Overcoming Shortcut Learning in a Target Domain by Generalizing Basic Visual Factors from a Source Domain

Piyapat Saranrittichai[1,2], Chaithanya Kumar Mummadi[1,2], Claudia Blaiotta[1], Mauricio Munoz[1], and Volker Fischer[1]

[1] Bosch Center for Artificial Intelligence
[2] University of Freiburg

Abstract. Shortcut learning occurs when a deep neural network overly relies on spurious correlations in the training dataset in order to solve downstream tasks. Prior works have shown how this impairs the compositional generalization capability of deep learning models. To address this problem, we propose a novel approach to mitigate shortcut learning in uncontrolled target domains. Our approach extends the training set with an additional dataset (the source domain), which is specifically designed to facilitate learning independent representations of basic visual factors. We benchmark our idea on synthetic target domains where we explicitly control shortcut opportunities as well as real-world target domains. Furthermore, we analyze the effect of different specifications of the source domain and the network architecture on compositional generalization. Our main finding is that leveraging data from a source domain is an effective way to mitigate shortcut learning. By promoting independence across different factors of variation in the learned representations, networks can learn to consider only predictive factors and ignore potential shortcut factors during inference.

1 Introduction

Humans seamlessly categorize objects in the real world by their basic visual factors (e.g., shape, texture, color). For example, we perceive a red fire truck as a object with *red* color and the shape of a *fire truck*. We are able to do this because we have learned abstract concepts of shape and color, which easily generalize to common objects including unseen, out-of-distribution (OOD) data [28]. Unlike humans, modern deep neural networks (DNNs) do not possess a generalized notion of basic visual factors and therefore tend to perform poorly on OOD data. This is especially true when networks exploit shortcuts inherent in the data, i.e., when they excessively rely on visual factors that are easy to learn and predictive on in-distribution data but fail to generalize on OOD samples [8,27]. For example, DNNs trained on a dataset in which all fire trucks are red might use color as a shortcut to recognize fire trucks while ignoring the more semantically meaningful attribute of shape. As a result, such a network may misclassify a yellow fire truck as a school bus.
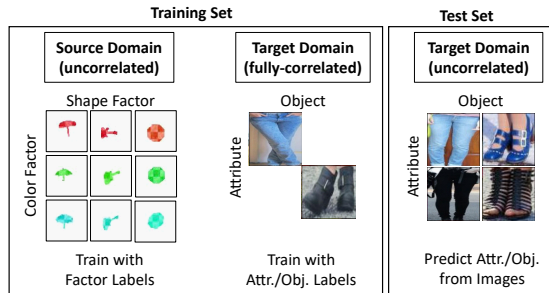
Fig. 1: Our task is to predict attribute/object labels in the target domain whose correlated training labels induce high shortcut opportunities. To improve generalization, we propose to augment the training set with a cheap source domain to learn representations of visual factors. In this given example, attribute and object in the target domain associate to color and shape factors respectively.

In this work, we study the ability of DNNs to recognize OOD samples in the context of compositional generalization, i.e., the ability to understand unseen combinations of known elements. While a few approaches have already been proposed to improve generalization in compositional zero-shot recognition tasks [22,2,23], the performance of all these methods heavily depends on the number of combinations observed during training. This is due to the fact that a low number of seen combinations generates more shortcut opportunities [8]. A naïve remedy to shortcut learning is to collect missing combinations but this could be costly as some combinations can be rare in practice. In this work, we aim to investigate the underpinning of compositional generalization in DNNs and hence focus on the challenging and unexplored regime in which different visual attributes are fully-correlated in the training data (e.g. where one attribute such as color is a deterministic function of another one, such as shape). In the rest of this paper, we will refer to the domain in which we aim to achieve compositional generalization as the *target domain*.

We propose a novel approach to mitigate shortcut learning in such a target domain by introducing a well-controlled *source domain* in the form of an additional training dataset, which can be cheaply generated. This source dataset is specifically designed to facilitate learning of basic visual factors by constraining them to be uncorrelated (all factor combinations appear uniformly during training). With this source dataset, we aim to learn independent representations of generic visual factors in order to improve OOD generalization in the target domain. Overview of our setup is shown in Figure 1. We show practical benefits of such a simple and easy-to-generate source domain for improving the performance across multiple, more complex, target domains. Our approach provides a low-cost and effective strategy to improve compositional generalization.

Considering our source dataset, we employ DiagVib [6], a framework to generate datasets whose visual factors (i.e., shape, color, texture, lightness and back-

ground) can be customized (see Figure 3). These factors are suitable for our purpose for the following reasons. Firstly, they are generic for common objects. Additionally, as suggested by previous studies [9,27], some visual factors (e.g., background and texture) are likely to introduce spurious correlations while other factors (e.g., shape) are more robust when used for recognizing objects. By including these key factors in the source dataset, our models can learn to focus more on important factors, ignore potential shortcut factors and thus improve the models' robustness to shortcuts.

Our contributions are as follows: firstly, we introduce a novel framework to improve compositional generalization in fully-correlated target domains. This is achieved by leveraging a source domain in order to alleviate shortcut learning. Secondly, we propose a simple network architecture exploiting the source domain to learn independent representations of visual factors. We also show that, if the target domain is not strictly fully-correlated, we can require less *a priori* knowledge for our approach. Lastly, we perform ablation studies to investigate effects of different source dataset configurations. Our main finding is that the source domain can act as a regularizer that encourages the internal representations of basic visual factors in DNNs to be less entangled. We show that this consistently improves compositional generalization across different target domains.

2   Related Works

Compositional Generalization (CG) We study CG in the context of compositional zero-shot learning, where the goal is to recognize images of unseen attribute-object combinations given only some combinations seen during training. A simple baseline VisProd [19,22] uses multiple classifiers to predict attribute and object labels. More recently, [22,23,2,14,21,17] learn to map images and labels (i.e., attribute-object combinations) to their joint feature space. It is commonly assumed that the same object can be seen in combination with certain number of attribute values during training. In our work, we study the corner case where the number of seen combinations is minimal, e.g., a fully-correlated attribute-object combinations setting, which introduces severe shortcut opportunities. A recent work [2] shows dramatic performance degradation of DNNs when reducing the number of seen combinations. The objective of this work is to mitigate this effect by incorporating a suitable source domain. We remark that, in most CG approaches, the label space has only two dimensions (i.e., attribute and object types). Instead, in our work the label space of the source domain is not restricted to two dimensions but can be as high-dimensional as the number of annotated factors. In particular, we consider basic visual factors as considered in [6], which are more likely to generalize across different tasks.

Domain Generalization The aim of domain generalization is to learn models that generalize to unseen domains at test time. The problem we address in this work is therefore related to this line of research. More specifically, our problem setting loosely resembles *Heterogeneous Domain Generalization* (HeDG)

[30,31,13,12,26], because we assume that the label distributions in the source and target domains have different, possibly disjoint, support (e.g., the labels represent objects in the source domain and animals in the target domain). Nevertheless, it should be noted that our work stems from a different motivation compared to the domain generalization literature. Unlike domain generalization, we do not assume the target data distribution to be unavailable during training. Instead, we assume to have access to a heavily biased sample from the target domain. From this perspective, our setup is also related to domain adaptation [7,5], where training is performed on both the source and target domains together.

Learning Independent Representations The topic of compositional generalization is also linked to the notion of disentanglement in generative modeling. The goal of disentangled representation learning is to construct a compact and interpretable latent representation, by discovering independent factors of variation (FoVs) in the data [4,11,24]. Most methods proposed in the disentanglement literature assume statistical independence between the factors of variation and perform learning without supervision. Thus, they are predominantly trained and evaluated on synthetic data where the ground truth FoVs are perfectly uncorrelated [16]. This is an idealized setting, which is almost never encountered in the real world. Therefore, the usefulness and generalization ability of these methods when the training data is biased remains unclear. For instance, a recent large-scale empirical study found that several state-of-the-art methods from the disentanglement literature fail to disentangle pairs of correlated factors [25]. Our work is also concerned with learning independent representations of basic visual factors, but, as opposed to prior works, we specifically focus on the problem of mitigating shortcut learning. It should be noted that, in contrast to unsupervised representation learning, we train representations of visual factors with factor annotations. However, we will discuss a scenario in which unsupervised representation learning can be integrated into our framework in section 4.3.

3   Methodology

3.1   Problem Formulation

Our task measures generalization performance in the presence of shortcuts from the perspective of compositional generalization. Specifically, let us consider a *target domain/dataset* t, where each sample consists of an image $x_t \in \mathcal{X}_t \subset \mathcal{I}$ containing a single object. Such object is associated with one attribute: $y_t = (a, o) \in \mathcal{Y}_t = \mathcal{A}_t \times \mathcal{O}_t$, where $\mathcal{A}_t = \{a_1, a_2, \ldots\}$ and $\mathcal{O}_t = \{o_1, o_2, \ldots\}$ are sets of attribute and object type values respectively. In our task, not all attribute-object combinations are seen during training. The goal is to learn a model for the joint probability distribution $p(a, o \mid x_t)$ which yields good predictive performance on images of both seen and unseen attribute-object combinations.

The target dataset may be heavily biased, and thus naively training a classifier to predict object and attribute types can lead to shortcut learning resulting in poor OOD generalization. We define a dataset in a target domain, to

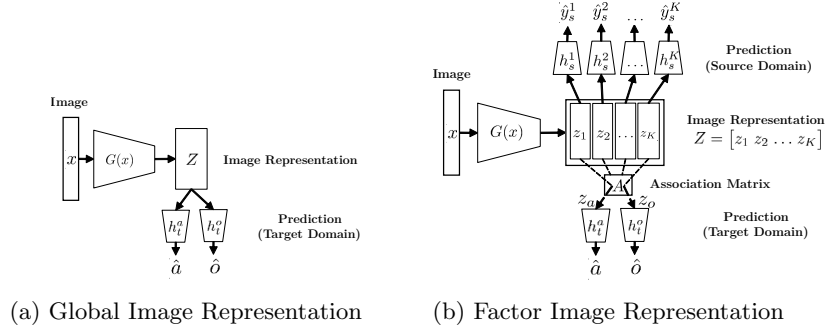(a) Global Image Representation   (b) Factor Image Representation

Fig. 2: Architectures with different image representations. a) Global: a single vector contains all visual clues. b) Factor: multiple factor representations encode different visual factors provided in source domain.

be *fully-correlated* when each annotated attribute label appears together with only one annotated object label or vice versa. We aim to investigate whether introducing an additional, specifically designed, dataset from a different *source domain s* can discourage DNNs from exploiting shortcuts when trained for attribute and object prediction in the target domain. For this purpose, each image $x_s \in \mathcal{X}_s \subset \mathcal{I}$ in the source domain s is assumed to be labeled with a tuple $y_s = (f_1, f_2, \ldots, f_K) \in \mathcal{F}_s^1 \times \mathcal{F}_s^2 \times \ldots \times \mathcal{F}_s^K$, where K is the number of factors and each $\mathcal{F}_s^k$ denotes the set possible values for an individual basic visual factor capturing generic image properties such as shape, color, texture, etc.

To utilize source factor information in the target domain, we build an association matrix that models the connection between different source factor representations to target attribute or object. For fully-correlated target domains, we assume that the association is given as *a priori* knowledge regarding the possible source factor(s) the model is expected to rely to avoid shortcuts. For example, in the Color-Fruit dataset (Figure 4c/4d), attribute and object can be manually associated to color and shape factors respectively. This *a priori* knowledge requirement can be relaxed in semi-correlated setting as presented in section 3.4.

3.2   Network Architecture

Architecture We study a simple baseline model (similar to VisProdNN [22]), which naturally lends itself to exploiting the availability of a source dataset. The baseline model (Figure 2a) consists of an encoder backbone G and multiple prediction heads. It maps an input image to a global latent representation, which is then used to predict both the attribute and object labels. A crucial limitation of this model is that, without additional inductive biases, its latent representation will encode all predictive image factors. This makes the model more vulnerable to learning shortcuts, since it will be free to rely on easy to learn, predictive signals irrespective of their (in)ability to generalize to novel combinations [10].

We extend the model of Figure 2a by splitting the latent representation into multiple factor representations, as in Figure 2b. As opposed to Figure 2a, the encoder G produces K non-overlapping factor representations. Each representation is intended to contain only information related to its corresponding basic factor (e.g., shape, color, texture). The encoder's output can then be written as $G(x) = Z = \begin{bmatrix} z_1 & z_2 & \dots & z_K \end{bmatrix} \in \mathbb{R}^{D \times K}$ (D is the size of a factor representation).

The prediction heads are divided in two subsets for predicting labels in source and target domains respectively. While in principle we could feed Z as an input to all the prediction heads, as discussed above, this approach leads to poor compositional generalization. Instead, we introduce an additional inductive bias, namely that each source prediction logit should only depend on a single factor representation. Therefore, predictions for the source data H_s are:

$$\hat{y}_s = H_s(Z) = \begin{bmatrix} h_s^1(z_1) & h_s^2(z_2) & \dots & h_s^K(z_K) \end{bmatrix} , \tag{1}$$

where $\hat{y}_s = \begin{bmatrix} \hat{y}_s^1 & \hat{y}_s^2 & \dots & \hat{y}_s^K \end{bmatrix}$ contains the predicted factor values for a sample from the source domain. Ideally, each $\hat{y}_s^k$ should only depend on z_k to discourage the latent representation from encoding information irrelevant to predict the k-th factor. Due to biases in the target domain, and in the absence of additional constraints, the architecture introduced above does not ensure invariance of every representation to the other factors. We explore different strategies to promote independence of the learned representations in section 3.3.

In fully-correlated scenario, the association matrix can manually be defined as a binary matrix $A \in \{0,1\}^{K \times 2}$ where A_{k1} and A_{k2} are set to 1 only if the k-th factor informs the attribute and object prediction respectively. The representations of attribute (z_a) and object (z_o) can be obtained by $\begin{bmatrix} z_a & z_o \end{bmatrix} = ZA$. The prediction on target data can then be computed as follows:

$$\hat{y}_t = H_t(ZA) = \begin{bmatrix} h_t^a(z_a) & h_t^o(z_o) \end{bmatrix} , \tag{2}$$

where $\hat{y}_t = (\hat{a}, \hat{o})$ is a tuple of the predicted attribute and object labels.

Loss To train the encoder and the predictors, we use a linear combination of the two loss terms $\mathcal{L}_{source} = \frac{1}{K} \sum_{\forall k} CE(\hat{y}_s^k, y_s^k)$ and $\mathcal{L}_{target} = \frac{1}{2} \sum_{l \in \{a,o\}} CE(\hat{y}_t^l, y_t^l)$, where CE denotes the cross-entropy loss. $\lambda \geq 0$ is a hyperparameter weighting the importance of the regularizing loss term $\mathcal{L}_{source}$, which encourages a factor representation via the source samples.

Training An equal number of samples from the source and target domains are sampled for every minibatch and fed to the network in order to compute $\mathcal{L}_{source}$ and $\mathcal{L}_{target}$ separately. The network is optimized via gradient-based minimization of the total loss. In this regard, all source samples will affect G and H_s and all target samples will affect G and H_t.

3.3   Additional Constraints

The factor representations $\{z_k\}_{k=1}^K$ may still be correlated when using the loss and model architecture described above. Consequently, in the target domain, z_a

(attribute representation) may be predictive of the object label, and z_o (object representation) may be predictive of the attribute label. These are unintended shortcuts that lead to poor compositional generalization. We explore two additional constraints to further encourage independence among factor representations: the *Isolated Latent Constraint* and the *Cross Independence Constraint.*

Isolated Latent (IL) Constraint completely prevents factor representations from being influenced by the target domain. While this suppresses the effect of biases in the target dataset, it may harm discriminative performance in the target domain. This constraint is implemented by stopping gradients from $\mathcal{L}_{target}$ to the encoder G.

Cross-Factor Independence (CI) Constraint promotes independence of factor representations, by adding a set of small auxiliary networks $\left\{H'_{k_1 k_2}\right\}_{k_1 \neq k_2}$ for cross-factor predictions. While each $H'_{k_1 k_2}$ is trained to predict $y_s^{k_2}$ from z_{k_1}, G is trained to produce Z such that all H' are poor predictors. More details are presented in the appendix A.3. Although the CI constraint only encourages independence with respect to the source domain, we investigate if this property can be transferred to target domains in section 4.2.

3.4   Learning Factor Association Matrix A

In non-fully-correlated target domains (e.g., semi-correlated setting mentioned in section 4.3), we can relax the requirement that the association matrix A must be manually given *a priori* and, instead, learn it in an end-to-end fashion together with the network parameters. To this end, we apply a continuous relaxation of the binary matrix A, and allow it to contain real numbers within $[0, 1]$ such that each column sums to 1 (i.e. overall weightage across source factors) to maintain scales of the attribute/object representations using the softmax function.

We found that naïvely learning A without any additional constraints can lead to poor properties of the association matrix. Ideally, the association matrix should match the target attribute or object type to only one (or a few) robust source factor(s), and ignore factors vulnerable to shortcuts. For this reason, we propose to add an additional regularization $\mathcal{L}_{Reg} = \alpha \mathcal{L}_{Entropy} + \beta \mathcal{L}_{Suppress}$.

$\mathcal{L}_{Entropy}$ is the sum of the entropy values of both columns of A. Minimizing this entropy loss will reduce the number of source factors used to predict target properties, encouraging only robust factors to be considered when minimizing together with cross-entropy losses. On the other hand, $\mathcal{L}_{Suppress}$ applies a regularization along the rows of A to make sure no same source factor is predictive of both target predictions. In particular, for each row i, if its maximum value $A_{ij^{max}}$ is higher than a threshold τ, all other entries will be suppressed by adding $\left(\sum_{j \neq j^{max}} A_{ij}\right) * (\mathrm{sg}(A_{ij^{max}}) - \tau)$ to the loss term. The symbol $\mathrm{sg}(A_{ij^{max}})$ indicates the stop gradient operation, such that minimizing $\mathcal{L}_{Suppress}$ only affects the cell whose values are not maximum in each row. Detailed experiments on the automatic learning of the association matrix are presented in section 4.3.
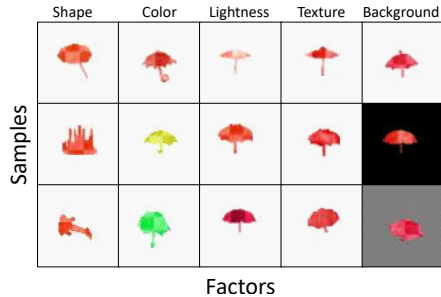
Fig. 3: Illustration of DiagVib-Caltech (main source dataset) showing its multiple independent factors. More details of each factor is presented in the appendix A.1.

4   Experiments

We conduct several experiments to understand how incorporating the source domain affects compositional generalization in scenarios which are vulnerable to shortcut learning. We start by describing our experiment setup below.

Datasets We use the following datasets in our experiments:

DiagVib [6] We extend the original framework to use shapes other than MNIST to increase variances. Our main configurations based on this framework are *a) DiagVib-Caltech*: With 50 non-animal shapes from Caltech101 [18] in 12 colors. 5 basic visual factors are available as shown in Figure 3. This dataset is our main source dataset. We will later show that compositional generalization can be improved by this low-cost dataset even in more complex target domains. *b) DiagVib-Animal*: With 10 animal shapes from [3] in 10 colors on 3 backgrounds and scales (see Figure 4a/4b).

Color-Fruit This dataset is comprised of real fruit images (of 5 types) from the Fruit-360 dataset [20]. Additionally, we control colors of the fruits using the recolorization approach from [29] (see Figure 4c/4d). More details on this dataset generation are described in the appendix A.10.

AO-CLEVR This dataset is proposed in [2] to benchmark compositional generalization. It contains 3 basic shapes in 7 different colors. We will use this dataset as an additional target domain. We simulate correlation between attributes and objects by limiting one color to appear with only one shape during training.

Color-Fashion This dataset is originally proposed in [15] in which each image sample depicts a person dressed with cloth combinations. For each sample, cloth type and color segmentations are provided. In this paper, original images are cropped so that only one cloth type is appeared in individual images (see Figure 4e/4f). 5 cloth types (T-shirt, skirt, jeans, shoes and dress) and 5 colors (Black, White, Yellow, Green and Blue) are selected from the original dataset.

For all target domains in our work, if a manual factor association matrix A is required for any algorithms, attribute and object types (i.e., shapes, fruit

(a)
DV-Animal
(Train)

(b)
DV-Animal
(Test)

(c)
Color-Fruit
(Train)

(d)
Color-Fruit
(Test)

(e)
Color-Fashion
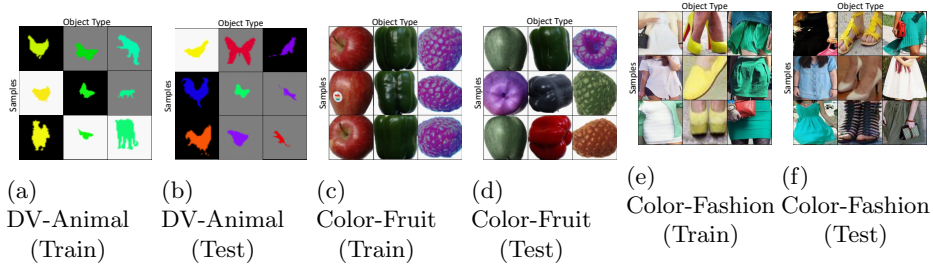(Train)

(f)
Color-Fashion
(Test)

Fig. 4: Samples from target domains. A column of each grid corresponds to an object type. Notice that each object (shape) corresponds to a single attribute (color) during training (see a, c and e) but not during testing (see b, d and f).

types, cloth types) are associated to color and shape factors respectively. This is intuitive since shape can robustly predict object types in general. Poorly assigned association can lead to poor performance as detailed in the appendix A.4.

Evaluation We evaluate compositional generalization in the open world setting as in [23] by computing predictive accuracies on seen and unseen attribute-object combinations as well as their harmonic mean (HM). One difference to [23] is that, for fair comparison, we do not use an additional hyperparameter (i.e. bias term) to calibrate the likelihood of predicting unseen attribute-object combinations (More details regarding the bias term are described in the appendix A.6). During testing, any attribute-object combinations in the target domain can appear. All quantitative results are averaged across 6 different random training seeds.

Network Configurations We follow the same convention as in [22,2,23] by using ImageNet pretrained features from ResNet-50 as network inputs. G, H_s and H_t are modeled as fully-connected networks. Factor-0 and FactorSRC refers to our architecture when a source domain is ignored and used, respectively. IL and CI constraints are appended as suffixes if the constraints are applied. Also, we benchmark the architecture with the global image representation (Figure 2a). Similarly, Global-0 and GlobalSRC denote its configurations where a source domain is not incorporated and incorporated respectively. More implementation details can be found in the appendix A.5. The code for our implementation is publicly available at `https://github.com/boschresearch/sourcegen`.

Baselines We compare our approach against the following baselines (1) **LabelEmbed+** [22]: a vanilla baseline, which performs recognition with a joint feature space for images and labels. (2) **TMN** [23], which employs automatic network rewiring conditioned on attribute-object pair hypotheses. (3) **CGE** [21], which exploits graph structure to regularize the joint feature space.

4.1   Compositional Generalization in Fully-Correlated Scenario

We investigate the impact of using an uncorrelated source domain (all factor combinations can appear uniformly) across different target domains whose labels are fully-correlated. We compare several variants of our approach against baselines that do not use a source domain. The results are shown in Table 1.

We begin by noting that, without a source domain (first five rows), baselines generally perform well only on the seen combinations but not on the unseen ones. E.g., Factor-0 on the Color-Fruit dataset has seen accuracy of 100% compared to only 2.9% on unseen combinations (Low seen accuracies of some baselines are discussed in the appendix A.6). This occurs as these networks are trained only on the target datasets with correlated combinations so that they learn to excessively exploit the easiest predictive visual factors present in the datasets. On the unseen combinations, these predictive factors do not necessarily generalize to the intended labels. This degrades the generalization of the networks.

In contrast, using a source domain (see FactorSRC variations) consistently improves the HM accuracy by increasing accuracy on unseen combinations, at the expense of a partial loss of performance on seen combinations. For example, the FactorSRC-IL baseline on the Color-Fruit dataset has a seen accuracy of 95.5%, which is lower compared to baselines that do not use the source domain, but in turn exhibits the highest unseen accuracy (40.7%). This shows a reduction of shortcut learning. The same trend holds for all other datasets we consider, with different seen/unseen accuracy trade-offs across datasets.

Compared to previous works, our results show that a *single* and *simple* source domain improves generalization performance on unseen combinations across different target domains. An alternative *naïve* solution would be to collect data corresponding to unseen combinations directly in the target domain and include them in the training set. However, this is in practice not a viable solution: not only is collecting data in a real-world target domain expensive but, perhaps more importantly, the biases that affect the training set are often unknown, which makes collecting an uncorrelated dataset difficult in practice. Rather, generalization improvement on those target domains can be achieved without much additional costs if we have a *universal* (a source domain can be used for multiple target domains) source domain at hand that can be *generated cheaply*. One open problem is the trade-off between generalization and in-distribution accuracy, which in some cases is still sub-optimal, especially when the target domain has a large domain shift from the source domain (see the drop of seen accuracy of the FactorSRC-IL on the Color-Fashion dataset). Improving this trade-off is an open research question leaving a scope for improvement in future works.

Global vs Factor Image Representations Our results suggest that a factor representation is essential to exploit the source domain for compositional generalization. In fact, all GlobalSRC variations, in spite of incorporating the source domain during training, exhibit significantly lower unseen accuracy compared to the FactorSRC variants. This is due to the fact that the global representation contains information about all visual factors that are relevant for the object-

Table 1: Accuracies on DiagVib-Animal, Color-Fruit, AO-CLEVR (each has random chance accuracy of 1%, 4% and 4.7% respectively) and Color-Fashion target domains. DiagVib-Caltech is used as the source domain.

Approach	Use Source?	DiagVib-Animal			Color-Fruit			AO-CLEVR			Color-Fashion		
		Seen	Unseen	HM	Seen	Unseen	HM	Seen	Unseen	HM	Seen	Unseen	HM
LabelEmbed+	✗	69.6	7.3	13.2	**100**	6.8	12.5	**100**	0.7	1.5	37.0	7.1	11.9
TMN	✗	95.5	0.1	0.3	**100**	0.0	0.0	**100**	0.0	0.0	86.5	0.7	1.4
CGE	✗	43.8	9.1	15.0	84.4	7.8	14.3	94.0	9.3	16.9	21.6	**20.5**	21.0
Global-0	✗	**96.1**	0.0	0.1	**100**	1.5	3.0	**100**	0.3	0.5	**93.6**	0.0	0.0
Factor-0	✗	95.2	0.8	1.5	**100**	2.9	5.5	**100**	2.2	4.3	92.7	1.8	3.6
GlobalSRC	✓	94.2	0.3	0.5	**100**	1.1	2.2	**100**	0.3	0.7	85.5	0.2	0.4
GlobalSRC-IL	✓	92.4	0.3	0.7	**100**	0.7	1.4	98.9	0.8	1.6	61.8	2.2	4.2
FactorSRC	✓	90.0	7.0	13.0	99.7	27.3	42.4	99.9	3.2	6.3	76.4	8.3	15.0
FactorSRC-CI	✓	91.2	7.9	14.5	**100**	10.9	19.6	**100**	2.3	4.5	87.3	8.2	14.8
FactorSRC-IL	✓	56.3	**32.6**	**41.3**	95.5	**40.7**	**57.0**	89.5	**19.6**	**32.1**	32.7	17.0	**22.3**

Table 2: Cross prediction accuracies on DiagVib-Animal. These are obtained by using each associated factor representation (z_a or z_o) to predict each target label (attribute or object) with a linear model. Ideal independence representations will have high direct prediction accuracies (predict their own labels well) but low cross prediction accuracies (predict others' labels poorly).

Approach	Direct-Prediction ↑		Cross-Prediction ↓	
	$z_a \to \hat{a}$	$z_o \to \hat{o}$	$z_a \to \hat{o}$	$z_o \to \hat{a}$
FactorSRC	**62**	**86**	77	44
FactorSRC-CI	54	83	73	35
FactorSRC-IL	58	85	**46**	**23**

attribute prediction task, thus offering easy-to-learn shortcuts that harm generalization performance. In contrast, by using factor representations, we promote factor disentanglement such that shortcuts are harder to learn.

4.2 Impact of Additional Constraints

We investigate the role of IL and CI constraints. We begin by noting that FactorSRC-IL gives the best HM accuracies across all target domains. Our hypothesis is that this result can be explained by a larger cross-factor information flow into the learned factor representations when adding the CI constraint compared to IL. To quantitatively measure the magnitude of cross-factor leakage, we extract z_k from all test samples and use them to predict all labels ($y_s^1, y_s^2, \ldots y_s^K$ for a source domain or $\hat{a}, \hat{o}$ for a target domain) with linear models. Ideally, z_k should predict its associated label (direct prediction) well but should fail to predict other labels (cross-prediction). We present these results in Table 2.

First, we are interested whether the CI constraint can encourage independence, not only in the source domain, but more importantly in the target domain.

Table 3: Accuracies on various target domains in semi-correlated scenarios.

Approach	DiagVib-Animal			Color-Fruit			Color-Fashion		
	Seen	Unseen	HM	Seen	Unseen	HM	Seen	Unseen	HM
TMN	**83.4**	3.2	6.2	**99.9**	3.1	6.0	**68.6**	3.7	6.9
CGE	51.8	1.9	3.6	72.0	8.4	15.0	42.5	13.0	19.9
FactorSRC-IL	52.8	**35.0**	**42.1**	93.7	**36.3**	**52.3**	32.5	**15.0**	**20.5**
FactorSRC-IL-LA	72.7	3.7	7.1	99.0	17.5	29.6	53.1	8.0	13.9
FactorSRC-IL-LA[R]	60.9	24.2	34.7	96.0	32.7	48.6	35.2	14.3	20.2

We observe that, while independence is promoted by FactorSRC-CI in the source domain well (see the appendix A.3), in the target domain, cross-prediction accuracies decrease only slightly compared to FactorSRC. Thereby, we can infer that the independence enforced by $\mathcal{L}_{H'}$ in the source domain is not necessarily transferable to the target domain. One possible reason is that factor representations are still affected by the dataset biases in the target domain via $\mathcal{L}_{target}$.

On the other hand, dataset bias in the target domain cannot affect the factor representations with the IL constraint. In Table 2, although an explicit independent constraint is not introduced, the factor representations are less entangled in the target domain, which is indicated by lower cross-prediction accuracies (while high direct-prediction accuracies are preserved). The independence which is indirectly encouraged only by $\mathcal{L}_{source}$ enables shortcut-robust factor representations resulting in better HM accuracies in Table 1.

4.3   Learning Association Matrix for Semi-Correlated Scenario

The association matrix A must be known *a priori* for fully-correlated target domains because the target data alone does not contain enough information to distinguish object types from their attributes. For a *semi-correlated* target domain, a more general solution is viable, in which the association matrix is learned. In this case, each object type is observed in combination with at least two attribute values. The additional combinations make it possible to distinguish attribute from object type. In order to learn the association matrix A, we adopt the algorithm introduced in section 3.4.

Table 3 reports performance of different algorithms. FactorSRC-IL-LA and FactorSRC-IL-LA[R] indicate algorithms that learn the association matrix without and with association regularization respectively. Results suggest that naïvely backpropagating through the matrix A is not an effective strategy as indicated by the large gap between HM accuracy of FactorSRC-IL and FactorSRC-IL-LA. This is due to an undesired property of the association matrix which we will later investigate. Fortunately, this undesired property can be alleviated by incorporating our proposed regularization constraints during training, as indicate by the comparable performance of FactorSRC-IL and FactorSRC-IL-LA[R].

To qualitatively assess the correctness of learned associations, we visualize A as heatmaps in Figure 5. In addition, we compare the association matrix that
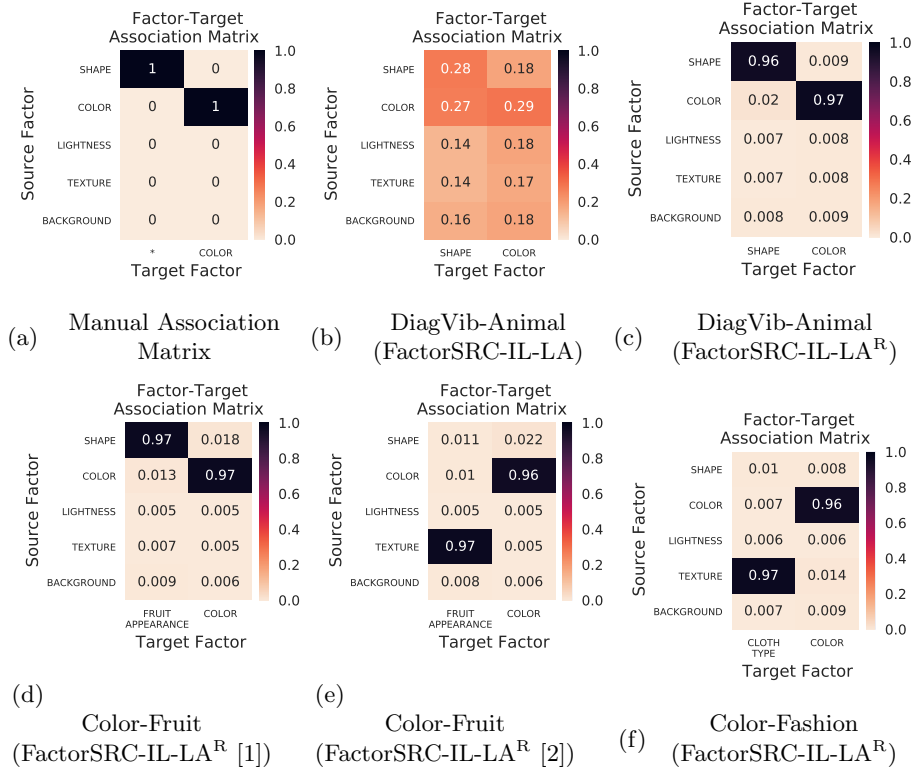
Fig. 5: Factor association matrices learned from different target datasets with various approaches resulted from representative runs. Manual Association Matrix in Figure a is only used for our manual association (not the groundtruth).

we manually assigned in FactorSRC* (Figure 5a) and the learned ones. First we consider the simplest target dataset, DiagVib-Animal. In this case, FactorSRC-IL-LA, which does not apply any regularization, fails to retrieve the association matrix (Figure 5b). Even if the association matrix has the high weights that associate shape factor to shape target and color factor to color target correctly, the matrix is still far from sparse, making the model vulnerable to shortcuts (see results in Table 3). On the other hand, when regularization constraints are introduced in FactorSRC-IL-LA^R, the learned matrix is more sparse and closer to the manually assigned matrix (see Figure 5c), which results in lower shortcut vulnerability of the model and higher compositional generalization performance.

In the more realistic case of the Color-Fruit dataset, we find that with multiple random seeds, FactorSRC-IL-LA^R can converge to two possible configurations for the estimated matrix A. The first configuration (Figure 5d) is close to our manual association as it matches the fruit type to the shape factor. Another configuration, on the other hand, associates the fruit type to the texture factor

(Figure 5e). This is not surprising, since in the color-fruit dataset, both shape and texture factors are predictive of fruit type. The same observation has already been made in the context of conventional image classification [9] especially with complex object shapes. We observed the same behaviour in the case of the Color-Fashion dataset where the learned matrix associates garment type to the texture factor (Figure 5f). Another advantage of our approach indicated by this observation is that it yields higher model interpretability, as we can understand which visual factors are important for network predictions.

The fact that learning of the association is possible when target datasets are not fully-correlated makes our approach well suited to practical applications. As the association is determined automatically, minimal *a priori* knowledge is required. In other words, factor representations can also be learned from images by any approaches including unsupervised representation learning that disentangle image representations into multiple independent factor representations [11].

4.4   Properties of the Source Domains

We also investigate the properties of source domains which encourage generalization. Our main findings can be summarized as follows: first, basic visual factors represented in the source domain should be sufficiently diverse and aligned with visual properties of the target data. Second, the fact that all factor combinations are available in the source domain during training is crucial. This allows deep networks to learn meaningful representation for each factor. Lastly, large intra-class variation of factors is also important to encourage better generalization. More details of our ablation studies can be found in the appendix A.1.

5   Conclusion

We study vulnerability of DNNs to shortcuts by evaluating their compositional generalization on target domains with correlated attribute-object combinations. We provide empirical evidence that incorporating an additional source domain can improve generalization on unseen combinations on target domains. The source domain enables certain networks to represent inputs in terms of multiple independent visual factors. From our findings, the impact of the source domain on compositional generalization relies on two major conditions: (1) Choice of network model: networks should have internal representations in which visual factors are disentangled and independent with respect to target domains (2) Choice of source domain: the source domain should be uncorrelated and cover main basic factors. The fact that our source domain is simple also shows a practical benefit that performance on certain target tasks can efficiently improve using an easy-to-generate source domain. This is relatively cheaper compared to acquiring samples from complex target domains. If target domains are not fully-correlated, some requirements of manual labels/annotations can be relaxed, leading to more practical applications of this work. We hope this work will serve as an inspiration to integrate inductive biases in the forms of datasets and network design.

References

1. Ahmed, F., Bengio, Y., van Seijen, H., Courville, A.: Systematic generalisation with group invariant predictions. In: International Conference on Learning Representations (2020)
2. Atzmon, Y., Kreuk, F., Shalit, U., Chechik, G.: A causal view of compositional zero-shot recognition. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 1462–1473. Curran Associates, Inc. (2020)
3. Bai, X., Liu, W., Tu, Z.: Integrating contour and skeleton for shape classification. In: 2009 IEEE 12th international conference on computer vision workshops, ICCV workshops. pp. 360–367. IEEE (2009)
4. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence **35**(8), 1798–1828 (2013)
5. Chang, W.G., You, T., Seo, S., Kwak, S., Han, B.: Domain-specific batch normalization for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7354–7362 (2019)
6. Eulig, E., Saranrittichai, P., Mummadi, C.K., Rambach, K., Beluch, W., Shi, X., Fischer, V.: Diagvib-6: A diagnostic benchmark suite for vision models in the presence of shortcut and generalization opportunities. arXiv preprint arXiv:2108.05779 (2021)
7. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International conference on machine learning. pp. 1180–1189. PMLR (2015)
8. Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. Nature Machine Intelligence **2**(11), 665–673 (2020)
9. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. arXiv preprint arXiv:1811.12231 (2018)
10. Hermann, K., Lampinen, A.: What shapes feature representations? exploring datasets, architectures, and training. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 9995–10006. Curran Associates, Inc. (2020)
11. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. International Conference on Learning Representations (2016)
12. Li, D., Zhang, J., Yang, Y., Liu, C., Song, Y.Z., Hospedales, T.M.: Episodic training for domain generalization (2019)
13. Li, Y., Yang, Y., Zhou, W., Hospedales, T.: Feature-critic networks for heterogeneous domain generalization. In: Chaudhuri, K., Salakhutdinov, R. (eds.) Proceedings of the 36th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 97, pp. 3915–3924. PMLR (09–15 Jun 2019)
14. Li, Y.L., Xu, Y., Xu, X., Mao, X., Lu, C.: Learning single/multi-attribute of object with symmetry and group. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
15. Liu, S., Feng, J., Domokos, C., Xu, H., Huang, J., Hu, Z., Yan, S.: Fashion parsing with weak color-category labels. IEEE Transactions on Multimedia **16**(1), 253–265 (2013)

16. Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., Bachem, O.: Challenging common assumptions in the unsupervised learning of disentangled representations. In: international conference on machine learning. pp. 4114–4124. PMLR (2019)
17. Mancini, M., Naeem, M.F., Xian, Y., Akata, Z.: Open world compositional zero-shot learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5222–5230 (2021)
18. Marlin, B., Swersky, K., Chen, B., Freitas, N.: Inductive principles for restricted boltzmann machine learning. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 509–516. JMLR Workshop and Conference Proceedings (2010)
19. Misra, I., Gupta, A., Hebert, M.: From red wine to red tomato: Composition with context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1792–1801 (2017)
20. Mureşan, H., Oltean, M.: Fruit recognition from images using deep learning. arXiv preprint arXiv:1712.00580 (2017)
21. Naeem, M.F., Xian, Y., Tombari, F., Akata, Z.: Learning graph embeddings for compositional zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 953–962 (2021)
22. Nagarajan, T., Grauman, K.: Attributes as operators: factorizing unseen attribute-object compositions. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 169–185 (2018)
23. Purushwalkam, S., Nickel, M., Gupta, A., Ranzato, M.: Task-driven modular networks for zero-shot compositional learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3593–3602 (2019)
24. Sauer, A., Geiger, A.: Counterfactual generative networks (2021)
25. Träuble, F., Creager, E., Kilbertus, N., Locatello, F., Dittadi, A., Goyal, A., Schölkopf, B., Bauer, S.: On disentangled representations learned from correlated data (2021)
26. Wang, Y., Li, H., Kot, A.C.: Heterogeneous domain generalization via domain mixup. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3622–3626. IEEE (2020)
27. Xiao, K., Engstrom, L., Ilyas, A., Madry, A.: Noise or signal: The role of image backgrounds in object recognition. arXiv preprint arXiv:2006.09994 (2020)
28. Zeithamova, D., Bowman, C.R.: Generalization and the hippocampus: more than one story? Neurobiology of Learning and Memory p. 107317 (2020)
29. Zhang, R., Zhu, J.Y., Isola, P., Geng, X., Lin, A.S., Yu, T., Efros, A.A.: Real-time user-guided image colorization with learned deep priors. ACM Transactions on Graphics (TOG) **9**(4) (2017)
30. Zhou, K., Yang, Y., Hospedales, T., Xiang, T.: Deep domain-adversarial image generation for domain generalisation (2020)
31. Zhou, K., Yang, Y., Hospedales, T., Xiang, T.: Learning to generate novel domains for domain generalization (2020)

A Appendix

In this section, we present arrays of ablation studies to understand crucial properties of the source domains. All experiments in this section are performed in the fully-correlated target domains.

A.1 DiagVib Dataset Configurations

As mentioned in the experiments in section 4, we use datasets based on the DiagVib framework which allows generation of synthetic datasets with custom configurations of basic visual factors. We consider five factors whose number of possible values are listed according to Table 4. It should be noted that DiagVib-Caltech and DiagVib-Animal have different number of available shapes.

Table 4: Different visual factors, which can be configured in the DiagVib framework

Factor	Description	No. of Classes
Shape	Object boundary defined by a silhouette	Caltech: 50 Animal: 10
Color	Hue value in HSV space	12
Lightness	Lighting condition (e.g., bright, dark)	4
Texture	Pattern drawn inside the object (e.g. wooden, checkerboard)	5
Background	Background color	3

A.2 Ablation Studies on the Source Domain

Table 5: Accuracies of FactorSRC-IL in the target domain (DiagVib-Animal) with variations of source domains to demonstrate the impact of their uncorrelated factors.

Source Setting	Images from	Correlated Factors	Target HM Acc.
Uncorrelated	DiagVib-Caltech	False	**33.5 ± 1.0**
Correlated	DiagVib-Caltech	True	2.5 ± 0.7
Target	DiagVib-Animal	True	1.7 ± 0.4

Impact of Uncorrelation of Factors In this study, we aim to investigate whether the improvement in generalization performance after incorporating the

source domain stems from uncorrelating visual factors. We compare the following source dataset settings: *a*) Uncorrelated: all factor combinations are available *b*) Correlated: shape and color factors are one-to-one correlated *c*) Target: use correlated data sampled from the target distribution (DiagVib-Animal) for training. For a fair comparison, the number of target-associated factors (shape/color) are reduced to 10 for Uncorrelated and Correlated settings, so as to match the Target setting. Results in Table 5 indicate that the Uncorrelated setting yields significantly higher accuracy compared to others. This empirically shows that this improvement in OOD generalization is indeed due to the uncorrelated nature of the source dataset and not just a mere result of the increased dataset size.

Impact of Shape Variations We conduct another experiments to understand if the complexity of the shapes provided in the source domain affects accuracies in the target domain. We modify the DiagVib-Caltech source domain to use MNIST shapes and compare it to the original setting with Caltech shapes (we use 10 shapes in both cases to be comparable). Table 6 shows that the setting with MNIST shapes has lower accuracies. We believe that this is due to the fact that MNIST has less intra-class shape variation compared to Caltech. For example, the shape of the number ones are not much different across different samples. This degrades the generality of the learned shape representation. This experiment suggests that a primary concern when constructing a source dataset should be intra-class variability of each factor.

Table 6: Accuracies of FactorSRC-IL with variations of shape in the source domain.

Shape	DiagVib-Animal HM. Acc	Color-Fruit HM. Acc	AO-CLEVR HM. Acc
Caltech	**33.5 ± 1.0**	**56.0 ± 2.7**	**36.4 ± 1.8**
MNIST	31.9 ± 0.7	46.9 ± 3.2	29.0 ± 1.4

Impact of Available Factors In this experiment, we study the effects of varying the number of basic visual factors represented in the source domain. According to the result in Table 7, while we find that increasing the number visual factors yields better performance overall, for some factors, the effect on different target domains is different. For instance, with DiagVib-Animal as a target, including the background as a factor in the source domain improves performance significantly, due to the fact that the target domain has variable background colors. In contrast, this effect is not observed on Color-Fruit, whose images have a constant background. Instead, learning lightness and texture can improve generalization performance since these two factors have high variation in this target
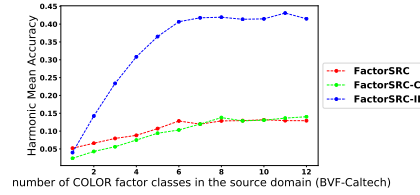
domain (DiagVib-Animal doesn't have variations of lightness and texture). We can infer from this result that the performance in the target domain tends to be better if the source domain captures basic factors which are represented in the target domain.

Table 7: HM Accuracies from FactorSRC-IL approach on DiagVib-Caltech source domain with different presence of factors (S, C, L, T, B correspond to Shape Color, Lightness, Texture and Background respectively).
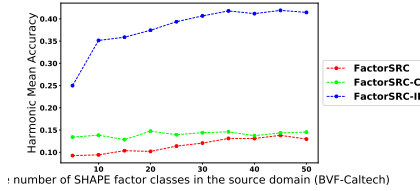
Factors	DV.-Animal	Color-Fruit
S/C	28.6 ± 1.2	49.5 ± 5.0
S/C/L	29.3 ± 1.1	53.2 ± 3.5
S/C/L/T	31.5 ± 0.5	**57.7 ± 3.3**
S/C/L/T/B	**41.3 ± 1.6**	57.0 ± 4.7

In summary, we have performed an array of ablation studies to analyze properties of the source domain which encourage better generalization in target domains. Firstly, we showed that visual factors in the source domain should be uncorrelated. This facilitates disentanglement of visual factors' representations, which in turn leads to less shortcut vulnerability. Secondly, we demonstrated that intra-factor variability is crucial in order for deep networks to learn generalizable representations. Lastly, visual factors encoded in the source domain should cover as many predictive features in the target domain as possible. We believe that these three aspects are among the most important criteria, which should guide practitioners towards choosing better source domains for augmenting biased training datasets.

Impact of Variations of the Number of Factor Classes From our experiments, we showed that FactorSRC-IL can learn factor representations from the source domain, which improve compositional generalization in several target domains. In this section, we would like to investigate how the number of factor values in the source dataset affects generalization performance in the target domain. For this purpose, we vary the number of factor values associated to each target label (shape and color in our setting) and measure compositional generalization in the DiagVib-Animal target domain. Results are shown in Figure 6 and indicate that a higher number of factor values generally leads to the better performance. This is intuitive since a higher number of classes should encourage networks to learn more general factor representations. An interesting observation is that the network needs only around 8 color classes to be close to optimal performance while around 35 classes are needed in the case of shape. We believe this is due to the fact that shape, as a basic visual factor, is more high-dimensional and thus more difficult to model than color.

(a) Varying the number of colors

(b) Varying the number of shapes

Fig. 6: Accuracies of FactorSRC-IL on the DiagVib-Animal with different number of factor classes (color and shape) while maintaining the same configuration for the other factors on the DiagVib-Caltech source domain.

A.3   Effect of the CI Constraint on the Source Domain

In our experiment section, we stated that the Cross-Factor Independence Constraint (CI) promotes independence of factor reresentations in the source domain. In this section, we provide experimental evidence supporting our claim. To this end, we compare cross-prediction accuracies with and without the CI constraint, for each factor among $z_1, z_2, \ldots, z_K$. Results are visualized in Figure 7. We can see that, while direct-prediction accuracies are comparable with and without the CI constraint, the cross-prediction performance decreases significantly when the CI constraint is introduced. This supports our hypothesis that the CI constraint induces a higher degree of independence among factor representations in the source domain.

A.4   Importance of Association Matrix Assignment

We hypothesize that the shape is a generic robust factor that can be used to predict object types. So, we manually associate the shape factor from the source domain to the object type of target domains in the association matrix A in all fully-correlated target domain scenarios. To validate this hypothesis, we perform an ablation study to evaluate network performance when different configurations of source factors are chosen in association matrix A. Performance of all configurations can be visualized as two-dimensional heatmaps for different datasets as in Figure 8. The value in each cell C_{ij} of a heatmap represents the average HM accuracy when target attribute and target object associate to source factor i

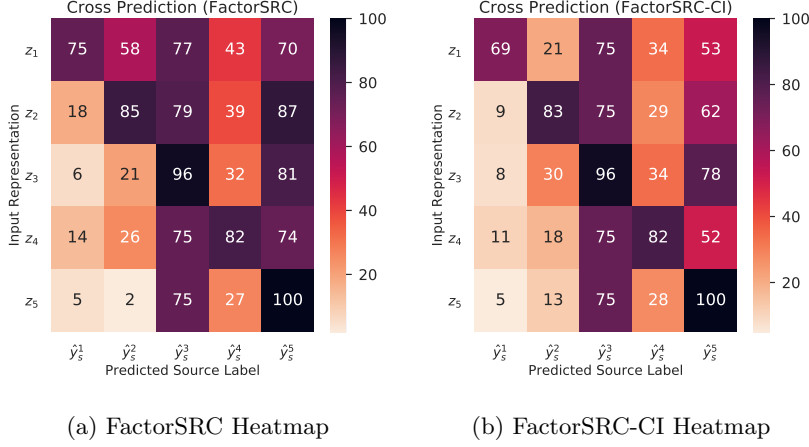(a) FactorSRC Heatmap   (b) FactorSRC-CI Heatmap

Fig. 7: Heatmaps displaying direct and cross-factor prediction accuracies using DiagVib-Caltech and DiagVib-Animal as source and target domain respectively. Each cell indicates the accuracy attained when a single factor representation (z_k in each row) is used to predict labels ($\hat{y}_s^l$) with a linear model. A higher degree of independence among factor representations is expected to yield similar diagonal values but lower off-diagonal ones (cross-prediction). Factor indices from 1 to 5 correspond to shape, color, lightness, texture and background respectively.

(row of heatmap) and j (column of heatmap) in the source domain respectively. In this regard, cell C_{21} in each heatmap represents HM accuracy of a configuration setting when target attribute/object associate with source color/shape factors.

First of all, considering the results from DiagVib-Animal and Color-Fruit target datasets in Figure 8a and 8b, the highest values of C_{21} (42% and 58%) for both datasets empirically support our hypothesis that the shape factor is a robust factor for predicting object types in the target domain. In the Color-Fruit dataset, an interesting observation can be made as texture factor is also a predictive of fruit type in addition to the shape factor (C_{13} of 40% in Figure 8b). This result shows capability of the texture to predict fruit type which aligns to the estimated association matrix in Figure 5e. From these results, we can empirically validate our hypothesis and show that a proper configuration of the association matrix is important to alleviate model vulnerability to shortcuts.

In a more challenge dataset Color-Fashion, even though our configuration C_{21} is among the best, there are other configuration settings that reach similar result (Figure 8c). This behavior can be explained intuitively: considering the target object type (garment type), models have high performance when associating the object type to either shape or texture factors (can be seen as cells of high values on the first and the fourth columns). This behaviour is similar to the case of the fruit type in Color-Fruit dataset emphasizing the fact that both shape or texture
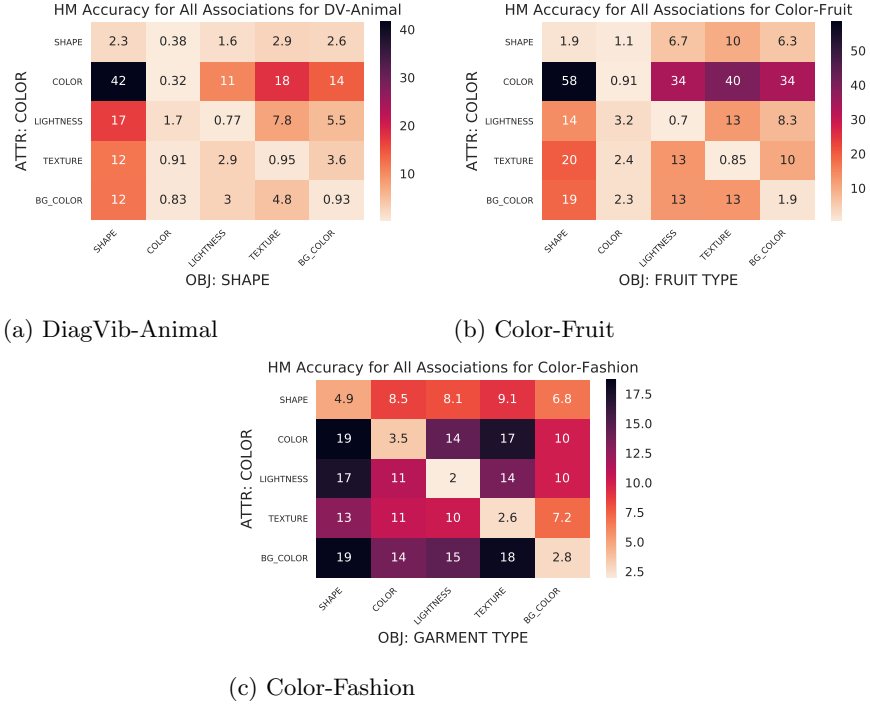
(a) DiagVib-Animal   (b) Color-Fruit

(c) Color-Fashion

Fig. 8: HM Accuracies with different configurations of association matrices in different target datasets. The cell C_{ij} in each heatmap corresponds to the accuracy when the association matrix associating target attribute and object type to the i-th and j-th factor respectively. The order of factors is shape, color, lightness, texture and background color.

can be a predictive factor for object types. For the target attribute type (garment color), its associations to color or background color produce high accuracies (can be seen as cells of high values on the second and the fifth rows). This implies that information of the garment color is contained in factor representations of both color and background color. The underlying reason can be due to the design of our source domain. In the DiagVib-Caltech source domain, boundaries between foregrounds and backgrounds are simple as backgrounds are only plain colors. However, in the case of the Color-Fashion target domain, its backgrounds are more complex representing realistic scenes. This suggests redesigning of the source domain. One possibility is to use more realistic backgrounds such as place images similar to [1].

A.5   Implementation Details

In this section, we provide details of our network design and training hyperparameters.

Table 8: Accuracies on DiagVib-Animal, Color-Fruit, AO-CLEVR and Color-Fashion target domains with the similar experiment setup as in Table 1. However, calibrated bias terms are incorporated before computing seen, unseen and HM accuracies.

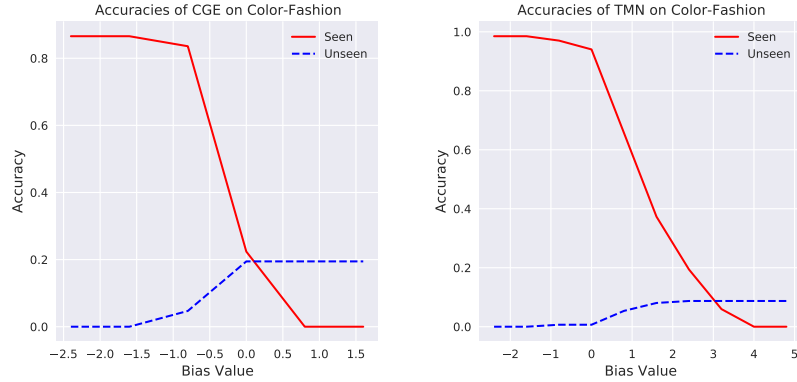| Approach	Use Source?	DiagVib-Animal			Color-Fruit			Color-Fashion		
		Seen	Unseen	HM	Seen	Unseen	HM	Seen	Unseen	HM
LabelEmbed+	✗	96.3	10.3	13.2	100	19.7	12.5	90.0	13.4	16.9
TMN	✗	95.7	7.0	12.2	100	17.9	29.2	89.4	7.0	8.9
CGE	✗	92.8	11.8	15.0	100	24.9	32.9	88.1	20.8	21.0

For all variants of the factorized architecture illustrated in Figure 2b (Factor-0, FactorSRC, FactorSRC-CI and FactorSRC-IL), the encoder G is a fully-connected network with 2 hidden layers, which outputs multiple factor representations, each one of length 64. All branches of H_s and H_t (i.e., all prediction heads in $\{h_s^k\}_{k=1}^K \cup \{h_s^o, h_s^a\}$) consist of a fully-connected network with 1 hidden layer. We set the hyperparameter λ equal to 10 when we include the source dataset for all experiments for fair comparison. In section 3.4, we introduce strategy to learn the factor association matrix with additional regularization constraints. Hyperparameters α, β and τ used for the regularization constraints are 5, 20 and 0.33 respectively.

For training we use Adam as an optimizer, a learning rate of 0.01 and weight decay equal to $5e^{-5}$. The optimal network is selected based on the loss on a validation split over 100 epochs.

A.6   Bias Terms for Adjusting Likelihood of Unseen Combinations

As mentioned earlier in section 4, unlike some prior works [23,21], we evaluate compositional generalization without bias terms to adjust the likelihood of unseen combinations (using higher bias makes the model more likely to predict unseen combinations). The reason is that tuning of the bias terms requires availability of samples from unseen combinations. This violates the zero-shot assumption. Additionally, bias terms are designed to be applicable only with certain baselines based on compatability scores (such as LabelEmbed+, TMN and CGE) but not the others leading to unfair comparison.

For completeness, we will also provide results when the calibrated bias terms are incorporated during evaluation for LabelEmbed+, TMN and CGE. The seen, unseen and HM accuracies reported here correspond to their maximum values when their optimal bias terms are used (maximum seen and maximum unseen accuracies usually employ different optimal values of bias terms). Adopting the same experiment setup similar to Table 1, baseline performance with calibrated bias terms is presented in Table 8. According to the results, the accuracies are higher when the calibrated biases are incorporated. However, the overall HM accuracies are still lower than results from our approaches. This still highlights vulnerability of these baselines to shortcuts.

(a) Seen/Unseen Accuracies of CGE   (b) Seen/Unseen Accuracies of TMN

Fig. 9: Seen/Unseen Accuracies of TMN and CGE baselines evaluated with different bias terms.

Here, we also investigate why seen accuracies of certain baselines are low in Table 1 (e.g., CGE on Color-Fashion). We can understand this behavior by observing seen/unseen accuracies using different bias terms. According to Figure 9a, the seen accuracy of CGE on Color-Fashion can be as high as 88.1 (similar to Table 8) when low bias term is used. However, in our experiment, we choose not to use bias terms for evaluation as per the reasons described above. Therefore, the reported seen accuracies on Table 1 are computed with bias terms of zero values. From Figure 9a, the seen accuracy of CGE on Color-Fashion is reduced to 21.6 (similar to Table 1). In contrast to CGE, the seen accuracy of TMN with zero bias term is already high (see Figure 9b). Therefore, we do not see low seen accuracy of TMN on Table 1.

A.7 Sweeping Weight of Loss for the Source Domain

The hyperparameter λ is used to weight the importance of $\mathcal{L}_{source}$ during training. Here we investigate its impact on the generalization performance attained in the target domain. Results of our analysis are shown in Figure 10. We note that, for FactorSRC and FactorSRC-CI, the harmonic mean of seen and unseen accuracies increases with higher λ values. This suggests that these two models are less sensitive to biases in the target dataset when λ is increased. High values of λ encourage FactorSRC and FactorSRC-CI to be more similar to FactorSRC-IL as $\mathcal{L}_{target}$ becomes less important to update G relative to $\mathcal{L}_{source}$. FactorSRC-IL, on the other hand, performs consistently when $\lambda > 0$. This result is reasonable since, when the IL constraint is introduced, $\mathcal{L}_{source}$ and $\mathcal{L}_{target}$ are independently used to update different parts of the network (they update $\{G, H_s\}$ and $\{H_t\}$ respectively). We note that, even though the higher λ leads to better performance, we reserve to use λ at 10 in our experiment so that we can study

effects from other loss terms. It should be noted that changing the value of λ here does not play a major role in our analysis since the key trends would be the same.
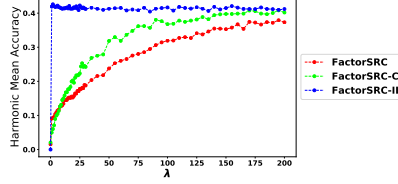

Fig. 10: HM Accuracies using the DiagVib-Animal target domain and the DiagVib-Caltech source domain with different λ values to weight the importance of $\mathcal{L}_{source}$. Notice that, higher λ values encourage models to behave closer to FactorSRC-IL. We sample λ on the low values with higher frequency to better highlight the faster increasing trends.

A.8 Cross-Factor Independence Constraint Algorithm

The Cross-Factor Independence constraint is implemented as a two-step optimization approach. In the first step, we update H' by minimizing the sum of cross entropy loss terms for all cross-factor predictions, i.e.,

$$\mathcal{L}_{H'} = \sum_{\forall k_1,k_2;k_1 \neq k_2} CE(H'_{k_1 k_2}(z_{k_1}), y_s^{k_2}) \ . \tag{3}$$

Subsequently, we update the whole network by minimizing the combination of $\mathcal{L}_{target}, \mathcal{L}_{source}$, and an additional independence loss $\mathcal{L}_{CI}$. In principle, $\mathcal{L}_{CI}$ could be formulated as $-\mathcal{L}_{H'}$ but we found that this leads to training instabilities due to the fact that such a loss is unbound. Instead, we minimize the cross entropy between the predictions of H' and a uniform label distribution. This encourages each factor representation to be uninformative with respect to all other factors. Mathematically, $\mathcal{L}_{CI}$ can be written as follows:

$$\mathcal{L}_{CI} = \gamma \sum_{\forall k_1,k_2;k_1 \neq k_2} CE\left(H'_{k_1 k_2}(z_{k_1}), \frac{\mathbf{1}^{N_{\mathcal{F}_s^{k_2}}}}{N_{\mathcal{F}_s^{k_2}}}\right) \tag{4}$$

, where $\mathbf{1}^N$ indicates a vector of ones with length N, $N_{\mathcal{F}_s^k}$ is the number of factor values of the k-th factor and $\gamma \geq 0$ is a hyperparameter (we use $\gamma = 5$).

A.9 Seen Accuracy form FactorSRC-IL on DiagVib-Animal

According to the result from Table 1, we notice that, on DiagVib-Animal, even though the HM accuracy of FactorSRC-IL is significantly higher than all other

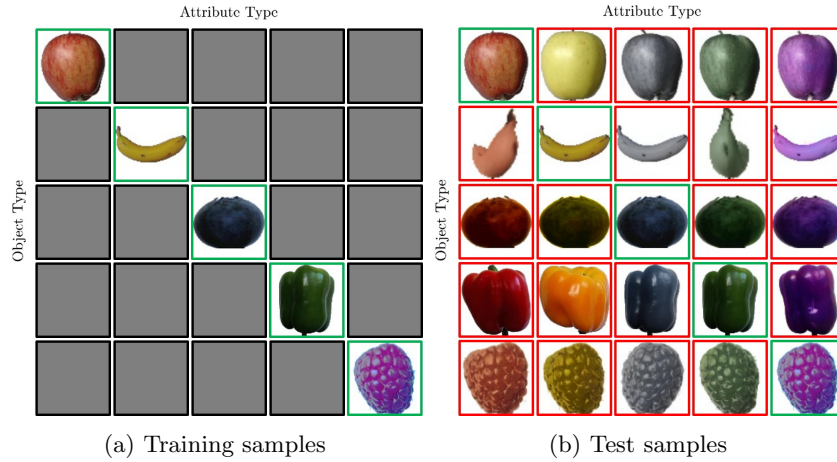(a) Training samples   (b) Test samples

Fig. 11: Examples of Color-Fruit dataset images. (a) Training samples containing images of fully-correlated attribute-object combinations denoted with green-bordered boxes (One object type always has one color and vice versa). (b) Test samples are, on the other hand, uncorrelated i.e., consisting of images with any attribute-object combinations (i.e., each object (fruit type) can appear with any attributes (color)). Fruit images whose colors are not available in the original Fruits 360 dataset are obtained by using the recolorization technique in [29]

approaches, the seen accuracy is dropped significantly (to 56.3%). The drop of the seen accuracy only presents in the case of DiagVib-Animal but not other target domains. We suspect that this behavior could stem from the lower random chance accuracy (1% on DiagVib-Animal compared to 4% and 4.7% on other target domains) or just the complexity of the DiagVib-Animal (with high intra-class variations and various backgrounds). In this regard, we conduct an experiment with reduced number of attribute/object labels from 10 to 5 so that it has the random chance accuracy of 4% which is the same as the one of Color-Fruit. In this regard, seen, unseen and HM accuracies on this reduced version of the DiagVib-Animal target domain are 74.2%, 52.6% and 61.4% respectively. Notice that, the seen accuracy is higher than the one on the original version but it is still relatively lower compared to the seen accuracies on other target domains. We can, therefore, conclude that the lower of the seen accuracy on DiagVib-Animal stems not only from its lower random chance accuracy but also from the complexity of the target domain itself.

A.10   Color-Fruit Dataset Generation

In order to generate the *Color-Fruit* dataset, used in our experiments, we use fruit images from the Fruits 360 dataset [20]. Five fruits (Apple, Banana, Blueberry, Pepper and Raspberry) are selected as they have distinct colors (red, yel-

low, blue, green and magenta), which facilitate the evaluation of compositional generalization in the case of fully-correlated seen combinations.

During evaluation, however, fruits with different colors are required. Thus, we perform recolorization of images in the test split using the approach described in [29]. Basically, an original test image is recolorized into median colors of all other fruits (e.g. a banana image is transformed such that it has a color similar to that of an apple, a blueberry, a pepper and a raspberry). More detailed visualization of the dataset is presented in Figure 11.