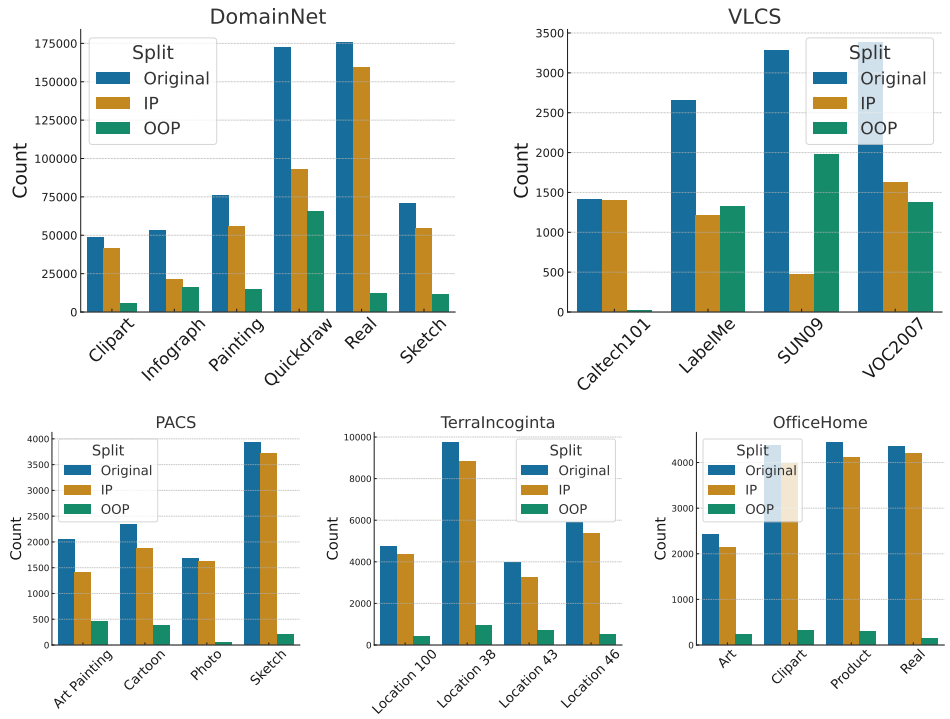


Figure 16: **DomainBed-(IP/OOP) Statistics**: Breakdown of DomainBed-IP and DomainBed-OOP counts, by dataset and domain. Overall, DomainNet and VLCS have the largest fraction of samples falling into DomainBed-OOP.

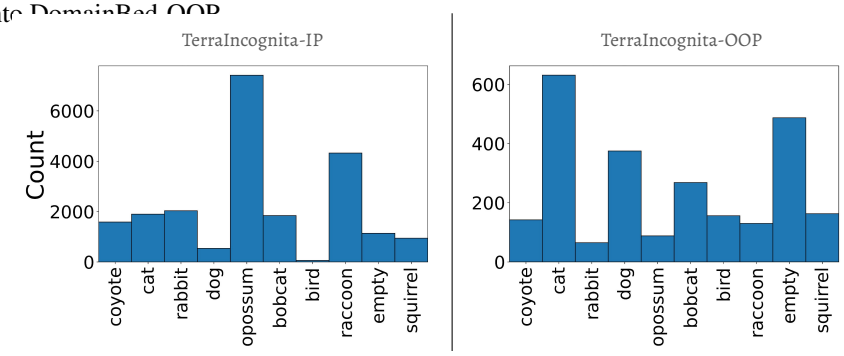


Figure 17: **Class-distribution shift:** TerraIncognita's class distribution differs between DB-IP and DB-OOP, indicating that some classes were better aligned during pretraining.

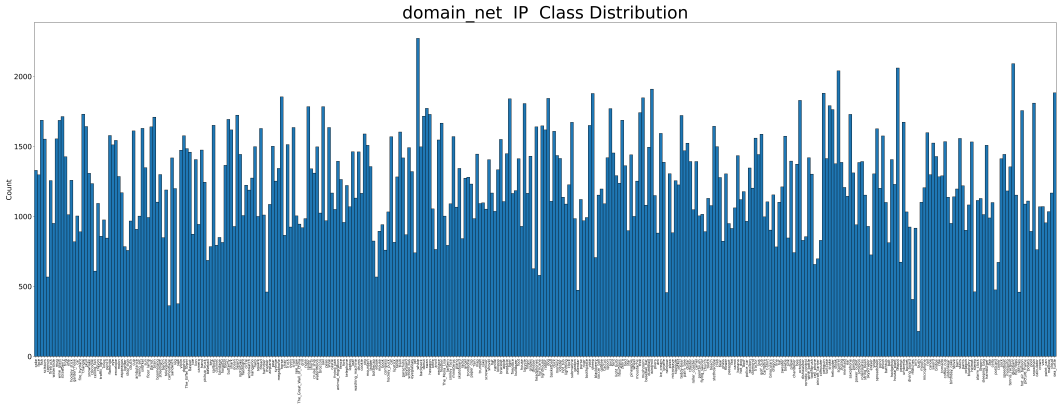


Figure 18: Class distribution of DomainNet-IP. Zoom in on pdf for best viewing.

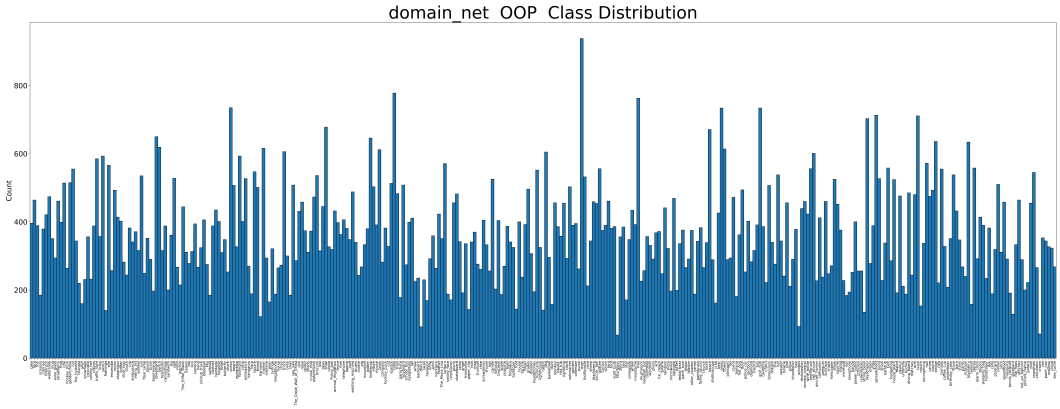


Figure 19: Class distribution of DomainNet-OOP. Zoom in on pdf for best viewing.

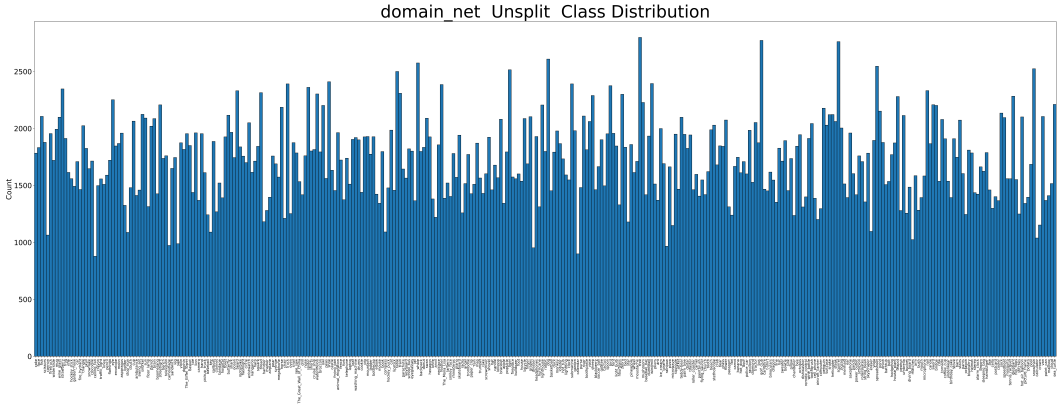


Figure 20: Class distribution of DomainNet before splitting. Zoom in on PDF before viewing.

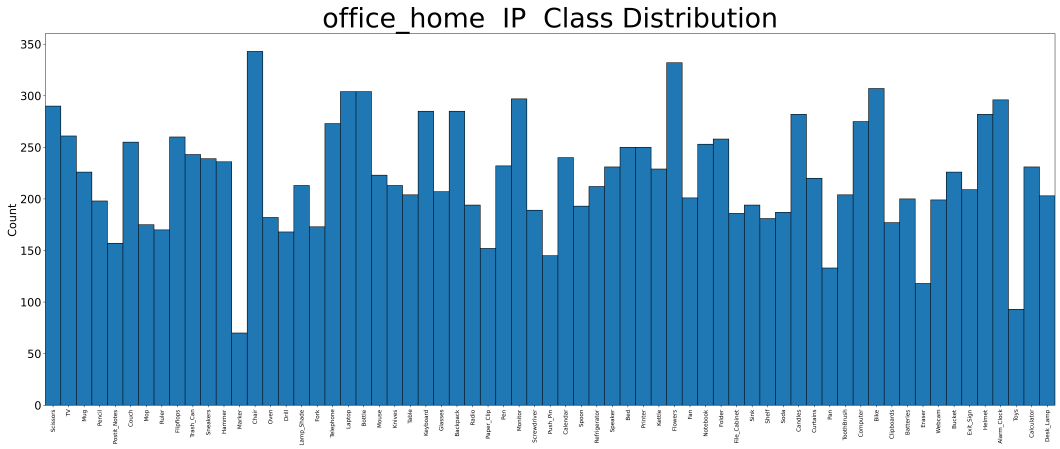


Figure 21: Class distribution of OfficeHome-IP. Zoom in on pdf for best viewing.

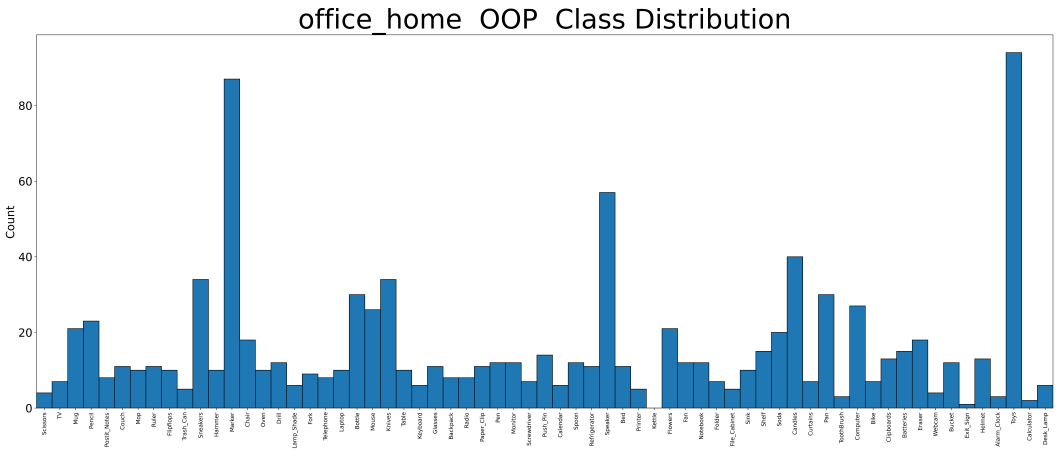


Figure 22: Class distribution of OfficeHome-OOP. Zoom in on pdf for best viewing.

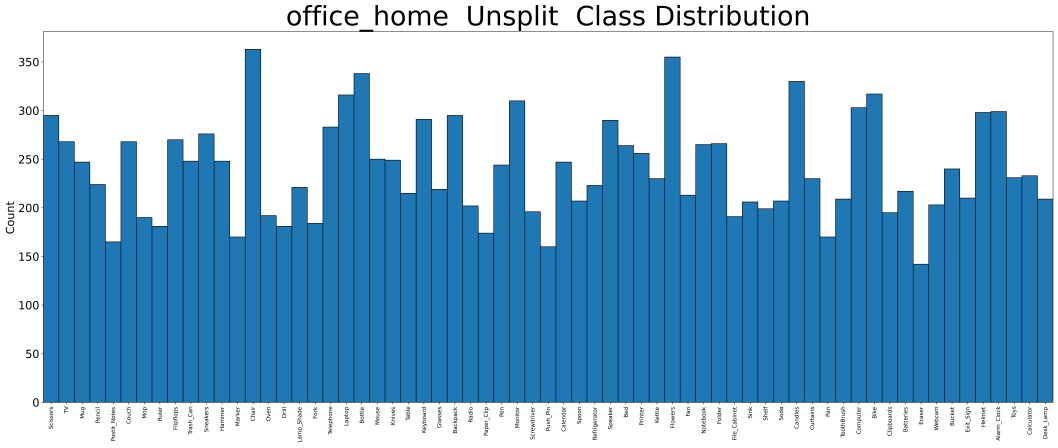


Figure 23: Class distribution of OfficeHome before Splitting. Zoom in on pdf for best viewing.

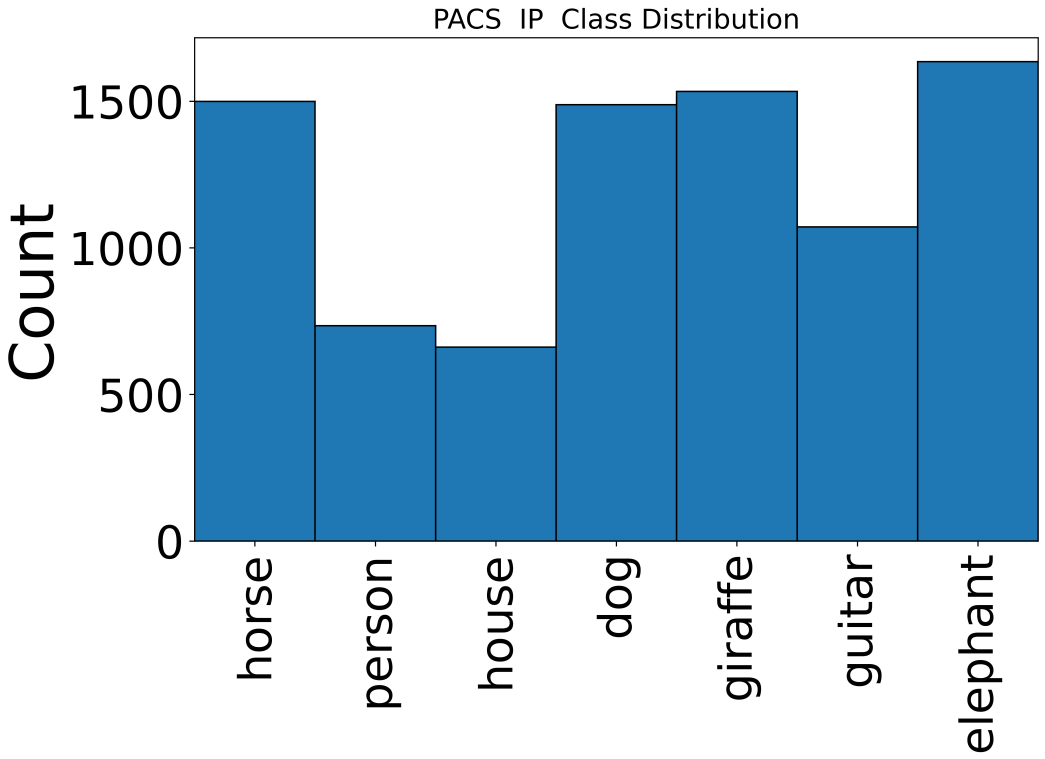


Figure 24: Class distribution of PACS-IP.

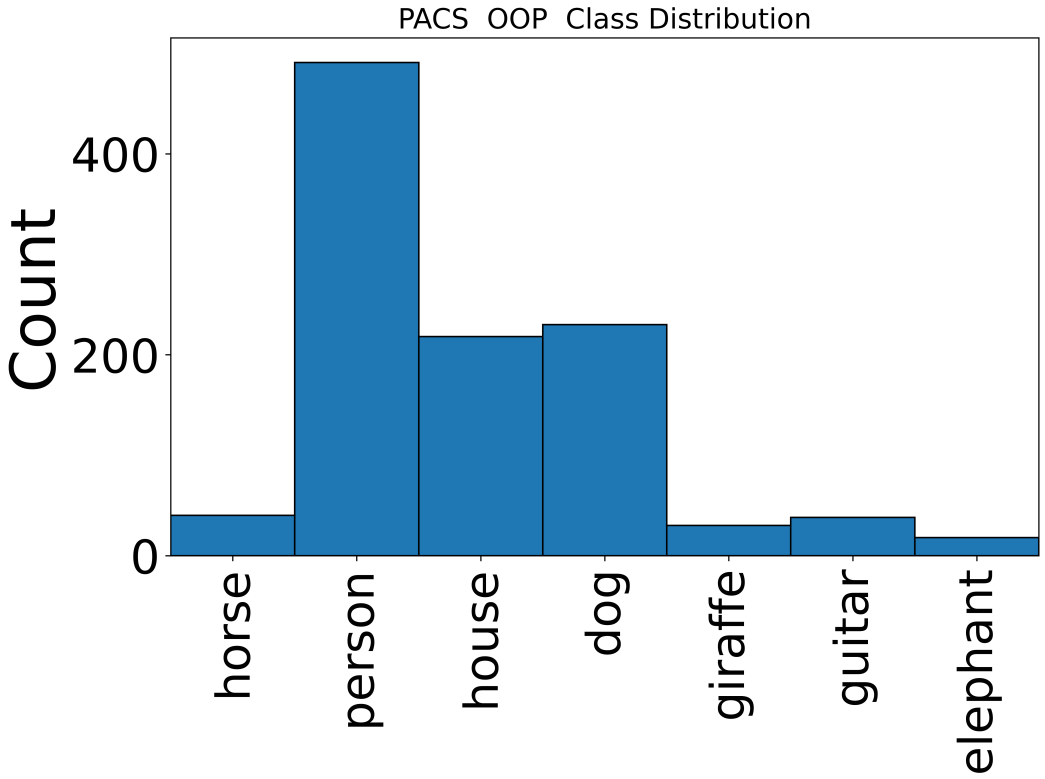


Figure 25: Class distribution of PACS-OOP.

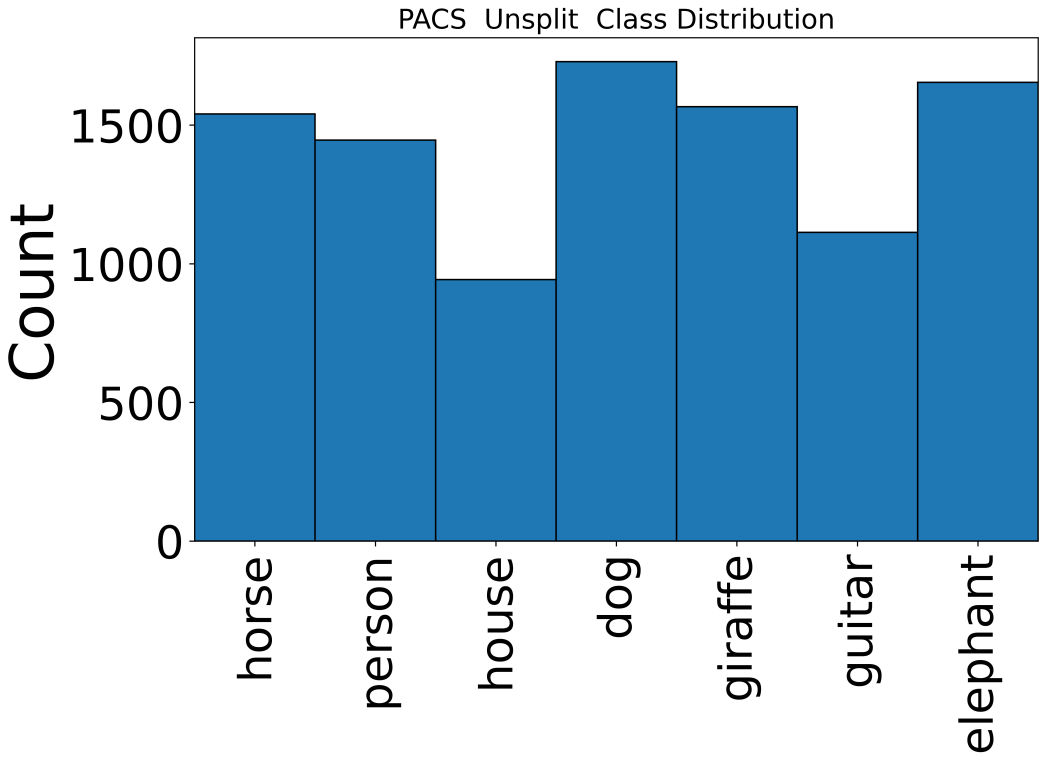


Figure 26: Class distribution of PACS before splitting.

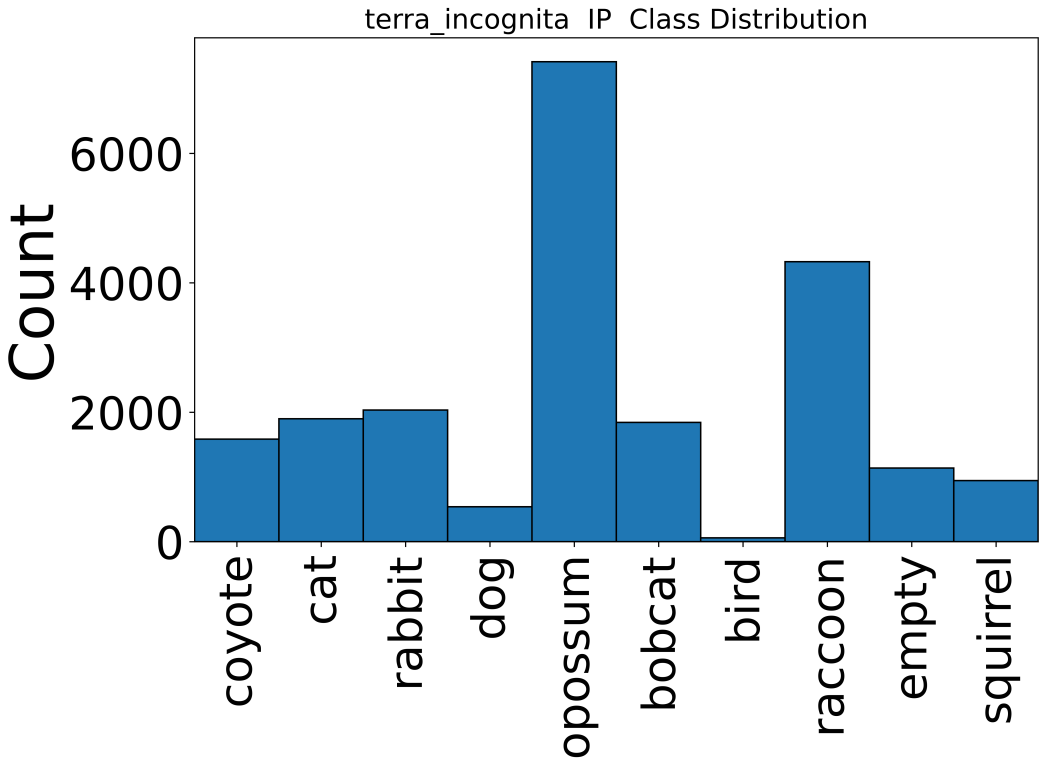


Figure 27: Class distribution of TerraIncognita-IP

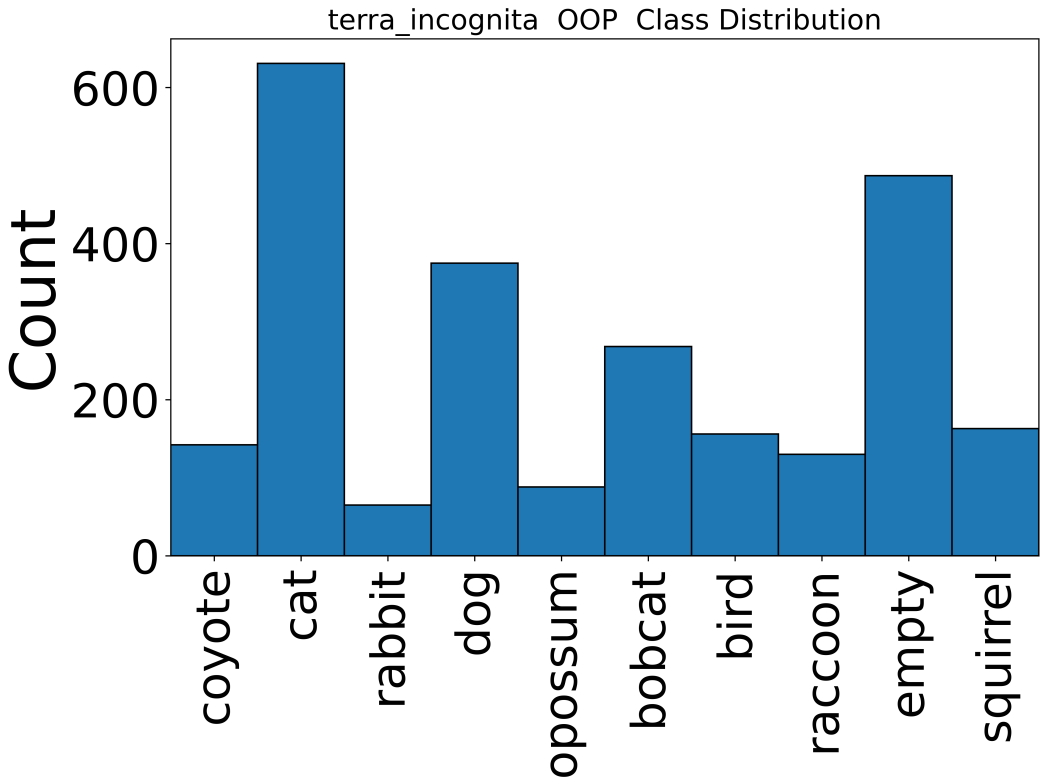


Figure 28: Class distribution of TerraIncognita-OOP

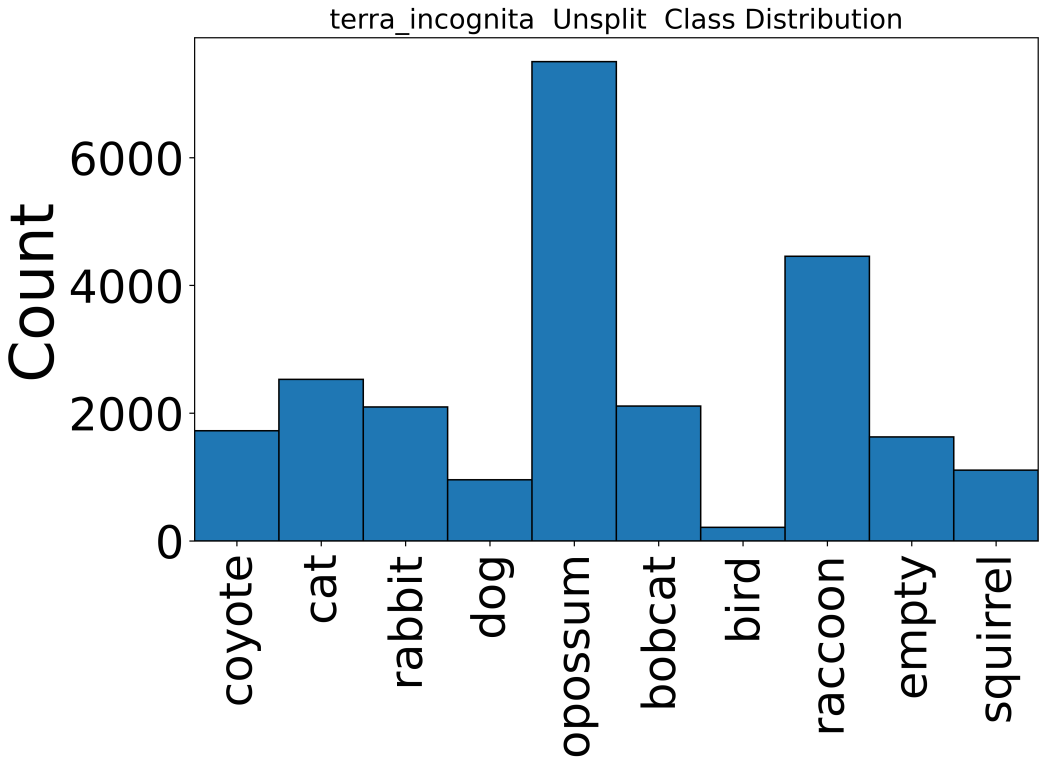


Figure 29: Class distribution of TerraIncognita before splitting.

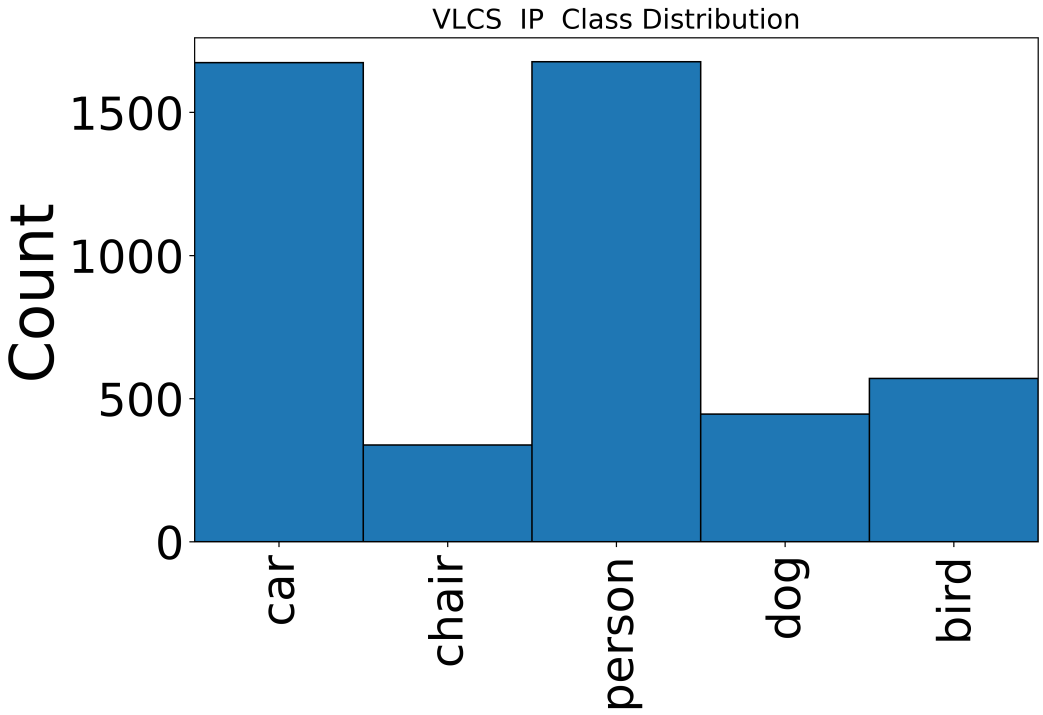


Figure 30: Class distribution of VLCS-IP

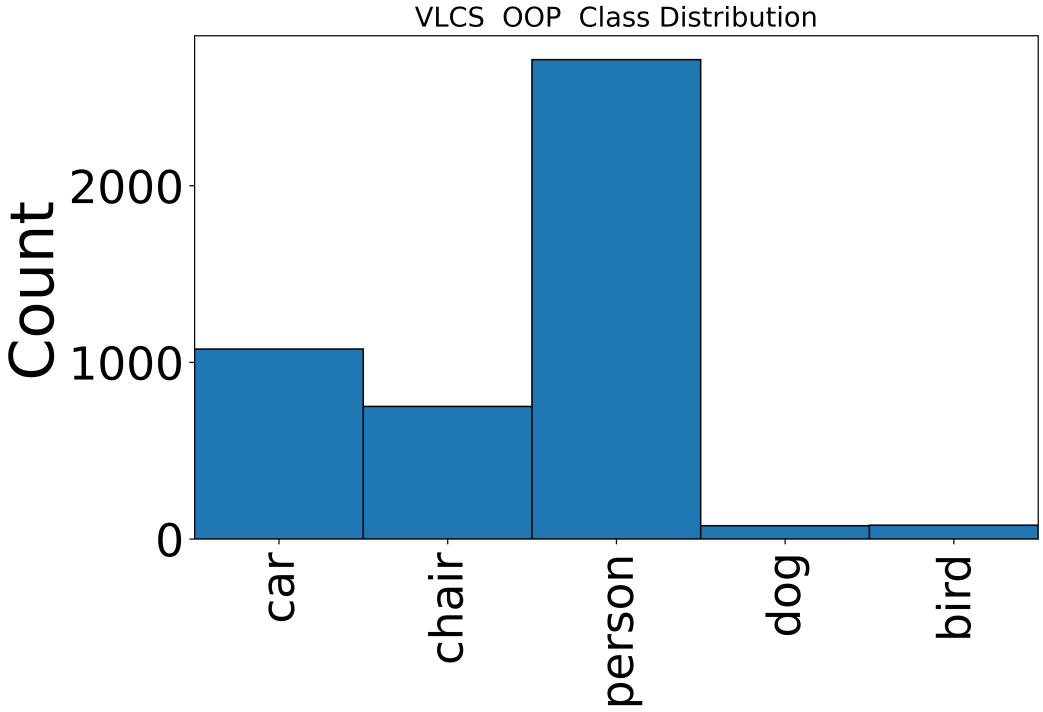


Figure 31: Class distribution of VLCS-OOP

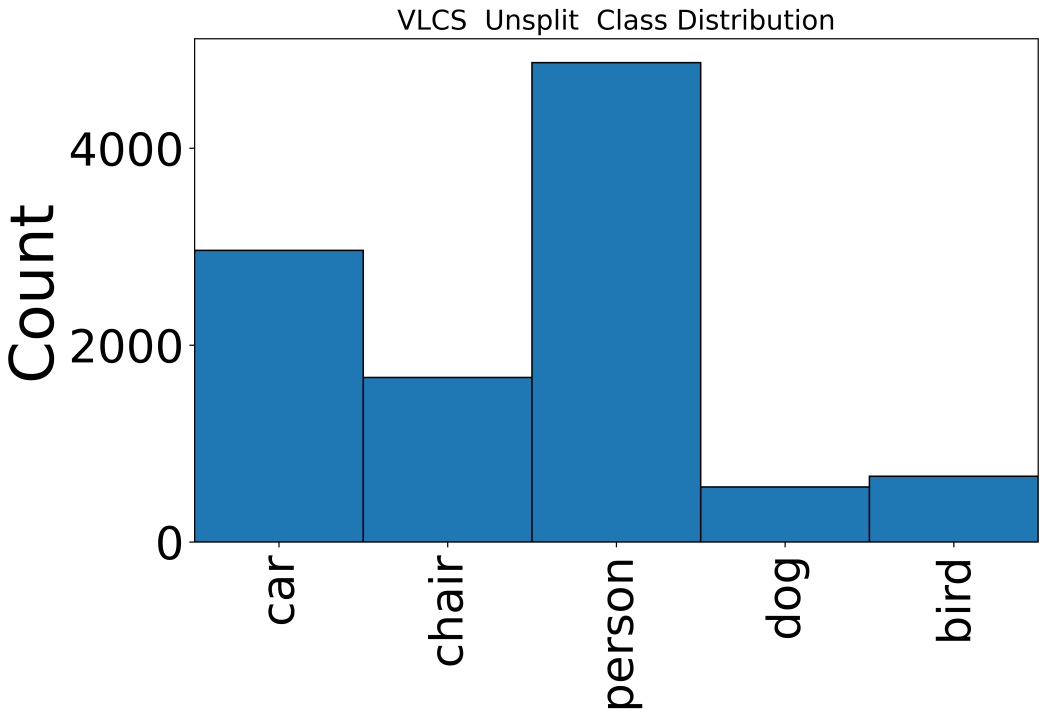


Figure 32: Class distribution of VLCS before splitting.