Appendix

This appendix is organized as follows:

- Section A.1 provides further interpretations of our proposed CaaM.
- Section A.2 presents theoretical evidences for the **Improper Causal Intervention** (Section 3.1), and for the convergence of **Adversarial Training** (Section 3.2).
- Section A.3 provides additional implementation details for **Invariant Loss** and **Adversarial Training** (Section 3.2).
- Section A.4 provides the methods used to generate OOD datasets (Section 4.1), additional training details (Section 4.2), the computation of attention accuracy (Section 4.4), and presents additional experimental results.

A.1. Interpretations

A.1.1. Invariant Loss is not Intervention?

Given the causal intervention formulation (Eq. (1)), readers may consider that the implementation of backdoor adjustment in robust classification can be simply achieved by optimizing the cross entropy loss in each data split $t \in \mathcal{T}$, rather than the invariant loss. Please note that this claim is actually to use the first objective item of the invariant loss and it acts just like the conventional cross entropy. The reason is, the backdoor adjustment in statistical causality theory is not designed for learning process in computer vision practice. The sum "$\sum$" of backdoor adjustment can not be implemented by summing up the cross entropy loss directly. We need the second term in Eq. (5) as a regularization to effectively implement the "$\sum$" by collecting the common invariant representation across different splits t. Then during inference, we just discard the second regularization term and forward the model to get the intervened prediction.

A.1.2. Mediator in Causal Graph

We have introduced the mediator M in the main paper and it is a part of the causal effect which need to be retained. In this paper we manually separate it out to better illustrate the improper intervention and explain why it hurts the causal effects. That is, non-accurate confounder set which falsely contains M will lead to the over adjustment of the mediator and hurt the causal representation. Then in Section 3.2 and 3.3 of the main paper, we give details of our proposed CaaM for obtaining better confounder set.

A.2. Theoretical Proofs

A.2.1. Proof of Improper Causal Intervention

We will show the derivation for the backdoor adjustment formula using the three rules of *do*-calculus [14], whose detailed proof can be found in [14, 13]. For a causal directed acyclic graph $\mathcal{G}$, let X, Y, Z and W be arbitrary disjoint sets of nodes. We use $\mathcal{G}_{\overline{X}}$ to denote the manipulated graph where all incoming arrows to node X are deleted. Similarly $\mathcal{G}_{\underline{X}}$ represents the graph where outgoing arrows from node X are deleted. We use lower case x, y, z and w for specific values taken by each set of nodes: $X = x, Y = y, Z = z$ and $W = w$. For any interventional distribution compatible with $\mathcal{G}$, we have the following three rules:

Rule 1 Insertion/deletion of observations. If $(Y \perp\!\!\!\perp Z|X, W)_{\mathcal{G}_{\overline{X}}}$:

$$P(y|do(x), z, w) = P(y|do(x), w), \tag{A1}$$

Rule 2 Action/observation exchange. If $(Y \perp\!\!\!\perp Z|X, W)_{\mathcal{G}_{\overline{X}\underline{Z}}}$,

$$P(y|do(x), do(z), w) = P(y|do(x), z, w), \tag{A2}$$

Rule 3 Insertion/deletion of actions. If $(Y \perp\!\!\!\perp Z|X, W)_{\mathcal{G}_{\overline{X}\overline{Z(W)}}}$,

$$P(y|do(x), do(z), w) = P(y|do(x), w), \tag{A3}$$

where $Z(W)$ is the set of nodes in Z that are not ancestors of any W-node in $\mathcal{G}_{\overline{X}}$.

In our causal graph, the desired interventional distribution $P(Y|do(X))$ can be derived by:

$$\begin{aligned}
&P(Y|do(X)) && \text{(A4)}\\
&= \sum_s P(Y|do(X), S = s)P(S = s|do(X)) && \text{(A5)}\\
&= \sum_s P(Y|do(X), S = s)P(S = s) && \text{(A6)}\\
&= \sum_s P(Y|X, S = s)P(S = s), && \text{(A7)}
\end{aligned}$$

where Eq. (A5) follows the law of total probability; Eq. (A6) uses Rule 3 given $S \perp\!\!\!\perp X$ in $\mathcal{G}_{\overline{X}}$; Eq. (A7) uses Rule 2 to change the intervention term to observation as $(Y \perp\!\!\!\perp X|S)$ in $\mathcal{G}_{\underline{X}}$. S When M and S are *disentangled* (or conditional independent), *i.e.*, $(S \perp\!\!\!\perp M)|X$, the backdoor adjustment formula can be further written as Eq. (2) by

$$\begin{aligned}
&P(Y|do(X)) && \text{(A8)}\\
&= \sum_s P(Y|X, S = s)P(S = s), && \text{(A9)}\\
&= \sum_s \sum_m P(Y|X, m, s)P(m|X, s)P(s), && \text{(A10)}\\
&= \sum_s \sum_m P(Y|X, m, s)P(m|X)P(s), && \text{(A11)}
\end{aligned}$$

where Eq. (A10) follows the law of total probability and Eq. (A11) is due to the conditional independence between M and S given X. This proves Eq. (2). However when M and S are *entangled* (as the "improper intervention"), $P(m|X,s) \neq P(m|X)$ and Eq. (A10) $\neq$ Eq. (A11). Therefore, in the case of entanglement, the optimization objective becomes Eq. (3).

A.2.2. Proof of Convergence of CaaM

In this section, we will first give the detailed intuition for the effectiveness of our CaaM adversarial training. Then we prove the positive feedback between the Maxi-Game and Mini-Game and the existence of the global optimal point of our CaaM, which ensures the convergence of our algorithm.

Intuitive Explanation. Consider the whole pipeline of the robust prediction. First, we need the complementary attention to disentangle the causal and confounder feature (*i.e.*, c and s) from the image representation $\mathbf{x}$ given a data partition $\mathcal{T}_i$; then we use the confounder feature s to generate better partition $\mathcal{T}_{i+1}$. The key is the mechanism for c and s to mutual promote each other and there exists a positive feedback between these 2 steps. With an accumulative confounder set, more accurate invariant representation c can be encoded, which will further bring a clearer picture of confounder feature s. Therefore we can then promote the partition $\mathcal{T}$ with the better s.

Positive Feedback. Here we assume that the image representation $\mathbf{x}$ is made up of the causal feature c and confounder feature s: $\mathbf{x} = c \circ s$, where $\circ$ is feature fusion. Here we use c^* and s^* to represent the oracle causal and confounder representation what we tend to find. $\mathcal{T}^*$ and θ^* are the corresponding oracle data partition. The initialized bias model is named as Ω which is trained with the conventional cross entropy loss. Therefore, we have $\Omega(x) = \mathbf{x}$ and it can be regarded as training with the random data partition. With Eq. (7), we have:

$$\theta_0 = \max_{\theta} \text{IL}(h, \Omega(x), \mathcal{T}(\theta)). \tag{A12}$$

θ_0 and its counterpoint $\mathcal{T}_0$ are the better partition compared to the random θ, approaching to $\mathcal{T}^*$ and θ^*. The behind reason can also be explained with the heterogeneous environment theorem [12]:

Theorem A1. *Denote image X and label Y, using the functional representation lemma [6], there exists random variable s such that $X = X(c, s)$ to make $P_t(Y|s)$ arbitrarily change across data splits t.*

Actually c and s are the robust causal and non-robust context feature. Theorem A1 indicates that a good data split t should reveal as much as possible spurious (or variant) feature to help to narrow the invariant feature. That means, if we can access more accurate bias feature, we can then achieve the better data split with Maxi-Game. Since the initial model Ω captures part of bias feature, the optimized θ_0 ($\mathcal{T}_0$) is better than random partition.

Then consider Mini-Game using Eq. (6), we can disentangle causal feature c_1 and confounder feature s_1 under current $\mathcal{T}_0$ with the proposed complementary attention module:

$$c_1, s_1 = \min_{\mathcal{A}_1, \overline{\mathcal{A}}_1, f, g} \text{XE}(f, \widetilde{x}, \mathcal{D}) + \text{IL}(g, \mathcal{A}_1(x), \mathcal{T}_0). \tag{A13}$$

Eq. (A13) directly optimize c and $\widetilde{x}$. With our assumption that $\mathbf{x} = c \circ s \approx \widetilde{x}$, better c and $\widetilde{x}$ leads to more accurate s. Therefore, confounder feature s_1 is a better approximation to s^* than previous feature $\mathbf{x}$ and the fitted bias model $h_1(\overline{\mathcal{A}}_1(x))$ is better than Ω using:

$$h_1 = \min_{h} \text{XE}(h, \overline{\mathcal{A}}_1(x), \mathcal{D}). \tag{A14}$$

We next update the new partition θ_1 and $\mathcal{T}_1$ with Eq. (A12). Since the bias model is better, the obtained θ_1 and $\mathcal{T}_1$ will be closer to $\mathcal{T}^*$ and θ^*. Moreover, since CaaM does not introduce new image label space (*e.g.*, continual learning [19]) but just update the partition, the previous $\mathcal{T}$ will still play its role partially. This results in an even better intervention with a comprehensive confounder set. Until now, we have illustrated a complete positive feedback of our CaaM. Next we give that the Mini-Game and Maxi-Game can both reach the global optimal point.

Global Optimal Point. Given the partition $\mathcal{T} = \mathcal{T}^*$, the learned c will be the oracle causal representation c^* when Eq. (A13) achieves the global minimal point. Then under the assumption of $\mathbf{x} = c \circ s$, s is equal to s^*. Therefore, the corresponding $\mathcal{T}$ can not be better. That is, Eq. (A12) also arrives to the global optimal point.

A.3. Implementation Details

A.3.1. Details of Invariant Loss

While having represented the core function of Invariant Loss (Eq. (5)) in the main paper, we aim to present more details regarding its motivation and practical implementation in this section. Invariant Loss aims to find an *invariant representation* (*i.e.*, the causal feature) robust for prediction, such that the optimal classifier over the representation is the same across dataset partition $\mathcal{T}$ [2]. This is formally given by:

Definition A1 (Invariant Representation)**.** *A representation $\mathcal{A}(x) \in \mathbb{R}^d$ is invariant across partition $\mathcal{T}$ if there exists a classifier $g : \mathbb{R}^d \to \mathbb{R}^k$ such that for $\forall t \in \mathcal{T}$, $g \in \arg\min_{\bar{g}} \text{XE}(\bar{g}, \mathcal{A}(x), t)$.*

This is achieved by the following objective function:

$$\begin{aligned} &\min_{\mathcal{A}, g} \sum_{t \in \mathcal{T}} \text{XE}(g, \mathcal{A}(x), t) \\ &\text{s.t.} g \in \arg\min_{\bar{g}} \text{XE}(\bar{g}, \mathcal{A}(x), t), \forall t \in \mathcal{T}, \end{aligned} \tag{A15}$$

where the definition of XE is given in Section 3.2. Intuitively, this objective optimizes the empirical risk minimization (ERM) representation $\mathcal{A}(x)$ subject to an invariant representation. However, this is a complex bi-level optimization problem, where each constraint corresponds to an inner-loop optimization. Therefore, [2] proposes a practical objective for approximation given in Eq. (5), *i.e.*,

$$\min_{\mathcal{A},g} \sum_{t\in\mathcal{T}} \mathrm{XE}(g, \mathcal{A}(x), t) + \lambda \|\nabla_{\mathbf{w}=\mathbf{1}.0} \mathrm{XE}(\mathbf{w}, \mathcal{A}(x), t)\|_2^2. \tag{A16}$$

In this paper, we exactly use Eq. (A16) for the Invariant Loss in Eq. (7), but for that in Eq. (6), we follow a more practical version [16] of Eq. (A16). Specifically, different linear classifier $\mathbf{W}_t$ is initialized for each data split t. The causal feature should be stable across splits. Therefore, the classifiers $\mathbf{W}_t$ are encouraged to converge to a common matrix, leading to a more practical version to replace the gradient penalty of Eq. (A16):

$$\min_{\mathcal{A},\{\mathbf{W}_t\}_{t=1}^m} \sum_{t\in\mathcal{T}} \mathrm{XE}(\mathbf{W}_t, \mathcal{A}(x), t) + \lambda \operatorname*{Var}_t(\mathbf{W}_t), \tag{A17}$$

where $\operatorname*{Var}_t(\mathbf{W}_t)$ controls the variance of classifier weights:

$$\operatorname*{Var}_t(\mathbf{W}_t) = (1/m) \sum_t (||\mathbf{W}_t - \overline{\mathbf{W}}||_2 / ||\mathbf{W}_t||_1)^2, \tag{A18}$$

where $\overline{\mathbf{W}}$ is the arithmetic mean of $\mathbf{W}_t$ over t. During testing, we use $\overline{\mathbf{W}}$ as classifier g to make the causal prediction $g(\mathcal{A}(x))$. Based on this invariant loss, we futher design a novel complementary attention to disentangle causal representation and adversarial training pipeline to update data partition.

A.3.2. Details of Adversarial Training

Mini-Game. The Mini-Game (Eq. (6)) can be further divided into 2 sub-steps. The first is the intervention training:

$$\min_{\mathcal{A},\overline{\mathcal{A}},f,g} \mathrm{XE}(f, \widetilde{x}, \mathcal{D}) + \mathrm{IL}(g, \mathcal{A}(x), \mathcal{T}). \tag{A19}$$

Then after achiving the confounder feature $\overline{\mathcal{A}}(x)$, we explicitly trains a biased classifier h using the confounding feature $\overline{\mathcal{A}}(x)$ and the original label y. To achieve this, we fix $\overline{\mathcal{A}}$ and optimize h by minimizing the CE loss:

$$\min_h \mathrm{XE}(h, \overline{\mathcal{A}}(x), \mathcal{D}). \tag{A20}$$

The Eq. (A19) and Eq. (A20) constitute the Mini-Game.

Maxi-Game. The core of Maxi-Game is to optimize Eq. (7), *i.e.*, maximizing the IL loss to update the partition $\mathcal{T}_i$:

$$\max_\theta \mathrm{IL}(h, \overline{\mathcal{A}}(x), \mathcal{T}_i(\theta)). \tag{A21}$$

We have introduced in the main paper that we define a set of optimizable parameters $\theta \in \mathbb{R}^{K\times m}$. Each parameter $\theta_{p,q}$ stores the current probability of the p-th sample belonging to the q-th split ($t_q \in \mathcal{T}_i$). $\mathcal{T}_i(\theta)$ is obtained by assign each data sample to the split index with largest probability. However, this process contains the `argmax` function which is not continuous in the backward pass. In this paper we use the Gumbel-Softmax [10] trick to relax the discrete sampling operation. Specifically, consider a k-dimensional categorical probabilities $\pi_1, ..., \pi_k$. The sample $\mathbf{y} = (y_1, ..., y_k)$ is given by:

$$y_v = \frac{\exp\left(\left(\log\left(\pi_v\right) + \mu_v\right)/\tau\right)}{\sum_{j=1}^k \exp\left(\left(\log\left(\pi_j\right) + \mu_j\right)/\tau\right)} \tag{A22}$$

where $\tau = 1.0$ is a temperature parameter. $\mu_v = -\log(-\log(u_v))$ and $u_v \sim \mathrm{Uniform}(0, 1)$. Here μ_v is named *Gumbel Noise*, perturbs each $\log(\pi_v)$ term so that taking the original `argmax` becomes equivalent to drawing a sample from the distribution $\pi_1, ..., \pi_k$. Moreover, to facilitate dataset grouping, the hard sample is needed. Therefore, we further adopt the Straight-Through (ST) Gumbel-Softmax [4, 10]. In the forward pass, it discretizes a continuous probability vector $\mathbf{y}$ sampled from the Gumbel-Softmax distribution into the one-hot vector $\mathbf{y}^{ST}$, where:

$$y_v^{ST} = \begin{cases} 1 & v = \arg\max_j y_j \\ 0 & \text{otherwise} \end{cases} \tag{A23}$$

And in the backward pass it simply uses the continuous $\mathbf{y}$, thus the loss signal is still able to backpropagate.

Initialization. For initialization, we train a bias model $h, \overline{\mathcal{A}}$ with the conventional cross entropy loss and assume this model is overfitted to some extent of confounding effects. Then we can get the initial $\mathcal{T}_0$ by replacing $h, \overline{\mathcal{A}}$ with $h_0, \overline{\mathcal{A}}_0$ in the Maxi-Game Eq. (A21). That means, we initialize the confounder representation with $h_0(\overline{\mathcal{A}}_0(x))$.

The overall training pipeline is summarized in Algorithm 1.

Algorithm 1 Adversarial CaaM Training

Input: Dataset $\mathcal{D}$
Output: $\mathcal{A}, \overline{\mathcal{A}}, \theta$
Initialize split assignment $\theta_0, \mathcal{T}_0$ with Eq. (A12)
Initialize $f, g, h, \mathcal{A}, \overline{\mathcal{A}}$
for $i \in \{1, 2, ..., N\}$ **do**
  for each $x \in \mathcal{D}$ **do**
    $\mathbf{c}_i \leftarrow \mathcal{A}_i(x)$
    $\mathbf{s}_i \leftarrow \overline{\mathcal{A}}_i(x)$
  end for
  Update $f_i, g_i, h_i, \mathcal{A}_i, \overline{\mathcal{A}}_i$ with Eq. (A19), Eq. (A20) and $\mathcal{T}_i$; {// Mini-Game}
  Update $\theta_i, \mathcal{T}_i$ with Eq. (A21); {// Maxi-Game}
until end

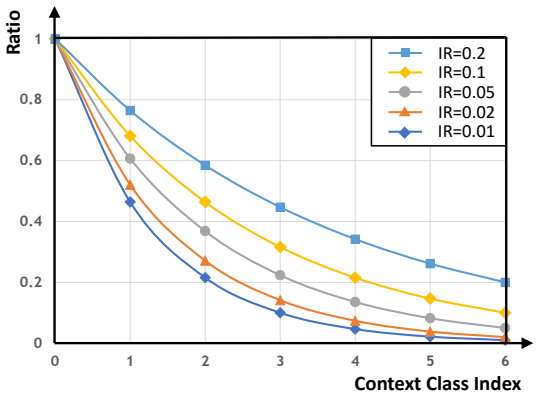

Figure A1. Plot of context class index against its corresponding ratio under various imbalance ratio (IR).

A.4. Experimental Details

A.4.1. NICO Dataset

In our experiment, we selected a subset of NICO animal dataset [9] as a challenging benchmark to test OOD robustness for proposed CaaM and baselines. Specifically, images in NICO are labeled with a context class (*e.g.*, "on grass"), besides the object class (*e.g.*, "dog"). As discussed in section 4.1, during training we chose 7 context classes (Long-Tailed Contexts as shown in Table A1) for each object class. Next, we formed a long-tailed training dataset by selecting part of the images in each context class with multiplying a ratio. In particular, the ratio for w-th context class ($w \in \{0, \dots, 6\}$) is given by

$$\text{ratio} = \text{IR}^{w/6}, \tag{A24}$$

where IR is a hyper-parameter that denotes the imbalance ratio. The effect of IR on ratio is shown in Figure A1 — lower ratio leads to the harder OOD problem. In the main paper we keep IR $= 0.02$. During testing, the number of test samples across the 7 context classes is balanced, *i.e.*, 50 samples per context. Moreover, we added 3 zero-shot context classes for each object class as shown in Table A1 (last three columns). These zero-shot context classes have the larger number of test samples (100 samples per context). Therefore, a model that performs well in our split must be robust to both long-tailed and zero-shot problems *w.r.t.* the context class. Figure A2 shows an example of our constructed subset for "cat" and "dog" during training and testing. Moreover, to construct m ground truth splits on NICO dataset for training in "w/ H.M. $\mathcal{T}$" setting, we first sort images in the context order and then divide them into m equal parts.

A.4.2. ImageNet-9 Dataset

Specifically, we elaborate the construction of the proxy context labels, respectively, for ImageNet-9 and ImageNet-A.

Proxy Context Label for ImageNet-9. Note that there is no ground truth context annotation in ImageNet-9 training and testing set. Therefore, we follow [3] to obtain the proxy ground truth context labels using texture feature clustering. Specifically, we extract the texture features from images by computing the gram matrices of low-layer feature maps [7, 11] from `relu1_2` of the ImageNet pre-trained VGG16 [15]. Then we run the mini-batch k-means algorithm with $k = m$ with batch size 1024, m is the number of data splits. For the construction of the unbias test set, we follow [3] to set $k = 9$ with batch size 1024. The clustering examples are shown in Figure A3 (left).

ImageNet-A. ImageNet-A is a dataset of natural adversarial examples for ImageNet classifiers, or real-world examples that fool current classifiers as shown in Figure A3 (right). The images consist of many failure modes of networks when "frequently appearing background elements" become erroneous cues for recognition (*e.g.*, a dragonfly on a yellow metal bracket is recognised as the banana).

A.4.3. Training Details

In Mini-Game, the optimizer was set to SGD with a learning rate of 0.05 for ResNet model; while for T2T-ViT, the learning rate was set as 0.001 with AdamW optimizer following ViT [5]. We trained the model with 200 epochs for NICO dataset and the learning rate was decreased by 5 at 80, 120, 160 epoch. While for ImageNet-9 we trained for 120 epochs with learning rate decreased by 5 at 50, 80, 100 epoch. For the bias classifier fitting Eq. (A20) and the Maxi-Game Eq. (A21), we trained each for 100 epochs with early stopping (accuracy no longer increases more than 5 epoch). The optimizer was set to SGD with learning rate as 0.1. For Mini-Game, λ in invariant loss was set to 5e4 or 5e5 following previous paper [16]. In Maxi-Game, λ is fixed to 1e6. N was set from 5 to 20. In Table (2), for results in first two row (**Num L.** and **Num S.**), we did not use the adversarial training. The default M was 2 for CNN-CaaM and 4 for ViT-CaaM, while the default m was set to 4.

A.4.4. Attention Accuracy (Q3)

In the question 3 of the main paper, we calculate the attention accuracy for proposed method and its comparisons to quantify the robustness of CaaM attention on ImageNet dataset. Here we detail how to compute. Given an Image, we can obtain its attention map with different architectures. Specifically, for ResNet+CBAM, we take the CAM activation [20] as the attention map following the original CBAM paper [17]. While for T2T-ViT, we use the Attention Rollout [1] following original ViT [5]. Briefly, we averaged at-

Class \ Context	Long-Tailed Contexts							Zero-shot Contexts		
Dog	on grass	in water	in cage	eating	on beach	lying	running	at home	in street	on snow
Cat	on snow	at home	in street	walking	in river	in cage	eating	in water	on grass	on tree
Bear	in forest	black	brown	eating grass	in water	lying	on snow	on ground	on tree	white
Sheep	eating	on road	walking	on snow	on grass	lying	in forest	aside people	in water	at sunset
Bird	on ground	in hand	on branch	flying	eating	on grass	standing	in water	in cage	on shoulder
Rat	at home	in hole	in cage	in forest	in water	on grass	eating	lying	on snow	running
Horse	on beach	aside people	running	lying	on grass	on snow	in forest	at home	in river	in street
Elephant	in zoo	in circus	in forest	in river	eating	standing	on grass	in street	lying	on snow
Cow	in river	lying	standing	eating	in forest	on grass	on snow	at home	aside people	spotted
Monkey	sitting	walking	in water	on snow	in forest	eating	on grass	in cage	on beach	climbing

Table A1. Construction of our NICO [9] subset for OOD multi-classification . **Context** denotes the context class name, while **Class** represents the object class name. "Long-Tailed Contexts" is the training contexts arranged by the sample number order (from more to less) and "Zero-shot Contexts" represents the context labels only appear in testing rather than training.

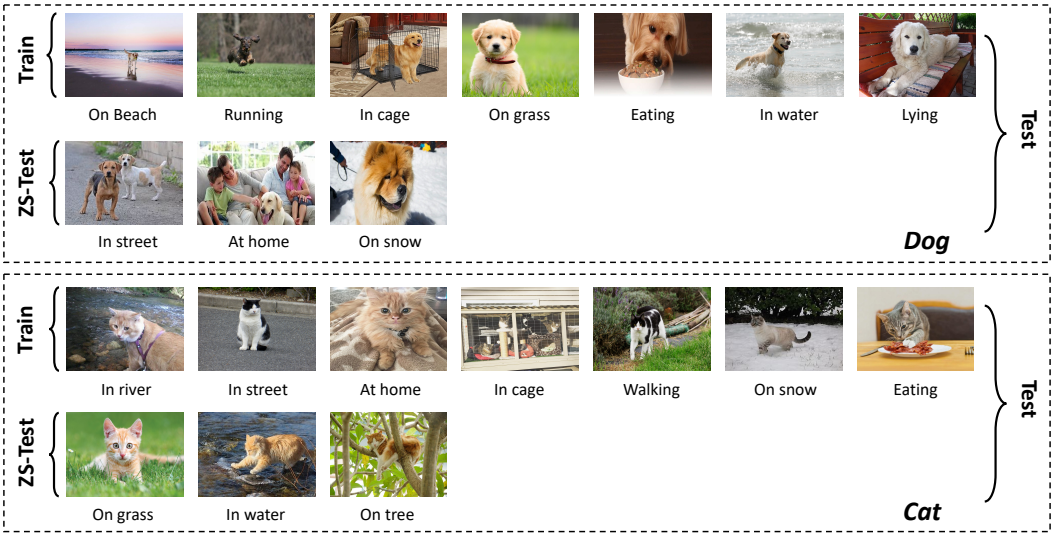

Figure A2. We list the sample images of each context class using "Dog" and "Cat" as the example in our constructed NICO dataset. **Train**, **Test** and **ZS-Test** denote samples for training, testing and zero shot testing respectively. Note that there is no overlap between training and testing images.

tention weights of T2T-ViT across all heads and then recursively multiplied the weight matrices of all layers. This accounts for the mixing of attention across tokens through all layers. Having the ground truth object location bounding box annotation B and the attention map A which is a weight matrix of image size, we can then compute the attention accuracy by:

$$\text{Att. Acc.} = \frac{sum((A \cap B) > \sigma)}{sum(A > \sigma)}, \tag{A25}$$

where $A \cap B$ denotes the attention map area in the bounding box and $\sigma = 0.9$ is the threshold. This function is similar to the Intersection over Union (IoU) score metric in object detection.

A.4.5. Additional Results

Complexities. We show the model sizes and the computational costs in Table A2. We can see that compared to baseline models ResNet18+CBAM and T2T-ViT7, using CaaM adds a small number of network parameters (overhead). This is because the complementary attention in CaaM does not rely on new network layers. In terms of computing speed, single-layer CaaM has the comparable Flops and MACs to baseline models. Multi-layer CaaM has linearly-increased Flops and MACs with respect to the number of layers, while the maximum costs are tolerable.

Visualizations. In Figure A4 and Figure A5, we supplement more visualization results for several comparable methods: our CaaM (unsupervised, *i.e.*, without using the labels of partition $\mathcal{T}$), an intervention method [16] (supervised, *i.e.*, using $\mathcal{T}$), and a convention attention method

Figure A3. The examples of the clustering results of ImageNet-9 dataset and the samples of ImageNet-A dataset.

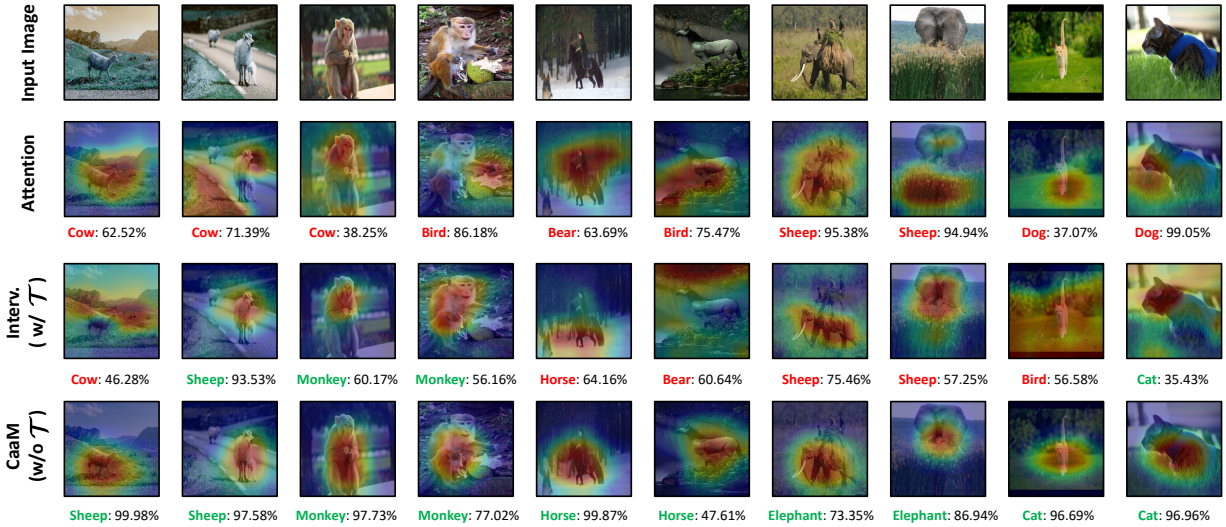

Figure A4. The attention activation maps based on CNN on NICO dataset. "Attention" and "Interv." denote the conventional attention model and intervention method [16] respectively. **Red** and **Green** text denote the false and correct prediction followed by the prediction confidence.

Models	Params (M)	Flops (G)	MACs (G)
ResNet18 [8]	11.18	3.63	1.81
ResNet18+CBAM [17]	11.27	3.64	1.82
ResNet18+CaaM	11.29	3.64	1.82
ResNet18+CaaM (M=2)	11.29	4.46	2.23
ResNet18+CaaM (M=4)	11.29	5.28	2.64
T2T-ViT7	4.00	1.95	0.97
T2T-ViT7+CaaM	4.01	2.08	1.04
T2T-ViT7+CaaM (M=2)	4.01	2.20	1.10
T2T-ViT7+CaaM (M=4)	4.01	2.46	1.23

Table A2. The model size and computational cost comparison between our proposed CaaM and baseline models.

(CBAM [17] for CNN and T2T-ViT [18] for ViT). Samples in Figure A4 are the results of CNN-based models. Samples in Figure A5 are from ViT-based models. Both are from the NICO dataset. For each sample, we report the heatmap of attention, the predicted label and the corresponding probability.

In Figure A4, we can observe that 1) the conventional attention model (the second row) produces many inaccurate attentions on images and false predictions of object labels; 2) the intervention method (the third row) partially tackle the problems by using the labeled partitions $\mathcal{T}$; and 3) impressively, our CaaM (the forth row) achieves both more accurate predictions and more precise attentions—mostly focused on the object bodies. Similar results can also be drawn from Figure A5 for methods based on ViT.

In addition to the quantitative attention accuracy results on ImageNet-9 dataset (Table 3) in the main paper, here we also visualize the attention map based on CNN on ImageNet-9 biased validation set in Figure A6, denoting the *IID* setting (NICO, ImageNet-9 unbiased set and ImageNet-A are the *OOD* setting). Compared to convention attention and intervention method, we can see that our CaaM can still clearly get tighter and more explainable attention maps in

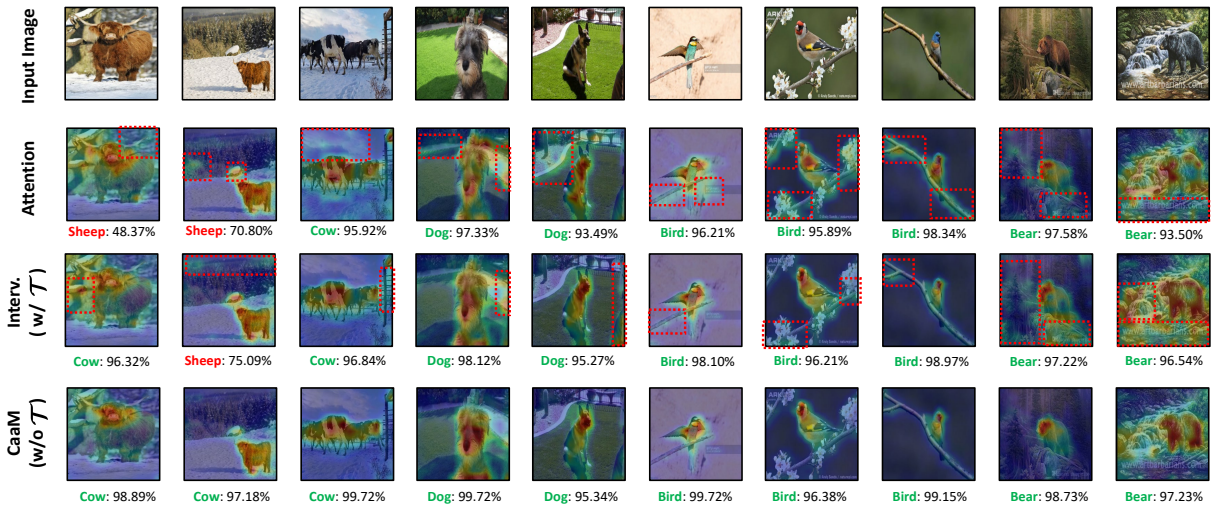

Figure A5. The attention activation maps based on T2T-ViT on NICO dataset. "Attention" and "Interv." denote the conventional attention model and intervention method [16] respectively. **Red** and **Green** text denote the false and correct prediction followed by the prediction confidence. The red dashed boxes highlight the worse attention activation for the comparisons.

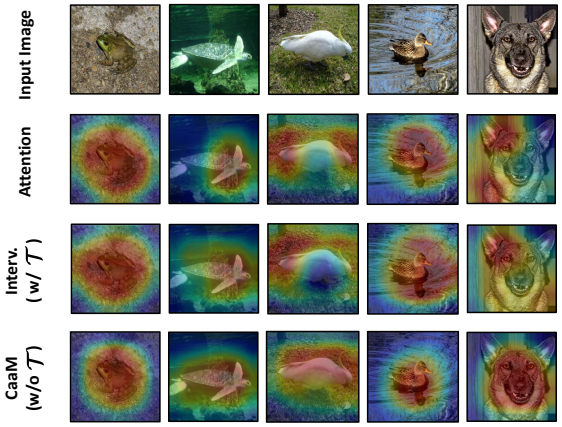

Figure A6. The attention activation maps based on CNN on ImageNet-9 dataset. "Attention" and "Interv." denote the conventional attention model and intervention method [16] respectively.

IID setting.

References

[1] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. *arXiv preprint arXiv:2005.00928*, 2020. 4

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

[3] 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. 4

[4] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 3

[5] 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. 4

[6] Abbas El Gamal and Young-Han Kim. *Network information theory*. Cambridge university press, 2011. 2

[7] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. *arXiv preprint arXiv:1505.07376*, 2015. 4

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

[9] Yue He, Zheyan Shen, and Peng Cui. Towards non-iid image classification: A dataset and baselines. *Pattern Recognit.*, 110:107383, 2021. 4, 5

[10] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 3

[11] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 4

[12] Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous risk minimization. *arXiv preprint arXiv:2105.03818*, 2021. 2

[13] Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995. 1

[14] Judea Pearl. *Causality*. Cambridge university press, 2009. 1

[15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[16] Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. Unshuffling data for improved generalization. *arXiv preprint*, 2020. 3, 4, 5, 6, 7

[17] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, pages 3–19, 2018. 4, 6

[18] 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. 6

[19] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017. 2

[20] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Computer Vision and Pattern Recognition*, 2016. 4