Causal Attention for Unbiased Visual Recognition

Tan Wang[1], Chang Zhou[2], Qianru Sun[3], Hanwang Zhang[1]
[1]Nanyang Technological University [2]Damo Academy, Alibaba Group [3]Singapore Management University
TAN317@e.ntu.edu.sg, zhouchang.zc@alibaba-inc.com, qianrusun@smu.edu.sg, hanwangzhang@ntu.edu.sg

Abstract

Attention module does not always help deep models learn causal features *that are robust in any confounding context, e.g., a foreground object feature is invariant to different backgrounds. This is because the confounders trick the attention to capture spurious correlations that benefit the prediction when the training and testing data are IID (identical & independent distribution); while harm the prediction when the data are OOD (out-of-distribution). The sole fundamental solution to learn causal attention is by causal intervention, which requires additional annotations of the confounders, e.g., a "dog" model is learned within "grass+dog" and "road+dog" respectively, so the "grass" and "road" contexts will no longer confound the "dog" recognition. However, such annotation is not only prohibitively expensive, but also inherently problematic, as the confounders are elusive in nature. In this paper, we propose a causal attention module (CaaM) that self-annotates the confounders in unsupervised fashion. In particular, multiple CaaMs can be stacked and integrated in conventional attention CNN and self-attention Vision Transformer. In OOD settings, deep models with CaaM outperform those without it significantly; even in IID settings, the attention localization is also improved by CaaM, showing a great potential in applications that require robust visual saliency. Codes are available at* https://github.com/Wangt-CN/CaaM.

1. Introduction

Do you think attention [59, 53] would always capture the salient regions in an image? No, as shown in Figure 1 (a, top), due to the lack of region-level labels, "learning to attend" is a *de facto* weakly-supervised task. Or do you think attention would always improve performance? Probably yes, after all, "Attention is All You Need" [14, 16, 44]. As shown in Figure 1 (a, top), even if the attended region is wrong, the model still makes correct predictions. In conventional IID settings, where the training and testing data are identically and independently distributed, the model equipped with attention is indeed better (red bar is higher than black bar in Figure 1 (b)).

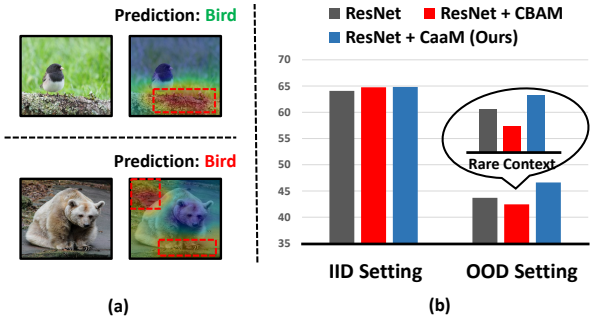

Figure 1. (a) The qualitative attention maps of two images in NICO [21] using ResNet18 with CBAM [57]. (b) The accuracies of three methods: ResNet18, ResNet18+CBAM [57] and + CaaM (Ours) in both IID and OOD settings.

However, few people realize that attention may do evil in OOD settings, where the testing data are out of the training distribution. For example, as shown in Figure 1 (a, top), the attention considers the "ground" region as the visual cue of the "bird" class, because most training "bird" images are in "ground" context; but, when the test image is "bear in ground" (bottom), the attention misleads the model to still predict "bird". Figure 1 (b) reports that the attention model is even worse than the non-attention baseline in OOD setting (red bar is lower than black bar), where the gap is further amplified by the rare object and context combination in training. Unfortunately, when we deploy such vision systems in critical domains such as car autopilot, it is often the rare case that causes fatal accidents, *e.g.*, recognizing a "white" truck as "white" clouds[1].

Astute readers who are knowledgeable in causality [27, 42] may point out that the key reason for the bipolar role of attention in IID and OOD is due to the confounding effect [55, 66, 65, 60]. In visual recognition, the causal pursuit between the input image X and the output label Y is confounded by a common cause: the context S. To see the effect, during data collection, X is usually found

[1]https://www.youtube.com/watch?v=X3hrKnv0dPQ

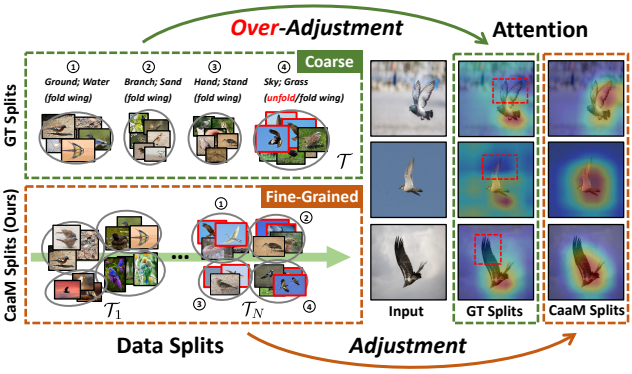

Figure 2. The comparison of attention maps between the partition-based intervention methods [4, 49] and our unsupervised CaaM with each training data splits. $\mathcal{T}_N$ denotes the N-th partition. Birds unfolding wings are highlighted with red boxes.

in S, and thus S is a contextual cue to recognize Y (*i.e.*, $X \leftarrow S \rightarrow Y$). After training, the model recklessly exploits the statistical cues of S as a feature of X to predict Y (*e.g.*, most training "bird" is on "ground" in Figure 1 (a, top)); however, in testing, if $X \not\rightarrow Y$ (*e.g.*, "bear" image $\not\rightarrow$ "bird"), seeing S misleads $X \rightarrow Y$ (*e.g.*, the "ground" pattern always recalls "bird"). In Section 3.1, we will revisit the above causality in detail.

The sole solution to mitigate the confounding bias is by *causal intervention* [42]. For example, Arjovsky *et al*. [4] and Teney *et al*. [49] suggest to collect "bird" images under every context (*i.e.*, adjusting the contexts of "bird"). In each context split (*e.g.*, "ground", "water" and "sky"), the attention does not attend to the context as it is no longer discriminative in the split. So, the combined "ground/water/sky + bird" attention will tend to focus on the "bird" unbiased towards any context. However, it is impractical to perform the causal intervention like above. Despite the expensive cost of extra annotations, it suffers from the following deficiency.

In practice, it is impossible to collect the samples of a class in any context, *e.g.*, it is hard to find "fish" in "sky". Technically, such absent context of a class violates the confounder positivity assumption of casual intervention [25] (see Section 4.4 for the poor performance caused by the violation). Therefore, we have to merge the ground-truth contexts into bigger splits to include all classes (*e.g.*, merging "water" and "ground" as one split in Figure 2 (left top)).

However, such coarser contexts will lead to the *over-adjustment* problem — the intervention not only removes the context, but also hurts beneficial causalities (*e.g.*, the object part). Figure 2 (left top) illustrates a real example. Recall that the aforementioned context split-based intervention removes the non-causal features of different contexts. Unfortunately, the causal feature of "bird"—"wing"—is also removed (see red dashed boxes in the attention). This is because all the birds in "sky" context unfold their wings: split ④ does not only represent "sky" and "grass", but also "wing". We formally formulate this problem in *improper causal intervention* in Section 3.1.

In this paper, we propose a causal attention module (CaaM \ka:m\) which generates data partition iteratively and self-annotates the confounders progressively to overcome the over-adjustment problem. Compared to the coarser contexts, multiple CaaM partitions are fine-grained and more exact to describe the comprehensive confounder. As shown in Figure 2 (left bottom), each split of partition $\mathcal{T}_N$ has images of "bird" unfolding "wings" (see images in red boxes). This encourages the model to capture the "wing" feature (see the improved visual attention), because "wing" no longer co-varies with the ④ "Sky; Grass" context. Technically, besides a standard attention that attends to the desired causal features (*e.g.*, foreground), CaaM has a complementary attention that deliberately captures the confounding effect (*e.g.*, background). The two disentangled attentions are optimized in an adversarial minimax fashion, which progressively constitute the confounder set and mitigates the confounding bias in unsupervised fashion.

We analyze how CaaM learns better causal features than existing baselines in Section 3.2. In Section 3.3, we show two deployment examples on the popular attention-based deep models: CBAM-based CNN [57] and Transformer-based T2T-ViT [63]. Extensive qualitative and quantitative experimental results in Section 4 demonstrate the consistent gain achieved by CaaM.

Our technical contributions are summarized as:

- A novel yet practical visual attention module CaaM who learns causal features that are robust in OOD settings without sacrificing the performance in IID settings.
- We offer a causality-theoretic analysis to guarantee the superiority of CaaM.
- The design of CaaM is generic to popular deep networks.

2. Related Work

Visual Attention. We consider both conventional attention [57, 28] and the recent self-attention [53, 16, 50, 63, 24, 52, 56]. Over the past years, although they had evolved into various models, the key mechanism is still to select the informative features (subject to a context or token query) [12, 39, 10]. Due to that the selection has no localized strong supervision, attention is inherently biased in OOD settings. Most recently, Yang *et al*. [61] also investigated the biased attention. However, our CaaM is fundamentally different: 1) Different assumptions. [61] is for visual-language tasks and assumes the mediator is visible from the vision-language context; however, in general visual recognition, this requirement is inapplicable. 2) Different methods. [61] uses front-door adjustment [43], while our CaaM is back-door adjustment. More importantly,

CaaM can self-annotate the confounder in an *unsupervised* way. In this view, CaaM is also technically different from recent visual causal inference works [55, 66, 47, 41, 64, 29].

OOD Generalization. Machine learning is always challenged by OOD problems [36, 22, 1], such as debiasing [18, 33, 30, 11, 54, 35], domain adaption [6, 40, 17, 51, 19] and long-tailed recognition [32, 38, 46]. We focus on the most challenging yet practical OOD setting [5, 21, 23] where the OOD visual semantics are unlabeled (different from long-tailed) and ubiquitous (different from domain adaptation). Moreover, we follow and reveal the recent progress of invariant risk minimization (IRM) [4, 34, 45, 13, 3, 2, 37, 62] as a kind of causal intervention, which however suffers from the over-adjustment discussed later. Our CaaM utilizes the complementary attention and an iterative adversarial training pipeline to overcome this problem.

3. CaaM: Causal Attention Module

3.1. Causal Preliminaries

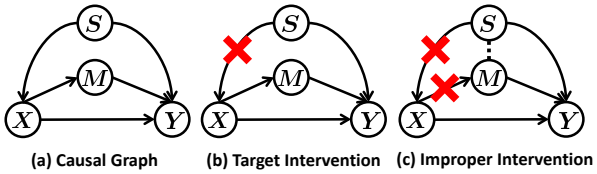

Figure 3. The causal graphs of visual recognition.

Causal View of Biased Recognition. We introduce the formulation of causality for visual recognition tasks by using a Structure Causal Model (SCM) [42]. We build this SCM by inspecting on the causalities among the key components: image X, label Y, mediator M, and confounder S. We illustrate the SCM in Figure 3 (a) where each direct link denotes a causal relationship between two nodes.

$\boldsymbol{X \rightarrow Y}$ denotes the desired causal effect from image content X to label Y, as image is labeled for its content. We call a recognition model is unbiased if it identifies $X \rightarrow Y$.

$\boldsymbol{X \leftarrow S \rightarrow Y}$. $S \rightarrow X$ denotes that unstable context S determines what to picture in image X [66]. For instance, S determines where to put "birds" and "ground" in an image. $S \rightarrow Y$ exists because model inevitablely uses the contextual cue to recognize Y. In the SCM, we can clearly see how S confounds X and Y via the backdoor path $X \leftarrow S \rightarrow Y$. Taking the bear-bird example again (Figure 1 (a)), though the "bear" image (X) has no causal relationship with the label of "bird" (Y), the backdoor path creates a spurious correlation between them (through S) and thus yields the wrong prediction of "bird" from a "bear" image.

$\boldsymbol{X \rightarrow M \rightarrow Y}$ is a beneficial causal effect for robust recognition, where M is a mediator that are invariant in different distributions. For example, M can be discriminative object parts, *e.g.*, "bird" has "wing". Note that M can be hidden in the causal path $X \rightarrow Y$. Here we define it as an explicit graph node for the convenience of following mathematical derivations.

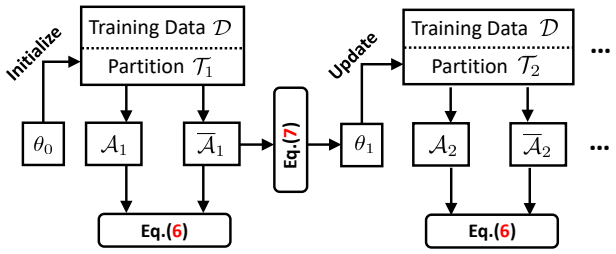

Figure 4. The training pipeline of our CaaM. In each iteration, it contains a Mini-Game: Joint Training (Eq. (6)) and a Maxi-Game: Partition Update (Eq. (7)). The subscript of $\mathcal{T}$, $\mathcal{A}$, and θ is iteration index. The resultant attention for unbiased recognition is $\mathcal{A}_N$ after N iterations.

Causal Intervention by Data Partition. Data partition (or environment split) [4] is an effective implementation of causal intervention. It first partitions the training data into a set of hard splits $\mathcal{T} = \{t_1, ..., t_m\}$, each of which represents a confounder stratum, allowing the model trained across different splits invariant to the confounder. We show that data partition is equivalent to the well-known backdoor adjustment [42]:

$$P(Y|do(X)) = \sum_{t \in \mathcal{T}} P(Y|X,t)P(t), \tag{1}$$

where $P(Y|X,t)$ denotes the prediction of the classifier trained in split t and $P(t) := 1/m$. We illustrate $P(Y|do(X))$ in Figure 3 (b). The interpretation is that $do(X)$ cuts off the confounding path $X \leftarrow S \rightarrow Y$, leaving only robust paths $X \rightarrow Y$ and $X \rightarrow M \rightarrow Y$. However, existing methods based on data partition [4, 49] only assumes a single yet small set of splits, which is far from sufficient for Eq. (1).

Improper Causal Intervention. In visual recognition, a perfect partition as in Eq. (1) is not easy to obtain because the conventional context-based partition annotation [4, 21] does not disentangle the confounder and mediator. Thus, straightforwardly adjusting the mediator hurts feature learning [47]. Below, we elaborate the reasons using causal formulations. Suppose that the data partition $\mathcal{T}$ only contains the confounder. Then, we can mitigate S without blocking M. By applying Bayes rules, Eq. (1) can be re-written as:

$$P(Y|do(X)) = \sum_{s \in S} \sum_{m \in M} P(Y|X,s,m)\underline{P(m|X)}P(s). \tag{2}$$

However, if each split in $\mathcal{T}$ contains both S and M, *i.e.*, $(S \not\perp\!\!\!\perp M) \mid X$. Eq. (1) will be re-derived to the false effect estimation:

$$P(Y|do(X)) = \sum_{s \in S} \sum_{m \in M} P(Y|X,s,m)\underline{P(m|X,s)}P(s), \tag{3}$$

where $P(m|X,s)$ is not equal to $P(m|X)$ in Eq. (2) due to $(S \not\perp\!\!\!\perp M) \mid X$. This means that the improper partition $\mathcal{T}$

indeed cuts off the robust mediation effect of $X \rightarrow M \rightarrow Y$, as shown in Figure 3 (c).

3.2. Training Pipeline

The iterative training pipeline of any model equipped with CaaM is illustrated in Figure 4. To enlarge the split number in Eq. (1), we discover partition $\mathcal{T}_i$ in each step. After N steps training, we can approximate Eq. (1) by $P(Y|do(X)) \approx \sum_i^N \sum_{t\in\mathcal{T}_i} P(Y|X,t)P(t)$. For the disentanglement of confounder and mediator, we design a pair of complementary attention modules $\mathcal{A}$ and $\overline{\mathcal{A}}$, where $\mathcal{A}$ is for attending to features of causal effect $X \rightarrow M \rightarrow Y$ and $X \rightarrow Y$, while $\overline{\mathcal{A}}$ is for attending to the confounding effect $X \leftarrow S \rightarrow Y$. Note that the roles of $\mathcal{A}$ and $\overline{\mathcal{A}}$ are adversarial, as the former aims to predict correctly using robust feature while the latter aims to capture bias. Therefore, the adversarial training encourages disentanglement, and we can use $\overline{\mathcal{A}}$ to update the partition $\mathcal{T}_{i+1}$. We illustrate the convergence of our training pipeline in Appendix. Next, we will detail the training losses.

Cross-Entropy Loss. This loss is to ensure that $\mathcal{A}$ and $\overline{\mathcal{A}}$ combined will capture the biased total effect from $X \rightarrow Y$ regardless of causal or confounding effects; otherwise, they may disrespect the training data generative causality as assumed in Figure 3 (a). Note that such biased training practice is widely adopted in unbiased models [8, 41, 48].

$$\mathrm{XE}(f, \widetilde{x}, \mathcal{D}) = \mathbb{E}_{(x,y)\in\mathcal{D}}\, \ell\left(f(\widetilde{x}), y\right), \tag{4}$$

where $\widetilde{x} = \mathcal{A}(x) \circ \overline{\mathcal{A}}(x)$ and $\circ$ denotes feature addition, f is a linear classifier, and ℓ is the cross-entropy loss function.

Invariant Loss [4]. This loss is for learning $\mathcal{A}$ that is split-invariant made by the causal intervention in Eq. (1) with incomplete confounder partition $\mathcal{T}_i$:

$$\begin{aligned}\mathrm{IL}(g, \mathcal{A}(x), \mathcal{T}_i) = &\sum_{t\in\mathcal{T}_i} \mathrm{XE}(g, \mathcal{A}(x), t) \\ &+ \lambda \|\nabla_{\mathbf{w}=\mathbf{1}.\mathbf{0}} \mathrm{XE}(\mathbf{w}, \mathcal{A}(x), t)\|_2^2,\end{aligned} \tag{5}$$

where t is a data split, g is a linear classifier for robust prediction, $\mathbf{w}$ stands for a dummy classifier [4] used to calculate gradient penalty across splits and λ is the weight. During inference, $g(\mathcal{A}(x))$ will be deployed for unbiased recognition. See Appendix for further details.

Adversarial Training. This training disentangles $\mathcal{A}$ and $\overline{\mathcal{A}}$ with a Mini-Game and a Maxi-Game. Intuitively, the Maxi-Game takes the confounder feature in $\overline{\mathcal{A}}(x)$ to generate the data partition $\mathcal{T}_i$ (causal feature does not contribute to maximization). While the Mini-Game exclude such confounder feature from $\mathcal{A}(x)$ with $\mathcal{T}_i$ (confounder feature does not contribute to minimization).

Mini-Game: It is a joint training with XE and IL, plus a new adversary classifier h that specializes in the confounding effect caused by $\overline{\mathcal{A}}(x)$:

$$\min_{\mathcal{A},\overline{\mathcal{A}},f,g,h} \mathrm{XE}(f, \widetilde{x}, \mathcal{D}) + \mathrm{IL}(g, \mathcal{A}(x), \mathcal{T}_i) + \mathrm{XE}(h, \overline{\mathcal{A}}(x), \mathcal{D}), \tag{6}$$

Maxi-Game: A good partition update should captures stronger confounder that is NOT split invariant:

$$\max_{\theta}\ \mathrm{IL}(h, \overline{\mathcal{A}}(x), \mathcal{T}_i(\theta)) \tag{7}$$

where $\mathcal{T}_i(\theta)$ denotes partition $\mathcal{T}_i$ is decided by $\theta \in \mathbb{R}^{K\times m}$, K is the total number of training samples and m is the number of splits in a partition. $\theta_{p,q}$ is the probability of the p-th sample belonging to the q-th split ($t_q \in \mathcal{T}_i$).

3.3. Implementations of CaaM

We implement the proposed CaaM on two popular attention-based deep models: CBAM-based CNN [57] and Transformer-based T2T-ViT [63]. We call the result models as CNN-CaaM and ViT-CaaM, respectively. For simplicity, in this section we use $\mathbf{c}$ and $\mathbf{s}$ to denote the causal and confounder feature (*i.e.*, $\mathbf{c} = \mathcal{A}(x)$ and $\mathbf{s} = \overline{\mathcal{A}}(x)$).

3.3.1 CNN-CaaM

CBAM [57] sequentially adopts the channel and spatial attention module for adaptive CNN feature refinement — one of the most fundamental ways of computing attention in CNN. Given an input feature $\mathbf{x}$, the attention feature $\mathbf{x}'$ is computed as:

$$\mathbf{z} = \mathrm{CBAM}(\mathbf{x}), \quad \mathbf{x}' = \mathrm{sigmoid}(\mathbf{z}) \odot \mathbf{x}, \tag{8}$$

where $\mathbf{z} \in \mathbb{R}^{w\times h\times c}$ and $\odot$ denotes the element-wise product. Therefore, our CaaM attention calculus based on CBAM is defined as:

$$\textbf{CaaM}: \begin{cases} \mathbf{z} = \mathrm{CBAM}(\mathbf{x}), \\ \mathbf{c} = \mathrm{Sigmoid}(\mathbf{z}) \odot \mathbf{x}, \\ \mathbf{s} = \mathrm{Sigmoid}(-\mathbf{z}) \odot \mathbf{x} \end{cases} \tag{9}$$

where $\mathrm{Sigmoid}(\mathbf{z})$ and $\mathrm{Sigmoid}(-\mathbf{z}) = 1 - \mathrm{Sigmoid}(\mathbf{z})$ are complementary such as to disentangle $\mathbf{c}$ and $\mathbf{s}$ from the input feature $\mathbf{x}$. Below we elaborate the details of plugging CaaM in residual blocks, as illustrated in Figure 5 (a).

Disentanglement Block (D-Block). D-Block is the block that contains CaaM calculus to generate two attention features $\mathbf{c}$ and $\mathbf{s}$. Note that before D-Block, there can be any number of standard residual blocks [20]. The formulation of D-Block^{j+1} with residual connection is thus as follows,

$$\textbf{D-Block}^{j+1}: \begin{cases} \hat{\mathbf{c}}^j, \hat{\mathbf{s}}^j = \mathrm{CaaM}(\mathbf{x}^j), \\ \mathbf{c}^{j+1} = \hat{\mathbf{c}}^{j+1} + \mathbf{c}^j \quad (\text{Skip Connection}), \\ \mathbf{s}^{j+1} = \hat{\mathbf{s}}^{j+1} + \mathbf{s}^j \quad (\text{Skip Connection}) \end{cases} \tag{10}$$

where $\mathbf{x}^j$ is the feature output by the j-th residual block, and note that as shown in Figure 5 (a), the first D-Block is denoted as D-Block (Init.), which is slightly different from the following D-Block: 1) The skip connection is connected from the output of the standard ResNet blocks. 2) We remove skip connection on confounder feature $\mathbf{s}^j$ to distinguish it from the causal feature $\mathbf{c}^j$.

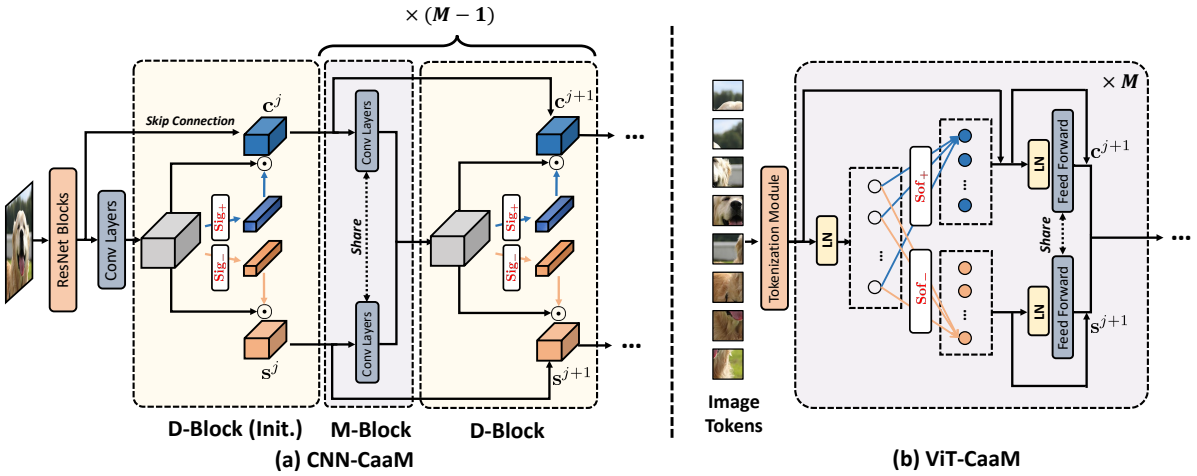

Figure 5. The network architectures of our CNN-CaaM based on CBAM [57] and our ViT-CaaM based on T2T-ViT [63]. Red formulas are used to generate our complementary attentions: Sig_+ denotes Sigmoid($\mathbf{z}$) and Sig_- for Sigmoid($-\mathbf{z}$). Sof_+ denotes Softmax($\mathbf{q}\mathbf{k}^{\mathrm{T}}/\sqrt{d_K}$) and Sof_- for Softmax($-\mathbf{q}\mathbf{k}^{\mathrm{T}}/\sqrt{d_K}$). For CNN-CaaM, D-Block is utilized to disentangle causal feature $\mathbf{c}$ (blue) and confounder feature $\mathbf{s}$ (orange) from the CNN feature $\mathbf{x}$. D-Block (Init.) denotes the first D-Block. While M-Block merges $\mathbf{c}$ and $\mathbf{s}$ with the convolution layer. Then M-Block and D-Block are stacked to progressively refine $\mathbf{c}$ and $\mathbf{s}$.

Merge Block (M-Block). As shown in Figure 5 (a), before D-Block, $\mathbf{c}$ and $\mathbf{s}$ are input into M-Block for feature fusion to get ready for the following D-Block. We denote the M-Block (the one left before D-Block) as M-Block^j, where $j+1$ is the index of D-Block, and introduce its formulation as follows:

$$\textbf{M-Block}^j : \mathbf{x}^j = \text{Conv}(\mathbf{c}^j) + \text{Conv}(\mathbf{s}^j). \tag{11}$$

Iterating Eq. (10) and Eq. (11) yields the multi-layer CaaM (M-Block→D-Block→M-Block). During inference, we use the final causal feature $\mathbf{c}^{j+M-1}$ for robust prediction.

3.3.2 ViT-CaaM

We build ViT-CaaM based on an advanced ViT model called Token-to-Token (T2T)-ViT [63] in which the T2T module aims to address the issue of simple tokenization in vanilla ViT [16]. Our CaaM is only plugged in the ViT attention modules of T2T-ViT, thus it is suitable for any ViT-based model, *e.g.*, DeiT [50] and VT [58].

As shown in Figure 5 (b), given the input feature $\mathbf{x} \in \mathbb{R}^{n\times d}$, CaaM first computes the query, key and value vectors $\mathbf{q}, \mathbf{k}, \mathbf{v} \in \mathbb{R}^{n\times d_k}$ using the standard self-attention, where n is the number of image patches, and d and d_k are feature dimensions. Then, it calculates the complementary attentions using Softmax functions. The overall formulation of CaaM in ViT is thus as follows:

$$\textbf{CaaM} : \begin{cases} \mathbf{q}, \mathbf{k}, \mathbf{v} = \mathbf{W}_q\mathbf{x}, \mathbf{W}_k\mathbf{x}, \mathbf{W}_v\mathbf{x}, \\ \mathbf{c} = \text{Softmax}(\frac{\mathbf{q}\mathbf{k}^{\mathrm{T}}}{\sqrt{d_K}})\mathbf{v}, \\ \mathbf{s} = \text{Softmax}(-\frac{\mathbf{q}\mathbf{k}^{\mathrm{T}}}{\sqrt{d_K}})\mathbf{v} \end{cases} \tag{12}$$

Different from CNN-CaaM, ViT-CaaM does not have D-Block and M-Block due to the transformer architecture. Given the input feature $\mathbf{x}^j$ (*i.e.*, the output feature of the j-th transformer module), the j+1-th module disentangles it to be intermediate features ($\hat{\mathbf{c}}^{j+1}$ and $\hat{\mathbf{s}}^{j+1}$) by applying CaaM attentions, as indicated by the blue and yellow links in Figure 5 (b). Then, $\hat{\mathbf{c}}^{j+1}$ and $\hat{\mathbf{s}}^{j+1}$ are fed into an MLP to generate the causal and confounder features, *i.e.*, $\mathbf{c}^{l+1}$ and $\mathbf{s}^{l+1}$. Note that 1) layer norm (LN) and skip connection are applied in every block, following the standard ViT [16, 63]; and 2) similar to CNN-CaaM, we omit the first skip connection when generating $\hat{\mathbf{s}}^{j+1}$ in order to avoid the entanglement between $\hat{\mathbf{c}}^{j+1}$ and $\hat{\mathbf{s}}^{j+1}$. Thus, the basic block of ViT-CaaM can be formulated as follows:

$$\begin{cases} \hat{\mathbf{c}}^{j+1}, \hat{\mathbf{s}}^{j+1} = \text{CaaM}(\text{LN}(\mathbf{x}^j)), \\ \mathbf{c}^{j+1} = \text{MLP}(\text{LN}(\hat{\mathbf{c}}^{j+1} + \mathbf{x}^j)) + \hat{\mathbf{c}}^{j+1} + \mathbf{x}^j, \\ \mathbf{s}^{j+1} = \text{MLP}(\text{LN}(\hat{\mathbf{s}}^{j+1})) + \hat{\mathbf{s}}^{j+1}, \\ \mathbf{x}^{j+1} = \mathbf{c}^{j+1} + \mathbf{s}^{j+1} \end{cases} \tag{13}$$

Iterating Eq. (13) yields the multi-layer ViT-CaaM. We use the final causal feature $\mathbf{c}^{j+M}$ for prediction in inference.

4. Experiments

4.1. Datasets and Settings

NICO [21] is a real-world image dataset designed for OOD settings. It contains 19 object classes, 188 contexts and nearly 25,000 images in total. Each image has an object label as well as a context label. It is thus convenient to "shift" the distribution of a class, *i.e.*, by adjusting the proportions of specific contexts for training and testing samples.

Our Settings: We use the animal subset of NICO. For each animal class, we randomly sample its images and make sure the context labels of those images are within a fixed set of 10 classes (*e.g.*, "snow", "on grass" and "in water"). Based on these data, we propose a challenging OOD setting including

three factors regarding contexts: 1) Long-Tailed—training context labels are in long-tailed distribution in each individual class, *e.g.*, "sheep" might have 10 images of "on grass", 5 images of "in water" and 1 image of "on road"; 2) Zero-Shot—for each object class, 7 out of 10 context labels are in training images and the other 3 labels appear only in testing; and 3) Orthogonal—the head context label of each object class is set to be as unique (dominating only in one object class) as possible. Please kindly refer to the Appendix for more details of our settings.

ImageNet-9 [31] Following the related work [5], we also evaluate our models on ImageNet-9, which is a subset of ImageNet containing 9 super-classes with 54,600/2,100 training and validation samples.

Our Settings: We have three settings to evaluate our model performance on the ImageNet-9. 1) Biased—This is a conventional metric that the accuracy is measured on the whole validation set, serving as a in-distribution testing. 2) Unbiased—This is taken as a proxy to the perfectly debiased test data. To achieve it, we follow [5] to categorize images into different contexts (*i.e.*, assign context labels to images) by clustering image textures into several groups. We compute the accuracy for each image cluster and average these accuracies as the final unbiased metric. 3) ImageNet-A [23]. It was proposed as a particularly challenging OOD testset of ImageNet. It contains 7,500 real-world images that fool the image classifiers trained on standard ImageNet. Such "fool" was caused by confounders— "the model heavily rely on the colors, textures and frequently appearing backgrounds" [23]. Therefore, this testing set exactly validates our model performance of deconfounding. Please kindly refer to the Appendix for more details.

4.2. Implementation Details

We implement CaaM on two backbones: ResNet18 [20] for CNN-CaaM and T2T-ViT7 [63] for ViT-CaaM. Compared to the original ViT, T2T-ViT introduces a layer-wise Tokens-to-token (T2T) transformation to progressively structurize the image to tokens by aggregating neighboring information. Below we introduce the comparable baseline methods, *i.e.*, debias methods and intervention methods.

Debias Methods. We compare our CaaM with two SOTA methods: RUBi [8] and ReBias [5]. RUBi explicitly learns a biased model using biased input, and then performs debiasing on a standard model by re-weighting its prediction logits where weights are generated by the biased model. The other SOTA is ReBias. It utilizes a small receptive fields CNN (BagNet [7]) to explicitly encode context bias in a model, and the debiased representation is encouraged to be statistically independent from it.

Intervention Methods. We compare our CaaM with three SOTA causal intervention methods: IRM [4], REx [34] and Unshuffle [49]. IRM claimed that the robust representation derives the image classifiers who can make invariant predictions across different contexts (for the same object class). REx is an improved version of IRM and its key is to encourage robustness over affine combinations for the training risks. Unshuffle [49] extended IRM to a real vision-language tasks—visual question answering. It trains an individual classifier for each data split and applies a variance regularizer on classifier weights.

Note that all above methods require the annotation of data splits: either from manual labeling [4, 9, 2] or from pre-defined clustering [49, 5]. While our CaaM does not need such annotations, leading to the intervention in an unsupervised fashion. To further show our superiority, we conduct two groups of comparisons to intervention methods (in the bottom blocks of Table 1): one allows all models to be trained with human annotated splits (w/ H.A. $\mathcal{T}$), and the other one without (w/o H.A. $\mathcal{T}$). On the NICO dataset, H.A. $\mathcal{T}$ is built by using context labels, *i.e.*, images containing same context are in the same split. On the ImageNet-9 dataset, H.A. $\mathcal{T}$ is built by clustering context features into several splits, following [5].

4.3. Comparing to SOTAs

Table 1 shows all comparisons we conduct on the NICO, ImageNet-9 and ImageNet-A(test only) datasets. It is obvious that our CaaM achieves the top performance across all settings. It is worth highlighting that 1) in the setting "w/ H.A. $\mathcal{T}$", our CaaM surpasses intervention methods by clear margins, *e.g.*, 2.1% and 1.2% higher than IRM and Unshuffle, respectively, on the challenging ImageNet-A (with CNN); and 2) these margins are even larger, *e.g.*, increased to 5% for ImageNet-A (with CNN), in the more difficult setting "w/o H.A. $\mathcal{T}$", where ours get improved but others' are reduced. These validate that our self-annotating for partitions—progressively updating $\mathcal{T}$ using CaaM—indeed achieves the superior representations of confounders than using any hard or manual splits as in [4, 49, 34].

4.4. Ablation Study

Q1: *What are the optimal hyperparameters of CaaM?* We replace the last M blocks of CNN (or the last M layers of ViT) and vary of value of M to find out how many attention blocks we need to get the best performance. Similarly, we use m context splits and vary its values to find out the optimal number of partitions.

A1: We can see from the top block of Table 2 that the performance of CaaM saturates around M=2 for CNN-CaaM (M=4 for ViT-CaaM). As we add CaaM layers from top to bottom, this is perhaps because the lower CNN feature maps do not emerge foreground and background semantics yet. On the middle block of Table 2, we find that m=4 is the best but the accuracy differences with other values are not significant, *i.e.*, not sensitive to m.

	Model	CNN-Based					ViT-Based				
		NICO		ImageNet-9 [31]		ImageNet-A [23]	NICO		ImageNet-9 [31]		ImageNet-A [23]
		Val	Test	Biased	Unbiased [5]	Test	Val	Test	Biased	Unbiased [5]	Test
Conv.	ResNet18 [20]	43.77	42.61	95.00	94.40	33.67	–	–	–	–	–
	ResNet18+CBAM [57]	42.15	42.46	94.81	94.09	34.31	–	–	–	–	–
	T2T-ViT7 [63]	–	–	–	–	–	36.23	35.62	88.76	88.35	31.28
	RUBi [8]	43.86	44.37	94.81	94.27	34.13	35.27	34.15	87.95	87.48	29.90
	ReBias [5]	44.92	45.23	95.20	94.89	34.26	35.28	35.74	88.99	88.32	29.33
	Cutout [15]	43.69	43.77	95.24	94.81	34.68	35.31	33.69	87.52	86.47	27.97
	Mixup [67]	44.85	41.46	95.43	94.79	**37.71**	37.85	34.31	89.72	88.66	30.73
w/ H.A. $\mathcal{T}$	IRM [4]	40.62	41.46	94.13	94.41	33.52	36.46	34.38	89.43	88.87	30.17
	REx [34]	41.00	41.15	94.15	94.28	33.18	36.23	33.46	88.52	87.26	29.18
	Unshuffle [49]	43.15	43.00	94.71	94.33	34.41	37.38	36.00	87.38	86.86	28.61
	CaaM (Ours)	**45.46**	**45.77**	**95.52**	**94.96**	35.60	**38.08**	**37.54**	**90.05**	**89.35**	**32.01**
w/o H.A. $\mathcal{T}$	IRM [4]	40.54	41.23	94.09	94.32	33.39	33.76	33.77	89.62	88.98	29.25
	REx [34]	40.85	41.52	93.26	93.79	32.84	35.62	34.00	88.68	87.01	28.72
	Unshuffle [49]	41.69	41.61	94.81	94.30	34.04	33.62	32.92	88.38	87.39	28.52
	CaaM (Ours)	**46.38**	**46.62**	**96.19**	**95.83**	**38.55**	**38.00**	**37.61**	**90.33**	**90.01**	**32.38**

Table 1. Recognition accuracies (%) based on ResNet18 and T2T-ViT7, on the NICO, ImageNet-9 and ImageNet-A datasets. "Conv.", "w/ H.A. $\mathcal{T}$", "w/o H.A. $\mathcal{T}$" denote conventional methods, causal intervention with human-annotated partitions $\mathcal{T}$ (*i.e.*, ground truth context splits) and intervention without partition annotations, respectively. Our results are highlighted. The **best** and **second best** accuracies are marked for all settings.

	Settings	CNN-CaaM		ViT-CaaM	
		Val	Test	Val	Test
Num L.	M=1	43.92	44.54	35.54	36.77
	M=2	45.46	45.77	37.89	37.46
	M=4	44.15	45.31	**38.08**	37.54
Num S.	m=2	43.98	44.92	37.87	37.32
	m=4	45.46	45.77	**38.08**	37.54
	m=8	45.23	45.74	37.94	37.23
T.S.	Reboot Training	44.23	44.46	36.69	35.46
	Randomize θ	44.46	43.38	37.12	36.15
	CaaM	**46.38**	**46.62**	38.00	**37.61**

Table 2. Ablation studies of our proposed CNN-CaaM and ViT-CaaM on NICO dataset. **Num L.**, **Num S.** and **T.S.** denote number of layers, number of splits and training schedule respectively.

Setting	Model	CNN-Based	ViT-Based
	Attention	69.73	55.36
w/ Partition	Interv. [49]	73.61	56.71
	CaaM (Ours)	77.52	58.24
w/o Partition	**CaaM (Ours)**	**78.37**	**58.83**

Table 3. The attention map accuracy (%) of using different models on ImageNet-9 test set with ground truth bounding boxes. "Attention" denotes the conventional attention model, *i.e.*, ResNet+CBAM and ViT. "Interv." is the abbreviation of the intervention method [49].

Q2: *What is the advantage of progressively updating and aggregating the effects of different partitions? Is the Maxi-Game indispensable?* We conduct two ablation studies: one is to omit the optimization in Eq. (6) and each phase we randomize the weights of those parameters, *e.g.*, $\mathcal{A}$ and $\bar{\mathcal{A}}$, which we denote as "Reboot Training"; the other is to omit the optimization in Eq. (7), and similarly, each phase we randomize the weights of θ, *i.e.*, "Randomize θ".

A2: We show the corresponding results on the bottom block of Table 2. It is clear that "Reboot Training" in each phase results in sharp performance drops for all models, by 2% on average. Besides, "Randomize θ" also brings clear performance drops while the reduced margins are smaller than those of "Reboot Training". These validate that our CaaM is a organic integrity composed of collaborative and optimizable modules.

Q3: *Can CaaM achieve robust attention?* To evaluate the exactness of the attention map generated by CaaM quantitatively, we calculate the attention accuracy of our CaaM and baseline methods with the groud truth object bounding box coordinates of ImageNet-9 test set. Specifically, the attention accuracy is given by the ratio between the attention area in bounding box and the whole attention area. Details are given in Appendix.

A3: We report the attention accuracy in Table 3. Compared to the conventional attention, the intervention method with ground truth context partition can achieve better performance; while our CNN-CaaM and ViT-CaaM largely outperform these two methods in both settings (*i.e.*, with and without partitions). This result fully demonstrates the both effectiveness of our multi-layer complementary attention and the adversarial training pipeline.

Q4: *Why merge the ground truth contexts into bigger splits?* Recall that in Section 1, we explain that the context absence of a class violates the positivity assumption. To evaluate the effect, we provide the detailed results on the NICO dataset with different number of data splits in Figure 7 (a).

A4: We can see that the accuracies of the intervention keep relatively stable for $2-4$ splits, but have a huge drop when grouping according to each context due to the violation.

Q5: *Does CaaM boost the recognition of both the samples with frequent contexts and with rare contexts?* We show the

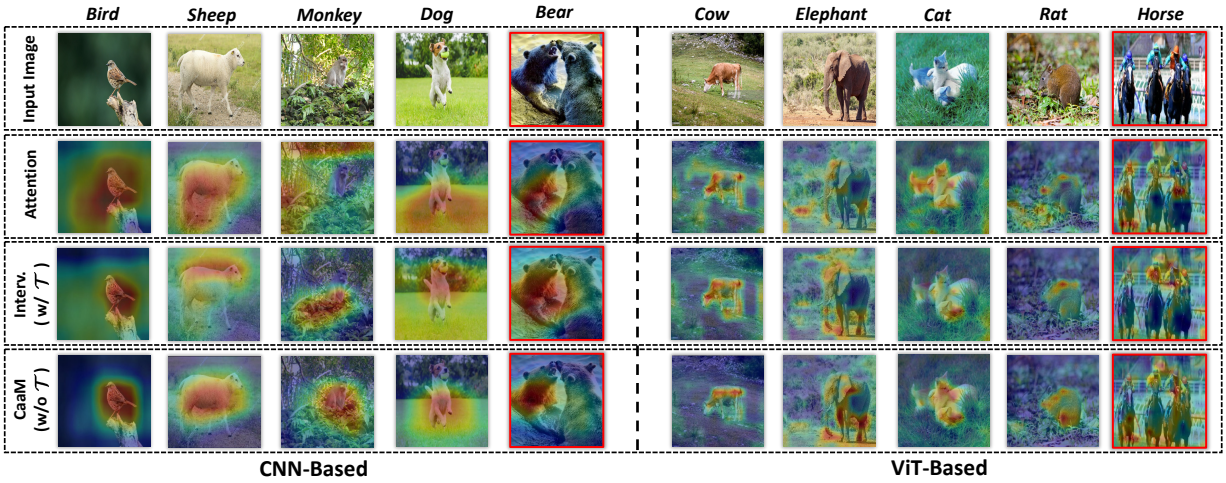

Figure 6. Visualization of the attention map with our CaaM and baseline methods based on CNN and ViT. "Attention" and "Interv." denote the conventional attention model and intervention method [49] respectively. The red box represents the failure case.

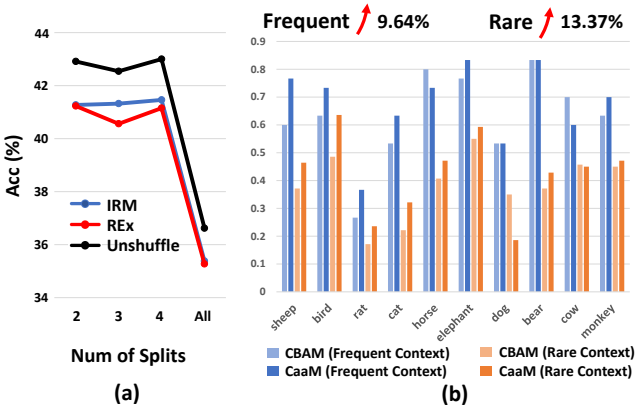

Figure 7. (a) The performance of intervention methods with different number of splits on NICO dataset. "All" denotes grouping data according to each context, *i.e.*, #Num of Splits=#Num of Contexts. (b) The per-class classification accuracy comparison between conventional CBAM [57] and our CaaM on images with frequent and rare context respectively .

performance of our CaaM and conventional CBAM attention model on test samples with frequent context and rare context respectively in Figure 7 (b).

A5: Specifically, the "frequent" denotes the top three context classes in training distribution, while "rare" represents the tailed seven context classes containing three zero-shot classes. Recall that as shown in Figure 1, the performance decreases more for the rare context classes using conventional attention model. Conversely, our CaaM can even receive a greater performance boost on rare contexts (13.37%) than that of the frequent (9.64%). Moreover, in relevant research fields (*e.g.*, long-tailed classification), it is well-known that the improvement of tail (rare) classes usually sacrifices the performance of the head (frequent). However, our CaaM can improve the accuracy of frequent and rare context simultaneously with a large margin.

4.5. Qualitative Results.

Figure 6 shows the qualitative attention map comparisons between our proposed CaaM (without the partition $\mathcal{T}$), intervention methods (with known partition $\mathcal{T}$) and convention attention model, *i.e.*, CBAM [57] and T2T-ViT7 [63]. Note that current ViT models are limited that the attention mechanism cannot be well trained without large-scale dataset. For attention visualization, the weights of T2T-ViT7 are initialized with ImageNet pretrained models. From Figure 6 we can see that, compared to the conventional attention (second row) and intervention methods (third row), our CaaM can achieve more accurate attention activation. Red boxes denote the failure cases. We find that our CaaM also cannot accurately attend to multiple objects (*e.g.*, two bears) or the single object co-existing with other ones (*e.g.*, horses and people). This inspires us to perform surrounding objects adjustment [55, 66] in future.

5. Conclusion

We demonstrated that the conventional attention module is particularly biased in OOD settings. We postulated that the reason is due to the confounder, whose effect should be removed by causal intervention. We theoretically showed that existing context-invariant methods suffer from improper causal intervention, which can be addressed by the proposed CaaM. Extensive experiments on three challenging benchmarks empirically demonstrated the effectiveness of CaaM. In future, we will seek a more powerful theory of causal effect disentanglement [26] and its implementations.

Acknowledgement. The authors would like to thank all reviewers for their constructive suggestions. This research is partly supported by the Alibaba-NTU Joint Research Institute, the A*STAR under its AME YIRG Grant (Project No. A20E6c0101), and the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 2 grant.

References

[1] Alessandro Achille and Stefano Soatto. Information dropout: Learning optimal representations through noisy computation. *IEEE TPAMI*, 40(12):2897–2905, 2018. 3

[2] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *ICML*, pages 145–155. PMLR, 2020. 3, 6

[3] Kartik Ahuja, Jun Wang, Amit Dhurandhar, Karthikeyan Shanmugam, and Kush R Varshney. Empirical or invariant risk minimization? a sample complexity perspective. *arXiv preprint*, 2020. 3

[4] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint*, 2019. 2, 3, 4, 6, 7

[5] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *ICML*, pages 528–539. PMLR, 2020. 3, 6, 7

[6] Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. Analysis of representations for domain adaptation. *NeurIPS*, 19:137, 2007. 3

[7] Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *ICLR*, 2019. 6

[8] Remi Cadene, Corentin Dancette, Hedi Ben-Younes, Matthieu Cord, and Devi Parikh. Rubi: Reducing unimodal biases in visual question answering. *NeurIPS*, 2019. 4, 6, 7

[9] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. Invariant rationalization. In *ICML*, pages 1448–1458. PMLR, 2020. 6

[10] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, pages 5659–5667, 2017. 2

[11] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *EMNLP*, 2019. 3

[12] Maurizio Corbetta and Gordon L Shulman. Control of goal-directed and stimulus-driven attention in the brain. *Nat. Rev. Neurosci.*, 3(3):201–215, 2002. 2

[13] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *ICML Workshop on Uncertainty and Robustness*, 2020. 3

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*, 2018. 1

[15] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 7

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint*, 2020. 1, 2, 5

[17] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, 2016. 3

[18] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *ICLR*, 2018. 3

[19] Mingming Gong, Kun Zhang, Tongliang Liu, Dacheng Tao, Clark Glymour, and Bernhard Schölkopf. Domain adaptation with conditional transferable components. In *ICML*, pages 2839–2848. PMLR, 2016. 3

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4, 6, 7

[21] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognit.*, 110:107383, 2021. 1, 3, 5

[22] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *ICLR*, 2017. 3

[23] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021. 3, 6, 7

[24] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. *ICCV*, 2021. 2

[25] Miguel A Hernán and James M Robins. Causal inference, 2010. 2

[26] Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *arXiv preprint*, 2018. 8

[27] Paul W Holland. Statistics and causal inference. *J. Am. Stat. Assoc.*, 81(396):945–960, 1986. 1

[28] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018. 2

[29] Xinting Hu, Kaihua Tang, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. Distilling causal effect of data in class-incremental learning. In *CVPR*, pages 3957–3966, 2021. 3

[30] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pages 1501–1510, 2017. 3

[31] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *NeurIPS*, 2019. 6, 7

[32] Salman H Khan, Munawar Hayat, Mohammed Bennamoun, Ferdous A Sohel, and Roberto Togneri. Cost-sensitive learning of deep feature representations from imbalanced data. *EEE Trans. Neural Netw. Learn. Syst.*, 29(8):3573–3587, 2017. 3

[33] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *CVPR*, pages 9012–9020, 2019. 3

[34] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). *arXiv preprint*, 2020. 3, 6, 7

[35] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. In *CVPR*, pages 9572–9581, 2019. 3

[36] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *ICLR*, 2018. 3

[37] Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous risk minimization. *arXiv preprint arXiv:2105.03818*, 2021. 3

[38] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, pages 181–196, 2018. 3

[39] Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. *NeurIPS*, 2014. 2

[40] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *ICML*, pages 10–18. PMLR, 2013. 3

[41] Yulei Niu, Kaihua Tang, Hanwang Zhang, Zhiwu Lu, Xian-Sheng Hua, and Ji-Rong Wen. Counterfactual vqa: A cause-effect look at language bias. *CVPR*, 2021. 3, 4

[42] Judea Pearl. *Causality*. Cambridge university press, 2009. 1, 2, 3

[43] Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016. 2

[44] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv preprint*, 2021. 1

[45] Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. *arXiv preprint arXiv:2010.05761*, 2020. 3

[46] Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *ECCV*, pages 467–482. Springer, 2016. 3

[47] Kaihua Tang, Jianqiang Huang, and Hanwang Zhang. Long-tailed classification by keeping the good and removing the bad momentum causal effect. *NeurIPS*, 2020. 3

[48] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *CVPR*, pages 3716–3725, 2020. 4

[49] Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. Unshuffling data for improved generalization. *arXiv preprint*, 2020. 2, 3, 6, 7, 8

[50] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint*, 2020. 2, 5

[51] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *CVPR*, pages 7167–7176, 2017. 3

[52] Ashish Vaswani, Prajit Ramachandran, Aravind Srinivas, Niki Parmar, Blake Hechtman, and Jonathon Shlens. Scaling local self-attention for parameter efficient visual backbones. In *CVPR*, pages 12894–12904, 2021. 2

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint*, 2017. 1, 2

[54] Haohan Wang, Zexue He, Zachary C Lipton, and Eric P Xing. Learning robust representations by projecting superficial statistics out. *ICLR*, 2019. 3

[55] Tan Wang, Jianqiang Huang, Hanwang Zhang, and Qianru Sun. Visual commonsense r-cnn. In *CVPR*, pages 10760–10770, 2020. 1, 3, 8

[56] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018. 2

[57] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, pages 3–19, 2018. 1, 2, 4, 5, 7, 8

[58] Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint*, 2020. 5

[59] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057. PMLR, 2015. 1

[60] Xu Yang, Hanwang Zhang, and Jianfei Cai. Deconfounded image captioning: A causal retrospect. *arXiv preprint*, 2020. 1

[61] Xu Yang, Hanwang Zhang, Guojun Qi, and Jianfei Cai. Causal attention for vision-language tasks. *arXiv preprint*, 2021. 2

[62] Nanyang Ye, Jingxuan Tang, Huayu Deng, Xiao-Yun Zhou, Qianxiao Li, Zhenguo Li, Guang-Zhong Yang, and Zhanxing Zhu. Adversarial invariant learning. In *CVPR*, pages 12446–12454, 2021. 3

[63] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *CVPR*, 2021. 2, 4, 5, 6, 7, 8

[64] Zhongqi Yue, Tan Wang, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. Counterfactual zero-shot and open-set visual recognition. In *CVPR*, pages 15404–15414, 2021. 3

[65] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. *NeurIPS*, 2020. 1

[66] Dong Zhang, Hanwang Zhang, Jinhui Tang, Xiansheng Hua, and Qianru Sun. Causal intervention for weakly-supervised semantic segmentation. *NeurIPS*, 2020. 1, 3, 8

[67] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 7