Don't Judge an Object by Its Context: Learning to Overcome Contextual Bias

Krishna Kumar Singh[1], Dhruv Mahajan[2], Kristen Grauman[2,3], Yong Jae Lee[1], Matt Feiszli[2], Deepti Ghadiyaram[2]

[1]University of California, Davis, [2]Facebook AI, [3]University of Texas at Austin

Abstract

Existing models often leverage co-occurrences between objects and their context to improve recognition accuracy. However, strongly relying on context risks a model's generalizability, especially when typical co-occurrence patterns are absent. This work focuses on addressing such contextual biases to improve the robustness of the learnt feature representations. Our goal is to accurately recognize a category in the absence of its context, without compromising on performance when it co-occurs with context. Our key idea is to decorrelate feature representations of a category from its co-occurring context. We achieve this by learning a feature subspace that explicitly represents categories occurring in the absence of context along side a joint feature subspace that represents both categories and context. Our very simple yet effective method is extensible to two multi-label tasks – object and attribute classification. On 4 challenging datasets, we demonstrate the effectiveness of our method in reducing contextual bias.

1. Introduction

Visual context serves as a valuable auxiliary cue for the human visual system for scene interpretation and object recognition [4]. Context can either be a co-occurrence of objects and scenes (e.g., "boat" is often present in "outdoor waters") or of two or more objects in a given scene (e.g., "skis" often co-occur with a "skier"). Context becomes especially crucial for our visual system when the visual signal is ambiguous or incomplete (e.g., due to occlusion, viewpoint of the scene capture, etc.). Past research explicitly models context and shows benefits on standard visual tasks such as classification [30] and detection [13, 3]. Meanwhile, convolution networks by design implicitly capture context.

Deep networks rely on the availability of large-scale annotated datasets [22, 12] for training. As highlighted in [32, 31], despite the best efforts of its creators, most prominent vision datasets are afflicted with several forms of *biases*. Let us consider an object category "microwave." A significant portion of images belonging to this category are likely to be captured in kitchen environments, where other objects such as "refrigerator," "kitchen sink," and "oven" frequently co-occur. This may inadvertently induce *contextual bias* in these datasets, which would consequently seep into models trained on them. Specifically, in the process of learning features that separate positive and negative instances in such a (biased) training dataset, a deep discriminative model can very often also strongly capture the context co-occurring with the category of interest. This issue is exacerbated in a setting where we do not have explicit location annotations (e.g., bounding boxes and segmentation masks) of such biased categories, and a model being trained has to rely solely on image-level annotations to perform multi-label classification. Having a model *implicitly* learn to localize such context-biased categories in the absence of location annotations is challenging.

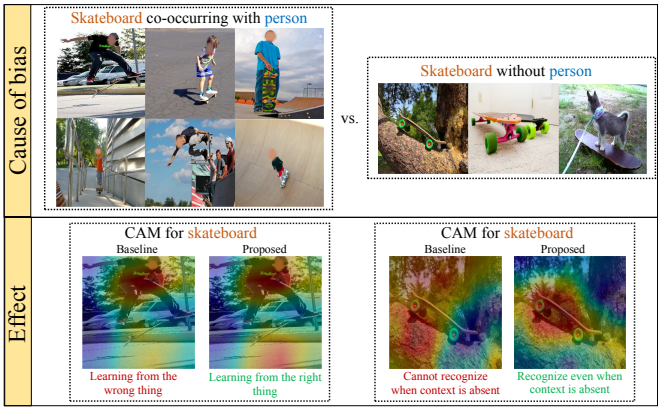

Figure 1. **Top (cause of contextual bias)**: Sample training images of the category "skateboard". Notice how it very often co-occurs with "person" and how all images are captured from similar viewpoints. In the rare cases where skateboard occurs exclusively, there is higher viewpoint variance. **Bottom (effect of such bias)**: Such data skew causes a typical classifier to rely on "person" to classify "skateboard" and worse, unable to recognize skateboard when person is absent. Our proposed approach overcomes such contextual bias by learning feature representations that decorrelate the category from its context.

Does it even matter if a model inadvertently learns such correlations? We believe this can cause problems on two fronts: (1) failing to identify "microwave" in a different context such as an "outdoor" scene or in the *absence* of "refrigerator" and (2) hallucinating "refrigerator" even in an indoor kitchen scene containing only "microwave." The issue of co-occurring bias is also prevalent in visual attributes [23, 35]. For example, in the Deep Fashion dataset [23], the attribute "trapeze" strongly co-occurs with "striped." This results in a less credible classifier that has a hard time recognizing "trapeze" in clothes with "floral." Recent research has identified far more serious mistakes made by trained models due to inherent biases in both language and vision datasets – learning correlations between ethnicity and certain sport activities [28], gender and profession [5, 16, 36], and age and gender of celebrities [2]. Such grave confusion caused due to biases in the data impedes the deployment of these models in real-world applications.

Given these issues, our goal is to train an unbiased visual classifier that can accurately recognize a category both in the presence and absence of its context. Specifically, given two categories with a strong co-occurring bias, our aim is to accurately recognize them when either one occurs *exclusively*, and at the same time not hurt the performance when they *co-occur*. To this end, we propose two key ideas. First, we hypothesize that a network should learn about a category by relying more on its corresponding pixel regions than those of its context. Since we only have class labels, we use class activation maps (CAM) [37] as "weak" location annotations and minimize their mutual spatial overlap.

Building on this, we devise a second method that learns feature representations to decorrelate a category from its context. While the entire feature space learned by the network jointly represents category and context, we explicitly carve out a subspace to represent categories that occur away from typical context. We learn this feature subspace only from training instances where a biased category occurs in the absence of its context. In all other cases, the model *should* also leverage context and thus the entire feature space. At test time, we make no such distinction and the entire feature space is equally leveraged. Therefore, in the example from Fig. 1, our goal is to learn a feature subspace to represent "skateboard" while the entire feature space jointly represents "skateboard" and "person."

Through extensive evaluation, we demonstrate significant performance gains for the hard cases where a category occurs away from its typical context. Crucially, we show that our framework does not adversely effect recognition performance when categories and context co-occur. To summarize, we make the following contributions:

- With an aim to teach the network to "learn from the right thing," we propose a method that minimizes the overlap between the class activation maps (CAM) of the co-occurring categories (Sec. 4.1).
- Building on the insights from the CAM-based method, we propose a second method that learns feature representations that decorrelate context from category (Sec. 4.2).
- We apply both methods on two tasks: object and attribute classification, and 4 datasets, and achieve significant boosts over strong baselines for the hard cases where a category occurs away from its typical context (Sec. 5).

2. Related work

Addressing biases: Prior work [32, 19, 33, 31] has shown that existing datasets suffer from bias and are not perfectly representative of the real world. Hence, a model trained on such data will have difficulty generalizing to non-biased cases. Attempts to reduce dataset bias include domain adaptation techniques [9] and data re-sampling [7, 21], e.g., so that minority class instances are better represented. One limitation of data re-sampling is that it can involve reducing the dataset, leading to sub-optimal models. Recent adversarial learning approaches [2, 20] try to mitigate bias from the learned feature representations while optimizing performance for the task at hand (e.g., removing gender bias while classifying age). However, these methods would not be directly applicable for mitigating *contextual* bias, as context (the bias factor) can still be useful for recognition—so it cannot be simply removed. Others study various forms of bias in the context of image captioning (e.g., gender bias) [16], image classification (e.g., ethnicity bias) [28], and object recognition (e.g., socio-economic bias) [11]. Overall, *contextual* bias in visual recognition remains relatively under explored.

Co-occurring-bias: Contextual bias is a well-studied problem in the field of natural language processing [25, 29], however, it is much less studied in the computer vision community. In vision, most efforts consider context as a useful cue [13, 3]. A few efforts have shown that a recognition model will fail to recognize an object without its co-occurring context, but do not propose a solution [8, 26].

A recent method reduces contextual bias in video action recognition [34], but it relies on temporal information and thus cannot be applied to the image recognition problems we tackle in this work. A pre-deep learning approach [17] reduces the correlation (bias) between visual attributes by leveraging additional knowledge in the form of semantic groupings of attributes. Recently [38] tried to reduce contextual bias for object detection by learning focused foreground features, but they require expensive bounding-box annotations. In contrast, our deep learning approach does not require any additional supervision apart from the object/attribute class labels. Most importantly, to our knowledge, there is no prior work focusing on mitigating contextual bias for object classification as we do in this paper.

Relation to few-shot learning: Lastly, contextual bias

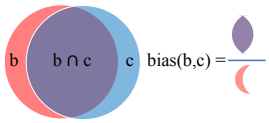

Figure 2. **Quantifying bias** in b due to its high co-occurrence with c.

could also be formulated as a few-shot [27, 18, 1] or class imbalance [14, 10] problem, since images in which objects appear without their usual co-occurring context (e.g., keyboard without a mouse next to it) are relatively rare. However, treating such rare (exclusive) images as a separate class or simply assigning them higher weight can be sub-optimal, as we show in our experiments.

3. Problem setup

Our method operates on the premise that the training data distribution corresponding to a few categories suffers from co-occurring bias. We henceforth refer to them as *biased categories*. We make no such assumptions about the test data distribution. For example, COCO-Stuff [6] has 2209 images where "ski" co-occurs with "person," but only has 29 images where "ski" occurs without "person." A model trained on such skewed data may fail to recognize when "ski" occurs in isolation. Our goal is to learn a feature space that is robust to such training data biases. In particular, given a (presumably) unbiased test dataset, our goal is to (1) correctly identify "ski" when it occurs in isolation and (2) not lose performance when "ski" co-occurs with "person." A key aspect of our approach is to identify most biased categories for a given dataset, which we describe next.

3.1. Identifying biased categories

Suppose we are learning a classifier on a multi-label training dataset with a vocabulary of M categories. Only a few of these categories suffer from context[1] bias; thus, a key aspect of our approach is to find this set of K category pairs $\mathbb{S} = \{(\mathrm{b_j}\ ,\ \mathrm{c_j}\ )\}$, where $0 \leq \mathrm{j} < \mathrm{K}$, which suffer the most from co-occurring bias[2]. Henceforth, $\mathrm{b_j}$ (e.g. "ski") denotes a class which is most biased with $\mathrm{c_j}$ (e.g. "person") due to its high co-occurrence.

Intuition: While there are several ways to construct $\mathbb{S}$, our method is built on the following intuition: a given category b is most biased by c if (1) the prediction probability of b drops significantly in the *absence* of c and (2) b co-occurs frequently with c.

We now define our method to identify c for a given b. For a given category z, let $\mathbb{I}_\mathrm{b} \cap \mathbb{I}_\mathrm{z}$ and $\mathbb{I}_\mathrm{b} \setminus \mathbb{I}_\mathrm{z}$ denote sets of images where b occurs with and without z respectively. Let $\hat{\mathrm{p}}(\mathrm{i}, \mathrm{b})$ denote the prediction probability of an image i for a category b obtained from training a standard multi-label classifier. We quantify the extent of *bias* between b and z as follows:

$$\mathrm{bias}(\mathrm{b}, \mathrm{z}) = \frac{\frac{1}{|\mathbb{I}_\mathrm{b} \cap \mathbb{I}_\mathrm{z}|} \sum\limits_{\mathrm{I} \in \mathbb{I}_\mathrm{b} \cap \mathbb{I}_\mathrm{z}} \hat{\mathrm{p}}(\mathrm{i}, \mathrm{b})}{\frac{1}{|\mathbb{I}_\mathrm{b} \setminus \mathbb{I}_\mathrm{z}|} \sum\limits_{\mathrm{I} \in \mathbb{I}_\mathrm{b} \setminus \mathbb{I}_\mathrm{z}} \hat{\mathrm{p}}(\mathrm{i}, \mathrm{b})}, \tag{1}$$

where $|.|$ denotes cardinality of a set. Eq (1) measures the ratio of average prediction probabilities of the category b when it occurs with and without z (see Fig. 2). A higher value indicates a higher dependency of b on z. We determine c as follows:

$$\mathrm{c} = \arg\max_{\mathrm{z}} \mathrm{bias}(\mathrm{b}, \mathrm{z}) \tag{2}$$

i.e., for each b, we identify a category c that (i) yields the highest value of bias and (ii) co-occurs at least $10 - 20\%$ times (see Sec. 4.3) with b. We then construct $\mathbb{S}$ with K most biased category pairs. We note that the above formulation is directional, i.e., it only captures the biases in b caused due to c. For instance, $\mathrm{bias}(\mathrm{ski}, \mathrm{person})$ only captures bias in "ski" due to "person" but not vice-versa.

We next propose two methods to combat co-occurring bias in the training data. The input to both methods is (1) training images and their associated weak (multiple) category labels and (2) the set $\mathbb{S}$ composed of the K most biased category pairs (identified from Eq. (1)). We stress that training images have only weak labels stating which categories are present; they have no spatial annotations to say *where* in the image each category is.

4. Approach

Our first method relies on class activation maps (CAM) as "weak" automatically inferred location annotations and minimizes their spatial overlap between biased categories (Sec. 4.1). Building on the observations from this CAM-based approach, we propose a second method which learns a feature space by encouraging context sharing when a biased category co-occurs with context while suppressing context when it occurs in isolation (Sec. 4.2).

4.1. CAM as "weak" location annotation

Our method operates on the following premise: as b almost always co-occurs with c, the network may learn to inadvertently rely on pixels corresponding to c to predict b. This is particularly problematic when the network is tested on images where b occurs in the absence of c. We hypothesize that one way to overcome this issue is to *explicitly* force the network to rely less on c's pixel regions, *without* using location annotations. While this may not succeed for occluding pairs like "person" and "shirt," it seems like a natural constraint for spatially-distinct categories like "person" and "skateboard."

[1] Throughout, we use context and co-occurring interchangeably.

[2] Although we consider pairs of co-occurring categories throughout, the proposed method is extensible for any number of co-occurring categories.

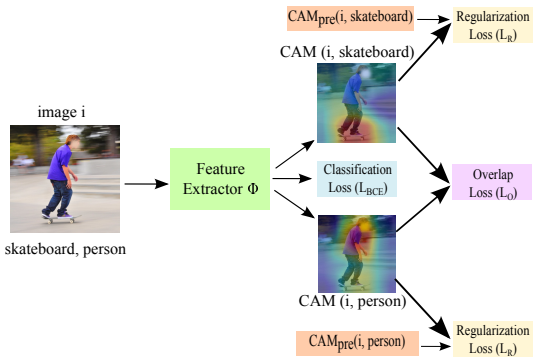

Figure 3. **Our CAM-based approach** operates on category labels and requires no ground-truth location annotations. Instead, we leverage CAMs as weak location annotations and propose to minimize the mutual overlap between a biased category and its co-occurring context.

Class Activation Maps: To this end, we propose to use class activation maps (CAM) [37] as a proxy for object localization information. For a given image i and class r, $\mathrm{CAM}(\mathrm{i}, \mathrm{r})$ indicates the discriminative image regions used by a deep network to identify r. Specifically, the final convolutional layer ($\mathrm{conv_f}$) of any typical network is followed by a global pooling and a fully connected (fc) layer which predicts a score for class r in image i. $\mathrm{CAM}(\mathrm{i}, \mathrm{r})$ is generated by *projecting back* the weights of the fc layer for r on $\mathrm{conv_f}$ and computing a weighted average of the feature maps. Though CAMs are typically used as a visualization technique, in this work, we also use them to reduce contextual bias as we describe next.

Formulation: In our setup, for each biased category pair (b, c) in $\mathbb{S}$ (defined in Sec. 3.1), we enforce minimal overlap of their CAMs via the loss function:

$$\mathrm{L_O} = \sum_{\mathrm{i} \in \mathbb{I}_\mathrm{b} \cap \mathbb{I}_\mathrm{c}} \mathrm{CAM}(\mathrm{i}, \mathrm{b}) \odot \mathrm{CAM}(\mathrm{i}, \mathrm{c}) \tag{3}$$

CAM offers two nice properties: (1) it is learned only through class labels without requiring any annotation effort and (2) it is fully differentiable, and thus can be integrated in an end-to-end network during training.

Ideally, Eq (3) should learn to reduce the spatial overlap between co-occurring categories, without hurting the classification performance. However, while attempting to minimize overlap, Eq (3) could also lead to a trivial solution where the CAMs of b and c drift apart from their actual pixel regions. To prevent this without strongly-supervised spatial annotations, we introduce a regularization term $\mathrm{L_R}$. Specifically, we pre-train a separate network (offline) for the standard classification task and generate $\mathrm{CAM_{pre}}$ from it for b and c. We then *ground* the CAMs of each category to be closer to its pixel regions predicted from $\mathrm{CAM_{pre}}$. $\mathrm{L_R}$ is thus defined as follows:

$$\mathrm{L_R} = \sum_{\mathrm{i} \in \mathbb{I}_\mathrm{b} \cap \mathbb{I}_\mathrm{c}} |\mathrm{CAM_{pre}}(\mathrm{i}, \mathrm{b}) - \mathrm{CAM}(\mathrm{i}, \mathrm{b})| + |\mathrm{CAM_{pre}}(\mathrm{i}, \mathrm{c}) - \mathrm{CAM}(\mathrm{i}, \mathrm{c})| \tag{4}$$

We use a standard binary cross-entropy loss ($\mathrm{L_{BCE}}$) for the task of multi-label classification. Thus, our final loss becomes:

$$\mathrm{L_{CAM}} = \lambda_1 \mathrm{L_O} + \lambda_2 \mathrm{L_R} + \mathrm{L_{BCE}}, \tag{5}$$

Fig. 3 for the entire approach. As we show in results (Sec. 5), our CAM-based method successfully learns to rely more on the biased category's pixel regions thereby improving recognition performance. Our method yields large gains when a biased category occurs in the absence of its typical context. However, it sometimes hurts performance when biased category co-occurs with context (discussed later in Fig. 7). One reason could be that the pixel regions surrounding the co-occurring category also offer useful complementary information for recognizing the biased category. By discouraging mutual spatial overlap, CAM-based approach may not be able to leverage this information. This key insight led to the formulation of our next approach, which splits the feature space into two and separately represents context and category, while posing no constraints on their spatial extents.

4.2. Feature splitting and selective context suppression

Rather than optimizing CAMs, we propose to learn a feature space that is robust to the inherent co-occurring biases in the training data. We observe that cases when a biased category co-occurs with context are often visually distinct from those where it occurs exclusively (see Fig. 1). This motivates us to learn a dedicated feature (sub) space to represent biased categories occurring away from their typical context. While the entire feature space learned by the model jointly represents context and category, this dedicated subspace should decouple the representations of a category from its context. We learn this feature subspace only from training instances where biased categories occur in the absence of their typical context. These modifications only affect training; at inference time the architecture is identical to the standard model.

Formulation: Given a deep neural network ϕ, let x denote the D-dimensional output of the final pooling layer just before the fully-connected layer (fc). Let the weight matrix associated with fc layer be $\mathrm{W} \in \mathrm{R}^{\mathrm{D} \times \mathrm{M}}$, where M denotes the number of categories in a given multi-label dataset. The predicted scores inferred by a classifier (ignoring the bias term) are

$$\hat{\mathrm{y}} = \mathrm{W}^\mathrm{T} \mathrm{x}. \tag{6}$$

Because we wish to separate the feature representations of a category from its context, we (row-wise) split W randomly into two disjoint subsets: $\mathrm{W_o}$ and $\mathrm{W_s}$, each of dimension $\frac{D}{2} \times M$. Consequently, x is split into $\mathrm{x_o}$ and $\mathrm{x_s}$ and the above equation can be rewritten as:

$$\hat{\mathrm{y}} = \mathrm{W_o^T x_o} + \mathrm{W_s^T x_s}. \tag{7}$$

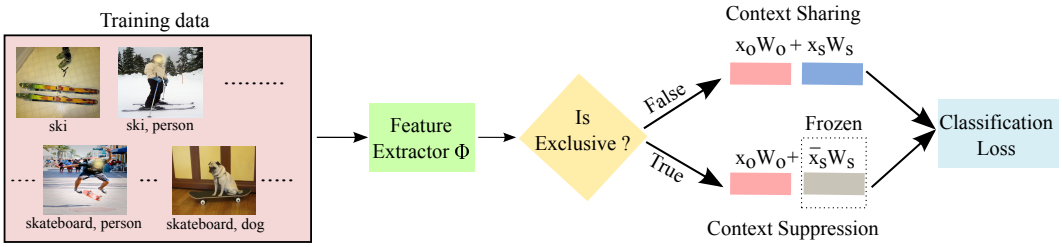

Figure 4. **Our feature splitting approach** where images and their associated category labels are provided as input. During training, we split the feature space into two equal sub spaces: x_o and x_s. If a training instance has a biased category occurring in the absence of context, we suppress x_s (no back-prop), forcing the model to leverage x_o. In all other scenarios, x_o and x_s are treated equally. At inference, the entire feature space is equally leveraged.

In scenarios where a biased category occurs in the absence of its context, we want to *enforce* the network to only rely on W_o by suppressing W_s. This step allows the network to explicitly capture the biased category-specific information when it occurs away from its context in W_o. On the other hand, when a biased category co-occurs with its context, we want to *encourage* the network to leverage both W_o and W_s. This would allow the network to jointly encode category and context in the full feature space.

To achieve this, we make two minor modifications to a standard classifier when a biased category occurs away from its typical context. First, we disable back propagation through W_s thereby forcing the network to learn only through W_o. Second, we set x_s to a constant value. We believe these two simple modifications allow us to suppress context in selective cases, i.e., when a biased category occurs away from its context. For instance, when *ski* occurs in the absence of its typical context *person*, our method suppresses W_s thereby encouraging W_o to encode its appearance; when *ski* co-occurs with *person*, both W_o and W_s are leveraged.

In practice, we set $x_s = \bar{x_s}$, where $\bar{x_s}$ is the average of x_s over the last 10 mini-batches, and allowed stabler training. Also, $\bar{x_s}$ is a closer approximation to the range of values x_s witnesses at test time.

Intuition behind weighted loss: An underlying aspect of our method is that the biased categories occur very rarely in the absence of their context, making the training data distribution skewed (see Sec. 3). This is a problem since W_o is learned solely from the (very few) samples with biased categories occurring in the absence of their typical context. We address this issue by associating a higher weight to such training samples. All other samples are weighed equally. Specifically, we define a weight α such that

$$\alpha = \begin{cases} \sqrt{\frac{|\mathbb{I}_b \cap \mathbb{I}_c|}{|\mathbb{I}_b \setminus \mathbb{I}_c|}}, & \text{when b occurs exclusively} \\ 1 & \text{otherwise} \end{cases} \quad (8)$$

Thus, α is the ratio of the number of training instances where category occurs in the presence vs. absence of context. A higher value of α for a given biased category indicates more data skewness. [3].

Given ground-truth label t and sigmoid function σ, our weighted binary cross-entropy loss is defined as follows:

$$L_{BCE} = -\alpha\,(t\log(\sigma(\hat{y})) + (1-t)\log(1-\sigma(\hat{y}))), \quad (9)$$

Figure 4 illustrates the proposed method. While a standard classifier jointly encodes category and context, it fails to recognize biased categories occurring without context. By contrast, our approach splits the feature space and represents biased categories occurring without context in a dedicated subspace. As we will show in results, due to selective context suppression, this feature subspace successfully captures category-specific information. Furthermore, in the second subspace, our method effectively leverages context when available and jointly encodes it with category.

As we show in results, leveraging context when available, distinguishes this method with the CAM-based method described in Sec. 4.1 and plays a key role in recognition performance. Further, while we selectively suppress context when a biased category occurs away from its context, the CAM-based method optimizes the mutual spatial overlap when a biased category co-occurs with context. We stress that both methods are applied only for the K biased category pairs; thus, misclassification loss for the other (non-biased) categories also plays an important role in learning. Finally, our method poses no constraints on the spatial extents of categories; thus, unlike our CAM-based approach, is extensible to attributes.

4.3. Training setup

Determining biased categories: For each category, we first identify other categories that occur frequently (at least $10\% - 20\%$ times, based on the dataset). Next, we partition the training data into non-overlapping $80 - 20$ split. We train a standard multi-class classifier with BCE loss on the 80% split and compute bias (Eq. 1) on the 20% split. While both methods proposed in this work can be applied to any number of biased category pairs, we found that setting $K = 20$ (Sec. 3.1) sufficiently captures biased categories in all the datasets we study here.

[3] In practice, we ensure α is at least α_{min} (a constant value > 1) when b occurs exclusively.

Datasets	Task	#Classes	#Train / #Test
MS COCO + Stuff [6]	object	171	82,783 / 40,504
UnRel [24]	object	43	- / 1,071
Deep Fashion [23]	attribute	250	209,222/40,000
AwA [35]	attribute	85	30,337 / 6,985

Table 1. **Properties of evaluation datasets.** For COCO-Stuff, we use object training and validation data from COCO-2014 split [22].

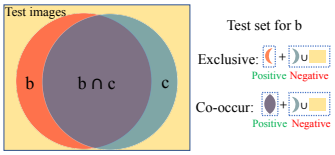

Figure 5. **Our evaluation setup** has two different test data distributions: (1) **exclusive** and (2) **co-occurring**. Our goal is to improve recognition performance on (1) without compromising on (2).

Optimization: We follow a two-stage training procedure: in the first stage, we start with a pre-trained network as a backbone and fine-tune it on all categories of a given dataset. This step ensures that the network learns useful context cues for the target task. In the second stage, we fine-tune our network and separately apply the modified loss defined in each proposed method. In the CAM-based approach, we reduce spatial overlap between the $|\mathrm{K}|$ category pairs; in the feature splitting method, we selectively suppresses context when the $|\mathrm{K}|$ biased categories occur exclusively and encourage context sharing in all other scenarios.

Implementation details: For both proposed methods, we use ResNet-50 [15] pre-trained on ImageNet as a backbone. For the first stage, an initial learning rate of 0.1 is used which is later divided by 10 following the standard step decay process for the learning rate. Following this, during the second stage of training, we train the network with a learning rate of 0.01 for both methods. For the CAM-based approach, we set λ_1 and λ_2 to be 0.1 and 0.01 respectively.

The input images are resized such that their shortest side is 256 and random crops of size 224×224 are used for training. To augment training data, we horizontal flip images. We use a batch size of 200 and stochastic gradient descent for optimization. Our model is implemented using PyTorch 1.0. Overall training time of both proposed methods is very close to that of a standard classifier and their inference time is exactly same as that of the standard classifier.

5. Experiments

In this section, we study the effectiveness of our approach across two tasks: object and attribute classification. We first describe our evaluation setup then report qualitative and quantitative performance on four image datasets against competitive baselines.

Datasets: We evaluate our approach on four multi-label datasets (summarized in Table 1). The choice of these datasets was driven by the fact that they exhibit strong co-occurrence bias. We summarize their co-occurrence statistics in the supplementary material. For DeepFashion [23], we only consider 250 most frequent attributes in the training data as other attributes do not have sufficient training samples. For Animals with Attributes (AwA) [17, 35], following common practice, we train an attribute prediction network on seen (40) animal categories and evaluate on unseen (10) categories. Finally, UnRel dataset [24] contains images of objects in unusual contexts, as they are obtained from rare and unusual triplet queries (e.g. "person ride giraffe," "dog ride bike"). We stress-test the generalizability of our model pre-trained on COCO-Stuff on this dataset.

Evaluation setup: We reiterate that our goal is to improve performance when highly biased categories occur exclusively, without losing much performance when they co-occur with other categories. Towards this end, for each dataset, we first determine the most biased category pairs $(\mathbb{S})$ following the approach in Sec. 3.1. Next, for these (b, c) category pairs, we report performance on two different test data distributions: (1) **exclusive:** b *never* occurs with c and (2) **co-occur:** b *always* co-occurs with c. We illustrate the two test distributions in Fig. 5. We report top-3 recall for DeepFashion [23] and mAP for all other datasets.

Baselines: Aside from a *standard* classifier trained with a binary cross-entropy loss for each category, we compare with the following state-of-the-art methods that tackle the issue of co-occurring bias: (1) *class balancing loss* [10] by treating the scenarios where biased categories occur exclusively as tail classes and (2) *attribute decorrelation* approach [17], where we replace the hand-crafted features with deep network features (`conv5` features of ResNet-50) for a fairer comparison. To further test the strength of our method, we designed the following competitive baselines:

1. *remove co-occur labels*, where we remove labels corresponding to c for each b in $\mathbb{S}$ during training. By removing supervision about co-occurring categories, we intend to soften the context-induced bias on the model.
2. *remove co-occur images* shares the same motivation as (2) but instead we remove training instances where the biased category and context co-occur.
3. *weighted loss*, where we apply 10 times higher weight to the loss when biased categories occur exclusively.
4. *negative penalty*, where we assign a large negative penalty if the network predicts co-occurring category in cases where a biased category occurs exclusively.

5.1. Object Classification Performance

5.1.1 Overall Results

In Table 2, we report performance on COCO-Stuff for the 20 most biased categories. First, we observe that the *standard* classifier has much better performance for co-occurring compared to exclusive test splits. This clearly demonstrates the inherent contextual bias present in COCO-Stuff, as *standard* classifier struggles when biased cate-

Methods	Exclusive	Co-occur
standard	24.5	**66.2**
class balancing loss [10]	25.0	66.1
remove co-occur labels	25.2	65.9
remove co-occur images	28.4	28.7
weighted loss	**30.4**	60.8
negative penalty	23.8	66.1
ours-CAM	26.4	64.9
ours-feature-split	28.8	66.0

Table 2. **Performance on COCO-Stuff** for the 20 most biased categories. Both our methods perform very well on all baselines except *weighted loss* and *remove co-occur images* on the exclusive test split, while successfully maintaining performance on the co-occurring test split.

gories do not co-occur with context. *class balancing loss* yields marginal gains indicating that weighing the rare exclusive cases alone cannot address contextual bias.

Next, we observe that both *ours-CAM* and *ours-feature-split* outperform *standard* by $\mathbf{1.9}\%$ and $\mathbf{4.3}\%$ respectively on the exclusive test set. *ours-feature-split* has a very marginal drop of 0.2% on the co-occurring split, compared to *standard*, while the performance drop is higher for *ours-CAM*. On categories such as "ski" and "skateboard" which have a very high co-occurrence bias with "person", the mAP boost from *ours-feature-split* is $\mathbf{24.2}\%$ and $\mathbf{19.5}\%$ respectively (per-class mAP for both methods in supp. material).

Comparison with other baselines: We note that *remove co-occur images* approach performs poorly as it relies only on the exclusive images of the biased categories and do not take advantage of the vast amount of co-occurring images which supply complementary visual information. *weighted loss* improves performance on the exclusive test split compared to *ours-feature-split* (30.4% vs. 28.8%), but significantly hurts performance on co-occurring split (60.8% vs. 66.0%). *negative penalty* does not hurt co-occurring split, but has inferior performance compared to our methods on the exclusive split. We also note that performance trends exhibited by these methods are consistent across all other datasets we test on; for all future experiments, we compare our methods with *standard* and *class balancing loss*.

Performance on the non-biased categories: We evaluate on the 60 non-biased object categories of COCO-Stuff and observe that both *ours-CAM* and *ours-feature-split* perform on par with *standard*, with a very mild drop of 0.2% overall mAP (details in supp. material). This indicates that our methods, while successfully improving performance for the biased categories, do not adversely effect the rest of the (non-biased) categories.

5.1.2 Qualitative Analysis

Next, we use CAM as a visualizing tool to analyze how our methods effectively tackle contextual bias.

***standard* vs. *ours-CAM*:** In Fig. 6, we present evidence where *standard* fails but *ours-CAM* succeeds[4] to recognize

[4]We determine 'success' when the predicted probability is $>= 0.5$ and 'failure' otherwise.

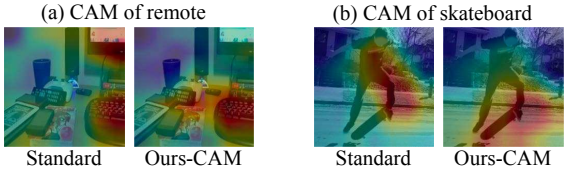

Figure 6. **Learning from the right thing: *ours-CAM*** (a) "remote" is contextually-biased by "person." In the absence of "person," *ours-CAM* focuses on the right pixel regions compared to *standard*. (b) "skateboard" co-occurs with "person." *standard* wrongly focuses on "person" due to contextual bias, while *ours-CAM* rightly focuses on "skateboard."

Figure 7. ***ours-CAM* vs. *ours-feature-split*** on the images for which *ours-feature-split* is able to recognize where as *ours-CAM* fails. *ours-CAM* primarily focuses on the object and does not use context whereas *ours-feature-split* makes use of context for better prediction.

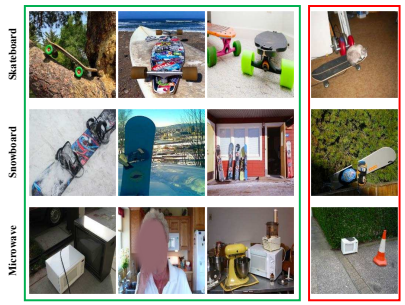

Figure 8. **Learning from the right thing: *ours-feature-split*** First 3 columns indicate **success cases** where *ours-feature-split* recognizes biased categories occurring away from their context while *standard* fails. Last column: **failure cases** where both *standard* and *ours-feature-split* fail.

biased categories. In both cases where a biased category co-occurs with context as well as occurs in its absence, *ours-CAM* focuses on the right category thus "learns from the right thing."

***ours-CAM* vs. *ours-feature-split*:** Fig. 7 presents cases where *ours-feature-split* succeeds but *ours-CAM* struggles to recognize biased categories. We observe that while *ours-CAM* rightly focuses on the category's pixel regions, *ours-feature-split* additionally leverages the available context and thus performs better.

***standard* vs. *ours-feature-split*:** The first 3 columns in Fig. 8 present evidence where the *standard* classifier fails but *ours-feature-split* succeeds. For example, our method is able to recognize "skateboard" and "snowboard" in the absence of "person", and "microwave" in the absence of "oven". By contrast, the *standard* classifier relies more on the context, thus fails on these images. The last column presents some failure cases where both *ours-feature-split* and *standard* fail when biased categories occur without context. Common failure cases are challenging scenarios when the image has poor lighting, the object is zoomed out and

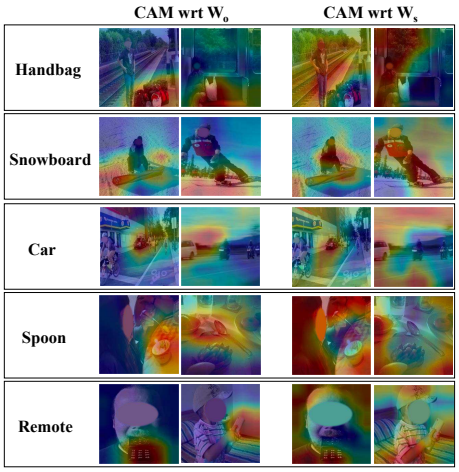

Figure 9. **Interpreting *ours-feature-split*** by visualizing CAMs with respect to W_o (left) and W_s (right). W_o has learnt to consistently focus on the actual category (e.g., car) while W_s captures context (e.g., road).

Methods	*standard*	*ours-CAM*	*ours-feature-split*
mAP	42.0	45.3	**52.1**

Table 3. **Cross-dataset experiment** where models trained on COCO-Stuff are applied without fine-tuning on UnRel. *ours-feature-split* yields huge boost over *standard* highlighting its generalizability on unseen data.

thus very small (e.g., microwave).

Analysing W_o and W_s: Recall that in Sec. 4.2, *ours-feature-split* is formulated with a goal to prominently capture biased category-specific features through W_o and context through W_s. We visually verify this by generating two distinct class activation maps: (i) x_o weighted by W_o and (ii) x_s weighted by W_s. From Fig. 9, it is evident that W_o learns to prominently focus on the category (e.g., handbag, car) and W_s on the co-occurring context (e.g., person, road).

5.2. Cross dataset experiment on UnRel

We next perform a *cross-dataset* experiment by taking our models trained on COCO-Stuff and testing them directly — without any fine-tuning — on UnRel dataset. UnRel has objects that are out-of-context (e.g., cat on a skateboard). Thus, a model that truly understands what the object is would be able to correctly classify it compared to a model that relies heavily on (or confuses the object with) context. Thus, this setting is a great testbed to evaluate our methods. Because we do not finetune, we evaluate only on the 3 categories of UnRel that overlap with the 20 biased categories of COCO-Stuff. From Table 3, we observe that both *ours-CAM* and *ours-feature-split* outperform *standard* by a large margins. This clearly demonstrates that both our methods learn from the right category and overcome contextual bias.

5.3. Attribute Classification

Here, we show that our approach of reducing contextual bias generalizes to attributes. Our CAM-based approach is not applicable to attributes, as they lack well-defined spatial extents (details in Sec. 4.1). As noted in Sec 5.1, the inherent contextual bias and difficulty in recognizing biased categories in the absence of their context leads to low scores on exclusive test split for all methods and datasets.

	DeepFashion (top-3 recall)		Animals with Attributes (mAP)	
Methods	Exclusive	Co-occur	Exclusive	Co-occur
standard	4.9	17.8	19.4	72.2
class balancing loss [10]	5.2	19.4	20.4	68.4
attribute decorrelation [17]	-	-	18.4	70.2
ours-feature-split	**9.2**	**20.1**	**20.8**	**72.8**

Table 4. **Attribute Classification Performance:** on DeepFashion and Animals with Attributes computed on the 20 most biased attributes. *ours-feature-split* offers boosts over all approaches for the exclusive test split, without hurting performance on the co-occurring split.

Results on DeepFashion: As is the common practice, we report per class top-3 recall on DeepFashion [23]. From Table 4, we note that *ours-feature-split* outperforms *standard* by a significant margin on both test splits. For attributes like *trapeze* and *bell* which exhibit strong co-occurrence with *striped* and *lace* respectively, *ours-feature-split* yields a boost of $\mathbf{21.2}\%$ and $\mathbf{17.4}\%$ top-3 recall respectively compared to *standard* classifier. We present per-attribute results and comparisons with other baselines in the suppl. material.

Results on Animals with Attributes: Animals with Attributes [35] suffers from severe bias among attributes, e.g. *blue* and *spots* are highly correlated to *coastal* and *long leg* respectively. In this task, the goal is to learn an attribute classifier on "seen" animal categories (e.g "spots" attribute from the animal category "dalmatian") and evaluate the model's generalizability on *unseen* animal categories (e.g. "spots" attribute on the unseen animal category "leopard"). From Table 4, we observe that *ours-feature-split* offers gains on the exclusive test split over other methods without hurting the co-occurring case. In particular, we outperform *attribute decorrelation* [17], which was specifically designed to decorrelate attributes.

6. Conclusion

We demonstrated the problem of contextual bias in popular object and attribute datasets by showing that standard classifiers perform poorly when biased categories occur away from their typical context. To tackle this issue, we proposed two simple yet effective methods to decorrelate feature representations of a biased category from its context. Both methods perform better at recognizing biased classes occurring away from their co-occurring context while maintaining the overall performance. More importantly, our methods generalize to new unseen datasets and perform significantly better than standard methods. Our current framework tackles contextual bias between pairs of categories; future efforts should leverage more available (scene or category) information and model relationships between them. Extending proposed methods to tasks like object detection and video action recognition is a worthy future direction.

Acknowledgments. This work was supported in part by NSF CAREER IIS-1751206.

References

[1] Amit Alfassy, Leonid Karlinsky, Amit Aides, Joseph Shtok, Sivan Harary, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Laso: Label-set operations networks for multi-label few-shot learning. In *CVPR*, 2019.

[2] Mohsan Alvi, Andrew Zisserman, and Christoffer Nellåker. Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *ECCV*, 2018.

[3] Ehud Barnea and Ohad Ben-Shahar. Exploring the bounds of the utility of context for object detection. *CVPR*, 2019.

[4] Irving Biederman, Robert J. Mezzanotte, and Jan C. Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, 1982.

[5] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *NIPS*, 2016.

[6] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, 2018.

[7] Nitesh Chawla, Kevin Bowyer, Lawrence Hall, and Philip Kegelmeyer. Smote: synthetic minority oversampling technique. *JAIR*, 2002.

[8] Myung Jin Choi, Antonio Torralba, and Alan S Willsky. Context models and out-of-context objects. *Pattern Recognition Letters*, 2012.

[9] Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*, 2017.

[10] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019.

[11] Terrance de Vries, Ishan Misra, Changhan Wang, and Laurens van der Maaten. Does object recognition work for everyone? In *CVPRW*, 2019.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[13] Santosh K Divvala, Derek Hoiem, James H Hays, Alexei A Efros, and Martial Hebert. An empirical study of context in object detection. In *CVPR*, 2009.

[14] Charles Elkan. The foundations of cost-sensitive learning.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[16] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *ECCV*, 2018.

[17] Dinesh Jayaraman, Fei Sha, and Kristen Grauman. Decorrelating semantic visual attributes by resisting the urge to share. In *CVPR*, 2014.

[18] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Repmet: Representative-based metric learning for classification and few-shot object detection. In *CVPR*, 2019.

[19] Aditya Khosla, Tinghui Zhou, Tomasz Malisiewicz, Alexei A Efros, and Antonio Torralba. Undoing the damage of dataset bias. In *ECCV*, 2012.

[20] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *CVPR*, 2019.

[21] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. In *CVPR*, 2019.

[22] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[23] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, 2016.

[24] Julia Peyre, Ivan Laptev, Cordelia Schmid, and Josef Sivic. Weakly-supervised learning of visual relations. In *ICCV*, 2017.

[25] Marta Recasens, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. Linguistic models for analyzing and detecting biased language. In *ACL*, 2013.

[26] Amir Rosenfeld, Richard Zemel, and John K Tsotsos. The elephant in the room. *arXiv preprint arXiv:1808.03305*, 2018.

[27] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NIPS*, 2017.

[28] Pierre Stock and Moustapha Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *ECCV*, 2018.

[29] Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing: Literature review. *arXiv preprint arXiv:1906.08976*, 2019.

[30] Kevin Tang, Manohar Paluri, Li Fei-Fei, Rob Fergus, and Lubomir Bourdev. Improving image classification with location context. In *CVPR*, 2015.

[31] Tatiana Tommasi, Novi Patricia, Barbara Caputo, and Tinne Tuytelaars. A deeper look at dataset bias. In *Domain adaptation in computer vision applications*. 2017.

[32] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR*. 2011.

[33] Emiel van Miltenburg. Stereotyping and bias in the flickr30k dataset. *arXiv preprint arXiv:1605.06083*, 2016.

[34] Yang Wang and Minh Hoai. Pulling actions out of context: Explicit separation for effective combination. In *CVPR*, 2018.

[35] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *TPAMI*, 2018.

[36] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *arXiv preprint arXiv:1707.09457*, 2017.

[37] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning Deep Features for Discriminative Localization. *CVPR*, 2016.

[38] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *CVPR*, 2019.