

---

# Robust Representation Learning via Perceptual Similarity Metrics

---

Saeid Asgari Taghanaki<sup>\*1</sup> Kristy Choi<sup>\*2</sup> Amir Khasahmadi<sup>1</sup> Anirudh Goyal<sup>3</sup>

## Abstract

A fundamental challenge in artificial intelligence is learning useful representations of data that yield good performance on a downstream task, without overfitting to spurious input features. Extracting such task-relevant predictive information is particularly difficult for real-world datasets. In this work, we propose Contrastive Input Morphing (CIM), a representation learning framework that learns *input-space transformations* of the data to mitigate the effect of irrelevant input features on downstream performance. Our method leverages a perceptual similarity metric via a triplet loss to ensure that the transformation preserves task-relevant information. Empirically, we demonstrate the efficacy of our approach on tasks which typically suffer from the presence of spurious correlations: classification with nuisance information, out-of-distribution generalization, and preservation of subgroup accuracies. We additionally show that CIM is complementary to other mutual information-based representation learning techniques, and demonstrate that it improves the performance of variational information bottleneck (VIB) when used together.

## 1. Introduction

At the heart of human intelligence is the property of *robust generalization* – given one or a handful of examples, we are typically able to learn a concept and apply it across a variety of tasks and conditions.

Modern machine learning aims to replicate this phenomenon with our artificial agents – one such way to do so is via representation learning, or extracting features from raw data that enable predictions with high accuracy (Hinton & Salakhutdinov, 2006; Vincent et al., 2010; Chen et al., 2016; Van

Den Oord et al., 2017; Oord et al., 2018). In particular, the recent successes of deep neural networks (Dean et al., 2012; LeCun et al., 2015) have been pivotal towards stepping closer to this goal. However, their rapidly growing size and large-scale training procedures, coupled with complex, high-dimensional data sources, pose significant challenges for learning models that perform well on a given task without overfitting to spurious input features (Zhang et al., 2016; Ilyas et al., 2019; Geirhos et al., 2020). As a result, trained networks have been shown to fail spectacularly on out-of-domain generalization tasks (Beery et al., 2018; Rosenfeld et al., 2018) and exhibit poor performance for rare subgroups present in data (Hashimoto et al., 2018; Goel et al., 2020), among others.

A wide range of methods tackle this problem, including regularization, data augmentation, leveraging causal explanations, and self-training (Srivastava et al., 2014; Chen et al., 2020b; Sagawa et al., 2019; Chen et al., 2020b; Gulrajani & Lopez-Paz, 2020). In particular, prior art places a heavy emphasis on *lossless access* to the input data during training, then constructing a high-level representation which extracts the necessary task relevant information. Yet it is reasonable to assume that in some cases, we desire access to only a *subset* of the input which is relevant to the task – for example, the background color in an image of a “7” is unnecessary for identifying its digit class. The fundamental challenge, then, is discerning which parts of the input are relevant without requiring access to *privileged information* (e.g. the nature of the downstream task) at training time.

Our approach, Contrastive Input Morphing (CIM), leverages labeled supervision to *learn lossy input-space transformations* of the data that mitigate the effect of irrelevant input features on downstream predictive performance. The key workhorse of CIM is an auxiliary network called the Transformation Network (TN). Drawing inspiration from the “robustness” of the human visual system (Geirhos et al., 2020), perceptual similarity metrics (Wang et al., 2004; Zhang et al., 2018), and metric learning (Goldberger et al., 2004; Weinberger & Saul, 2009; Schroff et al., 2015; Koch, 2015), the TN is trained via a triplet loss that computes the perceptual similarity between sets of transformed inputs, examples from the same class as the input (positive examples), and those from competing classes (negative examples). This measure of similarity is captured by the structural similarity

<sup>\*</sup>Equal contribution <sup>1</sup>Autodesk AI Lab <sup>2</sup>Computer Science, Stanford University <sup>3</sup>Mila, Université de Montréal. Correspondence to: Saeid Asgari Taghanaki <saeid.asgari.taghanaki@autodesk.com>.

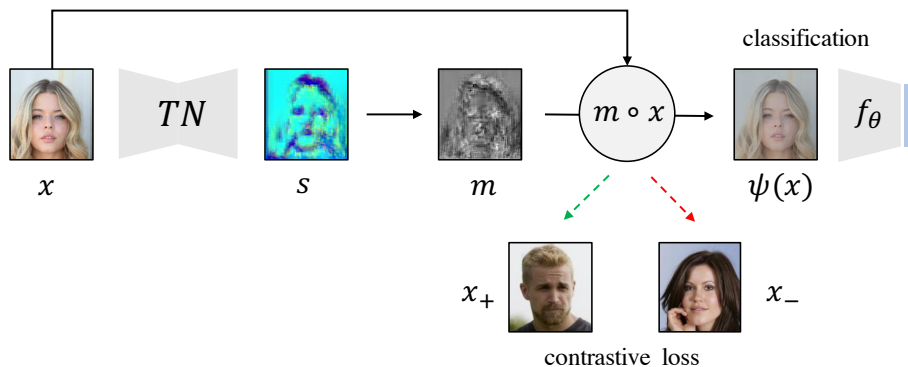


Figure 1. An end-to-end flowchart for the CIM training procedure. The Transformation Network (TN) maps an input image  $x$  into a weighting matrix  $m$ , then uses a triplet loss over the modified input using  $m$ , positive samples, and negative samples to facilitate learning task-relevant representations. The final transformed image  $\psi(x) = m \circ x$  is used to train the downstream classifier  $f_\theta$ .

metric, or SSIM (Wang et al., 2004) – a metric developed to extract features of an image that are striking to human perception. Intuitively, this objective uses the shared information from competing classes as a proxy for spurious correlations in the data, while leveraging the shared information within the same class as a heuristic for task-relevancy (Khosla et al., 2020).

The framework for CIM is quite general: it can be used as a plug-in module for training any downstream classifier, and we demonstrate a particular instance of its compatibility with variational information bottleneck (VIB) (Alemi et al., 2016), a mutual information (MI)-based representation learning technique. We emphasize that our method does not assume access to the exact nature of the downstream task, such as attribute labels for rare subgroups.

A flowchart of the CIM training procedure can be found in Figure 1.

Empirically, we evaluate our method on five different datasets under three settings that suffer from spurious correlations: classification with nuisance background information, out-of-domain (OOD) generalization, and improving accuracy uniformly across subgroups. In the first task, we show that when CIM is used with VIB (CIM+VIB), it outperforms ERM on colored MNIST and improves over the ResNet-50 baseline on the Background Challenge (Xiao et al., 2020). Similarly, CIM+VIB outperforms relevant baselines using ResNet-18 on the VLCS dataset (Torralba & Efros, 2011) for OOD generalization. For subgroup accuracies, our method outperforms unsupervised methods on CelebA (Liu et al., 2015) and the Waterbirds dataset (Wah et al., 2011) in terms of worst-group accuracy.

To the best of our knowledge, this work is the first to explore SSIM in a contrastive learning setup and to demonstrate its usefulness for learning robust representations on tasks which suffer from spurious correlations, unlike previous works

which were limited to image classification or adversarial robustness (Snell et al., 2017; Abobakr et al., 2019). In summary, our contributions in this work are as follows:

1. We propose CIM, a method demonstrating that lossy access to input data helps extract good task-relevant representations.
2. We show that CIM is complementary to existing methods, as the learned transformations can be leveraged by other MI-based representation learning techniques such as VIB.
3. We empirically verify the robustness of the learned representations to spurious correlations in the input features on a variety of tasks (Section 4).

## 2. Preliminaries

### 2.1. Notation and Problem Setup

We consider the supervised learning setting with inputs  $x \in \mathcal{X} \subseteq \mathbb{R}^d$  and corresponding labels  $y \in \mathcal{Y} = \{1, \dots, k\}$ . We assume access to samples  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$  drawn from an underlying (unknown) distribution  $p_{\text{data}}(x, y)$ , and use capital letters to denote random variables, e.g.  $X$  and  $Y$ . We use  $P(X, Y)$  to denote their joint distribution as well as  $P(\cdot)$  for the marginal distribution.

Our goal is to learn a classifier  $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ , where  $f_\theta \in \Theta$  achieves low prediction error according to some loss function  $\ell : \Theta \times (\mathcal{X} \times \mathcal{Y}) \rightarrow \mathbb{R}$ . Specifically, we minimize:

$$\mathcal{L}_{\text{sup}}(\theta) = \mathbb{E}_{x, y \sim p_{\text{data}}(x, y)}[\ell(f_\theta(x), y)] \approx \sum_{i=1}^n \ell(f_\theta(x_i), y_i) \quad (1)$$

In addition to good classification performance, we aim to learn representations of the data, which: (a) are highly pre-

dictive of the downstream task; and (b) do not rely on spurious input features. That is, the learned representations should be *task-relevant*.

## 2.2. Metric Learning

Metric learning (Goldberger et al., 2004) refers to a family of methods which learn a notion of similarity (the *metric* of interest) between sets of inputs to extract meaningful representations from data. Although several variations of the method exist, they all share a unifying principle in which examples from the same group are closer together in feature space, while those from opposing groups are further apart (Davis et al., 2007; Weinberger & Saul, 2009; Schroff et al., 2015; Koch, 2015; Hoffer & Ailon, 2015). Such methods typically operate over a triplet or contrastive loss (Khosla et al., 2020) which allows for gradient-based learning of an appropriate distance-based measure between the examples.

## 2.3. Structural Similarity Metrics

The key ingredient for training the TN is the SSIM metric (Wang et al., 2004), though we leverage its multi-scale variant (MS-SSIM) (Wang et al., 2003) for our experiments. Given a pair of images, SSIM metrics compute local, pixel-wise statistics across their luminance ( $l$ ), contrast ( $c$ ), and structure ( $s$ ) to assign a score for their perceptual image quality. Such comparison functions capture features that the human visual system are sensitive to (Hore & Ziou, 2010), making SSIM a more desirable candidate for comparing images in pixel space relative to others such as the  $\ell_2$  distance.

More concretely, for a pair of images  $x$  and  $y$ , we denote the mean pixel intensity of  $x$  as  $\mu_x$ , the standard deviation of its pixel intensity as  $\sigma_x$ , and the covariance between the pixels in  $x$  and  $y$  as  $\sigma_{xy}$ . The respective quantities for the second image  $y$  ( $\mu_y$  and  $\sigma_y$ ) are defined analogously. Then, the luminance, contrast, and structure are given by:

$$l(x, y) = \frac{2\mu_x\mu_y}{\mu_x^2 + \mu_y^2} \quad (2)$$

$$c(x, y) = \frac{2\sigma_x\sigma_y}{\sigma_x^2 + \sigma_y^2} \quad (3)$$

$$s(x, y) = \frac{\sigma_{xy}}{\sigma_x\sigma_y} \quad (4)$$

MS-SSIM (shortened to MS) considers multiple scales  $M$  by relaxing the assumption of a fixed image sampling density. Specifically, we compute the MS-SSIM metric as:

$$\text{MS}(x, y) = l_M(x, y) \prod_{j=1}^M c_j(x, y) s_j(x, y) \quad (5)$$

where  $c_j$  and  $s_j$  denote the contrast and structure of images  $x$  and  $y$  at scale  $j$  respectively, and  $l_M$  denotes the luminance only at scale  $M$ .

## 2.4. Information Bottleneck

Another way to measure “task-relevance” in random variables is to consider the total amount of information that a compressed (stochastic) representation  $Z$  contains about the input variable  $X$  and the output variable  $Y$ . In particular, (variational) information bottleneck (IB) (Tishby et al., 2000; Chechik et al., 2005; Alemi et al., 2016) is a framework which utilizes MI to quantify this dependence via the following objective:

$$\min_{P(Z|X)} I(X; Z) - \beta I(Z; Y) \quad (6)$$

where  $\beta > 0$  controls the importance of obtaining good performance on the downstream task. Given two random variables  $X$  and  $Z$ ,  $I(X; Z)$  is computed as

$$D_{\text{KL}}(P(X, Z) || P(X)P(Z)) = \mathbb{E}_{P(X, Z)} \left[ \log \frac{P(X, Z)}{P(X)P(Z)} \right] \quad (7)$$

We conjecture that because MS-SSIM is well-correlated with MI (Belghazi et al., 2018), inputs transformed by CIM should help MI-based representation learning methods such as VIB learn more task-relevant features; we further explore and empirically verify this hypothesis in Section 4.

## 3. Contrastive Input Morphing

We propose to approximate the information content between task-relevant and irrelevant features via local, pixel-level image statistics learned through a triplet loss. Our goal is to leverage both these statistics and labeled examples to *learn* desirable input-space transformations of the data for improved performance on various downstream tasks.

### 3.1. Transformation Network Training Procedure

**Network Architecture and Input Transformation.** We utilize a convolutional autoencoder for the Transformation Network (TN) to learn the appropriate transformation for each data point. The TN, which is parameterized by  $\phi$ , takes in an image  $x \in \mathbb{R}^{H \times W \times C}$  and produces a weight matrix  $m \in \mathbb{R}^{H \times W \times 1}$  normalized by the sigmoid activation function, where  $H \times W$  denotes the height and width of the image, and  $C$  denotes the number of channels. We then use this weight matrix  $m$  to transform the input samples by composing it with the learned mask via element-wise multiplication, which gives us the final transformed image  $\psi(x) = m \odot x$ . The classifier  $f_\theta(\cdot)$  is trained via the usual cross entropy loss on  $\psi(x)$ .

**Sampling Positive and Negative Examples.** We note that there exist several strategies for sampling triplets in Eq. 8. In CIM, we independently sample one  $x_+$  and one  $x_-$  for each  $x$ : that is, for each minibatch of anchor points

$\{x_i\}_{i=1}^n$  of size  $n$ , we sample  $n$  distinct positive examples and  $n$  distinct negative examples during training (for a total minibatch size of  $3n$ ). We further explore the effect of the sampling procedure on CIM’s performance in Section 4.4.

**Triplet Loss.** The TN is trained via a triplet loss that operates over sets of three examples at a time:  $(x, x_+, x_-)$ , where given an anchor point  $x$ ,  $x_+$  is a positive example from the same class as  $x$ , and  $x_-$  is a negative example from a different class as  $x$ . To encourage the transformed input’s pixel-wise statistics to be more similar to those of the positive examples (while remaining more dissimilar from those of the negative examples), we apply the MS-SSIM metric from Eq. 5. Therefore, our triplet loss is defined as:

$$\mathcal{L}_{\text{con}}(\phi) = \min_{\phi} \text{MS}(\psi(x), x_+) - \text{MS}(\psi(x), x_-) \quad (8)$$

**Learning Objective.** Therefore, our final objective is:

$$\mathcal{L}_{\text{CIM}}(\phi, \theta) = \lambda \mathcal{L}_{\text{con}}(\phi) + \mathcal{L}_{\text{sup}}(\theta) \quad (9)$$

where  $\lambda > 0$  is a hyperparameter which controls the contribution of the triplet loss from Eq. 8, and  $\mathcal{L}_{\text{sup}}(\theta)$  is the standard cross entropy loss for multi-class classification using the classifier  $f_{\theta}$ . The parameters for the transformation network ( $\phi$ ) and the classifier ( $\theta$ ) are trained jointly.

### 3.2. Additional Variants of CIM

We further explore different variants of CIM based on the observation that there exist various strategies for measuring task-relevant information. For brevity, we report the results of the additional variants in Appendix B.

**CIM<sub>g</sub>.** In this setup, we draw inspiration from the neural style transfer literature (Gatys et al., 2015; Li et al., 2017b; Sastry & Oore, 2019) and operate with Gram matrices (inner products) of the triplets’ features. Specifically, we modify CIM’s loss such that it encourages the positive examples’ Gram matrices to move closer together in the embedding space to those of the input, while ensuring that the negative examples’ representations are further apart. Therefore the loss is calculated as:

$$\mathcal{L}_{\text{con}}(\phi) = \min_{\phi} \{SX_+^T - SX_-^T\} \quad (10)$$

where  $S$  (Figure 1) denotes the learned representation by TN which in our setup has the same dimensions as the input.

**CIM<sub>f</sub>.** To assess whether working in feature space would be more beneficial than working directly in the output space, we also encode the negative and positive samples using the TN in addition to the input.

We then create three transformation matrices ( $m_1, m_2$ , and  $m_3$ ) that we use to modify the input, negative example, and

positive example, respectively. These three modified triplets are used to compute the loss during training as in Eq. 8.

**CIM + VIB.** Finally, we evaluate CIM’s compatibility with VIB to demonstrate that our method can be coupled with any (supervised) MI-based representation learning technique. In the CIM+VIB approach, in addition to modifying the input with CIM, we regularize the final feature vector  $Z$  of the classifier (the layer before the softmax) with the VIB objective. Therefore, the final loss is given by the following:

$$\mathcal{L}_{\text{CIM+VIB}}(\phi, \theta) = \lambda \mathcal{L}_{\text{con}}(\phi) + \mathcal{L}_{\text{sup}}(\theta) + D_{\text{KL}}(Q(Z|X) || P(Z)) \quad (11)$$

where  $P(Z) \sim \mathcal{N}(0, I)$  denotes the prior distribution over the stochastic representations  $Z$  and  $Q(Z|X)$  denotes the approximate posterior over the latent variables. We refer the reader to Appendix A for more details on the VIB architecture and hyperparameters.

## 4. Experimental Results

For our experiments, we are interested in empirically investigating the following questions:

1. Are CIM’s learned representations robust to spurious correlations in the input features?
2. Does the input transformation learned by CIM improve domain generalization?
3. How well can CIM preserve classification accuracy across subgroups?

**Datasets:** We consider various datasets and tasks to test the effectiveness of our method. We first construct a colored variant of MNIST (LeCun, 1998) to demonstrate that CIM successfully ignores nuisance background information in a digit classification task, then further explore this finding on the Background Challenge (Xiao et al., 2020) – a more challenging dataset. Next, we evaluate CIM on the VLCS dataset (Torralba & Efros, 2011) to demonstrate that the input transformations help in learning representations that generalize to OOD distributions. Then, we study two benchmark datasets, CelebA (Liu et al., 2015) and Waterbirds (Wah et al., 2011; Zhou et al., 2017), to show that CIM preserves subgroup accuracies.

**Models:** We use different classifier architectures depending on the downstream task. While ResNet-50 is the default choice for most datasets, we use ResNet-18 for a fair comparison with existing OOD generalization benchmarks. For the Colored MNIST experiment, we use a simple 3-layered multi-layer perceptron (MLP). The three fully-connected layers are of size 1024, 512, and 256 with ReLU activations.

We also experiment with VIB (Alemi et al., 2016) as both a competing and complementary approach to CIM. We

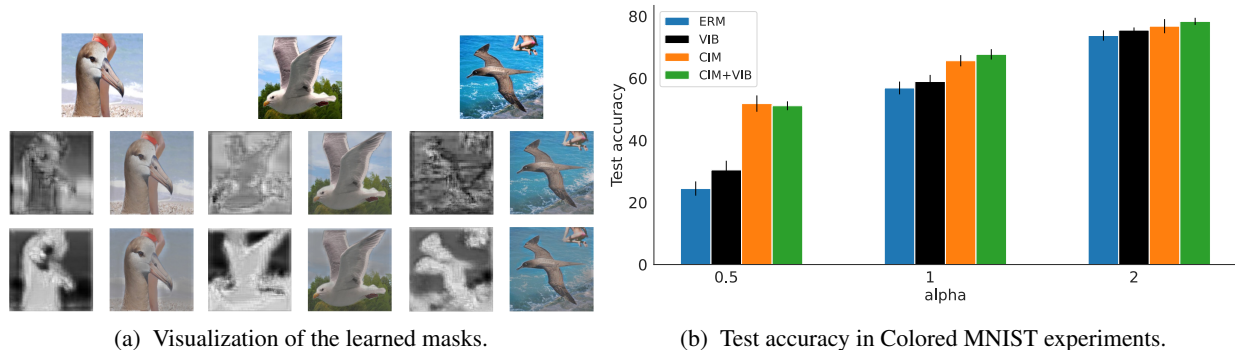


Figure 2. (a) Qualitative visualizations of the learned masks without and with CIM. The first, second, and third rows show the input, without CIM and with CIM weighting, respectively. The grayscale images are the learned per-pixel weights for each sample. (b) Colored MNIST classification results with different  $\alpha$  values, where  $\alpha$  denotes the proportion of digits colored with the opposite background color at test time. Both CIM and CIM+VIB significantly outperform ERM. Results are averaged over 3 runs.

use ResNet-50 as the encoder in VIB to be consistent with baselines and competing methods. We test three different settings for VIB, where we: (a) apply KL regularization on the last layer of the encoder.

(b) apply KL regularization on the last layer directly, but add 3 fully connected layers before calculating the cross-entropy loss; and (c) follow variant (a) after adding a fully connected layer after the last layer of the encoder. We found that (c) worked the best out of all three configurations, and refer the reader to Appendix A.1 for additional details.

As a strong contrastive learning baseline, we lightly modify the supervised contrastive learning (SCL) approach to make it comparable with our setup. That is, we train SCL end-to-end ( $SCL_{E2E}$ ) by learning the self-supervised representations jointly with the downstream classification task of interest. Therefore, the contrastive loss of  $SCL_{E2E}$  becomes:

$$\mathcal{L}_{\text{con}}(\phi) = \min_{\phi} \|E(x) - E(x_+)\|^2 - \|E(x) - E(x_-)\|^2 \quad (12)$$

where  $E(\cdot)$  is the last encoded feature vector of size 2048 in ResNet-50 and the model is parameterized by  $\phi$ . We refer the reader to Appendix A for additional experimental details and model hyperparameters.

#### 4.1. Classification with Nuisance Backgrounds

**Colored MNIST:** As a warm-up, we assess whether CIM can distinguish between two MNIST digit groups (0-4 and 5-9) in the presence of a spurious input feature (background color). We construct a dataset such that a classifier will achieve low accuracy by relying on background color. For a given value  $\alpha$ , we color  $\alpha\%$  of all digits labeled “0-4” and “5-9” in the training set with yellow and blue backgrounds, respectively. The remaining  $(100 - \alpha)\%$  of each of the digits’ backgrounds are then painted with the opposite color. We vary this proportion by  $\alpha = \{0.5, 1.0, 2.0\}$ .

At test time, we color all the digits labeled “0-4” in yellow, while coloring the “5-9” digits’ backgrounds in blue. Figure 2b shows that CIM+VIB is better able to utilize relevant information for the downstream task in comparison to ERM by 26.67%, 10.84%, and 4.55% on models trained with  $\alpha = \{0.5, 1.0, 2.0\}$  respectively. This suggests that the input transformations learned by CIM are indeed preserving task-relevant information that can be better leveraged by MI-based representation learning methods such as VIB.

**Background Challenge:** To evaluate whether the favorable results from Colored MNIST translate to a more challenging setup, we test CIM on the Background Challenge (Xiao et al., 2020). The Background Challenge is a public dataset consisting of ImageNet-9 (Deng et al., 2009) test sets with varying amounts of foreground and background signals, designed to measure the extent to which deep classifiers rely on spurious features for image classification. We compare CIM’s performance to relevant baselines in 2 test set configurations: *Mixed-random* (MR), where the foreground is overlaid onto a random background; and *Mixed-same* (MS), where the foreground is placed on a background from the same class. The background gap ( $BGP$ ), or the difference between these two scenarios, is a measure of average robustness to varying backgrounds from different image sources.

As shown in Table 2, CIM+VIB outperforms the baseline ResNet-50’s performance by 6.6% on *Mixed-random* (MR), 1.6% on the original test set, and 0.3% on *Mixed-same* (MS). More importantly, CIM+VIB reduces the background gap ( $BGP$ ) by 6.3% as compared to the baseline ResNet-50, and 1.4% as compared to VIB alone. These results demonstrate that our method indeed learns task-relevant representations without relying on nuisance sources of signal. We note that although  $SCL_{E2E}$  achieves slightly higher accuracy on the original test set, the background gap ( $BGP$ ) remains relatively large, which is undesirable. Additionally,

Method	Caltech	LabelMe	Pascal	Sun	Average
DeepC (Li et al., 2018b)	87.47	62.06	64.93	61.51	68.89
CIDDG (Li et al., 2018b)	88.83	63.06	64.38	62.10	69.59
CCSA (Motiian et al., 2017)	92.30	62.10	67.10	59.10	70.15
SLRC (Ding & Fu, 2017)	92.76	62.34	65.25	63.54	70.15
TF (Li et al., 2017a)	93.63	63.49	69.99	61.32	72.11
MMD-AAE (Li et al., 2018a)	94.40	62.60	67.70	64.40	72.28
D-SAM (D’Innocente & Caputo, 2018)	91.75	57.95	58.59	60.84	67.03
Shape Bias (Asadi et al., 2019)	98.11	63.61	74.33	67.11	75.79
VIB (Alemi et al., 2016)	97.44	66.41	73.29	68.49	76.41
SCL <sub>E2E</sub> (Ours)	95.56	66.72	73.16	65.10	75.14
CIM (Ours)	98.21	<b>67.80</b>	73.97	69.01	77.25
CIM + VIB (Ours)	<b>98.81</b>	66.49	<b>74.89</b>	<b>70.13</b>	<b>77.58</b>

Table 1. Multi-source domain generalization results (%) on the VLCS dataset with ResNet-18 as the base network for downstream classification. All reported numbers are averaged over three runs. CIM+VIB outperforms the state-of-the-art model (Asadi et al., 2019).

as shown in Figure 3b, SCL<sub>E2E</sub> is more computationally expensive as compared to other methods.

	OR (↑)	MS (↑)	MR (↑)	BGP (↓)
Res50 (Xiao et al., 2020)	96.3	89.9	75.6	14.3
VIB (Alemi et al., 2016)	97.4	89.9	80.5	9.4
CIM (Ours)	97.7	89.8	81.1	8.8
SCL <sub>E2E</sub> (Ours)	<b>98.2</b>	<b>90.7</b>	80.1	10.6
CIM + VIB (Ours)	97.9	90.2	<b>82.2</b>	<b>8.0</b>

Table 2. Results from the Background Challenge on ImageNet-9 using ResNet-50. Our method outperforms the relevant baselines across all three datasets. The difference between MS and MR is the background gap (BGP). OR corresponds to accuracy on the original test set.

## 4.2. Out-of-Domain Generalization

In this experiment, we evaluate CIM on OOD generalization performance using the VLCS benchmark dataset (Torralba & Efros, 2011). VLCS consists of images from five object categories shared by the PASCAL VOC 2007, LabelMe, Caltech, and Sun datasets, which are considered to be four separate domains. We follow the standard evaluation strategy used in (Carlucci et al., 2019), where we partition each domain into a train (70%) and test set (30%) by random selection from the overall dataset. We use ResNet-18 as the backbone to make a fair comparison with the state-of-the-art. As summarized in Table 7, CIM+VIB outperforms the state-of-the-art (Shape Bias (Asadi et al., 2019)) on each domain and by 1.79% on average, bolstering our claim that using a lossy transformation of the input is helpful for learning task-relevant representations that generalize across domains.

## 4.3. Preservation of Subgroup Performance

We investigate whether representations learned by CIM perform well on all subgroups on CelebA and Waterbirds datasets. Preserving good subgroup-level accuracy is challenging for naive ERM-based methods, given their tendency to latch onto spurious correlations (Kim et al., 2019; Arjovsky et al., 2019; Chen et al., 2020b). Most prior works leverage privileged information such as group labels to mitigate this effect (Ben-Tal et al., 2013; Vapnik & Izmailov, 2015; Sagawa et al., 2019; Goel et al., 2020; Xiao et al., 2020). As the TN is trained to capture task-relevant features and minimize nuisance correlations between classes, we hypothesize that CIM should perform well at the subgroup level *even without explicit subgroup label information*.

For a fair comparison, we use ResNet-50 as the backbone in all of our trained models. Table 3 shows that CIM+VIB outperforms unsupervised methods on CelebA and Waterbirds in terms of worst-group accuracy, while significantly improving over ERM by 42.49% and 17.23% on CelebA and Waterbirds datasets, respectively. We emphasize that the favorable performance of CIM+VIB is obtained *without using subgroup labels*, in contrast with supervised approaches. For additional results with different variants of our method, we refer the reader to Appendix B.

## 4.4. Ablation Studies

Next, we perform a series of ablation studies to assess the contributions of each component in CIM.

**Effect of the contrastive loss.** First, we investigate how much of the performance improvement in our method is due to the transformation learned via the *contrastive loss*, rather than a general attention-like mechanism that operates

Dataset	Method	Unsupervised (subgroup-level)	Worst group acc.	Average acc.
CelebA	GDRO (Sagawa et al., 2019)	✗	<b>88.30</b>	91.80
	ERM	✓	41.10	94.80
	Baseline (Ours)	✓	70.31	93.98
	SCL <sub>E2E</sub> (Ours)	✓	68.80	<b>95.80</b>
	VIB (Alemi et al., 2016)	✓	78.13	91.94
	CIM (Ours)	✓	81.25	89.24
	CIM + VIB (Ours)	✓	<u>83.59</u>	90.61
Waterbirds	GDRO (Sagawa et al., 2019)	✗	83.80	89.40
	CAMEL (Goel et al., 2020)	✗	<b>89.70</b>	90.90
	ERM	✓	60.00	<b>97.30</b>
	Baseline (Ours)	✓	62.19	96.42
	SCL <sub>E2E</sub> (Ours)	✓	64.10	96.50
	VIB (Alemi et al., 2016)	✓	75.31	95.39
	CIM (Ours)	✓	73.35	89.78
	CIM + VIB (Ours)	✓	<u>77.23</u>	95.60

Table 3. Average and worst-group accuracies for CelebA and Waterbird benchmark datasets. Methods without group-level supervision (✓) are preferable over those with group-level supervision (✗). CIMs + VIB outperforms unsupervised methods on both datasets, while achieving comparable performance against supervised approaches. Underline shows the best accuracy among the unsupervised methods.

directly over a single input (without any positive or negative examples). As shown in Figure 3a, we find that simply learning a reweighting matrix  $m$  via the TN with  $\lambda = 0$  in Eq. 9 leads to a significant performance degradation relative to CIM on the Waterbirds dataset. For a more in-depth qualitative analysis, we visualize the learned transformations both with and without CIM after they have been composed with the input. As shown in Figure 2a, we observe that the transformation learned by CIM places less emphasis on the task-irrelevant information (background).

**Effect of coupling VIB and CIM.** Next, we analyze how much of the performance improvement of CIM+VIB over relevant baselines is from the learned input transformation, rather than using VIB as a downstream feature extractor. As shown in Table 4, we observe that CIM significantly improves VIB’s performance (9.01%, 1.92%, 1.17% improvements on Colored MNIST, Waterbirds, and VLCS respectively), demonstrating that the learned transformation is indeed useful for better representation learning.

**Effect of positive and negative samples.** Additionally, we evaluate the effect of the positive and negative terms in our contrastive loss function on the Waterbirds dataset. Table 5 demonstrates that both the negative and positive samples contribute to improving the worst-group accuracies. In particular, the worst- and second worst-group accuracies improve by 5.07% and 5.48% respectively, relative to the model without the contrastive loss (i.e. only  $m$ ). We present additional results on sampling strategies for the positive and negative examples in Table 9 of the Appendix.

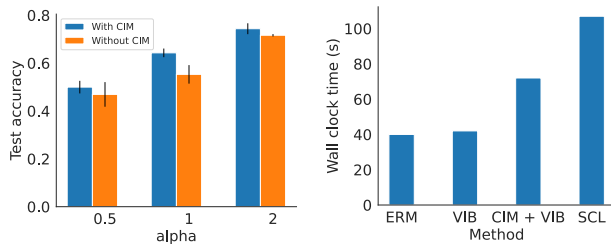
Method	Colored MNIST	Waterbirds	VLCS
Baseline	54.87	62.19	75.83
VIB	56.82	75.31	76.41
CIM	65.27	73.34	77.17
CIM+VIB	<b>65.83</b>	<b>77.23</b>	<b>77.58</b>

Table 4. Ablation study of CIM+VIB. Higher is better. We find that the learned input-space transformation helps VIB extract more predictive features for the downstream classification task of interest, as CIM+VIB outperformed all model configurations.

Method	Worst-group	2nd worst-group
Only $m$ (without CIM)	68.28	68.75
CIM (only $x_-$ )	69.29	71.25
CIM (only $x_+$ )	72.81	73.59
CIM (both $x_+$ and $x_-$ )	<b>73.35</b>	<b>73.75</b>

Table 5. Ablation study for the effect of positive and negative samples on the worst-group and second worst-group accuracy on Waterbirds dataset. The triplet loss, which leverages both positive and negative samples, outperformed the other configurations.

**Computational Overhead.** Finally, we compare the computational overhead of our approach relative to other baselines on the Waterbirds dataset in Figure 3b. Because CIM needs to calculate the triplet loss for every data point in a minibatch, our method is indeed slower than ERM or VIB.



(a) Effect of CIM on accuracy. (b) Wall clock time per epoch.

Figure 3. a) Ablation study investigating the effect of CIM, where we find that simply learning a reweighting matrix without a contrastive loss leads to worse performance. b) Training times per epoch on Waterbirds dataset. SCL is much slower than CIM+VIB, as it requires encoding all  $x$ ,  $x_-$ , and  $x_+$ , while CIM+VIB only needs to encode  $x$ .

However, we note that we are much faster than SCL, as CIM only requires encoding a single input  $x$  via the TN rather than SCL, which must encode all triplets  $x$ ,  $x_+$ , and  $x_-$ .

## 5. Related Work

**Contrastive representation learning.** There has been a flurry of recent work in contrastive methods for representation learning, which encourages an encoder network to map “positive” examples closer together in a latent embedding space while spreading the “negative” examples further apart (Oord et al., 2018; Hjelm et al., 2018; Wu et al., 2018; Tian et al., 2019; Arora et al., 2019; Chen et al., 2020a). Some representative approaches include triplet-based losses (Schroff et al., 2015; Koch, 2015) and variants of noise contrastive estimation (Gutmann & Hyvärinen, 2010). In particular, recent work (Tian et al., 2020; Wu et al., 2020) has shown that *minimizing* MI between views while maximizing predictive information of the representations with respect to the downstream task, leads to performance improvements, similar to IB (Chechik & Tishby, 2003). While most contrastive approaches are self-supervised, (Khosla et al., 2020) utilizes class labels as part of their learning procedure. We emphasize that CIM is not meant to be directly comparable to the aforementioned techniques, as our objective is to learn input transformations of the data that are task-relevant with *labeled supervision*, without relying on a two-stage pretraining approach.

**Robustness of representations.** Several works have considered the problem of learning relevant features that do not rely on spurious correlations with the predictive task (Heinze-Deml & Meinshausen, 2017; Sagawa et al., 2020; Chen et al., 2020b). Though (Wang et al., 2019) is similar in spirit to CIM, they utilize gray-level co-occurrence matrices as the spurious (textural) information of the input images, then regress out this information from the trained classifier’s output layer. Our method does not solely rely on textural

features and can learn any transformation of the input space that is relevant for the downstream task of interest. Although CIM also bears resemblance to InfoMask (Taghanaki et al., 2019), our method is not limited to attention maps. (Kim et al., 2019) uses an MI-based objective to minimize the effect of spurious features, while (Pensia et al., 2020) additionally incorporates regularization via Fisher information to enforce robustness of the features. In contrast, CIM uses an orthogonal approach to learn robust representations via the perceptual similarity metric in input space.

**SSIM-based loss functions.** SSIM and MS-SSIM (Wang et al., 2003; 2004) have seen a recent resurgence in neural network-based approaches. In particular, (Yang et al., 2020) adapted SSIM for single image dehazing and signal reconstruction, while (Lu, 2019) and (Snell et al., 2017) utilized the metric for image generation and downstream classification. Additionally, Abobakr et al. (2019) proposed a SSIM layer to make convolutional neural networks robust to noise and adversarial attacks. Though similar in spirit, CIM uses a triplet loss based on SSIM in order to learn input-space transformations of images for robust representations.

## 6. Conclusion

In this work, we considered the problem of extracting representations with *task-relevant* information from high-dimensional data. We introduced a new framework, CIM, which learns input-space transformations of the data via a triplet loss to mitigate the effect of irrelevant input features on downstream classification performance. Through experiments on (1) classification with nuisance background information; (2) out-of-domain generalization; and (3) preservation of subgroup performance, which typically suffer from the presence of spurious correlations in the data, we showed that CIM outperforms most relevant baselines. Additionally, we demonstrated that CIM is complementary to other mutual information-based representation learning frameworks such as VIB. A limitation of our work is precisely the need for labeled supervision, which may be difficult or prohibitively expensive to obtain during training. For future work, it would be interesting to test different types of distance metrics for the triplet loss, to explore whether CIM can be used as an effective way to *learn* views for unsupervised contrastive learning, and to investigate label-free approaches for learning the input transformations.

### ACKNOWLEDGEMENTS

We are thankful to Pang Wei Koh, Shiori Sagawa, and Aditya Sanghi for helpful discussions. KC is supported by the NSF GRFP, Stanford Graduate Fellowship, and Two Sigma Diversity PhD Fellowship.



## References

- Abobakr, A., Hossny, M., and Nahavandi, S. Ssimlayer: towards robust deep representation learning via nonlinear structural similarity. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pp. 1234–1238. IEEE, 2019.
- Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.
- Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.
- Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., and Saunshi, N. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.
- Asadi, N., Hosseinzadeh, M., and Eftekhari, M. Towards shape biased unsupervised representation learning for domain generalization. *arXiv preprint arXiv:1909.08245*, 2019.
- Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 456–473, 2018.
- Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, R. D. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.
- Ben-Tal, A., Den Hertog, D., De Waegenare, A., Melenberg, B., and Rennen, G. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.
- Carlucci, F. M., D’Innocente, A., Bucci, S., Caputo, B., and Tommasi, T. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2229–2238, 2019.
- Chechik, G. and Tishby, N. Extracting relevant structures with side information. In *Advances in Neural Information Processing Systems*, pp. 881–888, 2003.
- Chechik, G., Globerson, A., Tishby, N., and Weiss, Y. Information bottleneck for gaussian variables. *Journal of machine learning research*, 6(Jan):165–188, 2005.
- Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020a.
- Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I., and Abbeel, P. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pp. 2172–2180, 2016.
- Chen, Y., Wei, C., Kumar, A., and Ma, T. Self-training avoids using spurious features under domain shift. *arXiv preprint arXiv:2006.10032*, 2020b.
- Davis, J. V., Kulis, B., Jain, P., Sra, S., and Dhillon, I. S. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pp. 209–216, 2007.
- Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, M., Ranzato, M., Senior, A., Tucker, P., Yang, K., et al. Large scale distributed deep networks. In *Advances in neural information processing systems*, pp. 1223–1231, 2012.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.
- Ding, Z. and Fu, Y. Deep domain generalization with structured low-rank constraint. *IEEE Transactions on Image Processing*, 27(1):304–313, 2017.
- D’Innocente, A. and Caputo, B. Domain generalization with domain-specific aggregation modules. In *German Conference on Pattern Recognition*, pp. 187–198. Springer, 2018.
- Gatys, L. A., Ecker, A. S., and Bethge, M. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.
- Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. Shortcut learning in deep neural networks. *arXiv preprint arXiv:2004.07780*, 2020.
- Goel, K., Gu, A., Li, Y., and Ré, C. Model patching: Closing the subgroup performance gap with data augmentation. *arXiv preprint arXiv:2008.06775*, 2020.
- Goldberger, J., Hinton, G. E., Roweis, S., and Salakhutdinov, R. R. Neighbourhood components analysis. *Advances in neural information processing systems*, 17:513–520, 2004.
- Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.
- Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized

- statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 297–304, 2010.
- Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. *arXiv preprint arXiv:1806.08010*, 2018.
- Heinze-Deml, C. and Meinshausen, N. Conditional variance penalties and domain shift robustness. *arXiv preprint arXiv:1710.11469*, 2017.
- Hinton, G. E. and Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *science*, 313 (5786):504–507, 2006.
- Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.
- Hoffer, E. and Ailon, N. Deep metric learning using triplet network. In *International workshop on similarity-based pattern recognition*, pp. 84–92. Springer, 2015.
- Hore, A. and Ziou, D. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pp. 2366–2369. IEEE, 2010.
- Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pp. 125–136, 2019.
- Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.
- Kim, B., Kim, H., Kim, K., Kim, S., and Kim, J. Learning not to learn: Training deep neural networks with biased data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9012–9020, 2019.
- Koch, G. Siamese neural networks for one-shot image recognition. 2015.
- LeCun, Y. The mnist database of handwritten digits. <http://yann.lecun.com/exdb/mnist/>, 1998.
- LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436–444, 2015.
- Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5542–5550, 2017a.
- Li, H., Jialin Pan, S., Wang, S., and Kot, A. C. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5400–5409, 2018a.
- Li, Y., Wang, N., Liu, J., and Hou, X. Demystifying neural style transfer. *arXiv preprint arXiv:1701.01036*, 2017b.
- Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., and Tao, D. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 624–639, 2018b.
- Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.
- Lu, Y. The level weighted structural similarity loss: A step away from mse. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 9989–9990, 2019.
- Motiiian, S., Piccirilli, M., Adjeroh, D. A., and Doretto, G. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5715–5725, 2017.
- Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- Pensia, A., Jog, V., and Loh, P.-L. Extracting robust and accurate features via a robust information bottleneck. *IEEE Journal on Selected Areas in Information Theory*, 2020.
- Rosenfeld, A., Zemel, R., and Tsotsos, J. K. The elephant in the room. *arXiv preprint arXiv:1808.03305*, 2018.
- Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.
- Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. An investigation of why overparameterization exacerbates spurious correlations. *arXiv preprint arXiv:2005.04345*, 2020.
- Sastry, C. S. and Oore, S. Detecting out-of-distribution examples with in-distribution examples and gram matrices. *arXiv preprint arXiv:1912.12510*, 2019.
- Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

- Snell, J., Ridgeway, K., Liao, R., Roads, B. D., Mozer, M. C., and Zemel, R. S. Learning to generate images with perceptual similarity metrics. In *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 4277–4281. IEEE, 2017.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.
- Taghanaki, S. A., Havaei, M., Berthier, T., Dutil, F., Di Jorio, L., Hamarneh, G., and Bengio, Y. Infomask: Masked variational latent representation to localize chest disease. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 739–747. Springer, 2019.
- Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.
- Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.
- Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.
- Torralba, A. and Efros, A. A. Unbiased look at dataset bias. In *CVPR 2011*, pp. 1521–1528. IEEE, 2011.
- Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, pp. 6306–6315, 2017.
- Vapnik, V. and Izmailov, R. Learning using privileged information: similarity control and knowledge transfer. *J. Mach. Learn. Res.*, 16(1):2023–2049, 2015.
- Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.-A., and Bottou, L. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010.
- Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. 2011.
- Wang, H., He, Z., Lipton, Z. C., and Xing, E. P. Learning robust representations by projecting superficial statistics out. *arXiv preprint arXiv:1903.06256*, 2019.
- Wang, Z., Simoncelli, E. P., and Bovik, A. C. Multiscale structural similarity for image quality assessment. In *The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pp. 1398–1402. Ieee, 2003.
- Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
- Weinberger, K. Q. and Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *Journal of machine learning research*, 10(2), 2009.
- Wu, M., Zhuang, C., Mosse, M., Yamins, D., and Goodman, N. On mutual information in contrastive learning for visual representations. *arXiv preprint arXiv:2005.13149*, 2020.
- Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.
- Xiao, K., Engstrom, L., Ilyas, A., and Madry, A. Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*, 2020.
- Yang, H.-H., Yang, C.-H. H., and Tsai, Y.-C. J. Y-net: Multi-scale feature aggregation network with wavelet structure similarity loss function for single image dehazing. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2628–2632. IEEE, 2020.
- Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.
- Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.
- Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., and Torralba, A. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

## Appendix

### A. Additional Experimental Details

#### A.1. ARCHITECTURES

**TN Architectures.** In Figure 4, we show the detailed TN architectures used for the Colored MNIST experiment, as well as for the other models. Architecture details for the downstream classifiers that are specific to each task can be found in Section 4.

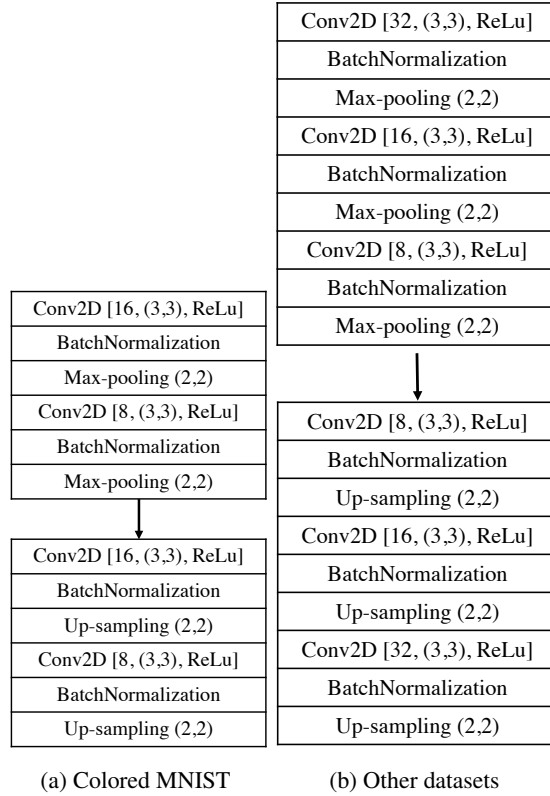


Figure 4. a) Architecture of the TN used for all experiments in the paper.

**VIB Architectures.** We tested 3 different approaches where we: (a) apply KL regularization on the last layer of the encoder which is of size (1, 2048) and then compute the cross entropy loss; (b) apply KL regularization on the last layer directly, but add 3 fully connected layers of (1024, ReLU, batch normalization), (512, ReLU, batch normalization), and (256, ReLU), before calculating the cross-entropy loss; and (c) follow variant (a) after adding a fully connected layer of size 512 after the last layer of the encoder. We found that (c) yielded the best performance.

A.2. HYPERPARAMETER CONFIGURATIONS

For all the methods including the baselines we have tuned the hyper-parameters such as the optimizer (SGD and Adam), learning rate (0.01, 0.001, 0.0001), and batch size (16, 32, 64). The input size for all the experiments was set to  $224 \times 224 \times 3$  except for Colored MNIST, which was  $28 \times 28 \times 3$ . For Colored MNIST and CelebA, we found that the Adam optimizer with a learning rate of 0.0001 and batch size of 64 worked best. For the Waterbirds and VLCS datasets, we found that the SGD optimizer with learning rate of 0.001 and momentum of 0.9 with batch size of 32 yielded the best performance. For the background challenge, we set the optimizer to SGD with learning rate of 0.001 and batch size to 32.

For CIM and VIB-based approaches, we tested  $\beta$  (the hyperparameter for VIB) and  $\lambda$  (for the contrastive loss) with a range of values within [1.0, 0.00001]. In Table 6, we report the best-performing hyperparameters for each method.

Dataset	Method	VIB's $\beta$	CIM's $\lambda$
CelebA	CIM	-	0.00001
	VIB	0.1	-
	CIM+VIB	0.001	0.0001
Waterbirds	CIM	-	0.0001
	VIB	0.001	-
	CIM+VIB	0.001	0.0001
Background challenge	CIM+VIB	0.001	0.0001
Color MNIST	CIM	-	0.00001
	VIB	0.00001	-
	CIM+VIB	0.00001	0.00001

Table 6.  $\beta$  and  $\lambda$  values for CIM, VIB, and CIM+VIB.

B. Additional Experimental Results

B.1. OUT-OF-DOMAIN GENERALIZATION

We present additional results from Section 4.2 with standard errors calculated over 3 runs for our methods. We note that the results from other papers did not include replicates.

Method	Caltech	LabelMe	Pascal	Sun	Average
DeepC (Li et al., 2018b)	87.47	62.06	64.93	61.51	68.89
CIDDG (Li et al., 2018b)	88.83	63.06	64.38	62.10	69.59
CCSA (Motiian et al., 2017)	92.30	62.10	67.10	59.10	70.15
SLRC (Ding & Fu, 2017)	92.76	62.34	65.25	63.54	70.15
TF (Li et al., 2017a)	93.63	63.49	69.99	61.32	72.11
MMD-AAE (Li et al., 2018a)	94.40	62.60	67.70	64.40	72.28
D-SAM (D’Innocente & Caputo, 2018)	91.75	57.95	58.59	60.84	67.03
Shape Bias (Asadi et al., 2019)	98.11	63.61	74.33	67.11	75.79
VIB (Alemi et al., 2016)	97.44±0.143	66.41±0.045	73.29±0.040	68.49±0.150	76.41±0.095
SCL <sub>E2E</sub> (Ours)	95.56±0.141	66.72±0.043	73.16±0.053	65.10±0.071	75.14±0.077
CIM (Ours)	98.21±0.004	<b>67.80±0.010</b>	73.97±0.003	69.01±0.003	77.25±0.005
CIM + VIB (Ours)	<b>98.81±0.003</b>	66.49±0.004	<b>74.89±0.007</b>	<b>70.13±0.008</b>	<b>77.58±0.006</b>

Table 7. Multi-source domain generalization results (%) on the VLCS dataset with ResNet-18 as the base network for downstream classification. All reported numbers are averaged over three runs. CIM+VIB outperforms the state-of-the-art model (Asadi et al., 2019).

B.2. ADDITIONAL VARIANTS OF CIM ON SUBGROUP PERFORMANCE

We present additional results on the other variants of CIM as mentioned in Section 3.2: CIM<sub>f</sub> and CIM<sub>g</sub>. We also experiment with a naive metric based on the  $\ell_2$  distance between two images in pixel space, which we name CIM<sub>mse</sub>, and find that it

leads to worse performance. We report the results of both these variants on Waterbirds and CelebA datasets in Table 8. We find that encoding only the input (i.e. CIM+VIB) to calculate the structural triplet loss outperforms both CIM<sub>g</sub> and CIM<sub>f</sub>.

Dataset	Method	unsupervised (group-level),	Worst-group acc.	Average acc.
CelebA	GDRO (Sagawa et al., 2019)	✗	<b>88.30</b>	91.80
	ERM (Sagawa et al., 2019)	✓	41.10	94.80
	Our baseline	✓	70.31	93.98
	VIB (Alemi et al., 2016)	✓	78.13	91.94
	SCL <sub>E2E</sub> (Ours)	✓	68.80	<b>95.80</b>
	CIM <sub>f</sub> + VIB (Ours)	✓	80.87	88.24
	CIM <sub>g</sub> + VIB (Ours)	✓	82.03	91.27
	CIM + VIB (Ours)	✓	<u>83.59</u>	90.61
Waterbirds	GDRO (Sagawa et al., 2019)	✗	83.80	89.40
	CAMEL (Goel et al., 2020)	✗	<b>89.70</b>	90.90
	ERM (Sagawa et al., 2019)	✓	60.00	<b>97.30</b>
	Our baseline	✓	62.19	96.42
	VIB (Alemi et al., 2016)	✓	75.31	95.39
	SCL <sub>E2E</sub> (Ours)	✓	64.10	96.50
	CIM <sub>f</sub> + VIB (Ours)	✓	76.65	95.31
	CIM <sub>g</sub> + VIB (Ours)	✓	73.79	94.91
CIM + VIB (Ours)	✓	<u>77.23</u>	95.60	

Table 8. Average and worst-group accuracies for CelebA and Waterbird benchmark datasets. Methods without group-level supervision (✓) are preferable over those with group-level supervision (✗). CIMs + VIB outperforms unsupervised methods on both datasets, while achieving comparable performance against supervised approaches. Underline shows the best accuracy among the unsupervised methods.

### B.3. CONNECTION TO WEIGHTED ATTENTION AND SALIENCY MAPS

Although our method bears resemblance to methods with (weighted) attention and saliency, we demonstrate that the TN in CIM is in fact learning a more sophisticated transformation. We experimented with two approaches: (1) learning an attention-like map (Only<sub>m</sub>) via the TN; and (2) GradCAM (Selvaraju et. al 2016) for saliency detection (Saliency) to see which method would yield perform more favorably on the downstream classification task.

For the GradCAM baseline, we first trained a ResNet-50 classifier, computed saliency maps using GradCAM after freezing the model, composed the saliency map with the inputs, and trained a new ResNet-50 classifier with the modified inputs (without nuisance background information). As in Table 9, we find that this saliency method (Saliency) improves over the baseline, but does not outperform either the learned attention map (Only<sub>m</sub>) or CIM. Figure 5 shows that this is because the saliency detection algorithm may incorrectly mask out a task-relevant region in the original image. By learning this transformation *jointly* with our classifier as in CIM, we are able to improve performance.

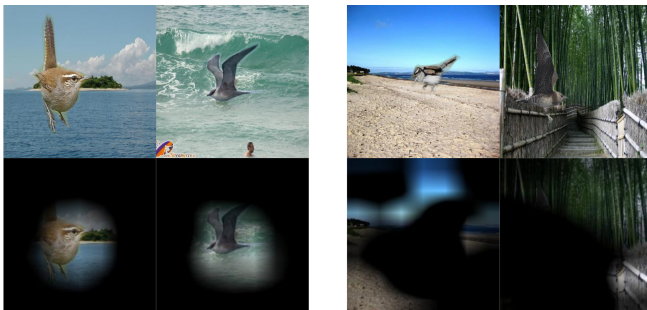


Figure 5. Learned saliency maps from the GradCAM baseline.

B.4. ALTERNATIVE SAMPLING STRATEGIES FOR CIM

Inspired by the contrastive learning literature which explores the impact of various positive and negative sampling strategies, we experimented with two alternative sampling approaches: `neg++` and `pos++` (sampling more negative/positive examples per batch respectively). In `neg++` we used 3 negative examples for each positive example, and vice versa. As shown in Table 9, we did not find a clear improvement.

Method	CelebA		Waterbirds	
	Worst-group	Average	Worst-group	Average
Baseline	70.31	<b>93.91</b>	62.19	<b>96.42</b>
Only $m$	82.81	90.06	75.31	95.14
Saliency	75.77	90.90	67.19	92.70
CIM <sub>mse</sub>	75.78	91.55	69.69	95.46
CIM <code>neg++</code>	78.12	91.90	70.62	96.10
CIM <code>pos++</code>	78.13	92.89	76.71	95.60
CIM	<b>83.59</b>	90.61	<b>77.23</b>	95.60

Table 9. Results for different baselines and sampling strategies.