Iterative Partition-based Invariant Risk Minimization for Self-Supervised Visual Representation Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

A good visual representation is an inference map from observations (images) to features (vectors) that faithfully reflects the structure and transformations of the underlying generative factors (semantics), who are invariant to environmental changes. In this paper, we formulate the notion of "good" representation from a group-theoretic view using Higgins' definition of ***disentangled representation*** [39], and show that existing Self-Supervised Learning (SSL) can only learn augmentation-related features such as lighting and view shifts, leaving the rest of high-level semantics entangled. To break the limitation, we propose an iterative SSL method: Iterative Partition-based Invariant Risk Minimization (IP-IRM), which successfully grounds the abstract group actions into a concrete SSL optimization. At each iteration, IP-IRM first partitions the training samples into subsets. In particular, the partition reflects an entangled semantic group action. Then, it leverages IRM to learn subset-invariant sample similarities, where the invariance guarantees to disentangle the corresponding semantic. We prove that IP-IRM converges with a full-semantic disentangled representation, and show its effectiveness on various feature disentanglement and SSL benchmarks. Codes are in Appendix.

1 Introduction

Deep learning is all about learning feature representations [5]. Compared to the conventional end-to-end supervised learning, Self-Supervised Learning (SSL) first learns a generic feature representation (*e.g.*, a network backbone) by training with unsupervised pretext tasks such as the prevailing contrastive objective [34, 16], and then the above stage-1 feature is expected to serve various stage-2 applications with proper fine-tuning. SSL for visual representation is so fascinating that it is the first time that we can obtain "good" visual features for free, just like the large-scale pre-training trend in NLP community [24, 8]. Most works only care how much stage-2 performance an SSL feature can improve, but overlook what features SSL is learning, why they can be learned, what cannot be learned, how far is SSL from supervised learning, and can SSL surpass it?

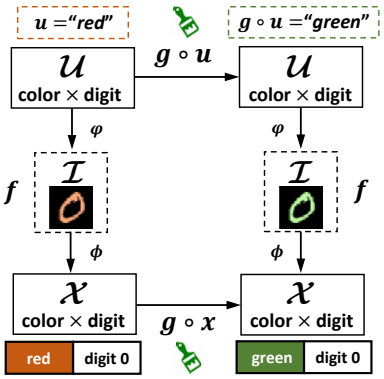


Figure 1: Disentangled representation is an equivariant map between the semantic space $\mathcal{U}$ and the vector space $\mathcal{X}$, which are decomposed into "color" and "digit".

The crux of answering those questions is to formally understand *what a feature representation is* and *what a good one is*. We postulate the classic world model of visual generation and feature representation [1, 65] as

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

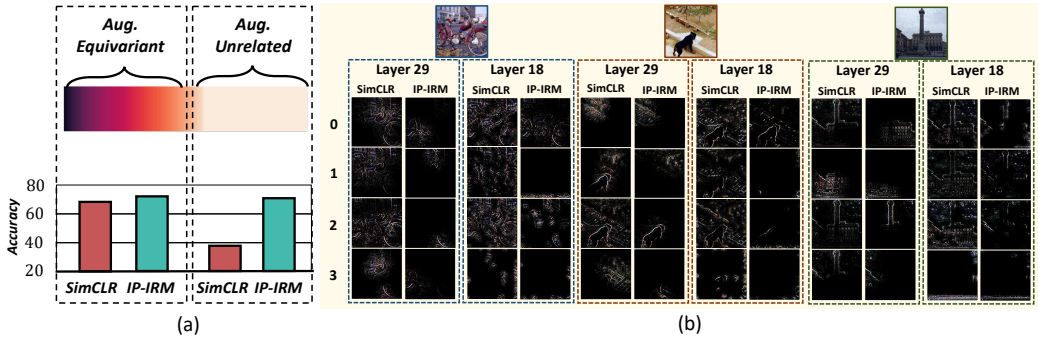


Figure 2: (a) The heat map visualizes feature dimensions equivariant to augmentations (aug. equivariant) and unrelated to augmentations (aug. unrelated), whose respective classification accuracy is shown in the bar chart below. Experiment was performed on STL10 [22] with representation learnt with SimCLR [16] and our IP-IRM. (b) Visualization of CNN activations [72] of 4 filters on layer 29 and 18 of VGG [71] trained on ImageNet100 [76]. The filters were chosen by first clustering the aug. unrelated filters with k-means ($k = 4$) and then selecting the filters corresponding to the cluster centers.

in Figure 1. Let $\mathcal{U}$ be a set of (unknown) *semantics*, *e.g.*, attributes such as "digit" and "color". There is a set of *independent and causal mechanisms* [62] $\varphi : \mathcal{U} \to \mathcal{I}$, generating images from semantics, *e.g.*, writing a digit "0" when thinking of "0" [70]. A **visual representation** is the inference process $\phi : \mathcal{I} \to \mathcal{X}$ that maps image pixels to vector space features, *e.g.*, a neural network. We define **semantic representation** as the functional composition $f : \mathcal{U} \to \mathcal{I} \to \mathcal{X}$. In this paper, we are only interested in the parameterization of the inference process for feature extraction, but not the generation process. We assume $\forall I \in \mathcal{I}, \exists u \in \mathcal{U}$, such that $I = \varphi(u)$ is fixed as the observation of each image sample. Therefore, we consider semantic and visual representations the same as **feature representation**, or simply **representation**, and we slightly abuse $\phi(I) := f\left(\varphi^{-1}(I)\right)$, *i.e.*, ϕ and f share the same trainable parameters.

We propose to use Higgins' definition of **disentangled representation** [39] to define what is "good". Note that this definition is a re-formulation of the model in Figure 1 using group representation theory [33] (See Appendix for the group theory review).

Definition 1 (Disentangled Representation). Let $\mathcal{G}$ be the group acting on $\mathcal{U}$, *i.e.*, $g \circ u$ transforms $u \in \mathcal{U}$ using $g \in \mathcal{G}$ (*e.g.*, changing color "red" to "green"). Suppose a direct product decomposition $\mathcal{G} = \mathcal{G}_1 \times \ldots \times \mathcal{G}_m$ and $\mathcal{U} = \mathcal{U}_1 \times \ldots \times \mathcal{U}_m$ such that $\mathcal{G}_i$ acts on $\mathcal{U}_i$ for $i = 1, \ldots, m$. A representation is disentangled if there exists a group $\mathcal{G}$ acting on $\mathcal{X}$ such that:

1. *Equivariant*: $\forall g \in \mathcal{G}, \forall u \in \mathcal{U}, f(g \circ u) = g \circ f(u)$, *e.g.*, changing the semantic "red" to "green" in $\mathcal{U}$ is equivalent to change the color vector in $\mathcal{X}$ from the value encoding "red" to "green".
2. *Decomposable*: There is a decomposition $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_m$, such that each $\mathcal{X}_i$ is fixed by the action of all $\mathcal{G}_j, j \neq i$ and affected[1] only by $\mathcal{G}_i$, *e.g.*, changing the color semantic in $\mathcal{U}$ does not affect the digit vector in $\mathcal{X}$.

The above definition indeed defines "good" features for classification in the common views: 1) *Robustness*: a good feature should be invariant to the change of environmental semantics, such as external interventions [44] or domain shifts [30]. By the above definition, a change is always retained in a subspace $\mathcal{X}_i$, while others are not affected. Hence, given the labelled data in a downstream task, the trained classifier will focus on the task-relevant features and ignore the ever-changing $\mathcal{X}_i$. 2) *Generalizability*: the structure ($\mathcal{G}$-module [33]) of the feature vector space $\mathcal{X}$ corresponds to the decomposition in $\mathcal{U}$ faithfully, and any change in $\mathcal{U}$ will be preserved in $\mathcal{X}$, even for unseen ones. So, the metrics of $\mathcal{X}$ trained by seen samples remains valid for zero-shot unseen samples [87]. We prove in Appendix that a disentangled representation always achieves lower supervised classification loss compared with an entangled one.

Are the existing SSL methods learning disentangled representations? No. We prove in Section 4 that they can only learn representations that are disentangled according to the hand-crafted augmentation semantics, *e.g.*, color, jitter, and rotation. For example, in Figure 2 (a), we show that one part of the SSL features is equivariant with augmentations, *i.e.*, it captures the augmentation-related semantics,

[1] See Appendix for a formal definition of "fixed" and "affected" using group action.

while the other part is not. Then, the classification accuracy of a standard SSL (SimCLR [16]) is mostly attributed first part, demonstrating that the other part is still useless as the features are still entangled. Moreover, in Figure 2 (b) the visualization of CNN activation in each layer indeed reveals entangled semantics (*e.g.*, tyre, motor and background in the motorcycle image). In contrast, our approach IP-IRM, to be introduced below, learns features useful for classification beyond augmentations as shown in Figure 2 (a) and disentangle the semantics in Figure 2 (b).

In this paper, we propose Iterative Partition-based Invariant Risk Minimization (**IP-IRM**) that guarantees to learn disentangled representations in an SSL fashion. We present the algorithm in Section 3, followed by the theoretical justifications in Section 4. In a nutshell, the core of IP-IRM is to partition training data into disjoint subsets. At each iteration, it first discovers a group $\mathcal{G}_i$ that is still entangled, whose action corresponds to identity mapping in each subset of the partition. Then, we adopt the **Invariant Risk Minimization (IRM)** [2] to implement a **partition-based** SSL, which disentangles the representation f *w.r.t.* $\mathcal{G}_i$. Iterating the above two steps eventually converges at a fully disentangled representation *w.r.t.* $\mathcal{G}_1 \times \ldots \times \mathcal{G}_m$. We conduct extensive experiments on various feature disentanglement and SSL benchmarks and show promising results as an empirical support.

2 Related Work

Self-Supervised Learning. SSL aims to learn representations from unlabeled data with hand-crafted pretext tasks [26, 59, 31]. Recently, Contrastive learning [61, 57, 37, 75, 16] prevails in most state-of-the-art methods. The key is to map positive samples closer, while pushing apart negative ones in the feature space. Specifically, the positive samples are from the augmented views [77, 3, 86, 41] of each instance and the negative ones are other instances. Along this direction, follow-up methods are mainly four-fold: 1) Memory-bank [83, 57, 34, 18]: storing the prototypes of all the instances computed previously into a memory bank to benefit from a large number of negative samples. 2) Using siamese network [7] to avoid representation collapse [32, 19, 78]. 3) Assigning clusters to samples to integrate inter-instance similarity into contrastive learning [11, 12, 13, 82, 52]. 4) Seeking hard negative samples with adversarial training or better sampling strategies [69, 20, 43, 45]. In contrast, our proposed IP-IRM jumps out of the above frame and introduces the *disentangled representation* into SSL with group theory to show the limitations of existing SSL and how to break through them.

Disentangled Representation. This notion dates back to [4], and henceforward becomes a high-level goal of separating the factors of variations in the data [79, 74, 81, 54]. Several works aim to provide a more precise description [25, 27, 68, 25] by adopting an information-theoretic view [17, 25] and measuring the properties of a disentangled representation explicitly [27, 68]. We adopt the recent group-theoretic definition from Higgins *et al.* [39], which not only unifies the existing, but also resolves the previous controversial points [73, 55]. Although supervised learning of disentangled representation is a well-studied field [91, 42, 10, 64, 66, 46], unsupervised disentanglement based on GAN [17, 60, 53, 67] or VAE [38, 15, 90, 47] is still believed to be theoretically challenging [55]. Thanks to the Higgins' definition, we prove that the proposed IP-IRM converges with full-semantic disentanglement using group representation theory. Notably, unlike all the existing unsupervised methods based on generative models, our IP-IRM is the first approach to learn an inference process, making it widely applicable even on large-scale datasets.

3 IP-IRM Algorithm

Notations. Our goal is to learn the feature extractor ϕ in an unsupervised way. We define a partition matrix $\mathbf{P} \in \{0,1\}^{N \times K}$ that partitions N training images into K disjoint subsets. $P_{i,k} = 1$ if the i-th image belongs to the k-th subset and 0 otherwise. Suppose we have a pretext task loss function $\mathcal{L}(\phi, \theta = 1.0, k, \mathbf{P})$ defined on the samples in the k-th subset ($k \in \{1, \ldots, K\}$) of the partition $\mathbf{P}$, where $\theta = 1.0$ is a "dummy" parameter used to evaluate the invariance of the SSL loss across the subsets (later discussed in Step 1). For example, for contrastive learning [34, 16], $\mathcal{L}$ is given by:

$$\mathcal{L}(\phi, \theta = 1.0, k, \mathbf{P}) = -\log \frac{\exp\left(\phi(I)^T \phi(I^*) \cdot \theta\right)}{\sum_{I' \in \mathcal{I}^- \cup I^*} \exp\left(\phi(I)^T \phi(I') \cdot \theta\right)} \tag{1}$$

where I^* is an augmented view of I and $\mathcal{I}^- = \{I_i | i \in \{1, \ldots, N\} \wedge P_{i,k} = 1\}$ is the negative samples that $I \notin \mathcal{I}^-$.

Input. N training images. Randomly initialized ϕ. A partition matrix $\mathbf{P}$ initialized such that the first column of $\mathbf{P}$ is 1, *i.e.*, all samples belong to the first subset. The set $\mathcal{P} = \{\mathbf{P}\}$.

Output. Disentangled feature extractor ϕ.

Step 1 [Update ϕ]. We update ϕ with the objective:

$$\min_{\phi} \sum_{\mathbf{P}\in\mathcal{P}} \sum_{k=1}^{K} \left[\mathcal{L}(\phi, \theta = 1.0, k, \mathbf{P}) + \lambda_1 \left\| \nabla_{\theta=1} \mathcal{L}(\phi, \theta = 1.0, k, \mathbf{P}) \right\|^2 \right], \tag{2}$$

where λ_1 is a hyper-parameter. In particular, the scalar $\|\nabla_\theta \mathcal{L}\|^2$ delineates how far the ϕ-induced similarity is from a fixed baseline $\theta = 1$. Therefore, this scalar regularizes ϕ to be invariant across the subsets in a partition, as the baseline is constant. See IRM [2] for more details. We use SGD for Eq. (2) using samples from the entire dataset for one epoch.

Step 2 [Update $\mathbf{P}$]. We fix ϕ and find a new partition $\mathbf{P}^*$ with:

$$\mathbf{P}^* = \arg\max_{\mathbf{P}} \sum_{k=1}^{K} \left[\mathcal{L}(\phi, \theta = 1.0, k, \mathbf{P}) + \lambda_2 \left\| \nabla_\theta \mathcal{L}(\phi, \theta = 1.0, k, \mathbf{P}) \right\|^2 \right], \tag{3}$$

where λ_2 is a hyper-parameter. To practically optimize $\mathbf{P}$ by back-propagation, we use a continuous partition matrix in $\mathbb{R}^{N\times K}$ during optimization and then threshold it to $\{0,1\}^{N\times K}$.

We update $\mathcal{P} \leftarrow \mathcal{P} \cup \mathbf{P}^*$ and iterate the above two steps until convergence.

4 Justification

We aim to justify the convergence of IP-IRM, where a more formal discussion and the detailed proof are given in Appendix.

Definition 2 ($\mathcal{G}_i$-Disentanglement). *A representation f is called $\mathcal{G}_i$-disentangled if f is equivariant under the action of $\mathcal{G}_i$ and there exists a decomposition $\mathcal{X} = \mathcal{X}_i \times \hat{\mathcal{X}}$, such that $\mathcal{X}_i$ is affected only by the action of $\mathcal{G}_i$, and $\hat{\mathcal{X}}$ is affected only by the action of $\prod \mathcal{G}_j, j \neq i$.*

Based on the definition, our goal is to gradually disentangle the representation, such that after each iteration of IP-IRM, the feature responds to the action of a new $\mathcal{G}_i$. This requires us to understand what has not been disentangled, which is given by the following lemma that enables the learning of $\mathcal{G}_i$-disentangled representation.

Lemma 1. *After the t-th iteration of IP-IRM, we have the group acting on the feature space $\mathcal{X}$ taking the form*

$$\{\prod_{i=1}^{t} \mathcal{G}_i \times \quad \mathcal{G}_{t+1} \times \prod_{j=t+2}^{m} \mathcal{G}_j\}, \tag{4}$$

with the corresponding decomposition $\mathcal{X} = \prod_{i=1}^{t} \mathcal{X}_i \times \tilde{\mathcal{X}}$, f is $\mathcal{G}_i$-disentangled $\forall i \in \{1, \dots t\}$ and equivariant under the action of $\mathcal{G}_{t+1}$, $\tilde{X}$ is affected only by the action of $\mathcal{G}_{t+1} \times \prod_{j=t+2}^{m} \mathcal{G}_j$. In particular, SSL and fully-supervised learning are two special cases of $\mathcal{G}_i$-disentanglement $\forall i \in \{1, \dots, t\}$.

Lemma 1 says that among the entangled groups, there exists $\mathcal{G}_{t+1}$ whose action on $\mathcal{X}$ is reflected faithfully as the changes in $\tilde{\mathcal{X}} \in \mathcal{X}$ through the equivariant representation f. In this way, by observing the distances in the feature space $\mathcal{X}$, we can gain information of $\mathcal{G}_{t+1}$, enabling us to partition the training images based on $\mathcal{G}_{t+1}$ as shown later.

If Lemma 1 does not hold, then f is not equivariant under the action of any $\mathcal{G}$ that essentially corresponds to a linear map in $\mathcal{X}$. There are two possible cases: 1) f is a collapsed representation that maps all inputs to a fixed vector, which is impossible in today's deep model training [88]. 2) $\mathcal{X}$ is a non-linear manifold. However, it has been shown that there always exists a linear projection [35], *i.e.*, it must have a linear map in a subspace of $\mathcal{X}$ that corresponds to at least one group, *e.g.*, $\mathcal{G}_{t+1}$, which contradicts with the former assumption. Therefore, Lemma 1 holds.

It is worth highlighting that Lemma 1 shows the limitations of SSL and supervised learning. In standard SSL, augmenting a sample corresponds to the action on augmentation semantics. By learning to recognize the augmentation-related action through SSL pretext tasks, we disentangle the corresponding group as $\prod_{i=1}^{t} \mathcal{G}_i$. Yet SSL fails to disentangle beyond augmentation. In fully-supervised learning, each class is essentially a point in the semantic space $\mathcal{U}$, and hence the transformations between classes corresponds to the action on class-related semantics, which is disentangled as the group $\prod_{i=1}^{t} \mathcal{G}_i$ with supervision on the classes (or transformation between them). However, it is impossible to use the transformations among limited classes (*e.g.*, 1,000 in ImageNet [23]) to cover the action of the *entire* group $\prod_{i=1}^{m} \mathcal{G}_i$ in complex visual tasks with large m, and hence the learned representation becomes unresponsive to the action of some group elements, leading to semantic loss [14, 87].

Theorem 1 (Convergence of IP-IRM)**.** *1) Maximization of Eq.* (3) *leads to a partition* $\mathbf{P}^*$*, such that within every subset of* $\mathbf{P}^*$*, the group action of* $\mathcal{G}_{t+1}$ *on the subspace of* $\mathcal{X}$ *spanned by the subset is the identity mapping. 2) When* $\mathbf{P}^* \in \mathcal{P}$*, minimization of Eq.* (2) *leads to a* $\mathcal{G}_{t+1}$*-disentangled* f*. 3) IP-IRM converges with a disentangled* f*.*

From Lemma 1, we show that maximizing Eq. (3) (*i.e.*, maximizing the sample similarity in $\mathcal{X}$ in each subset) leads to a series of sample exchanges between subsets (see Figure 3 top), until the action of $\mathcal{G}_{t+1}$ "disappears" in each subset of the resulting $\mathbf{P}^*$, *i.e.*, the only possible action is to remain as it is. Figure 3 shows an example with $\mathcal{G}_{t+1}$ corresponding to digit color, where in each subset of $\mathbf{P}^*$, there is no color change between any two samples.

Under $\mathbf{P}^*$, the action of $\mathcal{G}_{t+1}$ across subsets is *not* identity mapping (*e.g.*, changing from "green" to "red"). To minimize Eq. (2) and achieve invariance across subsets, the effect from $\mathcal{G}_{t+1}$ must be ignored. Since the SSL loss $\mathcal{L}$ uses $\tilde{\mathcal{X}}$ to compute sample-wise distances, any $\mathcal{G}_{t+1}$-entangled f will inevitably involve $\mathcal{G}_{t+1}$. By decomposing a separate subspace $\mathcal{X}_{t+1}$ from $\tilde{\mathcal{X}}$ and achieving $\mathcal{G}_{t+1}$-disentanglement, the value in $\mathcal{X}_{t+1}$ is the same in each subset of $\mathbf{P}^*$, as the action of $\mathcal{G}_{t+1}$ on $\mathcal{X}_{t+1}$ is identity mapping. In this way, $\mathcal{X}_{t+1}$ is not useful to $\mathcal{L}$ in each subset, and can be ignored to isolate the effect from $\mathcal{G}_{t+1}$. An example of the learnt disentangled representation in CMNIST dataset [2] is shown in Figure 5 (b), where each color and digit semantic correspond to its own set of feature dimensions.

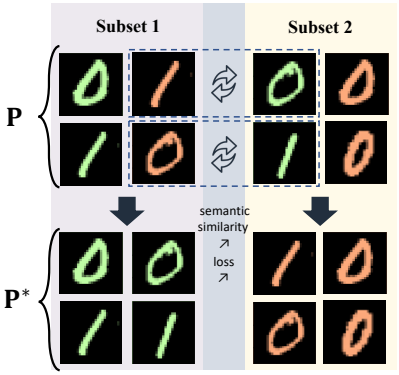


Figure 3: A series of "exchange" operation transforms a partition $\mathbf{P}$ to $\mathbf{P}^*$, while increasing the overall SSL loss by making the semantics in each subset more similar.

Overall, IP-IRM converges as we discover an entangled group in each iteration and m is finite.

5 Experiments

5.1 Unsupervised Disentanglement

Datasets. We used two datasets. **CMNIST** [2] has 60,000 digit images with semantic labels of digits (0-9) and colors (red and green). These images differ in other semantics (*e.g.*, slant and font) that are not labeled. Moreover, there is a strong correlation between digits and colors (most 0-4 in red and 5-9 in green), increasing the difficulty to disentangle them. **Shapes3D** [47] contains 480,000 images with 6 labelled semantics, *i.e.*, size, type, azimuth, as well as floor, wall and object color. Note that we only considered the first three semantics for evaluation, as the standard augmentations in SSL will contaminate any color-related semantics.

Settings. We adopted 6 representative disentanglement metrics: *Disentangle Metric for Informativeness* (DCI) [27], *Interventional Robustness Score* (IRS) [74], *Explicitness Score* (EXP) [68], *Modularity Score* (MOD) [68] and the accuracy of predicting the ground-truth semantic labels by two classification models called *logistic regression* (LR) and *gradient boosted trees* (GBT) [55]. Specifically, DCI and EXP measure the explicitness, *i.e.*, the values of semantics can be decoded from the feature using a linear transformation. MOD and IRS measure the modularity, *i.e.*, whether each feature dimension is equivariant to the shift of a single semantic. See Appendix for more detailed

	Method	DCI	IRS	MOD	EXP	LR	GBT	Average
CMNIST	VAE [48]	**0.952**	-	0.842	0.972	0.829	**0.952**	0.909
	β-VAE [40]	0.947	-	0.790	0.970	0.825	0.947	0.896
	β-AnnealVAE [9]	0.914	-	0.866	0.974	0.839	0.914	0.901
	β-TCVAE [15]	0.924	-	**0.982**	0.958	0.782	0.924	0.914
	Factor-VAE [47]	0.921	-	0.97	0.959	0.788	0.922	0.912
	SimCLR [16]	0.895	-	0.773	0.986	0.900	0.894	0.890
	IP-IRM (Ours)	0.921	-	0.830	**0.991**	**0.927**	0.920	**0.918**
Shapes3D	VAE [48]	0.322	0.295	0.807	0.764	0.368	0.321	0.480
	β-VAE [40]	0.347	0.303	**0.820**	0.824	0.468	0.353	0.519
	β-AnnealVAE [9]	0.351	**0.446**	0.718	0.636	0.269	0.352	0.462
	β-TCVAE [15]	0.410	0.287	0.806	0.742	0.304	0.413	0.494
	Factor-VAE [47]	0.369	0.333	**0.820**	0.786	0.371	0.371	0.508
	SimCLR [16]	0.512	0.402	0.594	0.942	0.639	0.514	0.601
	IP-IRM (Ours)	**0.533**	0.417	0.791	**0.953**	**0.734**	**0.532**	**0.660**

Table 1: Results on disentanglement metrics of existing unsupervised disentanglement methods, standard SSL (SimCLR [16]) and IP-IRM using CMNIST [2] and Shapes3D [47]. Note that IRS is based on intervening the semantics which requires access to the labels of all the semantics, and hence not applicable for CMNIST dataset.

formula of the metrics. In evaluation, we trained CNN-based feature extractor backbones with comparable number of parameters for all the baselines and our IP-IRM, and the full implementation details are in Appendix.

Results. In Table 1, we compared the proposed IP-IRM to the standard SSL method SimCLR [16] as well as several generative disentanglement methods [48, 40, 9, 15, 47]. On both CMNIST and Shapes3D dataset, IP-IRM significantly outperforms SimCLR regrading all metrics where the most relative gain is 19.7% for MOD. For this MOD, we notice that β-VAE and Factor-VAE perform better than our IP-IRM by 3 points, *i.e.*, 0.82 v.s. 0.79 for Shapes3D. This is because VAE explicitly pursues a high modularity score through regularizing the dimension-wise independence in the feature space. However, this regularization is adversarial to discriminative objectives [14, 87]. Indeed, we can observe from the column of LR (i.e., the performance of downstream linear classification) that VAE methods have clearly poor performance especially on the more challenging dataset Shapes3D. We can draw the same conclusion from the results of GBT. Different from VAE methods, our IP-IRM is optimized towards disentanglement without such regularization, and is thus able to outperform the others in downstream tasks while obtaining a competitive value of modularity.

What does IP-IRM feature look like? Figure 4 visualizes the features learned by SimCLR and our IP-IRM on two datasets: CMNIST in Figure 4 (a) and STL10 dataset in Figure 4 (b). In the following, we use Figure 4 (a) as the example, and can easily draw the similar conclusions from Figure 4 (b). On the left-hand side of Figure 4 (a), it is obvious that there is no clear boundary to distinguish the semantic of color in the SimCLR feature space. Besides, the features of the same digit semantic are scattered in two regions. On the right-hand side of (a), we have 3 observations for IP-IRM. 1) The features are well clustered and each cluster corresponds to a specific semantic of either digit or color. This validates the *equivariant* property of IP-IRM representation that it responds to any changes of the existing semantics, *e.g.*, digit and color on this dataset. 2) The feature space has the symmetrical structure for each individual semantic, validating the *decomposable* property of IP-IRM representation. More specifically, i) mirroring a feature (*w.r.t.* "*" in the figure center) indicates the change on the only semantic of color, regardless the other semantic (digit); and ii) a counterclockwise rotation (denoted by black arrows from same-colored 1 to 7) indicates the change on the only semantic of digit. 3) IP-IRM reveals the true distribution (similarity) of different classes. For example, digits 3, 5, 8 sharing sub-parts (curved bottoms and turnings) have closer feature points in the IP-IRM feature space.

How does IP-IRM disentangle features? *1) Discovered* $\mathbf{P}^*$: To visualize the discovered partitions $\mathbf{P}^*$ at each maximization step, we performed an experiment using the partitions with 2 subsets on a binary CMNIST (digit 0 and 1 in color red and green), and show the results in Figure 5 (a). Please kindly refer to Appendix for the full results on CMNIST. First, each partition tells apart a specific semantic into two subsets, *e.g.*, in Partition #1, red and green digits are separated. Second, besides of the obvious semantics—digit and color (labelled on the dataset), we can discover new semantics, *e.g.*, the digit slant shown in Partition #3. *2) Disentangled Representation*: In Figure 5 (b), we aim to visualize how equivariant each feature dimension is to the change of each semantic, *i.e.*, a darker color

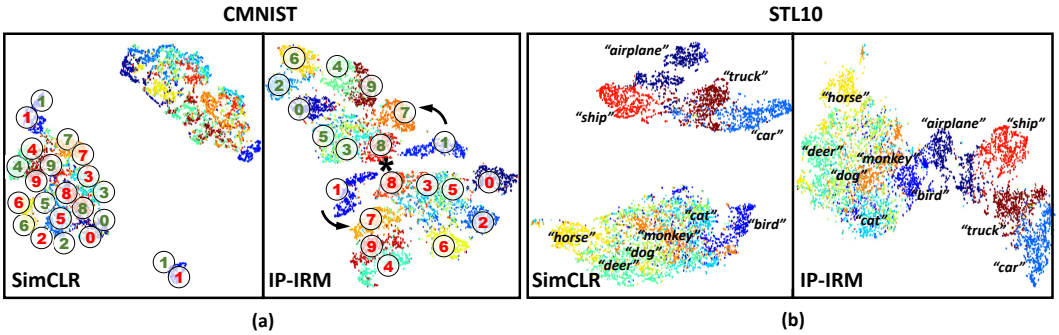


Figure 4: The t-SNE [80] visualizations of learned feature spaces using SimCLR [16] and IP-IRM on CMNIST [2] and STL10 [22]. For CMIST in (a), we annotate the digit and color near each cluster. We annotate only half of the feature points for SimCLR to avoid clutter. For STL10 in (b), we show the labels of the classes.

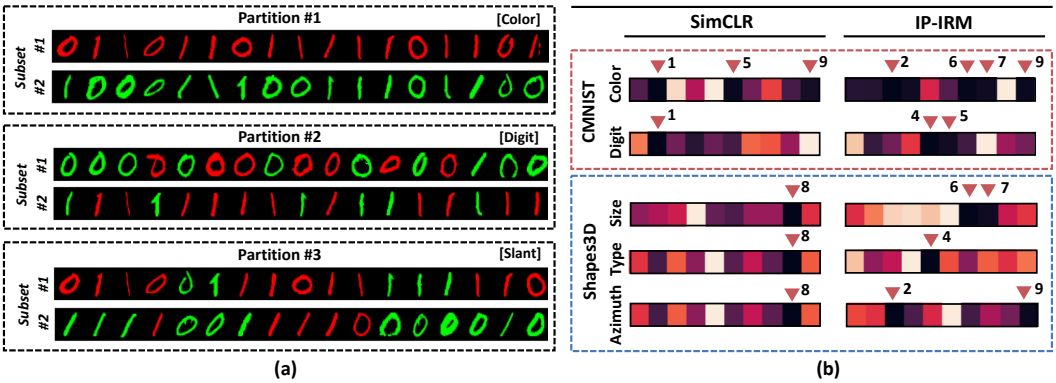


Figure 5: (a) Visualization of the obtained partitions $\mathbf{P}^*$ during training. Each partition has two subset and the displayed images are *randomly* sampled from each subset. (b) Visualization of the variance of each feature dimension when perturbing the semantic indicated on the left. The most equivariant dimensions are indicated by triangles and their corresponding indices.

shows that a dimension is more equivariant *w.r.t.* the semantic indicated on the left. We can see that SimCLR fails to learn the decomposable representation, *e.g.*, the 8-th dimension captures azimuth, type and size in Shapes3D. In contrast, our IP-IRM achieves disentanglement by representing the semantics into interpretable dimensions, *e.g.*, the 6-th and 7-th dimensions captures the size, the 4-th for type and the 2-nd and 9-th for azimuth on the Shapes3D. For summarization, the overall results support the justification in Section 4, *i.e.*, $\mathbf{P}^*$ corresponds to a semantic at each iteration and the training IP-IRM eventually converges with a disentangled representation.

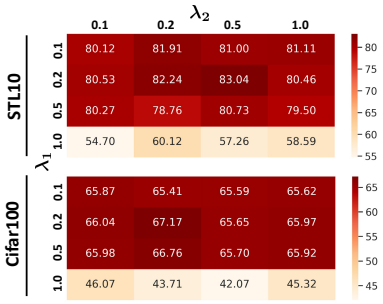
5.2 Self-Supervised Learning

Datasets and Settings. We conducted the SSL evaluations on 2 standard benchmarks, following [82, 20, 45]. **Cifar100** [51] contains 60,000 images in 100 classes and **STL10** [22] has 113,000 images in 10 classes. We used SimCLR [16], DCL [20] and HCL [45] as baselines, and learned the representations for 400 epochs. We evaluated both linear and k-NN ($k = 200$) accuracies for the downstream classification task.

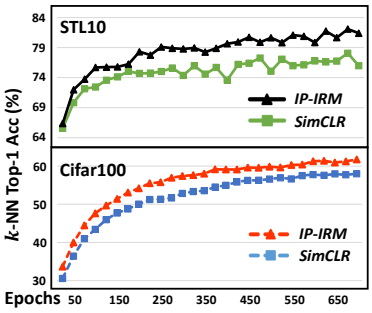
Results. We demonstrate our results and compare with baselines in Table 2. Incorporating IP-IRM to the 3 baselines brings consistent performance boosts to downstream classification models in all settings, *e.g.*, improving the linear models by 5.55% on STL10 and 2.92% on Cifar100. In particular, we observe that IP-IRM brings huge performance gain with k-NN classifiers, *e.g.*, 4.23% using HCL+IP-IRM on STL10, which validates that our algorithm further disentangles compared to the standard SSL, *i.e.*, the distance metrics in the IP-IRM feature space more faithfully reflecting the class semantic differences. Still, the quality of disentanglement cannot be fully evaluated by supervised classification—while the improved accuracy demonstrates that IP-IRM representation is more equivariant to class semantics, it does not reveal if the representation is decomposable. Hence we present an out-of-distribution (OOD) setting in Section 5.3 to further show this property.

K	k −NN	Linear
1	74.71	77.26
2	**78.31**	**83.04**
3	76.96	82.50
4	77.06	81.49
5	77.24	81.64
6	77.31	82.00

(a)



(b)



(c)

Figure 6: Our ablation study on the STL10 and Cifar100 datasets. (a) The effect of K, *i.e.*, the number of the subsets in $\mathbf{P}$, on the STL10 dataset. (b) The Top-1 accuracy (%) of linear classifiers using different values of λ_1 and λ_2 (in Eq. (2) and Eq. (3)), by training for 200 epochs on two datasets. (c) The Top-1 accuracy (%) of k-NN classifiers on two datasets, for which we trained the models for 700 epochs and updated $\mathbf{P}$ every 50 epochs.

Method	STL10		Cifar100	
	k-NN	Linear	k-NN	Linear
SimCLR [16]	73.60	78.89	54.94	66.63
DCL [20]	78.82	82.56	57.29	68.59
HCL [45]	80.06	87.60	59.61	69.22
SimCLR+IP-IRM	79.66	84.44	59.10	69.55
DCL+IP-IRM	81.51	85.36	58.37	68.76
HCL+IP-IRM	**84.29**	**87.81**	**60.05**	**69.95**

Table 2: Accuracy (%) of k-NN and linear classifiers on STL10 [22] and Cifar100 [51] using the representations of SimCLR [16], DCL [20], HCL [45] and those after incorporating our IP-IRM.

Is IP-IRM sensitive to the values of hyper-parameters? 1) *The number of subsets* K. In Figure 6 (a), we find that increasing K causes a decrease of the model performance, as it reduces the difficulty of pretext tasks, impairing the model learning (see Appendix for details). 2) λ_1 *and* λ_2 *in Eq.* (2) *and Eq.* (3). In Figure 6 (b), we observe that the best performance is achieved with λ_1 and λ_2 taking values from 0.2 to 0.5 on both datasets. All accuracies drop sharply if using $\lambda_1 = 1.0$. The reason is that a higher λ_1 forces the model to push the ϕ-induced similarity to fixed baseline $\theta = 1$, rather than decrease the loss $\mathcal{L}$ on the pretext task, leading to poor convergence. 3) *The number of epochs*. In Figure 6 (c), we plot the Top-1 accuracies of using k-NN classifiers along the 700-epoch training of two kinds of SSL representations—SimCLR and IP-IRM. It is obvious that IP-IRM converges faster and achieves a higher accuracy than SimCLR. It is worth to highlight that on the STL10, the accuracy of SimCLR starts to oscillate and grow slowly after the 150-th epoch, while ours keeps on improving. This is an empirical evidence that IP-IRM keeps on disentangling more and more semantics in the feature space, and has the potential of improvement through long-term training.

5.3 Potential on Large-Scale Data

Datasets. We evaluated on the standard benchmark of supervised learning **ImageNet ILSVRC-2012** [23] which has in total 1,331,167 images in 1,000 classes. To further reveal if a representation is decomposable, we used NICO [36], which is a real-world image dataset designed for OOD evaluations. It contains 25,000 images in 19 classes, with a strong correlation between the foreground and background in the train split (*e.g.*, most dogs on grass). We also studied the transferability of the learned representation following [28, 49]: FGVC Aircraft (**Aircraft**) [56], Caltech-101 (**Caltech**) [29], Stanford Cars (**Cars**) [85], **Cifar10** [50], **Cifar100** [50], **DTD** [21], Oxford 102 Flowers (**Flowers**) [58], Food-101 (**Food**) [6], Oxford-IIIT Pets (**Pets**) [63] and SUN397 (**SUN**) [84]. These datasets include coarse- to fine-grained classification tasks, and vary in the amount of training data (2,000-75,000 images) and classes (10-397 classes), representing a wide range of transfer learning settings.

Settings. For the ImageNet, all the representations were trained for 200 epochs due to limited computing resources. We followed the common setting [75, 34], using a linear classifier, and report Top-1 classification accuracies. For NICO, we fixed the ImageNet pre-trained ResNet-50 backbone and fine-tuned the classifier. See appendix for more training details. For the transfer learning, we followed [28, 49] to report the classification accuracies on Cars, Cifar-10, Cifar-100, DTD, Food, SUN and the average per-class accuracies on Aircraft, Caltech, Flowers, Pets. We call them uniformly as Accuracy. We used the few-shot n-way-k-shot setting for model evaluation. Specifically, we randomly sampled 2,000 episodes from the *test* splits of above datasets. An episode contains n

Method	ImageNet	NICO
InsDis [83]	56.5	41.6
PCL [52]	61.5	61.2
PIRL [57]	63.6	46.5
MoCo-v1 [34]	60.6	45.1
MoCo-v2 [18]	67.5	64.3
IP-IRM (Ours)	**69.1**	**67.2**

Table 3: ImageNet and NICO Top-1 Accuracy (%) of linear classifiers trained on the representations learnt with different SSL methods.

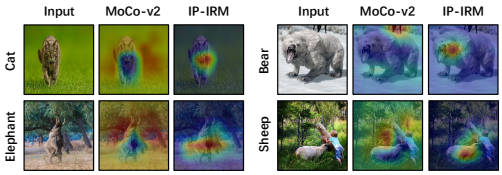


Figure 7: Visualization of CAM [89] on images from NICO [36] dataset using representations of the baseline MoCo-v2 [18] and our IP-IRM.

Method	Aircraft	Caltech	Cars	Cifar10	Cifar100	DTD	Flowers	Food	Pets	SUN	Average
InsDis [83]	35.07	75.97	37.49	51.49	57.61	69.38	77.35	50.01	66.38	74.97	59.57
PCL [52]	**36.86**	90.72	39.68	59.26	60.78	69.53	67.50	57.06	**88.31**	84.51	65.42
PIRL [57]	36.70	78.63	39.21	49.85	55.23	70.43	78.37	51.61	69.40	76.64	60.61
MoCo-v1 [34]	35.31	79.60	36.35	46.96	51.62	68.76	75.42	49.77	68.32	74.77	58.69
MoCo-v2 [18]	31.98	92.32	41.47	56.50	63.33	78.00	80.05	57.25	83.23	88.10	67.22
IP-IRM (Ours)	32.98	**93.16**	**42.87**	**60.73**	**68.54**	**79.30**	**82.68**	**59.61**	85.23	**89.38**	**69.44**

Table 4: Accuracy (%) of 5-way-5-shot few-shot evaluation using the image representation learned on ImageNet [23]. More detailed results are given in Appendix.

classes, each with k training samples and 15 testing samples, where we fine-tuned the linear classifier (backbone weights frozen) for 100 epochs on the training samples, and evaluated the classifier on the testing samples. We evaluated with $n = k = 5$ (results of $n = 5, k = 20$ in Appendix).

ImageNet and NICO. In Table 3 ImageNet accuracy, our IP-IRM achieves the best performance over all baseline models. Yet we believe that this does not show the full potential of IP-IRM, because ImageNet is a larger-scale dataset with many semantics, and it is hard to achieve a full disentanglement of all semantics within the limited 200 epochs. To evaluate the feature decomposability of IP-IRM, we compared the performance on NICO with various SSL baselines in Table 3, where our approach significantly outperforms the best baseline by 2.9%. This validates IP-IRM feature is more decomposable—if each semantic feature (*e.g.*, background) is decomposed in some fixed dimensions and some classes vary with such semantic, then the classifier will recognize this as a non-discriminative variant feature and hence focus on other more discriminative features (*i.e.*, foreground). In this way, even though some classes are confounded by those non-discriminative features (*e.g.*, most of the "dog" images are with "grass" background), the fixed dimensions still help classifiers neglect those non-discriminative ones. We further visualized the CAM [89] on NICO in Figure 7, which indeed shows that IP-IRM helps the classifier focus on the foreground regions.

Few-Shot Tasks. As shown in Table 4, our IP-IRM significantly improves the performance of 5-way-5-shot setting, *e.g.*, we outperform the baseline MoCo-v2 by 2.2%. This is in line with recent works [81] showing that a disentangled representation is especially beneficial in low-shot scenarios, and further demonstrates the importance of disentanglement in downstream tasks.

6 Conclusion

We presented an unsupervised disentangled representation learning method called Iterative Partition-based Invariant Risk Minimization (IP-IRM), based on Self-Supervised Learning (SSL). IP-IRM iteratively partitions the dataset into semantic-related subsets, and learns a representation invariant across the subsets using SSL with an IRM loss. We show that with theoretical guarantee, IP-IRM converges with a disentangled representation under the group-theoretical view, which fundamentally surpasses the capabilities of existing SSL and fully-supervised learning. Our proposed theory is backed by strong empirical results in disentanglement metrics, SSL classification accuracy and transfer performance. IP-IRM achieves disentanglement without using generative models, making it widely applicable on large-scale visual tasks. As future direction, we will continue to explore the application of group theory in representation learning and seek additional forms of inductive bias for faster convergence.

References

[1] Philip W Anderson. More is different. *Science*, 1972.

[2] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[3] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019.

[4] Yoshua Bengio. *Learning deep architectures for AI*. Now Publishers Inc, 2009.

[5] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[6] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, 2014.

[7] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, 6:737–744, 1993.

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

[9] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[10] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. Learning disentangled semantic representation for domain adaptation. In *IJCAI: proceedings of the conference*, 2019.

[11] 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)*, pages 132–149, 2018.

[12] 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*, pages 2959–2968, 2019.

[13] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[14] Long Chen, Hanwang Zhang, Jun Xiao, Wei Liu, and Shih-Fu Chang. Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In *CVPR*, 2018.

[15] Ricky TQ Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in neural information processing systems*, 2018.

[16] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[17] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, 2016.

[18] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[19] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020.

[20] Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *arXiv preprint arXiv:2007.00224*, 2020.

[21] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[22] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

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

[24] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019.

[25] Kien Do and Truyen Tran. Theory and evaluation metrics for learning disentangled representations. In *International conference on learning representations*, 2020.

[26] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430, 2015.

[27] Cian Eastwood and Christopher KI Williams. A framework for the quantitative evaluation of disentangled representations. In *International conference on learning representations*, 2018.

[28] Linus Ericsson, Henry Gouk, and Timothy M. Hospedales. How Well Do Self-Supervised Models Transfer? In *CVPR*, 2021.

[29] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, 2004.

[30] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 2016.

[31] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

[32] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

[33] W.F.J. Harris, W. Fulton, and J. Harris. *Representation Theory: A First Course*. 1991.

[34] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

[35] Xiaofei He and Partha Niyogi. Locality preserving projections. *Advances in neural information processing systems*, 2004.

[36] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognition*, 110:107383, 2021.

[37] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020.

[38] I. Higgins, Loïc Matthey, A. Pal, C. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

[39] Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.

[40] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *International conference on learning representations*, 2017.

[41] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[42] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. Learning to decompose and disentangle representations for video prediction. In *Advances in neural information processing systems*, 2018.

[43] Qianjiang Hu, Xiao Wang, Wei Hu, and Guo-Jun Qi. Adco: Adversarial contrast for efficient learning of unsupervised representations from self-trained negative adversaries. *arXiv preprint arXiv:2011.08435*, 2020.

[44] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, 2019.

[45] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *arXiv preprint arXiv:2010.01028*, 2020.

[46] Theofanis Karaletsos, Serge Belongie, and Gunnar Rätsch. Bayesian representation learning with oracle constraints. *International conference on learning representations*, 2015.

[47] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *International Conference on Machine Learning*, 2018.

[48] Diederik Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[49] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[50] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2012.

[51] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[52] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020.

[53] Zinan Lin, Kiran Thekumparampil, Giulia Fanti, and Sewoong Oh. Infogan-cr and modelcentrality: Self-supervised model training and selection for disentangling gans. In *International Conference on Machine Learning*, 2020.

[54] F. Locatello, M. Tschannen, S. Bauer, G. Rätsch, B. Schölkopf, and O. Bachem. Disentangling factors of variations using few labels. In *8th International Conference on Learning Representations (ICLR)*, 2020.

[55] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, 2019.

[56] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[57] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.

[58] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 2008.

[59] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer, 2016.

[60] Utkarsh Ojha, Krishna Kumar Singh, Cho-Jui Hsieh, and Yong Jae Lee. Elastic-infogan: Unsupervised disentangled representation learning in class-imbalanced data. In *Advances in neural information processing systems*, 2020.

[61] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[62] Giambattista Parascandolo, Niki Kilbertus, Mateo Rojas-Carulla, and Bernhard Schölkopf. Learning independent causal mechanisms. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4036–4044, 2018.

[63] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, 2012.

[64] Robin Quessard, Thomas Barrett, and William Clements. Learning disentangled representations and group structure of dynamical environments. *Advances in Neural Information Processing Systems*, 2020.

[65] Rajesh PN Rao and Daniel L Ruderman. Learning lie groups for invariant visual perception. *Advances in neural information processing systems*, 1999.

[66] Scott Reed, Kihyuk Sohn, Yuting Zhang, and Honglak Lee. Learning to disentangle factors of variation with manifold interaction. In *International conference on machine learning*, 2014.

[67] Xuanchi Ren, Tao Yang, Yuwang Wang, and Wenjun Zeng. Do generative models know disentanglement? contrastive learning is all you need. *arXiv preprint arXiv:2102.10543*, 2021.

[68] Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. In *Advances in neural information processing systems*, 2018.

[69] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020.

[70] Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning*, 2012.

[71] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations*, 2015.

[72] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

[73] Raphael Suter, Djordje Miladinovic, Stefan Bauer, and Bernhard Schölkopf. Interventional robustness of deep latent variable models. *arXiv*, 2018.

[74] Raphael Suter, Djordje Miladinovic, Bernhard Schölkopf, and Stefan Bauer. Robustly disentangled causal mechanisms: Validating deep representations for interventional robustness. In *International Conference on Machine Learning*, 2019.

[75] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

[76] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *European conference on computer vision*, 2020.

[77] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.

[78] Yuandong Tian, Xinlei Chen, and Surya Ganguli. Understanding self-supervised learning dynamics without contrastive pairs. *arXiv preprint arXiv:2102.06810*, 2021.

[79] Luan Tran, Xi Yin, and Xiaoming Liu. Disentangled representation learning gan for pose-invariant face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[80] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[81] Sjoerd Van Steenkiste, Francesco Locatello, Jürgen Schmidhuber, and Olivier Bachem. Are disentangled representations helpful for abstract visual reasoning? In *Advances in neural information processing systems*, 2019.

[82] Xudong Wang, Ziwei Liu, and Stella X Yu. Unsupervised feature learning by cross-level discrimination between instances and groups. *arXiv preprint arXiv:2008.03813*, 2020.

[83] 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*, pages 3733–3742, 2018.

[84] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, 2010.

[85] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[86] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6210–6219, 2019.

[87] Zhongqi Yue, Tan Wang, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Counterfactual zero-shot and open-set visual recognition. In *CVPR*, 2021.

[88] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, 2014.

[89] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[90] Yizhe Zhu, Martin Renqiang Min, Asim Kadav, and Hans Peter Graf. S3vae: Self-supervised sequential vae for representation disentanglement and data generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[91] Zhenyao Zhu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Multi-view perceptron: a deep model for learning face identity and view representations. *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, 2014.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] The theoretical justification can be found in Section 4 and the experiment results are included in Section 5.
   (b) Did you describe the limitations of your work? [Yes] See Section 6
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] We propose a practical self-supervised approach for representation disentanglement, which will help downstream tasks to remove bias, hence promoting fairness and transparency.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 4 with details in Appendix.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplementary materials.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See supplementary materials.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We reported the standard deviation for few-shot experiments in Appendix following the standard setting.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] See Appendix.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We included our code in the supplementary materials.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] We only used publicly available assets. Details in Appendix.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Appendix.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] No crowdsourcing or human subjects involved.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] No crowdsourcing or human subjects involved.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] No crowdsourcing or human subjects involved.