Adapting Self-Supervised Representations to Multi-Domain Setups

Neha Kalibhat[1]
nehamk@umd.edu
Sam Sharpe[2]
samuel.sharpe@capitalone.com
Jeremy Goodsitt[2]
jeremy.goodsitt@capitalone.com
Bayan Bruss[2]
bayan.bruss@capitalone.com
Soheil Feizi[1]
sfeizi@cs.umd.edu

[1] University of Maryland
College Park, MD, USA
[2] CapitalOne Research

Abstract

Current state-of-the-art self-supervised approaches, are effective when trained on individual domains but show limited generalization on unseen domains. We observe that these models poorly generalize even when trained on a mixture of domains, making them unsuitable to be deployed under diverse real-world setups. We therefore propose a general-purpose, lightweight Domain Disentanglement Module (DDM) that can be plugged into any self-supervised encoder to effectively perform representation learning on multiple, diverse domains with or without shared classes. During pre-training according to a self-supervised loss, DDM enforces a disentanglement in the representation space by splitting it into a domain-variant and a domain-invariant portion. When domain labels are not available, DDM uses a robust clustering approach to discover pseudo-domains. We show that pre-training with DDM can show up to 3.5% improvement in linear probing accuracy on state-of-the-art self-supervised models including SimCLR, MoCo, BYOL, DINO, SimSiam and Barlow Twins on multi-domain benchmarks including PACS, DomainNet and WILDS. Models trained with DDM show significantly improved generalization (7.4%) to unseen domains compared to baselines. Therefore, DDM can efficiently adapt self-supervised encoders to provide high-quality, generalizable representations for diverse multi-domain data.

1 Introduction

Self-supervised learning [7, 9, 10, 11, 12, 18, 21, 27] has become a popular paradigm for unsupervised representation learning as it shows impressive results on downstream tasks. However, we find that current self-supervised models when trained on a single-domain show very poor generalizability to domain shifts. This can hinder their deployment in large scale real-world settings where data almost always comes from multiple diverse domains. We

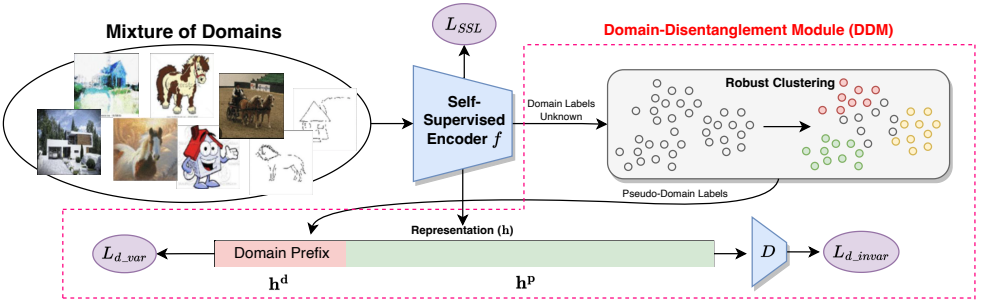


Figure 1: **Framework of our proposed Domain Disentanglement Module:** In our proposed DDM framework, the representation space (**h**) of any given self-supervised encoder is split into two portions, a domain prefix ($\mathbf{h}^d$) and a domain-invariant ($\mathbf{h}^p$) portion. Along with the self-supervised loss (L_{ssl}), $\mathbf{h}^d$ is trained to be distinguishable across domains (L_{d_var}) and $\mathbf{h}^p$ is trained to be invariant to any domain information (L_{d_invar}). DDM also supports scenarios when domain labels are not available using *robust clustering*, an iterative process that reduces outlier noise.

illustrate this issue in Figure 2, where we show that popular self-supervised models, SimCLR [11], MoCo [21] and BYOL [18], trained on individual domains of PACS [36] significantly under-perform on unseen domains. This means that a different self-supervised model needs to be trained for every new domain, which can add significant computational overheads given that training these models often require large batch sizes and a large number of training epochs [11, 21, 45].

One potential solution for self-supervised learning on multi-domain datasets is to train the models on the *union* of all input domains. We illustrate this in Figure 2 were we plot the multi-domain training results for each baseline on a mixture of PACS Photo, Sketch and Cartoon. We observe that this solution may show improved performance on the training domains, however they do not match the single-domain baselines in all cases. Moreover, they show poor generalization to unseen domains (PACS Painting). In Section 3, we study the representation space closely under multi-domain regimes to find that they can under perform compared to single-domain regimes because domain-related and content-related information overlap in the representation space, affecting their quality for instance classification.

To tackle these issues, we propose a **Domain-Disentanglement Module (DDM)**, that can be plugged in to any self-supervised model during multi-domain training. With DDM, we enforce a disentanglement in the representation space where a domain prefix is trained to be distinguishable across domains and the remaining portion is trained to be *domain-invariant* to produce better structured representations. This is achieved by minimizing the Wasserstein Distance [1] between the known and predicted domain label distributions. We also extend DDM to more realistic, entirely unsupervised multi-domain setups where domain labels are unknown. In such scenarios, we present a *robust clustering* approach that iteratively reduces outlier noise and detects pseudo-domain-labels that are used in DDM.

By pre-training with DDM, we show that we can improve the generalization capability of various state-of-the-art self-supervised baselines including SimCLR [11], MoCo [21], BYOL [18], DINO [10], SimSiam [12] and Barlow Twins [48]. We perform extensive experiments on generalization benchmarks including PACS [36], DomainNet [37] and WILDS [30]. Upon linear probing on unseen domains, we observe an improvement of 6.1% on

PACS, 7.4% on DomainNet and 5.9% on WILDS using DDM. In summary, we propose a lightweight module called DDM which can be simply attached to any self-supervised encoder to enable training over multiple diverse domains to produce well-structured, generalizable representations (See Figure 1).

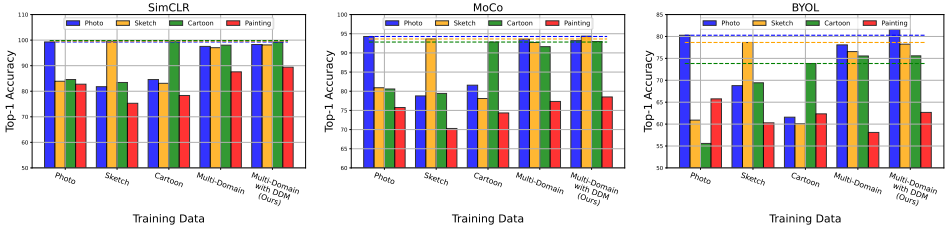


Figure 2: **Self-supervised baselines under single and multi-domain setups:** We plot 3 SOTA self-supervised baselines, SimCLR, BYOL and MoCo, trained individually on PACS Photo, Sketch and Cartoon and on their mixture. We observe that both single-domain and mult-domain training generalizes poorly to unseen domains on all baselines. These baselines when pre-trained with DDM (our method), outperforms even single-domain baselines and shows significantly improved generalization to the unseen domains.

2 Related Work

Building on the success of unsupervised learning techniques [4, 5, 6, 8, 15, 23, 47], self-supervised models have shown unprecedented capabilities when used in a range of downstream tasks. Among a number of self-supervised baselines, we focus on SimCLR [11], MoCo [21], BYOL [18], DINO [10], SimSiam [12] and Barlow Twins [48]. These are joint-embedding self-supervised learning methods, which involve taking two augmented views of the same input and ensuring their representations are close using the same encoder or two encoders sharing the same weights.

Extending these self-supervised methods to multiple diverse domains, other than ImageNet [38], is a relatively less explored topic [44]. Existing approaches [28, 29, 35] use pre-trained encoders and assume few source labels for unsupervised domain adaption and domain generalization. [41] uses available class information and novelty discovery to learn new samples in the wild. These works do not consider fully unsupervised multi-domain setups, where even domain label information is unavailable. [17] assumes domain labels and uses mutual information to encode common invariant information and domain-specific information for each image. [46] uses multiple domain-specific decoders to reconstruct images according to their domains such that the encoder is domain-invariant. This method may not be scalable and is contingent upon the number of available domains. [49] proposes a contrastive method that selects negatives across domains to train invariant representations. Our method reports better numbers on the PACS dataset compared to these baselines. Our method also does not assume domain labels and can be flexibly applied on any self-supervised setup.

In our paper, we focus on a general multi-domain setup with diverse related or unrelated domains, with and without shared classes, and evaluate on individual domain-specific tasks. We make it possible to efficiently pre-train a single encoder on any existing state-of-the-art self-supervised setup, over multiple domains, to significantly improve their generalizability.

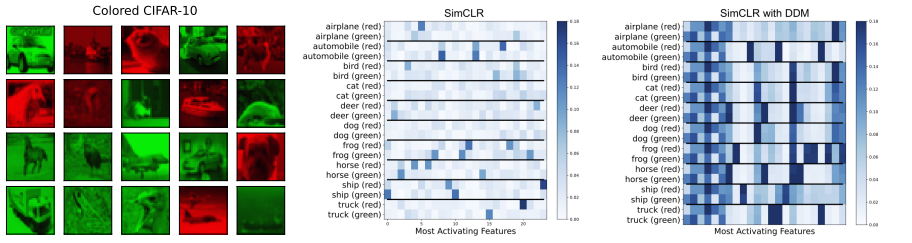


Figure 3: **Visualizing Colored-CIFAR representations:** We prepare Colored-CIFAR, multi-domain version of CIFAR-10 [37] where the images are randomly colored red or green. We visualize the top activating features of the class-averaged representations of both domains. In the SimCLR baseline, we observe a clear difference in feature distribution within the same classes, across domains. DDM enables representations to have a shared domain-specific prefix while the remaining portion is domain-invariant and almost identical across classes. This structure significantly improves linear evaluation performance (See Figure 4).

3 Self-Supervised Models under Multi-domain Setups

We observed in Figure 2, that state-of-the-art self-supervised learning methods like SimCLR [11], MoCo [13] and BYOL [18] show low transfer performance on unseen domains on both single-domain and multi-domain regimes. In this section, we take a closer look at the learned representation space under these regimes to explain this behavior.

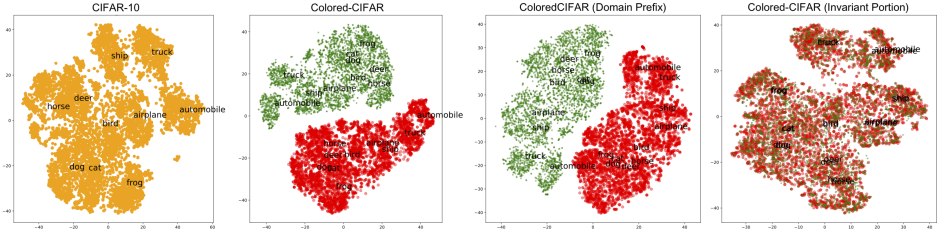
We first define some notations. Let us consider a self-supervised model with a base encoder $f(.)$. We apply data transformations and pass the input samples, $\mathbf{x}_i \in \mathbb{R}^n$, through the base encoder to get self-supervised representations denoted by $f(\mathbf{x}_i) = \mathbf{h}_i \in \mathbb{R}^r$ where r is the size of the representation space.

Let us take the example of SimCLR [11] trained on CIFAR-10 [37] dataset. In the first t-SNE [43] plot in Figure 4(a), we observe that the representations are naturally clustered based on their classes, which allows us to achieve a top-1 accuracy of 90.18 after linear probing. Let us now define a multi-domain version of CIFAR-10 called *Colored-CIFAR* where, each sample is randomly colored either red or green as shown in the first panel of Figure 3. In this dataset, the domains refer to the colors of the image, while the labels are of the objects. When SimCLR is trained on Colored-CIFAR, there is a significant drop in top-1 accuracy (78.52). We observe that the representation space is divided into two large clusters, corresponding to the domains (red or green) as shown in 4(b). We attribute the loss in accuracy to this significant change in representation structure.

We now study the SimCLR representation space of Colored-CIFAR to further understand and explain multi-domain behavior. In Figure 3, in the second panel, we show a heatmap of the domain-wise averaged representations of each class in CIFAR-10. Each column corresponds to specific feature indices of the class-averaged representations. The darker the column, the higher the magnitude of the feature. For fair comparison, we L2 normalize every feature. For ease of visualization, we display only the subset of feature indices (called *most activating features*) that are strongly deviated from the mean in at least one row. The remaining features show low activation across the board and are omitted from visualization [24, 25]. Top activating features correspond to important physical attributes discovered from the training data [25, 40]. Two images of a car, one in each domain, would share all physical

attributes except for the color. An ideal self-supervised encoder is expected to encode all physical attributes independent of any domain shift.

However, in multi-domain SimCLR, we observe that there is almost no overlap between the most activating features of each class between the red and green domains. This suggests that the domain information (color) and instance information (actual content of the image) are somewhat interleaved in these representations, causing different sets of features to be strongly activated for the same class based on the domain. In single-domain SimCLR on CIFAR-10 (no colors), the representations only encode content information, which results in linearly separable representations by class. In multi-domain SimCLR on Colored-CIFAR, a combination of both domain and content information is encoded in every representation which directly affects linear classification performance. Therefore, to achieve comparable performance to single-domain setups, we propose to **disentangle** domain information from representations by plugging in a general-purpose a Domain-Disentanglement Module (DDM) for Self-Supervised Models which is discussed in the next section.



(a) Accuracy: 90.18 (b) Accuracy: 78.52 (c) Accuracy: 87.06 (DDM) (d) Accuracy: 87.06 (DDM)

Figure 4: **SimCLR Representation t-SNE before and after DDM:** CIFAR-10 representations are naturally clustered by class, however, Colored-CIFAR representations are clustered by domain which leads to a significant reduction in classification performance. When SimCLR is trained with DDM, the prefix alone has domain-distinguishable representations, while the remaining portion of the representation is domain-invariant, clustered by class. This structure notably improves the classification performance.

4 Domain-Disentanglement Module for Self-Supervised Representations

As described in the previous section, self-supervised models in their current state, are not trained to learn content and domain information independently. We hypothesize that disentangling domain information from the learned representations can improve the performance of existing state-of-the-art SSL models in multi-domain setups. We therefore propose a general-purpose Domain-Disentanglement Module (DDM) that can be simply attached at any SSL encoder during its pre-training. In this work focus on joint-embedding (involving two transformed views) self-supervised encoders [3, 9, 10, 11, 12, 13, 18, 48] and not masked image models [20].

Recall that for a given sample $\mathbf{x}_i$, its representation is denoted by $f(\mathbf{x}_i) = \mathbf{h}_i \in \mathbb{R}^r$. Let y_i denote the domain of the i^{th} representation. We allocate the first k features of the representation as the domain prefix, $\mathbf{h}_{i,0..k}$, denoted by $\mathbf{h}_i^d$ for ease of notation. The remaining portion of the representation $\mathbf{h}_{i,k..r}$ is denoted by $\mathbf{h}_i^p$. We call $\mathbf{h}_i^d$ as the *domain-variant* portion and

$\mathbf{h}_i^p$ as the *domain-invariant* portion. We train the domain prefix of the i^{th} sample according to the following contrastive optimization, $L_{i_{d\_var}} = \log \frac{\sum_{j=1}^{2N} \mathbb{1}_{j \neq i} \mathbb{1}_{y_i = y_j} sim(\mathbf{h}_i^d, \mathbf{h}_j^d)}{\sum_{j=1}^{2N} \mathbb{1}_{y_i \neq y_j} sim(\mathbf{h}_i^d, \mathbf{h}_j^d)}$.

where $sim(a,b) = \exp\left(\frac{1}{\tau} \frac{a^T b}{\|a\|\|b\|}\right)$. This loss maximizes the similarity of the domain prefixes within each domain and minimizes the similarity of domain prefixes across domains. $\mathbf{h}_i^p$ is learned according to any self-supervised loss like SimCLR, MoCo, DINO etc., denoted by $L_{i_{ssl}}$. Splitting the representation in this manner helps us control each portion independently. L_{ssl} ensures that all content information is encoded in a self-supervised manner such that representations can be utilized for downstream tasks. L_{d_var} ensures that the domain prefixes across samples of different domains are distinguishable.

We next ensure that $\mathbf{h}_i^p$ does not contain any domain-related information (domain-invariance constraint). In other words, it should not be possible to predict the the domain label y_i from the representation $\mathbf{h}_i^p$. To achieve this, we pass each $\mathbf{h}_i^p$ through a domain discriminator $D(.)$ and minimize the Wasserstein distance (using the dual form as proposed in [1]), $L_{i_{d\_invar}} = D(\mathbf{h}_i^p, y_i) - D(\mathbf{h}_i^p, y_{rand})$, where $y_{rand} \sim \mathbb{P}(y)$, i.e., randomly drawn from the distribution of domain labels. The final optimization for the encoder ($f(.)$) and the discriminator ($D(.)$) is,

$$\max_f \sum_{i=1}^{2N} \left[L_{i_{ssl}} + \lambda_1 L_{i_{d\_var}} + \lambda_2 L_{i_{d\_invar}} \right] \tag{1}$$

where λ_1, λ_2 are tunable hyperparameters. We optimize both the encoder $f(.)$ and the discriminator D using alternating gradient descent ascent. We train $D(.)$ using gradient penalty to improve its stability as proposed in [19]. This formulation is similar to [26, 34], except that we use Wasserstein Distance to disentangle domain information from the remaining portion of the representation space. In summary, our module DDM consists of splitting the representation space into two parts and applying two additional loss terms, L_{d_var} and L_{d_invar}. Note that, DDM can be plugged in while training any existing state-of-the-art self-supervised model.

In Figure 3, in the last panel, we show the representation space of SimCLR trained on Colored-CIFAR using DDM. We observe that among the most activating features, the first few features (which are part of the domain prefix) are equivalent for all classes within a domain and clearly distinguishable between both domains. The remaining portion of the representation is completely invariant to any domain information as each class shows very similar feature distribution in both red and green domains. In the t-SNE plots (Figure 4(c) and (d)), we observe that the domain prefix is separable by domain whereas the domain-invariant portion shows natural class clusters with overlapping red and green images. This update in structure leads to a significant improvement in top-1 accuracy from 78.52 to 87.06.

4.1 Experimental Setup

We use ViT-S [16] as the base encoder ($f(.)$) for all of our experiments. Our domain discriminator ($D(.)$) is an MLP with LeakyReLU activations. The representations are 384-dimensional with a 24-dimensional domain prefix. We train the encoder according to various self-supervised baselines including SimCLR [11], MoCo [21], BYOL [18], DINO [10], SimSiam [12] and Barlow Twins [48]. We use the same optimization and scheduling for the encoder as the respective papers. While training with DDM, we use the Adam optimizer

for the domain discriminator with a learning rate of 0.005 and cosine-annealing scheduling and $\lambda_1 = \lambda_2 = 0.5$. We experiment with PACS [36], DomainNet [37] and the WILDS [30] multi-domain benchmarks. We use Nvidia GeForce RTX A4000 GPUs for pre-training. We evaluate representations using the linear evaluation protocol [2, 31, 42] where we train a linear classifier on top of frozen representations and compute the top-1 accuracy over the training and unseen domains.

4.2 Self-Supervised Baselines Trained with DDM

In Figure 2, we observed that self-supervised baselines (SimCLR, BYOL and MoCo), when trained on a single domain or multiple domains, generalize poorly to unseen domains. These baselines, when pre-trained with DDM, show improved performance on the training domains (PACS Photo, Sketch and Cartoon) as well as significantly improved generalization to the unseen domain (PACS Art Painting). Pre-training on multiple domains with DDM outperforms every self-supervised baseline as shown in Table 1 with a maximum of 2.6% improvement on average top-1 accuracy on SimSiam. We also tabulate our results on DomainNet using Painting, Real and Sketch as training domains and Clipart, Infograph and Quickdraw as the unseen domains in Table 2. We observe that pre-training with DDM improves upon each self-supervised baseline with a maximum of 3.5% improvement on average top-1 accuracy on BYOL. DDM generalizes significantly better than its baselines showing a 6.1% (SimSiam) increase in PACS (Painting) and a 7.4% (DINO) in DomainNet (Clipart).

To further evaluate the generalization of self-supervised baselines with DDM, we utilize the WILDS benchmark [30]. In this benchmark, pre-train on iWildCam (200K samples, 182 classes, 323 domains), Camelyon17 (456K samples, 2 classes, 5 domains), FMoW (141K samples, 62 classes, 80 domains) and RxRx1 (125K samples, 1139 classes, 51 domains). We summarize our results in Table 3. On each benchmark, we observe that DDM outperforms the baselines on the unseen validation set. The accuracy in rxrx1 is low since it is a very hard classification task as it contains 1139 classes and 51 domains. We observe a 5.9% increase linear classification accuracy on iWildCam on SimCLR

Table 1: SSL baselines trained on PACS (Photo, Sketch and Cartoon) with DDM

Model	Top-1 Accuracy (Baseline / with DDM)				
	Photo	Sketch	Cartoon	Painting (Unseen)	Average
SimCLR	97.54 / **98.28**	**98.12** / 97.04	98.03 / **99.24**	87.59 / **89.42**	95.32 / **96.00**
MoCo	**93.59** / 93.19	92.71 / **94.36**	91.63 / **92.98**	77.34 / **78.51**	88.81 / **89.76**
BYOL	78.08 / **81.61**	76.55 / **78.24**	75.55 / **75.58**	58.10 / **62.67**	72.07 / **74.53**
DINO	93.67 / **95.25**	94.33 / **96.42**	79.44 / **81.77**	72.12 / **74.43**	85.89 / **86.97**
SimSiam	83.68 / **84.71**	80.97 / **85.44**	**93.75** / 92.59	57.98 / **64.09**	79.09 / **81.71**
Barlow Twins	**85.09** / 83.94	85.44 / **88.07**	92.0 / **92.83**	59.01 / **62.67**	80.39 / **81.89**

Table 2: SSL baselines trained on DomainNet (Painting, Real and Sketch) with DDM

Model	Top-1 Accuracy (Baseline / with DDM)						
	Painting	Real	Sketch	Clipart (Unseen)	Infograph (Unseen)	Quickdraw (Unseen)	Average
SimCLR	74.49 / **75.99**	79.31 / **82.02**	85.86 / **86.26**	68.60 / **70.48**	34.75 / **39.25**	22.98 / **24.38**	60.99 / **63.06**
MoCo	70.20 / **73.08**	**89.79** / 86.37	86.66 / **88.15**	65.10 / **68.91**	34.56 / **34.75**	19.89 / **22.12**	61.03 / **62.23**
BYOL	56.87 / **59.82**	77.60 / **79.67**	71.43 / **75.21**	50.67 / **55.86**	27.4 / **30.68**	19.33 / **22.85**	50.55 / **54.02**
DINO	**79.53** / 79.11	86.46 / **86.88**	75.8 / **76.50**	66.32 / **73.76**	30.83 / **32.12**	27.71 / **29.08**	61.11 / **62.90**
SimSiam	77.55 / **78.78**	82.02 / **85.88**	86.52 / **88.38**	67.43 / **71.53**	27.03 / **30.56**	22.29 / **25.67**	60.47 / **63.47**
Barlow Twins	56.78 / **61.18**	79.06 / **80.16**	71.56 / **73.90**	60.40 / **64.33**	26.11 / **28.82**	18.67 / **21.70**	52.09 / **55.01**

Table 3: SSL baselines trained on WILDS with DDM

Model	Top-1 Accuracy (Baseline / with DDM) iWildCam	Camelyon17	FMoW	RxRx1
SimCLR	66.01 / **71.87**	95.19 / **95.68**	38.94 / **41.23**	8.43 / **11.20**
MoCo	67.05 / **69.12**	91.45 / **93.47**	40.04 / **40.23**	5.67 / **5.93**
BYOL	71.69 / **74.88**	95.15 / **96.38**	38.74 / **39.78**	4.39 / **6.20**
DINO	64.55 / **68.07**	94.38 / **95.38**	33.57 / **34.52**	7.32 / **7.66**
SimSiam	60.45 / **61.16**	88.37 / **89.16**	39.27 / **40.05**	6.39 / **7.26**
Barlow Twins	63.17 / **63.84**	96.38 / **97.62**	44.40 / **47.46**	5.79 / **6.65**

5 DDM without Domain Labels

Most real-world multi-domain datasets are unlabelled (i.e., domain label information is not available). In this section, we develop an extension of DDM for such setups by identifying pseudo domain labels via a clustering approach in the representation space. As it is common in clustering, we assume the number of domains (denoted by M) is known. Depending on the multi-domain setup, we can also approximate the number of domains by studying any available meta-data like data sources, geo-location, quality, etc. We can also estimate the number of domains empirically through clustering and visualization.

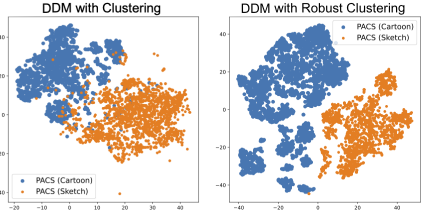


Figure 5: **DDM with clustering:** When domain labels are not available, we perform DDM with clustering to identify pseudo-domain-labels. In the above plots we show the t-SNE of the SimCLR representations trained on "Cartoon" and "Sketch" domains in PACS. We observe that DDM with robust clustering produces a better separation between domains.

Domain labels are required in both DDM losses (L_{d_var}, L_{d_invar}) as described in the previous section. Let us consider a fully unlabelled setup, with no domain labels while the number of domains M is known. We first warm up our self-supervised encoder $f(.)$ treating it as a single-domain setup for a few iterations to get somewhat distinguishable representations by domain. We next cluster the representations into M clusters using K-Means clustering [20]. Using the cluster assignments as pseudo-domain-labels (y), we continue training the encoder $f(.)$ along with a discriminator using the DDM optimization, to learn domain-disentangled representations.

In practice, clustering does not discover 100% accurate domain labels, especially for datasets that are distributionally similar. We therefore use a **robust clustering** approach coupled with DDM to prevent outlier clustering noise from affecting the pseudo-domain-labels. Suppose we discover M clusters with centroids $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_M$, before assigning pseudo-domain-labels to each sample, we first determine if they are outliers or not. If so, we ignore these samples in the next stages of training to prevent assigning a noisy label to them. We say a representation is *not* an outlier if it is significantly closer to one of the clustering centroids compared to another. Concretely, $\mathbf{h}_i$ is not an outlier if

$$\max\left\{\frac{\|\mathbf{h}_i-\mathbf{c}_m\|^2}{\|\mathbf{h}_i-\mathbf{c}_n\|^2} : 1\leq m\leq M, 1\leq n\leq M\right\} > 1+\varepsilon \tag{2}$$

where $\varepsilon \geq 0$ is defined as the *outlier threshold*. When ε is high, it means that the given sample is very close to its respective centroid. When ε approaches 0, it indicates that the sample is almost equidistant from at least two centroids and therefore, may not be reliably assigned one pseudo-label. We ignore such samples going forward in training. When we perform clustering for the first time, we start with $\varepsilon = 1$. We repeat the clustering at regular intervals of training on the representations $\mathbf{h}$ to get improved cluster centroids. Each time we repeat clustering, we decay the value of ε exponentially such that it approaches 0. By the end of training, all samples will contribute to the training of the self-supervised encoder with DDM. In Figure 5, we illustrate the difference between regular clustering and robust clustering with MDSSL trained on the PACS dataset [36] ("Cartoon" and "Sketch" domains). We observe that robust clustering helps in identifying more accurate and distinguishable clusters.

To evaluate DDM with robust clustering, we combine CIFAR-10 [32], CIFAR-100 [33] and STL-10 [14] to form a multi-domain dataset. The constituent datasets are distributionally similar with several shared classes (CIFAR-10 and STL-10 share 9 out of 10 classes). With this setup, we try to simulate a real-world scenario where data arises from various domains however the actual domains are undefined. We therefore apply DDM with robust-clustering to identify pseudo-domain-labels. We then evaluate the pre-trained representations by linear probing the validation portion of each constituent dataset. We include Tiny-ImageNet [34] as an unseen domain to test generalization.

In Table 4, we tabulate the results on this prepared multi-domain dataset on various self-supervised baselines with and without DDM and robust clustering. We observe an improvement in the average top-1 accuracy across all baselines with 1.7% improvement in MoCo. DDM shows improved generalization on Tiny-ImageNet with a 2.9% increase in DINO.

Table 4: SSL baselines trained on a mixture of CIFAR-10, STL-10 and CIFAR-100 using DDM and robust clustering

Model	Top-1 Accuracy (Baseline / with DDM and robust clustering)				
	CIFAR-10	STL-10	CIFAR-100	Tiny-ImageNet (Unseen)	Average
SimCLR	89.43 / **90.03**	79.77 / **81.01**	63.33 / **64.90**	49.58 / **51.22**	70.53 / **71.79**
MoCo	**90.80** / 90.69	80.02 / **81.60**	61.57 / **64.28**	37.16 / **39.55**	67.38 / **69.03**
BYOL	88.31 / **89.68**	75.07 / **75.72**	64.82 / **65.56**	50.04 / **51.10**	69.56 / **70.52**
DINO	90.61 / **92.96**	**84.7** / 82.35	62.63 / **63.57**	49.52 / **52.46**	71.87 / **72.84**
SimSiam	87.02 / **87.38**	72.15 / **73.78**	**62.08** / 61.90	33.11 / **34.78**	63.59 / **64.46**
Barlow Twins	88.31 / **89.01**	75.59 / **76.11**	65.03 / **66.89**	40.27 / **41.31**	67.30 / **68.33**

6 Conclusion

We proposed a Domain Disentanglement Module (DDM) for self-supervised encoders that provide better structured representations, domain-invariant representations that can be used for diverse multi-domain tasks. DDM also supports training over setups where domain labels are not available by using a robust clustering technique that reduces outlier noise. With DDM, we prevent the need for having to train multiple single-domain encoders and instead leverage a single encoder to perform comparably on multiple domains. The benefit of invariant representations is better generalization which we show on various benchmarks including PACS, DomainNet and WILDS.

7 Acknowledgement

This project was supported in part by a grant from Capital One, an NSF CAREER AWARD 1942230, ONR YIP award N00014-22-1-2271, ARO's Early Career Program Award 310902-00001, Meta grant 23010098, HR001119S0026 (GARD), Army Grant No. W911NF2120076, NIST 60NANB20D134, the NSF award CCF2212458 and an Amazon Research Award.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

[2] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/ddf354219aac374f1d40b7e760ee5bb7-Paper.pdf.

[3] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning, 2021.

[4] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Tikhoncheva, and Bjorn Ommer. Cliquecnn: Deep unsupervised exemplar learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/65fc52ed8f88c81323a418ca94cec2ed-Paper.pdf.

[5] Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 517–526. PMLR, 06–11 Aug 2017. URL http://proceedings.mlr.press/v70/bojanowski17a.html.

[6] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[7] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. *Lecture Notes in Computer Science*, page 139–156, 2018. ISSN 1611-3349. doi: 10.1007/978-3-030-01264-9_9. URL http://dx.doi.org/10.1007/978-3-030-01264-9_9.

[8] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[9] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information*

Processing Systems, volume 33, pages 9912–9924. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf.

[10] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9650–9660, October 2021.

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/chen20j.html.

[12] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15750–15758, June 2021.

[13] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning, 2020.

[14] Adam Coates, Honglak Lee, and Andrew Y. Ng. Stanford stl-10 image dataset. URL https://cs.stanford.edu/~acoates/stl10/.

[15] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/07563a3fe3bbe7e3ba84431ad9d055af-Paper.pdf.

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[17] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning from multi-domain data. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3244–3254, 2019.

[18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21271–21284. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/f3ada80d5c4ee70142b17b8192b2958e-Paper.pdf.

[19] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans, 2017.

[20] J. A. Hartigan and M. A. Wong. A k-means clustering algorithm. *JSTOR: Applied Statistics*, 28(1):100–108, 1979.

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, 2020. doi: 10.1109/CVPR42600.2020.00975.

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollar, and Ross Girshick. Masked autoencoders are scalable vision learners. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022. doi: 10.1109/cvpr52688.2022.01553. URL http://dx.doi.org/10.1109/CVPR52688.2022.01553.

[23] Jiabo Huang, Qi Dong, Shaogang Gong, and Xiatian Zhu. Unsupervised deep learning by neighbourhood discovery. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2849–2858. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/huang19b.html.

[24] Li Jing, Pascal Vincent, Yann LeCun, and Yuandong Tian. Understanding dimensional collapse in contrastive self-supervised learning. *ArXiv*, abs/2110.09348, 2021.

[25] Neha Kalibhat, Kanika Narang, Hamed Firooz, Maziar Sanjabi, and Soheil Feizi. Measuring self-supervised representation quality for downstream classification using discriminative features, 2022.

[26] Priyatham Kattakinda, Alexander Levine, and Soheil Feizi. Invariant learning via diffusion dreamed distribution shifts, 2022.

[27] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020.

[28] Donghyun Kim, Kuniaki Saito, Tae-Hyun Oh, Bryan A. Plummer, Stan Sclaroff, and Kate Saenko. Cross-domain self-supervised learning for domain adaptation with few source labels, 2020.

[29] Donghyun Kim, Kuniaki Saito, Tae-Hyun Oh, Bryan A. Plummer, Stan Sclaroff, and Kate Saenko. Cds: Cross-domain self-supervised pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9123–9132, October 2021.

[30] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts, 2021.

[31] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1920–1929, 2019. doi: 10.1109/CVPR.2019.00202.

[32] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). . URL http://www.cs.toronto.edu/~kriz/cifar.html.

[33] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-100 (canadian institute for advanced research). . URL http://www.cs.toronto.edu/~kriz/cifar.html.

[34] Ya Le and X. Yang. Tiny imagenet visual recognition challenge. 2015.

[35] Bo Li, Yifei Shen, Yezhen Wang, Wenzhen Zhu, Colorado J. Reed, Jun Zhang, Dongsheng Li, Kurt Keutzer, and Han Zhao. Invariant information bottleneck for domain generalization, 2021.

[36] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. doi: 10.1109/iccv.2017.591. URL http://dx.doi.org/10.1109/ICCV.2017.591.

[37] 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 International Conference on Computer Vision*, pages 1406–1415, 2019.

[38] 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.

[39] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation, 2017.

[40] Sahil Singla and Soheil Feizi. Salient imagenet: How to discover spurious features in deep learning?, 2021.

[41] Yiyou Sun and Yixuan Li. Opencon: Open-world contrastive learning, 2022.

[42] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2019.

[43] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

[44] Bram Wallace and Bharath Hariharan. Extending and analyzing self-supervised learning across domains. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 717–734, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58574-7.

[45] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[46] Haiyang Yang, Meilin Chen, Yizhou Wang, Shixiang Tang, Feng Zhu, Lei Bai, Rui Zhao, and Wanli Ouyang. Domain invariant masked autoencoders for self-supervised learning from multi-domains, 2022.

[47] Asano YM., Rupprecht C., and Vedaldi A. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=Hyx-jyBFPr.

[48] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction, 2021.

[49] Xingxuan Zhang, Linjun Zhou, Renzhe Xu, Peng Cui, Zheyan Shen, and Haoxin Liu. Towards unsupervised domain generalization. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022. doi: 10.1109/cvpr52688.2022.00486. URL http://dx.doi.org/10.1109/CVPR52688.2022.00486.