SemAlign: Language Guided Semi-supervised Domain Generalization

Muditha Fernando Kajhanan Kailainathan Krishnakanth Nagaratnam
Isuranga Senavirathne Ranga Rodrigo

Abstract

It is practically useful to have deep learning models capable of learning generalizable features that can perform well on unseen target domains in a data-efficient manner. This challenge is addressed by semi-supervised domain generalization (SSDG). Existing domain generalization (DG) and semi-supervised learning (SSL) methods demonstrate suboptimal performance in the SSDG setting compared to fully supervised DG methods, leaving room for improvement. Existing SSDG methods highlight the importance of achieving high pseudo-labeling (PL) accuracy and preventing model overfitting as the main challenges in SSDG. We highlight that the SSDG literature's excessive focus on PL accuracy, without considering the benefits of maximally utilizing data during training, limits potential performance improvements. We propose a novel approach to the SSDG problem by aligning the intermediate features of our model with the semantically rich and generalized feature space of a Vision Language Model (VLM) in a way that promotes domain-invariance. We also propose a simple but highly effective set of image-level augmentation strategies to encourage domain invariant feature learning. We also adopt changes at the model output level to enable the model to utilize all data samples in the training process and avoid overfitting. Extensive experimentation across four benchmarks against existing SSDG baselines suggests that our method achieves SOTA results both qualitatively and quantitatively. The code will be made publicly available.

1. Introduction

Most visual recognition models suffer substantial performance drop when there is shift across training and test data distributions [2, 32, 47]. The domain generalization (DG) research direction addresses this problem [2, 28, 34, 56]. In the DG setting, the training data consists of multiple source domains and the test data comes from domain(s) entirely unseen during training, referred to as target domain(s). Most methods that tackle the domain shift problem assume the availability of an abundance of labeled examples from all source domains. Since labeled data is costly to obtain, solutions that address domain shifts with limited labeled data could be highly utilitarian in practice. However, DG methods struggle in the limited data setting [14, 73].

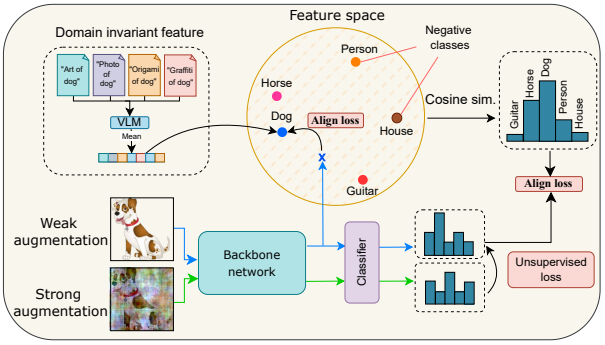


Figure 1. A conceptual diagram illustrating the proposed domain-invariant semantic alignment approach, where features are aligned with semantically rich representations from the text encoder of a VLM to achieve semi-supervised domain generalization.

We study the semi-supervised domain generalization (SSDG) problem. In addition to DG, this problem is also related to semi-supervised learning (SSL). Literature from SSL aims to maximize performance with limited labeled data and a large amount of unlabeled data [18, 26, 51, 52]. SSL methods such as FixMatch [51] demonstrate relatively better performance than DG methods in the SSDG setting. Hence, the majority of SSDG methods build on top of FixMatch with the objective of improving model-generalization [14, 15, 42, 73].

Although these methods have significantly improved upon FixMatch, room for improvement remains. Commonly cited limitations in SSDG literature are pseudo-labeling accuracy and overfitting to limited labeled data [14, 73]. Obtaining accurate pseudo-labels (PL) is challenging under multiple domain shifts [15, 73]. Therefore, alleviating domain-specific feature learning and promoting domain-invariant feature learning is crucial. We further analyze the PL statistics of SSL-based-SSDG methods and observe that the correct PLs retained after thresholding as a proportion of the entire dataset, representing the number of samples the model effectively utilizes for the "correct"

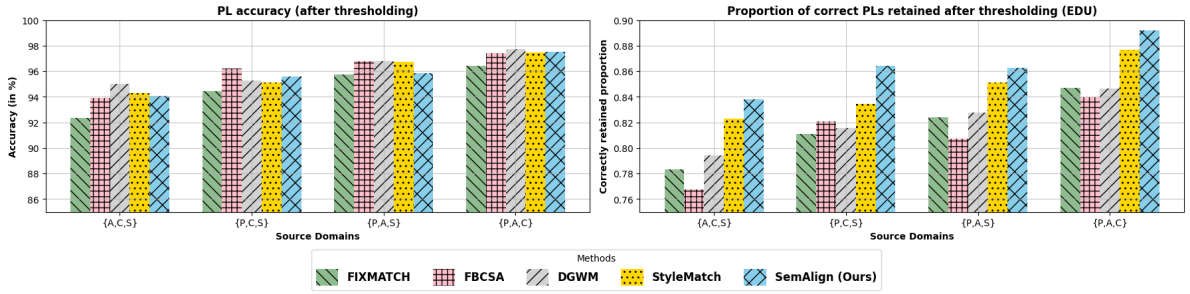


Figure 2. Left: PL accuracy after thresholding, Right: The correct pseudo-labels retained after thresholding as a proportion of the entire dataset (EDU) of the baselines FixMatch [51], FBC-SA [14], DGWM [15], StyleMatch [73], and ours. Our method achieves significant gains in proportion of samples retained after thresholding while maintaining comparable PL accuracy to SOTA SSL-based-SSDG methods. Here A, C, P, and S denote Art-painting, Cartoon, Photos, and Sketch domains, respectively.

learning, is a key limitation [35], as shown in Fig. 2. We coin the term Effective Data Utilization (EDU) to refer to this metric. In addition to correct PLs, valuable information exists in rejected samples as well. The inability of SSDG methods to leverage them is also a key limitation.

We propose a new SSDG approach towards addressing the above limitations from three perspectives: feature space, model output space, and input/pixel space. The feature space of the model is leveraged to tackle the pseudo-labeling accuracy problem by aligning visual features with class-label semantic priors from a pre-trained language encoder. This allows the model to map the visual features to a rich semantic space [12, 35]. We use the CLIP text encoder to generate these priors, using a technique that enables these class features to be domain-invariant [21]. CLIP has been trained on a large-scale, diverse dataset, giving it the capability to generalize across domains [8, 21]. Also, the CLIP text encoder has a robust understanding of image features, since it has been aligned with the CLIP image encoder [46]. Therefore, we utilize the label embedding space from the CLIP text encoder and refine it following PathCLIP [24], to strike a balance between task-specific adaptation and preserving CLIP's rich visual understanding. We further regularize the feature space alignment by contrasting with negative class label embeddings. We also attend to the model output space to address some of the aforementioned limitations. We introduce Entropy Meaning Loss (EML) following FullMatch [7], which improves PL accuracy by reducing the competitiveness of negative classes. This also improves the EDU by enabling more samples to pass the thresholding criterion [7]. To incorporate samples rejected at thresholding to the learning process, we introduce Adaptive Negative Learning (ANL) following FullMatch [7]. ANL allows the model to leverage all unlabeled data for training. To tackle the overfitting issue we follow StyleMatch [73] and use a stochastic classifier [4, 13]. This prevents the model from producing excessive incorrect PLs. To facilitate domain invariant feature learning, StyleMatch [73] uses style-transferred images from multiple domains. However, this adds a large overhead at training time (Tab. 5). Instead, in pixel space we introduce a set of specialized data augmentations such as Fourier-based augmentations [61, 64] and textural feature reduction. By these augmentations, we remove the domain-specific components of the input data, thus forcing the model to learn the semantic features instead of relying on spurious correlations, with a significantly lower overhead.

In summary, we make the following key contributions.

1. To the best of our knowledge, we are the first to leverage the rich output space of a Vision Language Model (**VLM**) as domain-invariant class priors to enhance pseudo-labeling accuracy in SSDG.
2. We highlight the importance of considering both the pseudo-labeling accuracy and the **Effective Data Utilization (EDU)** in SSDG and show significant gains in the latter.
3. We propose a novel approach to address the SSDG problem from three perspectives: (1) in the **feature space**, through alignment strategies; (2) in the model **output space**; and (3) in the **pixel space**, leveraging innovative data augmentation techniques. Our approach **does not rely on domain labels**.
4. We conduct experiments on four different DG datasets: PACS [27], OfficeHome [54], DigitsDG [67], and miniDomainNet [71], a subset of DomainNet [40], demonstrating **state-of-the-art (SOTA) performance** compared to other SSDG methods in the literature.

2. Related Work

2.1. Domain Generalization

DG is an extensively studied field. In this review we limit focus to methods closely related to our approach. A broader introduction to DG can be found in [72]. A common approach in DG is domain alignment [22, 29, 30, 33, 34, 37]. Here, the central idea is to minimize the distance between

features across source domains in order to produce invariant feature representations that are adaptable to the target domains as well. Another popular approach is to use data augmentations to minimize overfitting, thus ensuring that the model does not rely on domain-specific spurious correlations [43, 44, 50, 56, 57, 62, 66, 68–70]. In DG, augmentation techniques at the pixel level focus on approximating the meta-domain of all visual domains by retaining the semantic components and inducing variance in domain-specific components [56, 61]. However, most DG algorithms assume the availability of an abundance of labeled data. Their performance degrades rapidly as the access to labeled data is gradually limited [14, 73].

2.2. Semi-supervised Learning

SSL is a well-established area with numerous methods, such as entropy minimization [18], consistency learning [31, 52, 59, 60], and pseudo-labeling [26, 41]. Consistency learning takes the approach of making similar predictions for different views of the same input. Pseudo-labeling assigns labels to unlabeled instances by selecting the class with the highest predicted probability and then trains the model using these estimated pseudo-labels. Fixmatch [51], combines consistency learning and pseudo-labelling, suggesting a rather simple recipe for SSL. Its conceptual simplicity and superior results made this work highly influential in SSL, inspiring numerous methods to build upon its framework and achieve improved results [5, 7, 35, 58, 63]. However, the performance of stand-alone SSL methods are sub-optimal when there is a substantial domain shift in between training and test data [73].

2.3. Semi-supervised Domain Generalization

Stylematch [73] introduces the SSDG problem by pointing out the shortcomings when DG and SSL methods are naively adopted to the SSDG setting. One of the main issues addressed by most SSDG methods is the substantial drop in PL accuracy when multiple domains are found in training data [14, 15, 42, 73]. One approach in SSDG exploits domain-specific features in the input to improve PL accuracy [15, 42]. DGWM [15] learns a domain guided weight masking algorithm to achieve this, while MultiMatch [42] uses a criterion to fuse results from multiple domain-specialized classifiers and a global classifier. A second approach uses a shared classifier similar to the ERM baseline [49] where the penultimate features are expected to be domain invariant. StyleMatch replaces the classifier in FixMatch with a stochastic classifier [4] to minimize model overfitting, and a multiview consistency framework is used to encourage the feature encoder outputs to be domain-invariant. UPLM [23] improves PL accuracy by using an uncertainty parameter to filter out uncertain PLs. FBC-SA [14] leverages domain-aware prototypes derived from the labeled data to generate posteriors from the feature space, which are then aligned with the model output space. However, domain alignment is used to subsequently promote domain invariance in the feature space as training progresses. Our method too follows this second approach. Unlike the above methods, we utilize all three of input pixel space, feature space, and output space to improve PL accuracy and promote domain invariance. We also learn from all the samples and show its effectiveness in the SSDG setting.

2.4. VLMs in Domain Generalization

VLMs have been utilized for DG in recent works [8, 9, 21, 65]. This is due to (1) a deep understanding of the visual-semantic world, including nuances of domain shifts, acquired through a large dataset with richly descriptive image captions [45], and (2) the ability to simulate domain shifts in the visual space simply by using linguistic descriptions of domain-label names [46]. One approach uses the language encoder of the VLM, guided by learnable prompts, to make generalized predictions [1, 9, 38, 65]. Another approach is to use manual text prompts and obtain rich semantic features to align a vision encoder [8, 21, 24, 55]. The most related to our work are RISE [21], which leverages the CLIP text encoder to generate domain-invariant representations for feature alignment, and Path-CLIP [24], which proposes a residual feature refinement module to adapt the CLIP vision encoder to the domain of pathological images while mitigating the catastrophic forgetting issue. Despite the effectiveness of VLMs in DG, it has not yet been leveraged by the under-explored research direction of SSDG.

3. Method

Our SemAlign architecture includes a pre-trained text encoder, a Residual Feature Refinement (RFR) module, an image feature extractor, and a stochastic classifier. The text encoder generates domain-invariant class prototypes, and the feature extractor outputs are aligned with these prototypes. Further, objectives are used to encourage more PLs to clear the thresholding criterion and to learn from rejected samples. Enforcing consistency between weakly and strongly augmented image views further promotes domain-invariant learning. Figure 3 provides an overview of our method. The algorithm is provided in the suppl.

3.1. Problem Setting

In our work, $\mathcal{X}$ is the input image feature space and $\mathcal{Y} = \{1, 2, ..., C\}$ is the output space, a finite set of C possible classes. A domain is defined as a joint probability distribution P_{XY} over $\mathcal{X} \times \mathcal{Y}$. During training time, we will have access to D source datasets $\{\mathcal{S}^{(d)}\}_{d=1}^{D}$, sampled from related but distinct joint probability distributions, $\{P_{XY}^{(d)}\}_{d=1}^{D}$, each representing a distinct domain. Note

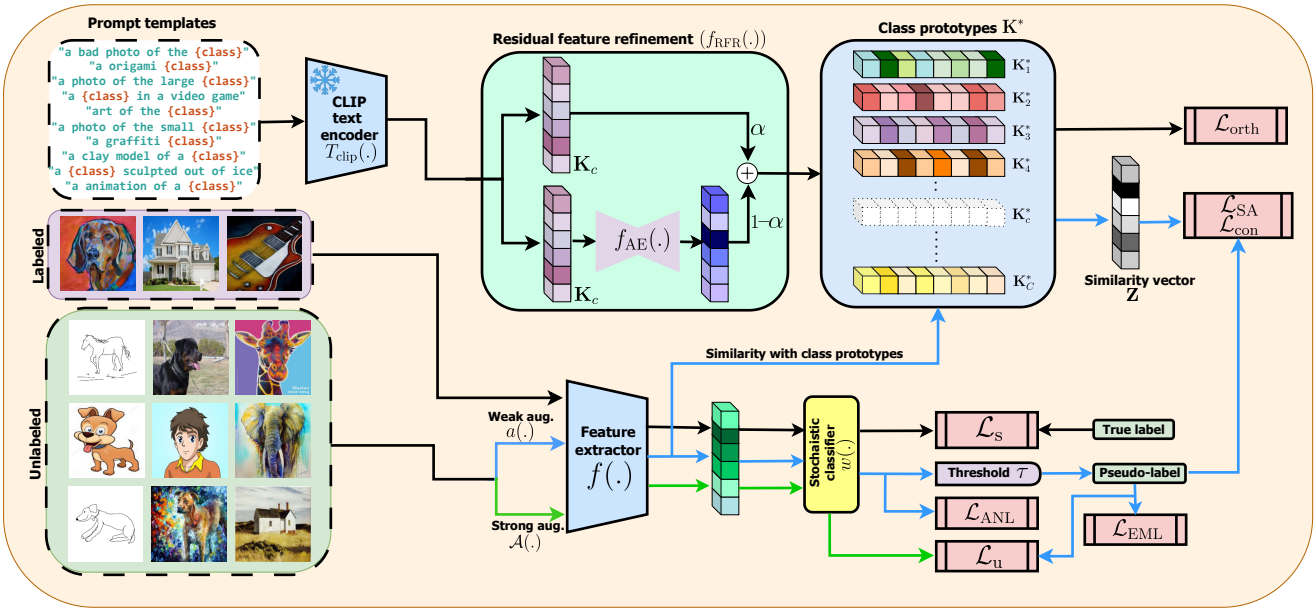


Figure 3. For class labels of the classification problem we generate domain invariant representations using CLIP [45] text encoder and further refine them using the RFR module to create the class prototype matrix ($\mathbf{K}^*$). The orthogonality of these class prototypes is encouraged using $\mathcal{L}_{\text{orth}}$. For a weakly augmented view, we align the feature extractor output with the corresponding class prototype guided by the PL generated by the classifier. Alignment is constrained using $\mathcal{L}_{\text{SA}}$ and $\mathcal{L}_{\text{con}}$. To reduce overfitting we use a stochastic classifier. At the classifier level, we minimize competition from competitive classes using $\mathcal{L}_{\text{EML}}$ and allow learning from rejected samples at the pseudo-labeling using $\mathcal{L}_{\text{ANL}}$. $\mathcal{L}_{\text{s}}$ and $\mathcal{L}_{\text{u}}$ are same as in FixMatch [51].

that $\forall d, d' \in \{1, 2, ..., D\}, d \neq d' \Rightarrow P_{XY}^{(d)} \neq P_{XY}^{(d')}$. Each source dataset $\mathcal{S}^{(d)}$ consists of labeled and unlabeled data, denoted $\mathcal{S}_\ell^{(d)}$ and $\mathcal{S}_u^{(d)}$ respectively. That is, $\mathcal{S}^{(d)} = \mathcal{S}_\ell^{(d)} \cup \mathcal{S}_u^{(d)}$, with $\mathcal{S}_\ell^{(d)} = \{(\mathbf{x}_i^{(d)}, y_i^{(d)})\} \sim_{i.i.d} P_{XY}^{(d)}$ and $\mathcal{S}_u^{(d)} = \{(\mathbf{x}_i^{(d)})\} \sim_{i.i.d} P_X^{(d)}$, where $P_X^{(d)}$ is the marginal distribution of $P_{XY}^{(d)}$ over $\mathcal{X}$. The source datasets contain n_ℓ labeled samples and n_u unlabeled samples. Generally, $n_\ell \ll n_u$ in SSDG (and SSL) settings. Each minibatch for training will comprise of B_ℓ labeled samples and B_u unlabeled samples. Our objective is to learn a model $\mathcal{F}$ using the source datasets, such that $\mathcal{F}(\mathbf{x}_i^{(\mathcal{T})}) = y_i^{(\mathcal{T})}$ with high probability, generalizing well to an unseen target domain $P_{XY}^{\mathcal{T}} = \{(\mathbf{x}_i^{(\mathcal{T})}, y_i^{(\mathcal{T})})\}$. Here, $\forall d, P_{XY}^{(\mathcal{T})} \neq P_{XY}^{(d)}$. We decompose $\mathcal{F} = w \circ f$, where $f : \mathbf{x} \to \mathbf{h}$ is a feature extractor and $w : \mathbf{h} \to y$ is a classifier. We also note that, for our implementation, meaningful class label names are required, while domain labels are not required.

3.2. The Preliminary Framework

Since the FixMatch framework [51] performs better than DG methods and other SSL methods in the SSDG setting [73], we build our method on top of FixMatch, similar to other work in the SSDG literature. FixMatch is an SSL algorithm that leverages consistency regularization and pseudo-labeling to improve model training with limited labeled data. Given a batch of labeled and unlabeled images, FixMatch first applies weak augmentation ($a(.)$) (*e.g.*, random flips and shifts) to the labeled samples and computes the standard cross-entropy loss ($\mathcal{L}_s$) in a supervised manner.

$$\mathcal{L}_s = \frac{1}{B_\ell} \sum_{b=1}^{B\ell} H(P_b \, , Q_b^w), \tag{1}$$

where P_b is the one-hot distribution of the given label, and $Q_b^w = P(y \mid w, f(a(\mathbf{x}_b)))$ is the model probability output distribution for input $\mathbf{x}_b$. For the unlabeled data, it generates PLs using the model's predictions on weakly augmented versions of the images, retaining only those with high confidence. These PLs are then used as targets for images after applying strong augmentation ($\mathcal{A}(.)$) (*e.g.*, RandAugment [10], CTAugment [3]), enforcing consistency through a cross-entropy loss.

$$\mathcal{L}_u = \frac{1}{B_u} \sum_{b=1}^{B_u} \mathbb{1}(\max(Q_b^w) \geq \tau) H(\hat{Q}_b^w, \, P(y \mid w, f(\mathcal{A}(\mathbf{x}_b)))), \tag{2}$$

where τ is a scalar hyper-parameter denoting the threshold above which we retain a PL and $\hat{Q}_b^w = \arg\max(Q_b^w)$ is the PL. For simplicity, we assume that $\arg\max$ applied to a probability distribution produces a corresponding "one-hot" distribution. The overall objective is a combination of the supervised and unsupervised losses.

FixMatch demonstrates suboptimal performance in the SSDG setting due to the distributional shifts found in the training data and the limited labeled data per domain [73]. Following StyleMatch [73], we replace the original classifier in the FixMatch framework with a stochastic classifier [4], to improve the generalization capability of the model by minimizing overfitting to limited labeled data. To further enable this framework to perform well in the SSDG setting, we extend this by providing solutions in feature space (see Sec. 3.3), model output space (see Sec. 3.4), and pixel space (see Sec. 3.5).

3.3. Aligning with CLIP Text Embeddings

We incorporate additional domain-invariant and semantically rich class prior information from the CLIP text encoder [45] to guide the training process. This promotes domain-invariant feature learning which leads to improved PL accuracy, which improves the model performance.

Domain-Invariant Class Prototypes: To utilize prior information from the CLIP text encoder, we create C domain-invariant class prototypes $\{\mathbf{K}_c\}_{\mathbf{c}=1}^{C}$ corresponding to the C classes of the classification problem, at the beginning of training. This is done by averaging M different prompt templates (t_c^m) for each class c (see Fig. 3), inspired by RISE [21]. That is,

$$\mathbf{K}_c = \frac{1}{M}\sum_{m=1}^{M} T_{\text{clip}}(t_c^m), \tag{3}$$

where $T_{\text{clip}}(.)$ is the pre-trained CLIP text encoder with frozen weights. We use a subset of the recommended list of eighty templates of text prompts by CLIP [21, 45] to span the entire meta-domain of images. Empirical results show that this approach is superior to using a single template [21]. We also point out that this averaging across domain-descriptive templates leads to domain invariance, since it has been demonstrated that domain shifts can be simulated in the CLIP vision embedding space by making corresponding simple algebraic manipulations in the CLIP text embedding space [46].

Residual Feature Refinement: Despite having a broad understanding of the visual world, humans pay increased attention to specific features to disambiguate among confusing classes of objects. Inspired by this observation, we hypothesize that reorganizing CLIP text embeddings in feature space can improve adaptation to classification tasks, particularly for confusing classes. However, this adaptation should happen without significant catastrophic forgetting of the rich vision-language embedding space. We achieve this through an RFR module, which mitigates forgetting of previously learned knowledge while optimizing an orthogonality loss term ($\mathcal{L}_{\text{orth}}$) that encourages class features to become mutually orthogonal. The inspiration for the RFR module is taken from Path-CLIP [24], however, we use it for a different objective.

$$\mathbf{K}_c^* = f_{\text{RFR}}(\mathbf{K}_c) = (1-\alpha)\cdot f_{\text{AE}}(\mathbf{K}_c) + \alpha\mathbf{K}_c, \tag{4}$$

where $f_{\text{RFR}}(.)$ describes the RFR module, $\mathbf{K}_c^*$ is the refined class prototype where prototypes of distinct but potentially similar classes are encouraged to be orthogonal, $f_{\text{AE}}(.)$ is a 2-layer fully connected auto-encoder module, and the hyperparameter α is the ratio between preserving original knowledge from CLIP and adapting to fine-tuned knowledge. α is typically a high value (*e.g.*, 0.9). The orthogonality loss is calculated as follows:

$$\mathcal{L}_{\text{orth}} = \frac{1}{C^2-C}||\mathbf{K}^*\mathbf{K}^{*\mathbf{T}} - \mathbf{I}_{\mathbf{C}\times\mathbf{C}}||_F^2, \tag{5}$$

where $\mathbf{K}^* = [\mathbf{K}_1^*; \mathbf{K}_2^*; ...; \mathbf{K}_C^*]$ is the class prototype matrix, $||.||_F^2$ is the Frobenius norm, and the denominator term is the number of non-diagonal elements in the gram matrix, $\mathbf{K}^*\mathbf{K}^{*\mathbf{T}}$. This helps us to to better distinguish similar and confusing classes, as we demonstrate in Sec. 4.2.

Feature Similarity and Alignment: We compute posterior class similarity scores for each weakly augmented image feature embedding using domain-invariant class priors from the CLIP text encoder, following the same approach as in CLIP zero-shot classification. The similarity vector $\mathbf{z} \in \mathbb{R}^C$ is computed for input $\mathbf{x}_b$ as follows:

$$\mathbf{z} = \mathbf{K}^{*\mathbf{T}} f(a(\mathbf{x}_b)), \tag{6}$$

By computing $\text{softmax}(\mathbf{z})$, we obtain the semantic class posterior probability distribution, $Q_b^{\text{sem}} = P_b(y \mid \mathbf{K}^*, f(a(\mathbf{x}_b)))$. This distribution is used to align the class posterior probabilities $Q_b^w = P_b(y \mid w, f(a(\mathbf{x})))$ from the model output space. The alignment is done by minimizing a cross-entropy loss:

$$\mathcal{L}_{\text{SA}} = \frac{1}{B_u}\sum_{b=1}^{B_u} \mathbb{1}(\max(Q_b^w) \geq \tau) H(Q_b^{\text{sem}} \mid Q_b^w), \tag{7}$$

This alignment with semantic and domain-invariant class representation $\mathbf{K}^*$ enhances the representation quality of the image feature representation $f(a(\mathbf{x}_b))$.

Improving Feature Discrimination: Taking inspiration from FBC-SA [14], we resort to sharpening the posterior class similarity scores $\mathbf{z}$, by increasing the contrast of the similarity of the predicted class, against the similarity of other classes. Dedicated objectives that discourage competition with the predicted class improve model performance. The following loss function ($\mathcal{L}_{\text{con}}$) is used to achieve this objective.

$$\mathcal{L}_{\text{con}} = 1 - \mathbf{z}[y_{\text{pred}}] + \frac{1}{C-1}\sum_{y_c \neq y_{\text{pred}}} \mathbf{z}[y_c], \tag{8}$$

where y_{pred} is the model prediction of the weakly augmented view, $w(f(a(\mathbf{x}_b)))$. The effectiveness of this module is shown in Tab. 2.

We define the feature-level constraint $\mathcal{L}_{\text{feat}}$ as the summation of the three complementary losses $\mathcal{L}_{\text{SA}}$, $\mathcal{L}_{\text{orth}}$, and $\mathcal{L}_{\text{con}}$.

$$\mathcal{L}_{\text{feat}} = \mathcal{L}_{\text{SA}} + \mathcal{L}_{\text{orth}} + \mathcal{L}_{\text{con}} \tag{9}$$

3.4. Maximizing Learning from Unlabeled Data

To simultaneously improve the quality of PLs and to utilize all the data samples in the training process, we adopt training losses such as EML ($\mathcal{L}_{\text{EML}}$) and ANL ($\mathcal{L}_{\text{ANL}}$) from FullMatch [7], a method that addresses the SSL problem.

Entropy Meaning Loss: For the PLs that meet the thresholding criterion, EML reduces competition for the pseudo-label class from competing classes with significantly large probability values. This is done by neutralizing the competitiveness of these competing classes by enforcing a uniform distribution among these classes, which will improve the quality of PLs. This will increase the likelihood that the data samples will overcome the thresholding criterion [7].

Adaptive Negative Learning: ANL helps to learn useful information from all the data samples, even if they fail to pass the thresholding criterion. Even for ambiguous examples with multiple classes having competitive probability values, some classes will have extremely low probabilities. An adaptive k-value is calculated by ensuring that the top-k error rate is close to zero, and a uniform distribution is enforced on the remaining classes, since the model is highly certain that these classes do not correspond to the given example. The ANL objective exploits this knowledge by learning it using a negative learning approach [6, 48]. This allows the model to learn from all the data samples.

We ablate and demonstrate that these objectives not only enhance performance but also increase the EDU (Fig. 2).

We define the output-level constraint $\mathcal{L}_{\text{out}}$ as the summation of the two complementary losses $\mathcal{L}_{\text{EML}}$ and $\mathcal{L}_{\text{ANL}}$.

$$\mathcal{L}_{\text{out}} = \mathcal{L}_{\text{EML}} + \mathcal{L}_{\text{ANL}} \tag{10}$$

Our overall loss has four different terms, the supervised and unsupervised losses from [51], feature-level, and output-level constraints,

$$\mathcal{L} = \mathcal{L}_{\text{s}} + \mathcal{L}_{\text{u}} + \mathcal{L}_{\text{feat}} + \mathcal{L}_{\text{out}} \tag{11}$$

3.5. Data Augmentations

We devise data augmentation methods that increase the semantic component of images, thus enabling the model to solely rely on these features, encouraging domain-invariance in the feature space. We randomly apply our augmentations in addition to the FixMatch strong augmentations. This will reduce the effect of the model overfitting to the training data by producing substantially varied views of the image, at each epoch.

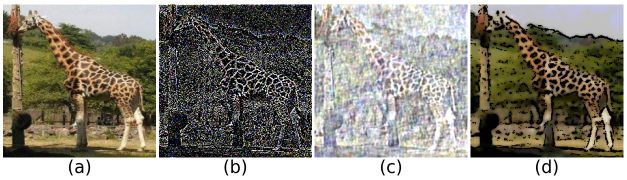


Figure 4. Our augmentation methods. (a) Raw image, (b) Phase-only image reconstruction, (c) Amplitude swapping, (d) Texture reduction. We apply these augmentations randomly along-side FixMatch [51] augmentations as the strong augmentation.

Augmentations in the Fourier Domain: Most semantic information resides in the phase component of a signal in the Fourier domain while the amplitude component primarily encodes non-semantic attributes such as texture and color [39]. Since semantic features tend to remain invariant across visual domains, we leverage this insight to generate data augmentations. While inspired by prior work [61, 64], our approach differs in its methodology and application. We use two methods: (1) We retain only the phase component while setting the amplitude to a constant to preserve only the semantic information (see Fig. 4 (b)). (2) We maintain a dynamically updated amplitude bank from previous samples and randomly swap the amplitude components of training samples with those from the bank to simulate novel visual domains and enhance generalization (see Fig. 4 (c)).

Texture Reduction: CNNs are known to have a bias towards textural components in images over semantic and shape attributes [17]. This inherent bias affects model generalization. To mitigate it, we (1) reduce the image's color space resolution to flatten nuanced variations and (2) apply smoothing using small kernels to distort textural features. However, this smoothing might blur the edges of the objects, distorting useful semantic information. To overcome this, we perform this smoothing after overlaying an edge mask of the image, which would prevent smoothing across these edges (see Fig. 4 (d)).

The effect of these augmentations are shown in Tab. 2 and Tab. 5. A detailed description is provided in suppl.

4. Experiments

Datasets: We use the commonly used DG datasets, PACS [27], OfficeHome [54], DigitsDG [67], and miniDomainNet, a subset of DomainNet [40] for our experiments. A detailed description of the datasets is provided in suppl.

Training and Implementation: We follow the same training setup used in StyleMatch [73]. Experiments are done with 5 and 10 labeled data samples per class, and the labeled data is sourced randomly for each batch. A batch

Table 1. SSDG accuracy on target domain (%). For each dataset, the average over 5 independent trials is reported. For each SSDG setting (5, 10 labels) the average over all datasets is reported. Best results are in **bold** and the second best is underlined. We achieve an average performance gain of **+2**% and **+2.6**% in the 5 and 10 labels case, over the second best model.

# labeled data			5 labels					10 labels		
Model	**PACS**	**OfficeHome**	**DigitsDG**	**miniDomainNet**	**Avg**	**PACS**	**OfficeHome**	**DigitsDG**	**miniDomainNet**	**Avg**
FixMatch [51]	75.2 ± 1.3	55.2 ± 0.9	55.0 ± 2.8	34.6 ± 1.3	55.0	77.1 ± 2.0	58.2 ± 1.0	64.5 ± 1.4	39.8 ± 0.6	59.9
StyleMatch [73]	$\underline{78.4 \pm 0.9}$	$\underline{56.3 \pm 0.6}$	55.9 ± 1.6	$\underline{36.6 \pm 1.0}$	56.8	$\underline{79.7 \pm 1.9}$	$\underline{59.8 \pm 0.7}$	66.1 ± 1.6	40.6 ± 1.2	61.6
FBC-SA [14]	75.4 ± 1.1	55.8 ± 0.7	$\underline{64.3 \pm 0.5}$	36.1 ± 1.2	$\underline{57.9}$	78.1 ± 2.4	59.3 ± 1.0	$\underline{69.4 \pm 1.4}$	$\underline{41.3 \pm 1.1}$	$\underline{62.0}$
DGWM [15]	78.0 ± 0.8	$\underline{56.3 \pm 0.7}$	55.9 ± 0.6	36.4 ± 1.4	56.7	78.8 ± 1.6	59.6 ± 1.0	65.6 ± 1.5	$\underline{41.3 \pm 1.5}$	61.3
SemAlign (Ours)	$\mathbf{79.6 \pm 0.5}$	$\mathbf{57.0 \pm 0.8}$	$\mathbf{66.3 \pm 0.6}$	$\mathbf{36.7 \pm 1.2}$	**59.9**	$\mathbf{81.8 \pm 1.7}$	$\mathbf{60.3 \pm 0.9}$	$\mathbf{74.2 \pm 1.4}$	$\mathbf{42.2 \pm 1.3}$	**64.6**

Table 2. Ablation study on PACS with 10 labels per class. SC: stochastic classifier, $\mathcal{L}_{\text{out}}$: Sec. 3.4, Data aug: Sec. 3.5, $\mathcal{L}_{\text{SA}}$, $\mathcal{L}_{\text{orth}}$, $\mathcal{L}_{\text{con}}$: Sec. 3.3).

Method	Average
Baseline [51]	77.1
Baseline + SC	78.1
Baseline + SC + $\mathcal{L}_{\text{out}}$	78.3
Baseline + SC + $\mathcal{L}_{\text{out}}$ + Data aug	79.5
Baseline + SC + $\mathcal{L}_{\text{out}}$ + Data aug + $\mathcal{L}_{\text{SA}}$	80.8
Baseline + SC + $\mathcal{L}_{\text{out}}$ + Data aug + $\mathcal{L}_{\text{SA}}$ + $\mathcal{L}_{\text{con}}$	81.1
Baseline + SC + $\mathcal{L}_{\text{out}}$ + Data aug + $\mathcal{L}_{\text{SA}}$ + $\mathcal{L}_{\text{orth}}$	81.2
Baseline + SC + Data aug + $\mathcal{L}_{\text{SA}}$ + $\mathcal{L}_{\text{orth}}$ + $\mathcal{L}_{\text{con}}$	81.0
Baseline + SC + $\mathcal{L}_{\text{out}}$ + Data aug + $\mathcal{L}_{\text{SA}}$ + $\mathcal{L}_{\text{orth}}$ + $\mathcal{L}_{\text{con}}$ (Ours)	**81.8**

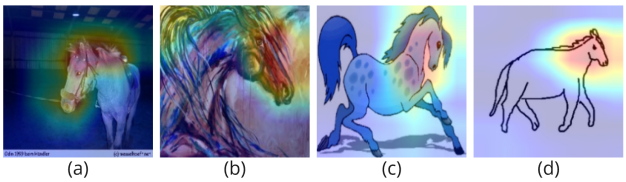
consists of 16 labeled and 16 unlabeled data samples. We use ResNet18 [20] as our feature extractor with ImageNet [11] pretrained weights. The SGD optimizer is used and the feature extractor and the classifier head have initial learning rates 0.003 and 0.01 respectively, which eventually decay following the cosine annealing rule. We train all the models for 20 epochs on all datasets. We report top-1 accuracy averaged over 5 independent trials.

Evaluation Protocol: We use the leave-one-domain out protocol [19] that has been adopted by most prior work in SSDG [14, 15, 73].

Baselines: We choose [51] as a baseline from the SSL domain, for interpretation. From the SSDG domain, we choose recent SOTA methods such as StyleMatch [73], FBC-SA [14], and DGWM [15]. For a fair comparison, we only use work that has publicly available code bases, as it is imperative to use the same labeled/unlabeled data split for experiments since these models are highly sensitive to these splits.

4.1. Results

Model Generalization: For the settings with 5 and 10 labeled data samples per class, the model accuracies for the target domain for our method and the other baselines are reported in Tab. 1. As shown, our method achieves the best results across all experiments, while the second-best results are distributed among methods such as StyleMatch [73], FBC-SA [14], and DGWM [15]. While identifying the second-best method is ambiguous due to varying performance across different datasets, our superiority is evident, as our model consistently leads across all the experiments by considerable margins. Our model leads the average performance of the second-best model by 2% and 2.6% in the 5 and 10 labels case. To further illustrate our model's superiority, in the 5 and 10 labels setting, its average performance gain over the FixMatch baseline is 1.68 and 2.23 times that of the second-best model.



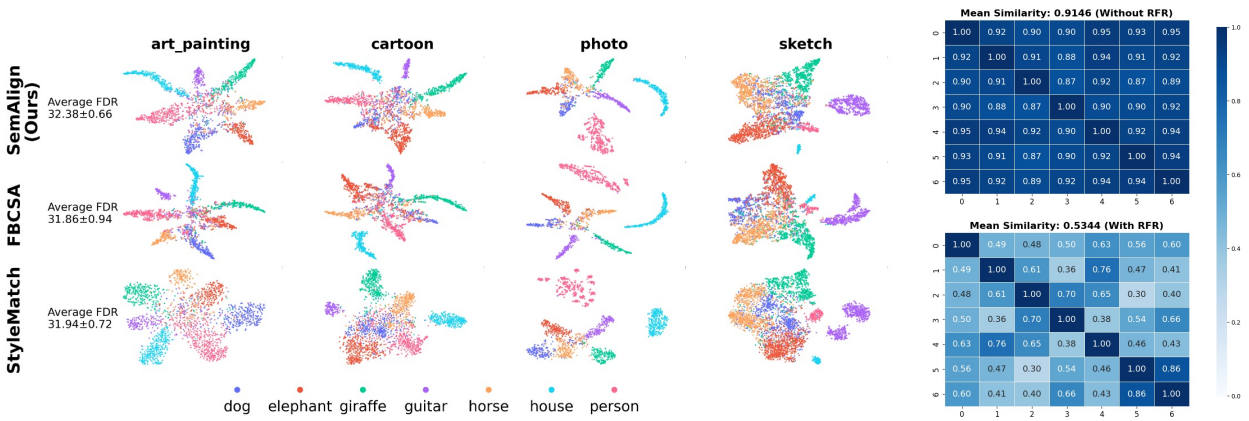
(a) (b) (c) (d)

Figure 5. Grad-CAM visualization of model focus in (a) photo, (b) art-painting, (c) cartoon, and (d) sketch domains for horse class in PACS dataset. Our model consistently attends to the head of the horse and discriminates it from other classes.

Effective Data Utilization: Fig. 2 shows the PL accuracy and the EDU for the PACS dataset for all combinations of source domains in the 10 labels per class setting. As shown, our PL accuracy is competitive with the other SSDG baselines, averaging just 0.45% lower than the best method (FBC-SA). However, the EDU is significantly higher than other methods, exceeding FBCSA, which had the best PL accuracy, by an average of 5.25% and StyleMatch, which has the second best EDU by, 2.00%. In the absence of $\mathcal{L}_{\text{out}}$, the EDU of our method drops by 1.25%.

t-SNE Visualizations: As shown in Fig. 6a our approach achieves a higher Fisher Discrimination Ratio (FDR) score, indicating that the features are more effective in distinguishing between classes. We only visualize the methods with the highest average FDR scores and the visualizations for the remaining methods can be found in suppl.

Grad-CAM Visualization: As shown in Fig. 5, our model has learned invariant features across multiple domains, indicating the generalization capability of our model. More



(a) (b)

Figure 6. (a) t-SNE visualization of the feature space for PACS dataset (10 labels per class). The average log Fisher Discriminant Ratio (FDR) (±1 standard deviation) was calculated over 5 random seeds, over all 4 domains. Higher scores indicate more compact and well-separated classes in the higher dimensional space. (b) Similarity matrices for the class prototype embeddings of the PACS dataset, before and after adding the RFR module. RFR reduces the similarity between class prototypes while preserving the semantics.

visualizations can be found in suppl.

4.2. Ablation Studies

Impact of Different Components: In Tab. 2 we report the contribution of each key component of our approach by sequentially adding them to FixMatch [51], because most components are complementary to each other. Each component is capable of improving the average accuracy over all target domains. Similar to StyleMatch [73] we also report the importance of stochastic classifier as it gives 1% improvement over FixMatch. Our data augmentation scheme gives a 1.2% gain, while the feature alignment gives a 1.3% gain.

RFR Module: Refinement of CLIP features using the RFR module and $\mathcal{L}_{\text{orth}}$ gains 0.4% improvement in the presence of $\mathcal{L}_{\text{SA}}$ and gains 0.7% in the presence of both $\mathcal{L}_{\text{SA}}$ and $\mathcal{L}_{\text{con}}$. As shown in Fig. 6b, this reduces the similarity between prototype embeddings, suggesting improved discriminability.

4.3. Comparing Components with Similar Work

Feature Space Constraints: Both FBC-SA [14] and our method impose feature space consistency constraints. As demonstrated in the t-SNE plot, both the methods result in more discriminative and well-separated feature spaces. However, our approach achieves a higher FDR score, showing its better discriminability.

StyleMatch Multiview Consistency vs Ours: StyleMatch and our method use data augmentations to promote domain-invariance. However, the computational overhead of our method (56.24%) is singificantly lesser than that of StyleMatch (153.15%). We report the average time per epoch in seconds on a single Quadro RTX 6000 GPU for the PACS dataset for the 10 labels setting in Tab. 5.

Table 3. Training overhead due to augmentations in comparison with FixMatch

Method	Average time/epoch (sec)	Overhead
FixMatch [51]	38.53	-
StyleMatch [73]	97.54	153.15%
SemAlign (Ours)	60.2	**56.24%**

5. Conclusion

We present a novel approach that addresses the SSDG problem. We map intermediate features of the model to the semantically rich feature space of a VLM in a domain-invariant manner encouraging domain-invariant feature learning. Further, we introduce a set of simple data augmentation methods to encourage domain invariant feature learning and reduce overfitting. Our method is also capable of utilizing all data samples into the learning process. Results on four benchmarks, against four top-performing SSDG methods, show that our model has notable gains. A potential limitation of our method is the requirement of meaningful class labels which is left for future work.

References

[1] Shuanghao Bai, Yuedi Zhang, Wanqi Zhou, Zhirong Luan, and Badong Chen. Soft prompt generation for domain generalization. In *ECCV*, pages 434–450. Springer, 2025. 3

[2] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010. 1

[3] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019. 4

[4] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. pages 1613–1622. PMLR, 2015. 2, 3, 5

[5] Hao Chen, Ran Tao, Yue Fan, Yidong Wang, Jindong Wang, Bernt Schiele, Xing Xie, Bhiksha Raj, and Marios Savvides. Softmatch: Addressing the quantity-quality trade-off in semi-supervised learning. *arXiv preprint arXiv:2301.10921*, 2023. 3

[6] John Chen, Vatsal Shah, and Anastasios Kyrillidis. Negative sampling in semi-supervised learning. pages 1704–1714. PMLR, 2020. 6

[7] Yuhao Chen, Xin Tan, Borui Zhao, Zhaowei Chen, Renjie Song, Jiajun Liang, and Xuequan Lu. Boosting semi-supervised learning by exploiting all unlabeled data. In *CVPR*, pages 7548–7557, 2023. 2, 3, 6

[8] Zining Chen, Weiqiu Wang, Zhicheng Zhao, Fei Su, Aidong Men, and Hongying Meng. Practicaldg: Perturbation distillation on vision-language models for hybrid domain generalization. In *CVPR*, pages 23501–23511, 2024. 2, 3

[9] Junhyeong Cho, Gilhyun Nam, Sungyeon Kim, Hunmin Yang, and Suha Kwak. Promptstyler: Prompt-driven style generation for source-free domain generalization. In *ICCV*, pages 15702–15712, 2023. 3

[10] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR*, pages 702–703, 2020. 4

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 7

[12] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. *NeurIPS*, 26, 2013. 2

[13] Yarin Gal and Zoubin Ghahramani. Bayesian convolutional neural networks with bernoulli approximate variational inference. *arXiv preprint arXiv:1506.02158*, 2015. 2

[14] Chamuditha Jayanga Galappaththige, Sanoojan Baliah, Malitha Gunawardhana, and Muhammad Haris Khan. Towards generalizing to unseen domains with few labels. In *CVPR*, pages 23691–23700, 2024. 1, 2, 3, 5, 7, 8

[15] Chamuditha Jayanaga Galappaththige, Zachary Izzo, Xilin He, Honglu Zhou, and Muhammad Haris Khan. Domain-guided weight modulation for semi-supervised domain generalization. *arXiv preprint arXiv:2409.03509*, 2024. 1, 2, 3, 7

[16] Y Ganin and V Lempitsky. Unsupervised domain adaptation by backpropagation. arxiv 2014. *arXiv preprint arXiv:1409.7495*, 2022. 1

[17] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018. 6

[18] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *NeurIPS*, 17, 2004. 1, 3

[19] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020. 7

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 7

[21] Zeyi Huang, Andy Zhou, Zijian Ling, Mu Cai, Haohan Wang, and Yong Jae Lee. A sentence speaks a thousand images: Domain generalization through distilling clip with language guidance. In *ICCV*, pages 11685–11695, 2023. 2, 3, 5

[22] Xin Jin, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Feature alignment and restoration for domain generalization and adaptation. *arXiv preprint arXiv:2006.12009*, 2020. 2

[23] Adnan Khan, Mai A Shaaban, and Muhammad Haris Khan. Improving pseudo-labelling and enhancing robustness for semi-supervised domain generalization. *arXiv preprint arXiv:2401.13965*, 2024. 3

[24] Zhengfeng Lai, Joohi Chauhan, Brittany N Dugger, and Chen-Nee Chuah. Bridging the pathology domain gap: Efficiently adapting clip for pathology image analysis with limited labeled data. In *ECCV*, pages 256–273. Springer, 2024. 2, 3, 5

[25] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 1

[26] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. page 896. Atlanta, 2013. 1, 3

[27] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, pages 5542–5550, 2017. 2, 6, 1

[28] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, 2018. 1

[29] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *CVPR*, pages 5400–5409, 2018. 2

[30] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *ECCV*, pages 624–639, 2018. 2

[31] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE TPAMI*, 41(8):1979–1993, 2018. 3

[32] Jose G Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern recognition*, 45(1):521–530, 2012. 1

[33] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, pages 5715–5725, 2017. 2

[34] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. pages 10–18. PMLR, 2013. 1, 2

[35] Islam Nassar, Samitha Herath, Ehsan Abbasnejad, Wray Buntine, and Gholamreza Haffari. All labels are not created equal: Enhancing semi-supervision via label grouping and co-training. In *CVPR*, pages 7241–7250, 2021. 2, 3

[36] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NeurIPS*, page 4. Granada, 2011. 1

[37] Toan Nguyen, Kien Do, Bao Duong, and Thin Nguyen. Domain generalisation via risk distribution matching. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2790–2799, 2024. 2

[38] Hongjing Niu, Hanting Li, Feng Zhao, and Bin Li. Domain-unified prompt representations for source-free domain generalization. *arXiv preprint arXiv:2209.14926*, 2022. 3

[39] Alan V Oppenheim and Jae S Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69(5):529–541, 1981. 6

[40] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, pages 1406–1415, 2019. 2, 6, 1

[41] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *CVPR*, pages 11557–11568, 2021. 3

[42] Lei Qi, Hongpeng Yang, Yinghuan Shi, and Xin Geng. Multimatch: Multi-task learning for semi-supervised domain generalization. *ACM MM*, 20(6):1–21, 2024. 1, 3

[43] Fengchun Qiao and Xi Peng. Uncertainty-guided model generalization to unseen domains. In *CVPR*, pages 6790–6800, 2021. 3

[44] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *CVPR*, pages 12556–12565, 2020. 3

[45] 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. pages 8748–8763. PMLR, 2021. 3, 4, 5

[46] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2, 3, 5

[47] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? pages 5389–5400. PMLR, 2019. 1

[48] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv preprint arXiv:2101.06329*, 2021. 6

[49] Stephan R Sain. The nature of statistical learning theory, 1996. 3, 2

[50] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*, 2018. 3

[51] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 33:596–608, 2020. 1, 2, 3, 4, 6, 7, 8

[52] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *NeurIPS*, 30, 2017. 1, 3

[53] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR*, pages 1521–1528. IEEE, 2011. 2

[54] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5018–5027, 2017. 2, 6, 1

[55] Vidit Vidit, Martin Engilberge, and Mathieu Salzmann. Clip the gap: A single domain generalization approach for object detection. In *CVPR*, pages 3219–3229, 2023. 3

[56] Riccardo Volpi and Vittorio Murino. Addressing model vulnerability to distributional shifts over image transformation sets. In *ICCV*, pages 7980–7989, 2019. 1, 3

[57] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *NeurIPS*, 31, 2018. 3

[58] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, et al. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*, 2022. 3

[59] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *NeurIPS*, 33:6256–6268, 2020. 3

[60] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, pages 10687–10698, 2020. 3

[61] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *CVPR*, pages 14383–14392, 2021. 2, 3, 6

[62] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. *arXiv preprint arXiv:2007.13003*, 2020. 3

[63] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *NeurIPS*, 34:18408–18419, 2021. 3

[64] Lei Zhang, Ji-Fu Li, and Wei Wang. Semi-supervised domain generalization with known and unknown classes. *NeurIPS*, 36, 2024. 2, 6

[65] Xin Zhang, Shixiang Shane Gu, Yutaka Matsuo, and Yusuke Iwasawa. Domain prompt learning for efficiently adapting clip to unseen domains. *Transactions of the Japanese Society for Artificial Intelligence*, 38(6):B–MC2_1, 2023. 3

[66] Zheyuan Zhang, Bin Wang, Debesh Jha, Ugur Demir, and Ulas Bagci. Domain generalization with correlated style uncertainty. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2000–2009, 2024. 3

[67] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, pages 13025–13032, 2020. 2, 6, 1

[68] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *ECCV*, pages 561–578. Springer, 2020. 3

[69] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, pages 13025–13032, 2020.

[70] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. *arXiv preprint arXiv:2104.02008*, 2021. 3

[71] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. *IEEE TIP*, 30:8008–8018, 2021. 2, 1

[72] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE TPAMI*, 45(4):4396–4415, 2022. 2

[73] Kaiyang Zhou, Chen Change Loy, and Ziwei Liu. Semi-supervised domain generalization with stochastic stylematch. *IJCV*, 131(9):2377–2387, 2023. 1, 2, 3, 4, 5, 6, 7, 8

SemAlign: Language Guided Semi-supervised Domain Generalization

Supplementary Material

6. Pseudocode

The detailed pseudocode of our algorithm is provided in Algorithm 1.

7. Datasets Description

PACS: PACS [27] comprises of 4 domains $d \in$ {photos, art, cartoons, sketches}, 7 classes $c \in$ {dog, elephant, giraffe, guitar, horse, house, person}, and contains 9, 991 examples in total. The image size used is (224 x 224).

OfficeHome: OfficeHome [54] comprises of 4 domains $d \in$ {art, clip art, product, real}, 65 object categories typically found in office and home environments, and 15, 588 examples. The image size used is (224 x 224).

DigitsDG: DigitsDG [67] comprises of 4 domains $d \in$ {MNIST, MNIST-M, SVHN, SYN} which are each individual datasets, 10 classes for each digits, and 6, 000 examples with 600 examples for each class. MNIST [25] contains hand-written digit images, MNIST-M [16] is a variant of MNIST that is produced by blending MNIST digits with random color patches. SVHN [36] contains street-view house number images. SYN [16] consists of synthetic digit images with varying fonts, backgrounds, and stroke colors. This dataset utilizes a smaller image size (32×32).

miniDomainNet: This is a subset of DomainNet [40], the largest dataset available for domain generalization experiments, with 6 domains, 345 classes, and 586, 575 examples. miniDomainNet [71] has 4 domains $d \in$ {clip art, painting, real, sketch}, 126 classes, and 140, 007 examples and utilizes a smaller image size (96×96).

It is worth noting that, we use ImageNet pretrained weights for the backbones used, following similar SSDG work [14, 15, 73]. Since these are pretrained on image size (224x224), better performance can be achieved in DigitsDG and miniDomainNet datasets by resizing images to this resolution.

8. Additional Pseudo-labeling staistics

Figure 7 shows the entropy distribution for each source dataset combination in the PACS dataset. The entropy is computed from the distribution of correctly retained pseudo-labels (PLs) within the training set, where higher entropy indicates greater dataset diversity. As shown, our model achieves the highest distributional diversity when the photo domain is included in the source dataset. We hypothesize that this effect is observed due to these models capitalizing on spurious correlations found in the photo domain.

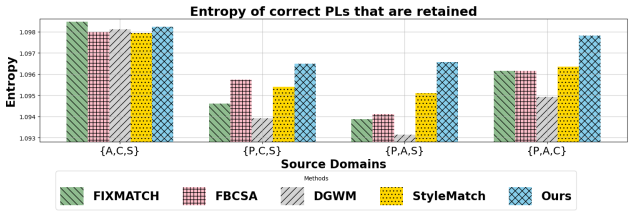


Figure 7. Distributional Diversity expressed as entropy for each of the baselines and our method.

Thus, without a dedicated objective to maximize learning from unlabeled data, increased focus on PL accuracy could lead to the model learning from a distribution consisting mostly of samples from "easier-to-learn" domains, consequently impairing model generalization.

9. Confusion Matrices

We plot the confusion matrices for the PACS (see Fig. 8) and DigitsDG (see Fig. 9) datasets, for the 10 labels per class setting. We compare our method with the existing SSDG methods. Our approach shows class-wise improvement on both datasets.

10. Additional Feature Visualizations

We visualize the features for the PACS dataset (see Fig. 10), and DigitsDG (see Fig. 11) datasets using t-SNE feature visualizations.

We also show Grad-CAM visualization (see Fig. 12) to show the features focused by different models. Our model is able to focus on invariant features across domains.

11. Scaling Performance with Number of labels per class

Top-1 accuracy for PACS dataset with an increasing number of per-class labels is shown in Tab. 6. We observe that the performance of our method improves as the number of per-class labels increases. Notably, across all per-class label settings, our method consistently outperforms the FixMatch baseline. Moreover, with only 10 labels per-class labels, our method surpasses the fully supervised ERM baseline (with all labels), which achieves 80.0±0.5.

12. Qualitative Analysis of Data Augmentations

Visualizations of our data augmentation method are shown in Fig. 13. As seen in these examples, phase-only image

reconstruction, only retains semantic information of the image, removing textural information completely. Amplitude swapping gives the image a random textural pattern. Texture reduction augmentation helps reduce noise and remove fine-grained, unwanted textures in an image. This can prevent the model from focusing on irrelevant details or artifacts.

13. Class Imbalance of VLCS Dataset

We noticed that the VLCS [53] dataset has a significant class imbalance than most DG datasets (see Tab. 4) . Under this class imbalance, all the baselines considered happened overfit for the 5 labels per class setting on the majority of the 5 independent trials we conducted, resulting in a very high variance in the results. Due to the high variance in the results, we don't report this dataset in our main results.

Table 4. VLCS Dataset Class-imbalance

Domain	# Samples in largest class	# Samples in smallest class
Caltech	809	62
LabelMe	1124	39
Pascal	1394	307
SUN	1175	19

Table 5. VLCS Dataset SSDG accuracy on target domain. Average over 5 independent trials is reported

Model	5 labels	10 labels
FixMatch [51]	55.6 ± 14.0	62.4 ± 12.2
StyleMatch [73]	60.5 ± 15.1	70.6 ± 3.1
FBCSA [14]	54.5 ± 14.7	66.2 ± 11.7
DGWM [15]	$\mathbf{61.0 \pm 15.3}$	69.0 ± 6.9
SemAlign (Ours)	59.1 ± 14.7	$\mathbf{72.4 \pm 1.8}$

Table 6. Results with different per class labels on PACS dataset. Note that the fully-supervised ERM model achieves 80.0 ± 0.5

.

Model	**5**	**10**	**25**	**50**
ERM [49]	51.2 ± 1.0	59.8 ± 2.5	66.7 ± 2.2	71.2 ± 1.9
FixMatch [51]	75.2 ± 1.3	77.1 ± 2.0	77.6 ± 1.4	78.7 ± 1.5
SemAlign (Ours)	$\mathbf{79.6 \pm 0.5}$	$\mathbf{81.8 \pm 1.7}$	$\mathbf{82.2 \pm 1.2}$	$\mathbf{83.3 \pm 1.2}$

Algorithm 1 Semantic Alignment

Require: Number of epochs E, class names $\{\mathrm{n}_1 \dots \mathrm{n}_C\}$, prompt templates $\{t^1(.) \dots t^M(.)\}$, weak augmentation a, strong augmentation $\mathcal{A}$, pseudo labeling threshold τ, EML(.), ANL(.) our model $\mathcal{F} = w \circ f$, RFR network f_{RFR}, CLIP text encoder $T_{\mathrm{clip}}(.)$

# Calculate domain-invariant prototypes
for class names $\mathrm{n}_c \in \{\mathrm{n}_1 \dots \mathrm{n}_C\}$ **do**
  for templates $t^m(.) \in \{t^1(.) \dots t^M(.)\}$ **do**
    # Get class templates
    $t_c^m \leftarrow t^m(\mathrm{n}_c)$
  end for
  # Get domain-invariant class prototypes
  $\mathbf{K}_c \leftarrow \frac{1}{M} \sum_{m=1}^{M} T_{\mathrm{clip}}(t_c^m)$
end for
for epochs $1, \dots, E$ **do**
  for minibatch indices (B_ℓ, B_u) **do**
    # Compute refined class prototypes
    $\mathbf{K}_c^* \leftarrow f_{\mathrm{RFR}}(\mathbf{K}_c)$
    # Compute the intermediate feature vectors
    $\mathrm{feat}_a \leftarrow f(a(\mathbf{x}_b))$
    $\mathrm{feat}_{\mathcal{A}} \leftarrow f(\mathcal{A}(\mathbf{x}_b))$
    # Compute model output
    $Q_b^w \leftarrow w(\mathrm{feat}_a)$
    # Calculate supervised loss
    if If the true label P_b is available **then**
      $\mathcal{L}_s \leftarrow H(P_b \mid Q_b^w)$
    end if
    # Generate pseudo-label
    $\hat{Q}_b^w \leftarrow \arg\max Q_b^w$
    if $max(Q_b^w) \geq \tau$ **then**
      # Calculate the unsupervised loss
      $\mathcal{L}_u \leftarrow H(\hat{Q}_b^w \mid w(\mathrm{feat}_{\mathcal{A}}))$
      # Calculate semantic similarity $\mathbf{z} \in \mathbb{R}^C$
      $\mathbf{z} \leftarrow [\mathbf{K}_1^* \mathrm{feat}_a; \dots; \mathbf{K}_C^* \mathrm{feat}_a]$
      $Q_b^{\mathrm{sem}} \leftarrow \mathrm{softmax}(\mathbf{z})$
      # Calculate the semantic alignment loss
      $\mathcal{L}_{\mathrm{SA}} \leftarrow H(Q_b^{\mathrm{sem}} \mid Q_b^w)$
      # Calculate the orthogonality loss
      $\mathcal{L}_{\mathrm{orth}} \leftarrow \frac{1}{C^2 - C} ||\mathbf{K}^* \mathbf{K}^{*\mathbf{T}} - \mathbf{I}_{\mathbf{C} \times \mathbf{C}}||_F^2$
      # Calculate the contrastive loss
      $\mathcal{L}_{\mathrm{con}} \leftarrow 1 - \mathbf{z}[y_{\mathrm{pred}}] + \frac{1}{C-1} \sum_{y_c \neq y_{\mathrm{pred}}}^{C} \mathbf{z}[y_c]$
      $\mathcal{L}_{\mathrm{feat}} \leftarrow \mathcal{L}_{\mathrm{SA}} + \mathcal{L}_{\mathrm{orth}} + \mathcal{L}_{\mathrm{con}}$
      # Calculate EML loss following [7]
      $\mathcal{L}_{\mathrm{EML}} \leftarrow EML(w(\mathrm{feat}_a))$
    end if
    # Calculate the ANL loss following [7]
    $\mathcal{L}_{\mathrm{ANL}} \leftarrow ANL(w(\mathrm{feat}_a))$
    $\mathcal{L}_{\mathrm{out}} \leftarrow \mathcal{L}_{\mathrm{EML}} + \mathcal{L}_{\mathrm{ANL}}$
    $\mathcal{L} \leftarrow \mathcal{L}_s + \mathcal{L}_u + \mathcal{L}_{\mathrm{feat}} + \mathcal{L}_{\mathrm{out}}$
    Update $\mathcal{F}$, f_{RFR}
  end for
end for
return Trained model $\mathcal{F}$

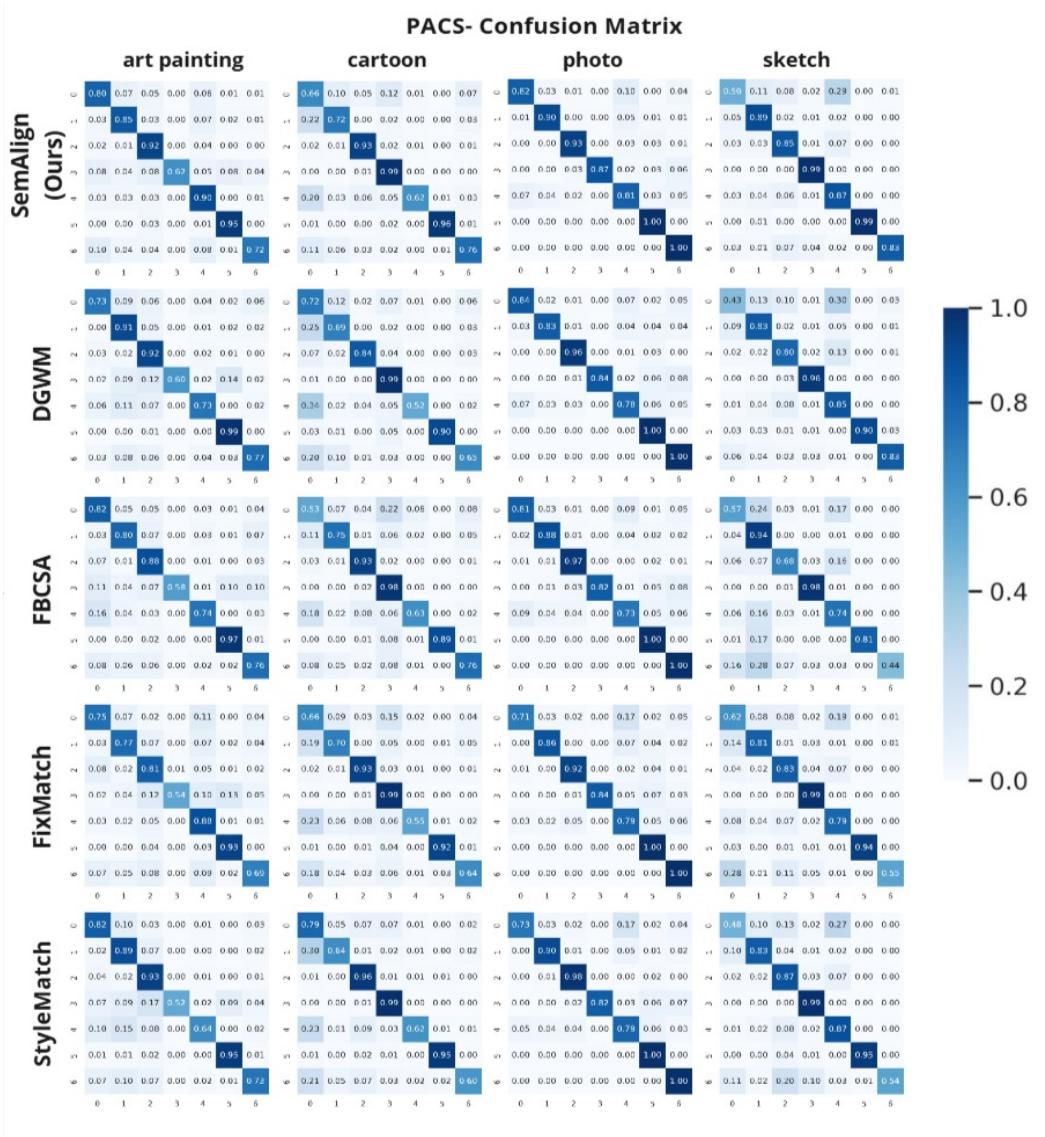


Figure 8. Confusion matrix for PACS dataset under 10-label setting.

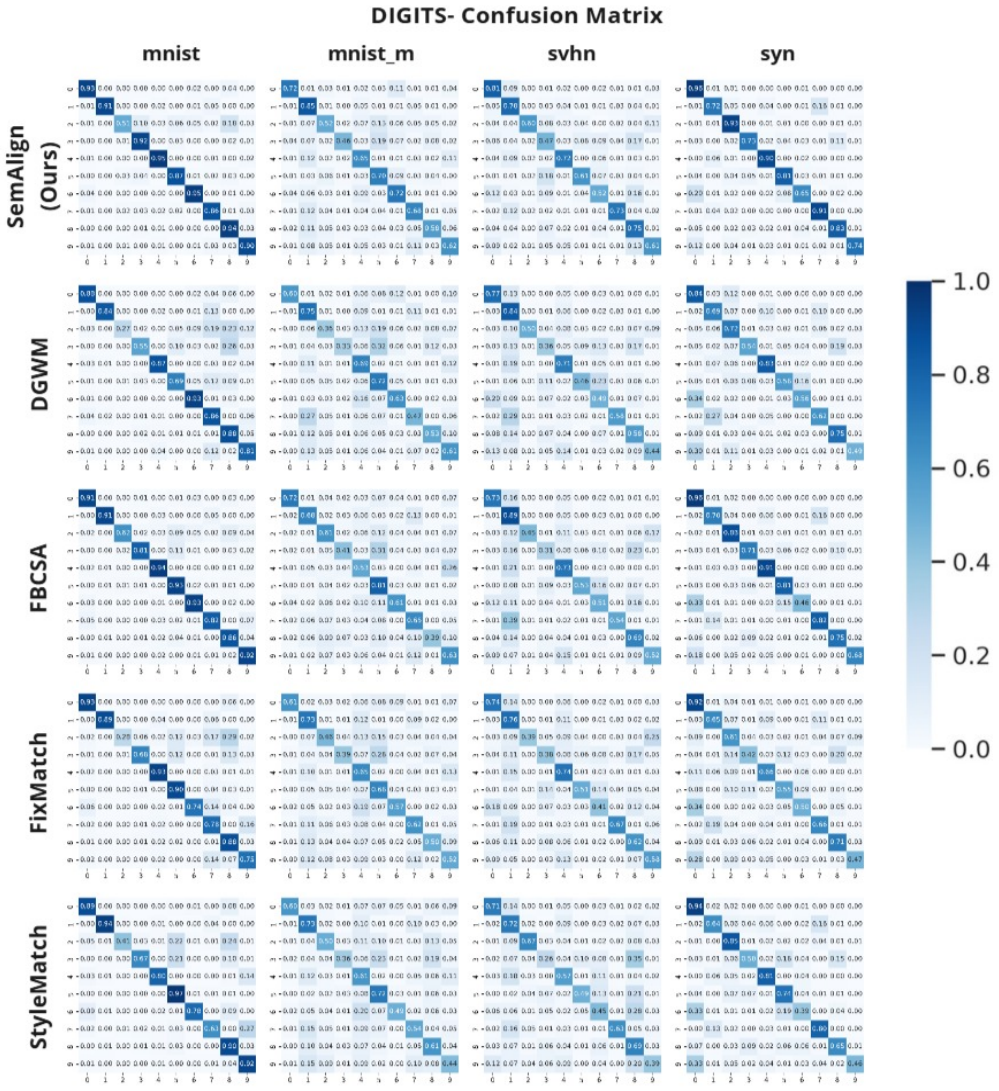


Figure 9. Confusion matrix for Digits-DG under 10-label setting.

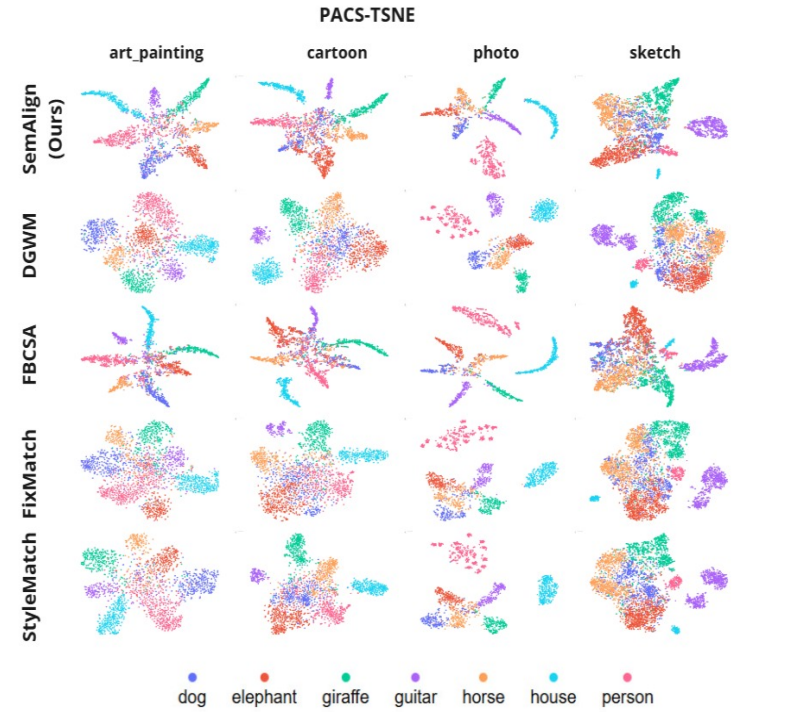


Figure 10. Feature visualization using tSNE for PACS under 10-label setting.

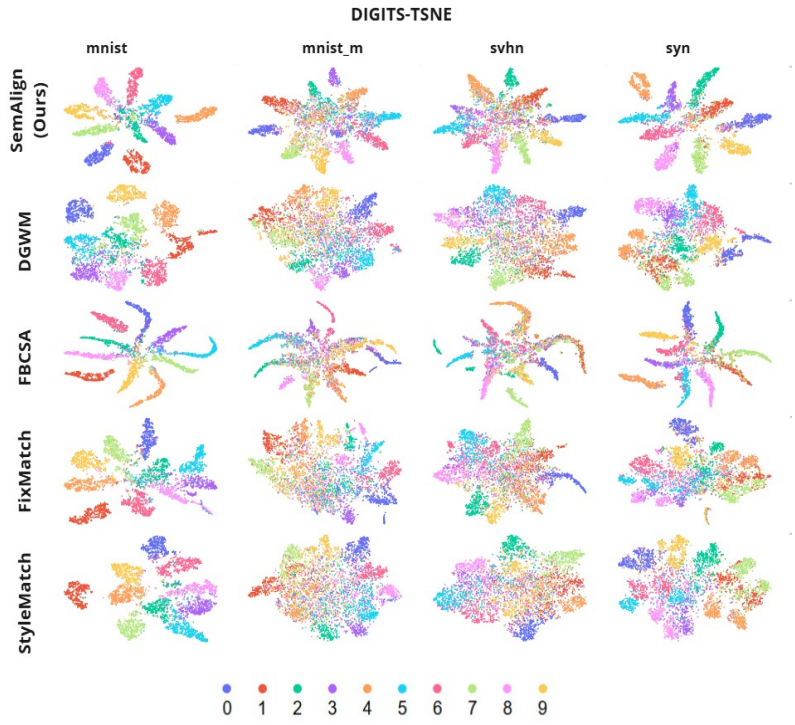


Figure 11. Feature visualization using tSNE for Digits-DG under 10-label setting.

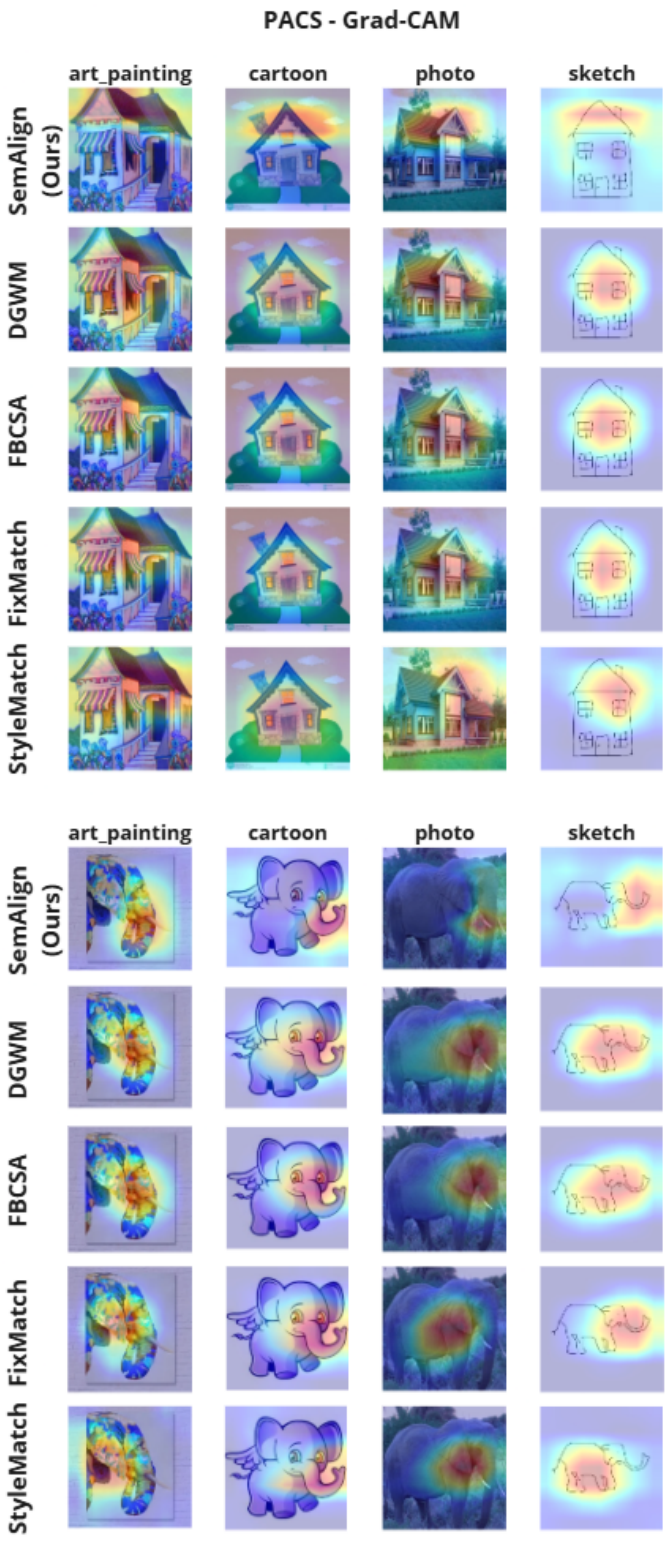


Figure 12. Grad-CAM visualization of model focus across the domains in PACS dataset

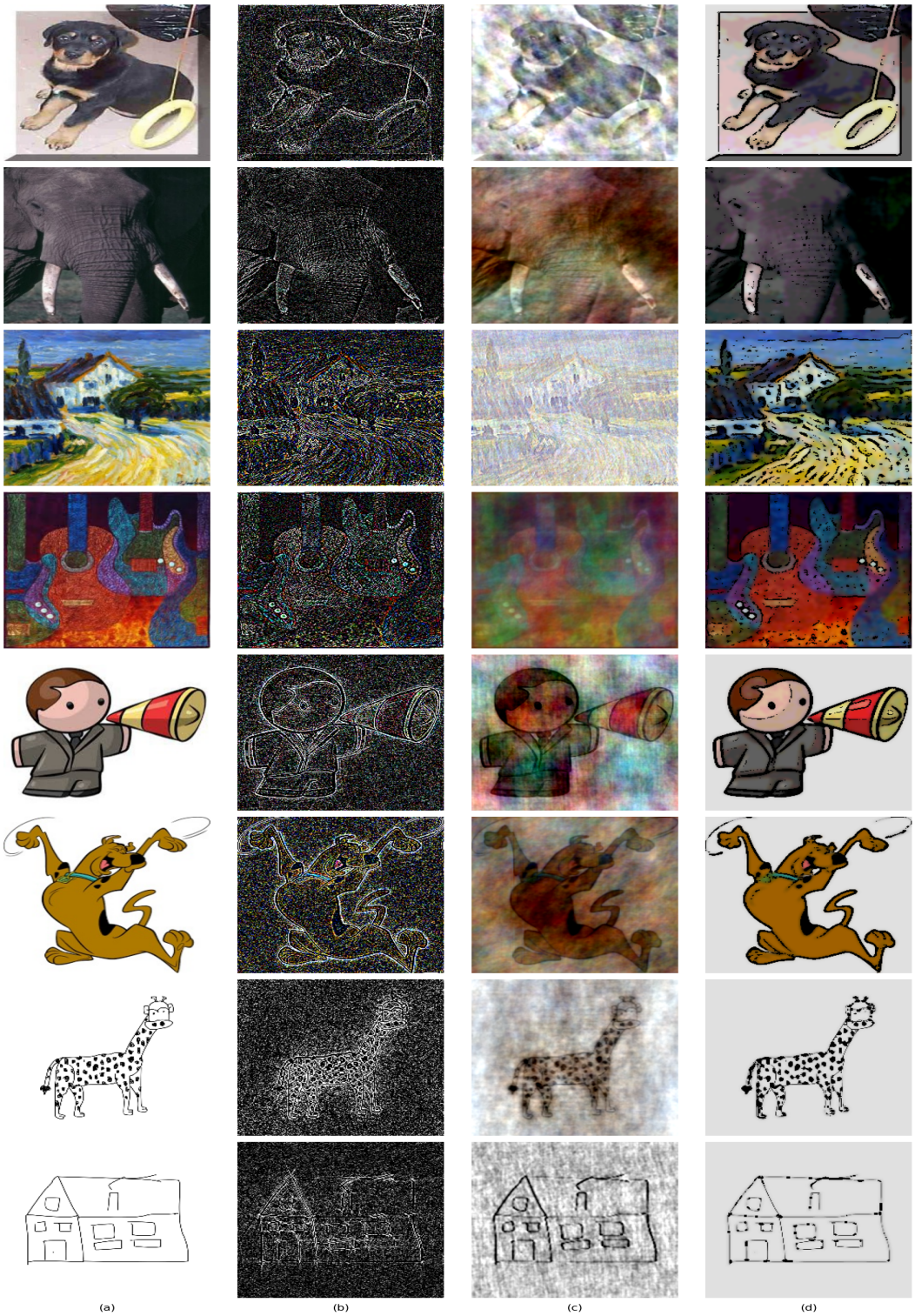


Figure 13. Data augmentations