ASPIRE: Language-Guided Data Augmentation for Improving Robustness Against Spurious Correlations

Anonymous ACL submission

Abstract

Neural image classifiers can often learn to make predictions by overly relying on non-predictive features that are spuriously correlated with the class labels in the training data. This leads to poor performance in real-world atypical scenarios where such features are absent. This paper presents **ASPIRE** (Language-guided Data **A**ugmentation for **SP**ur**I**ous correlation **RE**moval), a simple yet effective solution for supplementing the training dataset with images *without* spurious features, for robust learning against spurious correlations via better generalization. ASPIRE, guided by language at various steps, can generate non-spurious images without requiring any group labeling or existing non-spurious images in the training set. Precisely, we employ LLMs to first extract foreground and background features from textual descriptions of an image, followed by advanced language-guided image editing to discover the features that are spuriously correlated with the class label. Finally, we personalize a text-to-image generation model using the edited images to generate diverse in-domain images *without* spurious features. ASPIRE is complementary to all prior robust training methods in literature, and we demonstrate its effectiveness across 4 datasets and 9 baselines and show that ASPIRE improves the worst-group classification accuracy of prior methods by 1% - 38%. We also contribute a novel test set for the challenging Hard ImageNet dataset.

1 Introduction

Spurious correlations are unintended associations or biases learned by models, between the input image and the target label, often resulting from factors like data selection biases (Torralba and Efros, 2011; Jabri et al., 2016). The repeated co-occurrence of certain features (like foreground objects or backgrounds), with a more than *average* chance, within instances of a particular class leads the model to learn shortcuts and focus on these spurious non-predictive features for prediction than core ones. For example, most of the images in ImageNet dataset (Deng et al., 2009) labeled as *Dog Sled* also show a dog, and image classifiers trained on ImageNet fail to correctly identify an image of a dog sled *without* a dog in it.

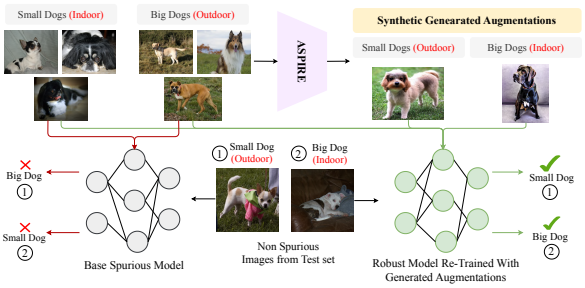


Figure 1: Overview of **ASPIRE**. Given a training dataset, ASPIRE automatically detects non-predictive spuriously correlated features for each class (e.g., indoor background for small dogs) and generates synthetic images without them (small dogs in an outdoor background). These images can then be added to the train set to learn a more robust image classifier.

Instances of a class in the training set where the co-occurring spurious features are present are commonly known as *majority groups*, while atypical instances where such spurious features are absent are known as *minority groups*. Deep neural networks trained on these datasets poorly generalize on minority groups (naturally due to their scarcity) and thus can exhibit significant performance degradation on minority groups in the test (Sagawa et al., 2019), or in real-world scenarios when encountering domain shift (Arjovsky et al., 2019). When training over-parameterized deep neural networks, there are multiple solutions with the same loss values at any given training stage, and the optimizer usually gravitates towards a solution with lesser complexity (or tends to learn a shortcut) (Wilson et al., 2017; Valle-Perez et al., 2018; Arpit et al., 2017; Kalimeris et al., 2019). When faced with co-occurring spurious features, the optimizer may pref-

erentially utilize them, as they often require less complexity than the anticipated semantic signals of interest (Bruna and Mallat, 2013; Bruna et al., 2015; Brendel and Bethge, 2019; Khani and Liang, 2021). *Even powerful classifiers like CLIP and ViT undergo a significant drop in performance when exposed to minority group images in the test (Yang et al., 2023; Kirichenko et al., 2023).*

Motivation. Learning classifiers robust to spurious correlations is an active area of research (Sagawa* et al., 2020; Liu et al., 2021a; Kirichenko et al., 2023), and has the potential to improve various Computer Vision (CV) applications such as visual question-answering (Liu et al., 2023d), retrieval (Kong et al., 2023; Kim et al., 2023), classification (Liu et al., 2021a), etc. have shown to consistently In prior work, researchers generally employed different learning techniques with the assumption that annotated data for the minority groups existed in the training dataset. Most of these works are built on the same base principle: improved generalization on minority groups can lead to a more robust classifier. Despite extensive research in deep learning indicating that more data may lead to better generalization, little effort has been made to leverage this principle specifically for building robust classifiers. Additionally, we argue that it is impractical to manually collect and label minority group images for real-world, large-scale datasets. For example, in more complex datasets like the Hard ImageNet, beyond the commonly evaluated CelebA (Liu et al., 2015) and Waterbirds (Welinder et al., 2010), a single class of images may have multiple spuriously correlated features. Thus, identifying all such features through human perception to collect and label minority group images is a difficult task.

Main Contributions. In this paper, we present ASPIRE, a novel technique to augment existing image classification datasets with diverse non-spurious images for building robust image classifiers. Intuitively, our solution exploits the fact that more data can lead to *better generalization* on minority groups (Sagawa et al., 2020; Liu et al., 2021b). Guided by language, ASPIRE does not depend on any additional image annotations or human-labeled non-spurious data and only requires a training dataset and a standard model trained using Empirical Risk Minimization (ERM) to identify most of the spurious correlated features for each class in the training dataset. ASPIRE first selects a small portion of the total instances in the training set, misclassified by a classifier after ERM training. These selected images are then captioned, and an LLM extracts the tokens from the caption that describe the foreground objects and background. This is followed by editing the image using advanced language-guided image editing pipelines to remove or replace one object at a time and predicting the class of the edited image using the standard ERM classifier. We attribute the objects or background features that lead to the highest miss-prediction (due to its absence) as *plausible* spurious correlations learned by the model. Finally, we personalize a diffusion model on the edited images to generate diverse in-domain synthetic images for each class with our desired features, i.e., without the *plausible* spurious correlations detected by ASPIRE. To summarize, our main contributions are as follows:

- We propose ASPIRE, a method to expand existing datasets with non-spurious images to build more robust image classifiers. ASPIRE is dataset-agnostic (works with any dataset with one or multiple spuriously correlated features per class), training-method agnostic (complements all other methodologies proposed in prior work), and does not need any additional labeled supervision of spurious features or non-spurious images.
- We extensively evaluate ASPIRE on 4 datasets and 9 baselines and show that augmentations generated by ASPIRE improve the worst-group accuracy of all baselines. Additionally, we perform extensive qualitative analysis to prove the effectiveness of ASPIRE.
- We contribute a novel test set for the Hard ImageNet dataset (Moayeri et al., 2022) equally balanced with spurious and non-spurious images to promote research in this space.

2 Methodology

Preliminaries. In this section, we provide an overview of our proposed approach. Fig. 2 pictorially describes the various steps in ASPIRE. Let's assume we have a training dataset $\mathcal{D}_{train}$ = $\{x_i,y_i\}$, where every group of images belonging to a particular class predominantly has images with co-occurring spurious features, also known as the majority group. $\mathcal{D}_{train}$ might have a much smaller number of non-spurious images, or might not, which is also known as the minority group.

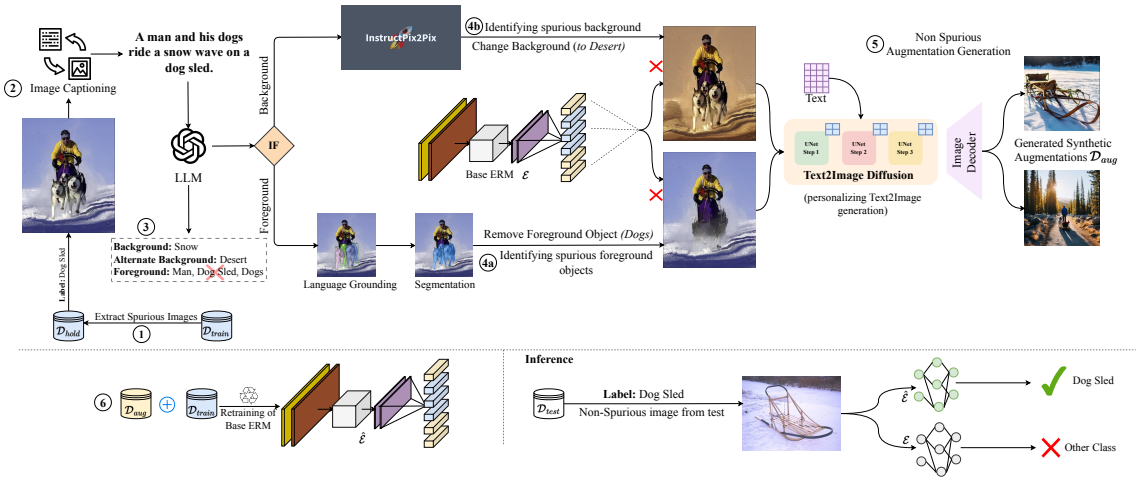


Figure 2: Illustration of **ASPIRE**: ASPIRE follows a 6-step process to improve the robustness against spurious correlations. (1) We first train a base classifier $\mathcal{E}$ using ERM on the entire training set and extract images with features that are spuriously correlated to construct $\mathcal{D}_{hold}$. (2) We caption each image in $\mathcal{D}_{hold}$. (3) We feed the caption to a LLM and extract the foreground objects and background for each image. (4a) We remove one foreground object at a time and predict the class of the edited image $\mathcal{E}$. If $\mathcal{E}$ predicts incorrectly, we consider the object as a plausible spurious correlation learned by $\mathcal{E}$ for that class. (4b) We edit the image to change its original background with an alternative background suggested by the LLM and follow the process to similar to (4a). (5) We personalize a text-to-image diffusion model using edited images from the previous step for the top-k unique items leading to the highest number of wrong predictions. (6) We re-train $\mathcal{E}$ using the generated augmentations to obtain $\hat{\mathcal{E}}$.

We do not assume our training dataset to have any group labeling or additional supervision, like labeling for spurious objects. We also have a test dataset $\mathcal{D}_{test} = \{x_i, y_i\}$ where both groups are represented equally. Additionally, we have a model $\mathcal{E}$ trained on $\mathcal{D}_{train}$, using naive Empirical Risk Minimization (ERM). Thus, $\mathcal{E}$ would already identify the majority group images in $\mathcal{D}_{test}$ with remarkable accuracy; our primary objective is to improve the performance of the classifier on the minority group without hurting the model's overall performance. The next subsections describe each step in detail.

(1) Extracting $\mathcal{D}_{hold}$ using $\mathcal{E}$. We use $\mathcal{E}$ to extract a small hold-out set from $\mathcal{D}_{train}$, which we denote as $\mathcal{D}_{hold}$. $\mathcal{D}_{hold}$ should consist of images with spurious correlations in the train (or the majority group), which we will use in our later stages to detect the specific features that are the spurious correlations. Precisely, we first identify training examples that are correctly classified by a standard ERM model and then randomly select p% of them for constructing $\mathcal{D}_{hold}$. We are inspired by prior work in this space and build on the heuristic that a well-trained classifier tends to have low majority group loss (and subsequently high majority group accuracy) (Liu et al., 2021a; Nam et al., 2020).

(2) Image Captioning on $\mathcal{D}_{hold}$. As mentioned earlier, ASPIRE depends on language guidance to achieve its primary objective of generating synthetic, non-spurious images. Thus, in this step, we generate a textual description of each image in $\mathcal{D}_{hold}$, which can capture foreground and background information in the image. To accomplish this, we use a state-of-the-art image captioning model, GIT (Wang et al., 2022). We expect our image description to include information about most of the visible foreground objects and the predominant background, and we found captions generated by GIT to meet these requirements and not suffer from spurious correlations themselves. As captioning tools get better, we acknowledge that replacing GIT with its improved counterparts will improve the performance of ASPIRE even further.

(3) Extracting objects and backgrounds from captions. After captioning, we use LLMs to extract the phrases in the caption that correspond to foreground and background objects and the single predominant background. We assume our search space for identifying spurious correlations to be bounded within them, which is a reasonable assumption in most real-world cases and also in line with most prior work in this space (Joshi et al., 2023). Recent LLMs have been shown to possess superior reasoning abilities, and we employ GPT-4 for our task (OpenAI, 2023). LLaMa-2 70B (Touvron et al., 2023) also proved to be competitive in this task. However, GPT-4 made fewer mistakes. An example of the input and output of this step of ASPIRE is as follows: *Original Caption*:

"A man with two dogs and a sled in the snow.", *Original Label*: "Dog Sled". *Output:* foreground: ["man","dogs"], background:["snow"], alternate background:["desert"]. For simplicity, let us denote the list of identified foreground and background objects for image x_i as $\mathcal{F}_i$ and the predominant background as $\mathcal{B}_i$ (more about the alternate background in Step **4.b.**). The task of extracting objects and backgrounds from text captions is effectively an information extraction task that involves understanding the structure of the sentence and the relationship between the words, and we found LLMs to deal better with anomalies and out-of-distribution text scenarios than traditional NLP methodologies (algorithmic details about the traditional NLP method can also be found in Appendix A.2). Additionally, we want the identified objects or background to ignore the actual class label. This is crucial for the ASPIRE pipeline, as we do not want to edit the core feature in the image (discussed in detail in the next subsections). This can also be challenging as the class label may or may not exactly appear in the caption. However, we found LLMs to complete this task with remarkable accuracy and ASPIRE to be able to handle minor errors (due to top-*k* selection described later in this section). We use a single generic prompt with annotated exemplars for all datasets, which can be found in Appendix A.6.

(4.a.) Identifying spurious foreground objects. The primary objective of this step is to identify one or several unique features per class that are plausible spurious correlations. To achieve this, we build on recent advancements in language-guided image in-painting to remove one object at a time identified in $\mathcal{F}_i$ followed by allowing $\mathcal{E}$ to predict the class of the edited image. If $\mathcal{E}$ predicts the image correctly, we do not do anything with that image. If $\mathcal{E}$ predicts an image incorrectly, we identify the removed object as a plausible spurious correlation for the class c in the dataset and add the image to a set $\mathcal{D}_{synth}$ (which we later use to personalize text-to-image generation). Additionally, we add the text phrase of the spurious object to another set $\mathcal{T}_{synth}$.

Precisely, for every fore-ground object f in $\mathcal{F}_i$, we first localize the object using Grounding DINO (Liu et al., 2023c), which takes as input the text phrase of f identified from the caption and outputs a bounding box bb for f. This is followed by extracting the segmentation map $\mathcal{M}$ for f using Segment Anything (Kirillov et al., 2023), which accepts bb as the segmentation prompt. $\mathcal{M}$ is then used to remove f from x_i using LaMa image inpainting (Suvorov et al., 2022). For detailed information on the workings of Grounding DINO, Segment Anything, and LaMa, we request our readers refer to the original paper.

(4.b.) Identifying spurious backgrounds. The primary objective of this step is to identify if the predominant background of the image x_i serves as a spurious correlation for the particular class c of images in the dataset to which x_i belongs. Following a hypothesis similar to (**4.a.**), we assume that if removing the background b in $\mathcal{B}_i$ from x_i can lead $\mathcal{E}$ to a wrong prediction, b can be a plausible spurious correlation. However, removing the background altogether (and keeping just the foreground items) not only disrupts the image semantics but is also not representative of real-world cases. Prior work also shows that removing the background forces the model to pay attention to foreground objects (Kirichenko et al., 2023) which are not suitable for our use case. Thus, we employ recent advances in instruction-based image editing to achieve this task. We first leverage the superior reasoning abilities of an LLM to suggest an alternate contrasting background $\tilde{b}$ for the image from its caption. Next, we instruct InstructPix2Pix (Brooks et al., 2023) to convert the background of x_i from b to $\tilde{b}$. Similar to the previous step, if $\mathcal{E}$ predicts the image correctly, we do not do anything with that image. However, if $\mathcal{E}$ predicts an image incorrectly, we identify the original background as a plausible spurious correlation and add the edited image to $\mathcal{D}_{synth}$ while we add the original text phrase of the background from the caption to $\mathcal{T}_{synth}$.

(5) Non-spurious augmentation generation. The primary objective of this step is to generate in-domain non-spurious images $\mathcal{D}_{aug}$ for every class in the dataset $\mathcal{D}_{train}$. These generated augmentations can then be used to supplement the training dataset $\mathcal{D}_{train}$ followed by re-training $\mathcal{E}$ to reduce its reliance on the spurious correlations. Generating *in-domain* augmentations without non-spurious features is crucial to the success of our approach as out-of-distribution samples may adversely affect model performance (Trabucco et al., 2023). The most trivial approach would be to generate $\mathcal{D}_{aug}$ by prompting any open-source text-to-image model. However, there exist two primary roadblocks to this approach: **(1)** Open-source diffusion models trained on internet-scale data generate diverse im-

ages for diverse prompts. Thus, prompting these models does not confirm the consistency of generations with the underlying distribution. **(2)** These models also posses spurious correlations or biases themselves (Trabucco et al., 2023). For example, prompting Stable Diffusion with the prompt: *"picture of a dog sled"* generates dog sleds with dogs most of the time. Attaching negative words with the prompts (like *"picture of a dog sled without a dog"*) often leads to the same spurious images.

To overcome the aforementioned problems and generate in-domain images with the desired non-spurious features, we resort to personalizing a text-to-image generation model. Specifically, we train Stable Diffusion using textual-inversion (Gal et al., 2023) with samples from top-k phrases in $\mathcal{T}_{synth}$, and their corresponding images in $\mathcal{D}_{synth}$. Textual-inversion effectively learns concepts and style from a small set of images for each class in $\mathcal{D}_{synth}$ by just fine-tuning a single token in the embedding layer (which in our case is just the original class label) without over-fitting the generation model. $\mathcal{D}_{synth}$ is the perfect candidate for extracting this small set as it contains non-spurious images, i.e., images without spurious features and concepts. Finally, we prompt the fine-tuned model to generate $n\times$ diverse samples for $\mathcal{D}_{aug}$.

Top-k selection. Recall that $\mathcal{D}_{synth}$ and their corresponding text phrases in $\mathcal{T}_{synth}$ represent *all* wrongly predicted edited instances, i.e., they have a diverse set of foreground objects and backgrounds for each class. Thus, we attribute only the top-k unique items in $\mathcal{T}_{synth}$ with the highest frequencies as the spurious correlation associated with that class and use images from only the top-k items for diffusion personalization. However, due to diversity in generated captions, text phrases corresponding to the same type of objects and backgrounds may be represented in $\mathcal{T}_{synth}$ in diverse forms, for e.g., ["dogs","dog","two dogs", $\cdots$]. Thus, before selecting the top-k items, we first collapse all the similar phrases to one by first finding the root for all phrases in $\mathcal{T}_{synth}$ by stemming and then calculating the cosine similarity between the glove embedding of the roots (to account for dissimilar roots, for e.g., "snow" and "snowy mountain"). Items with a cosine similarity of ≥ 0.90 are collapsed into one.

(6) Re-training the base classifier $\mathcal{E}$. Once we have generated $\mathcal{D}_{aug}$, we add the generated images to the existing $\mathcal{D}_{train}$ to re-train our standard classifier $\mathcal{E}$. As mentioned earlier, the ASPIRE augmentation methodology is training-method-agnostic, and the augmentations generated can be coupled with any existing training approach from literature. The next Section describes how we add ASPIRE augmentations to our baseline training pipelines.

3 Experimental Setup

Datasets. To evaluate the effectiveness of ASPIRE, we experiment on 4 benchmark datasets, including Waterbirds (Sagawa et al., 2019), CelebA (Liu et al., 2015), SPUCO Dogs (Joshi et al., 2023) and Hard ImageNet (Moayeri et al., 2022). The Waterbirds dataset, generated synthetically by combining images of birds from the CUB dataset (Wah et al., 2011) and backgrounds from the Places dataset (Zhou et al., 2017), has 4 groups of images in training and testing datasets including waterbirds on water background, waterbirds on land background, landbirds on water background and landbirds on land background. The minority groups for the dataset (groups with the least number of samples in the training set) are waterbirds on land and landbirds on water. The main challenge is correctly identifying the minority groups in the test. For CelebA, we perform the hair color prediction task, which has 4 groups of images, including blond and non-blond males and blond and non-blond females. The minority group is blond males. SPUCO Dogs has 4 groups of images, including big dogs in indoor and outdoor settings and small dogs in indoor and outdoor settings. The minority groups are big dogs indoors and small dogs outdoors. The Hard ImageNet dataset has images from 15 ImageNet synsets and is more complex than the other 3 datasets, does not have group labeling, and has multiple spurious correlations for each class. For more details, we request our readers to refer to Moayeri et al. (2022). Since the dataset does not have a test set, we contribute a novel expert-annotated test dataset with 25 spurious and 25 non-spurious images per class. The spurious and non-spurious features for each class were inspired by the original paper. More details about dataset statistics and annotation can be found in Appendix A.1.

Baselines. To prove the efficacy of ASPIRE augmentations, we add ASPIRE augmentations to the original training pipeline for various robust training methods proposed in literature. Precisely, we employ Group DRO (Sagawa et al., 2019), SUBG (Idrissi et al., 2022), Just Train Twice (JTT), Learning from Failure (LfF) (Nam et al., 2020), Correct-n-Contrast (CnC) (Zhang et al., 2022),

Method	Waterbirds		CelebA		SpucoDogs		Hard ImageNet	
	Worst-group Acc. (%)	Avg Acc. (%)	Worst-group Acc. (%)	Avg Acc. (%)	Worst-group Acc. (%)	Avg Acc. (%)	Worst-group Acc. (%)	Avg Acc. (%)
ERM	74.4	96.9	43.4	95.5	42.3	74.5	12.6	74.3
ERM + Azizi et al.	71.8	**97.1**	36.2	**96.7**	39.6	75.4	10.7	76.7
ERM + Gowal et al.	75.7	85.6	45.7	96.4	46.8	73.7	23.3	83.4
ERM + ASPIRE	**78.7**$_{\pm 1.31}$ (+4.3)	89.6$_{\pm 1.10}$	**50.5**$_{\pm 0.79}$ (+7.1)	95.4$_{\pm 1.08}$	**51.6**$_{\pm 0.86}$ (+9.3)	**75.5**$_{\pm 1.18}$	**50.1**$_{\pm 1.26}$ (+37.5)	**96.5**$_{\pm 1.32}$
LfF (Nam et al., 2020)	78.0	91.2	77.2	85.1	70.2	80.8	58.8	92.5
LfF + Azizi et al.	74.2	**92.3**	74.4	85.7	67.5	**81.6**	54.3	92.6
LfF + Gowal et al.	81.0	89.3	78.2	85.8	72.9	80.9	60.3	92.7
LfF + ASPIRE	**83.2**$_{\pm 0.20}$ (+5.2)	91.4$_{\pm 1.12}$	**81.7**$_{\pm 0.43}$ (+4.5)	**86.3**$_{\pm 1.25}$	**75.4**$_{\pm 0.38}$ (+5.2)	80.9$_{\pm 0.31}$	**63.8**$_{\pm 0.30}$ (+5.0)	**92.7**$_{\pm 0.21}$
Group DRO (Sagawa et al., 2019)	91.4	93.5	88.9	92.9	75.4	82.8	65.6	91.8
Group DRO + Azizi et al.	88.2	94.1	85.6	93.2	71.7	84.1	62.8	**92.9**
Group DRO + Gowal et al.	91.6	94.2	89.8	93.7	76.3	83.4	65.5	91.7
Group DRO + ASPIRE	**92.8**$_{\pm 0.49}$ (+1.4)	**94.6**$_{\pm 0.49}$	**90.1**$_{\pm 1.08}$ (+1.2)	**94.3**$_{\pm 0.92}$	**78.7**$_{\pm 1.26}$ (+3.3)	**84.3**$_{\pm 0.58}$	**67.4**$_{\pm 1.01}$ (+1.8)	92.4$_{\pm 0.59}$
JTT (Liu et al., 2021b)	86.7	93.3	81.1	88.0	73.0	80.4	63.5	90.6
JTT + Azizi et al.	83.2	**94.9**	78.3	90.2	71.8	**82.2**	61.4	92.4
JTT + Gowal et al.	87.5	94.2	83.8	89.6	74.1	81.1	64.1	91.9
JTT + ASPIRE	**90.2**$_{\pm 1.16}$ (+3.5)	94.6$_{\pm 1.24}$	**85.7**$_{\pm 0.64}$ (+4.6)	**91.6**$_{\pm 0.75}$	**75.5**$_{\pm 1.33}$ (+2.5)	81.7$_{\pm 1.12}$	**65.2**$_{\pm 0.54}$ (+1.7)	**92.9**$_{\pm 0.82}$
DivDis (Lee et al., 2022)	85.6	87.3	55.0	90.8	39.3	65.5	15.5	71.8
DivDis + Azizi et al.	84.2	**88.6**	53.7	**92.2**	37.5	66.4	13.7	77.2
DivDis + Gowal et al.	86.3	87.4	56.1	91.2	42.1	66.3	23.9	76.9
DivDis + ASPIRE	**87.2**$_{\pm 0.49}$ (+1.6)	87.8$_{\pm 0.84}$	**57.4**$_{\pm 1.13}$ (+2.4)	91.6$_{\pm 0.66}$	**43.6**$_{\pm 1.48}$ (+4.3)	**67.1**$_{\pm 1.22}$	**35.5**$_{\pm 0.82}$ (+20.0)	**77.6**$_{\pm 0.34}$
SUBG (Idrissi et al., 2022)	88.9	91.2	86.2	89.1	74.2	81.5	62.3	90.9
SUBG + Azizi et al.	86.5	91.8	85.4	**91.3**	72.3	81.6	60.5	**92.9**
SUBG + Gowal et al.	89.7	91.7	88.2	89.9	75.6	81.7	64.8	91.6
SUBG + ASPIRE	**90.7**$_{\pm 0.62}$ (+1.8)	**92.1**$_{\pm 0.88}$	**88.6**$_{\pm 1.37}$ (+2.4)	90.1$_{\pm 0.64}$	**77.5**$_{\pm 0.73}$ (+3.3)	**83.5**$_{\pm 0.92}$	**66.7**$_{\pm 1.22}$ (+4.4)	92.4$_{\pm 0.63}$
Correct-n-Contrast (Zhang et al., 2022)	88.7	90.6	88.1	89.4	73.7	81.2	60.5	91.7
Correct-n-Contrast + Azizi et al.	84.3	**93.4**	85.2	91.3	70.8	**85.6**	58.7	**93.3**
Correct-n-Contrast + Gowal et al.	89.1	91.7	88.7	90.6	74.9	82.6	63.2	92.1
Correct-n-Contrast + ASPIRE	**90.8**$_{\pm 1.18}$ (+2.1)	92.6$_{\pm 1.48}$	**89.9**$_{\pm 1.45}$ (+1.8)	**91.3**$_{\pm 0.28}$	**76.8**$_{\pm 1.10}$ (+3.1)	83.1$_{\pm 1.04}$	**65.9**$_{\pm 0.94}$ (+5.4)	91.9$_{\pm 1.11}$
MaskTune (Taghanaki et al., 2022)	78.0	91.2	77.9	92.5	31.6	59.2	33.0	58.5
MaskTune + Azizi et al.	75.8	**93.4**	73.3	**93.5**	26.3	**63.4**	28.9	**61.3**
MaskTune + Gowal et al.	79.3	85.2	78.8	88.1	35.2	60.7	35.3	55.8
MaskTune + ASPIRE	**81.6**$_{\pm 1.28}$ (+3.6)	91.3$_{\pm 0.54}$	**81.2**$_{\pm 0.22}$ (+3.3)	92.8$_{\pm 0.38}$	**37.5**$_{\pm 0.33}$ (+5.9)	61.3$_{\pm 1.05}$	**41.0**$_{\pm 0.61}$ (+8.0)	60.2$_{\pm 0.37}$
DFR (Kirichenko et al., 2023)	81.7	90.1	80.5	85.3	78.8	83.2	33.3	95.7
DFR + Azizi et al.	78.6	**92.7**	78.3	88.4	72.1	85.1	29.5	**96.3**
DFR + Gowal et al.	83.1	86.5	83.4	86.2	81.0	84.4	35.2	92.0
DFR + ASPIRE	**85.3**$_{\pm 1.34}$ (+3.6)	91.7$_{\pm 0.79}$	**85.5**$_{\pm 0.64}$ (+5.0)	**89.5**$_{\pm 0.51}$	**84.2**$_{\pm 0.83}$ (+5.4)	**87.5**$_{\pm 0.57}$	**37.5**$_{\pm 0.39}$ (+4.2)	96.2$_{\pm 0.91}$

Table 1: Average and worst-group test accuracies of all baselines trained with and without ASPIRE augmentations. ASPIRE substantially improves the worst-group accuracy of all baselines (in the range of 1% - 38%) with just 1× more augmentations.

Deep Feature Reweighting (DFR). (Kirichenko et al., 2023) and MaskTune (Asgari et al., 2022), To this list, we add the standard Empirical Risk Minimization (ERM) baseline, trained using SGD without any additional modifications. Additionally, we compare ASPIRE augmentations with augmentations generated using the methods proposed by Gowal et al. (2021) and Azizi et al. (2023). More details on baselines and how ASPIRE augmentations were added for training can be found in Appendix A.4. *We do not experiment with CLIP (Radford et al., 2021) or other LVLMs like LLaVa (Liu et al., 2023a) as there is no simple method to fine-tune them for robustness against spurious correlations proposed in literature. ASPIRE is only meant to complement methods proposed on the standard framework of fine-tuning vision encoders for image classification (all baselines mentioned), and we only experiment with the systems proposed under this framework.*

Hyper-parameters. For training the base ERM model, we train the model for 100 epochs with a learning rate of $1e^{-3}$ using the SGD optimizer with a weight decay of $1e^{-4}$. For training all other baselines, we use the original hyper-parameter settings proposed by the authors in their original paper. This includes the seed settings and the number of runs for every model. We use just 1 × augmentations of non-spurious images. Though this is possible for us as all our current datasets are also annotated with group labels, the number of ASPIRE augmentations to be added can be decided using hyper-parameter search, and we noticed no signs of over-fitting till 3× augmentations (see Appendix 1). For top-k, we resort to k=3 post a hyper-parameter search among k={1,2,3,4,5}. k=3 seemed to capture the most major spurious correlations while ignoring the minor ones. Examples of extracted top-k can be found in Figure 4 and Appendix 1. For prompting InstructPix2Pix, we use Text CFG=7.5 and Image CFG=1.5. Prompt in Appendix A.6.

4 Results and Analysis

4.1 Quantitative Analysis

Table 1 compares the results of 9 baselines trained with and without ASPIRE augmentations. Worst-group accuracy corresponds to the accuracy of minority groups (or non-spurious images) in the test.

	Worst-group Acc.(%)	Avg. Acc.(%)
ASPIRE - Step 4.a.	70.65	86.54
ASPIRE - Step 4.b.	66.40	82.67
ASPIRE - Step 5.	65.75	81.44
ASPIRE	71.80	87.39

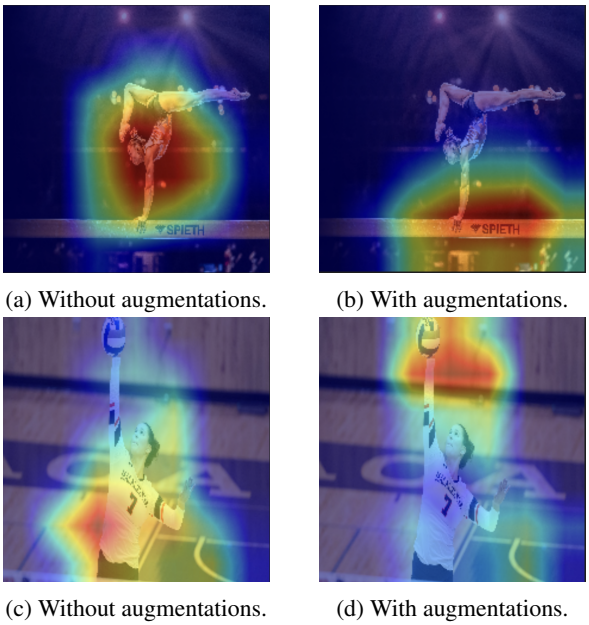
Table 2: Ablation study of ASPIRE. "-" indicates that the step was removed from the ASPIRE pipeline. All results are averaged across all datasets.

As we clearly see, with just 1× augmentations, ASPIRE improves the average accuracy of our baselines by 0.1%-22.2% and the worst-group accuracy of our baselines by 1.2%-37.5%. *ASPIRE consistently achieves higher gains in worst-group accuracy and only undergoes a slight drop in average accuracy in some settings, which is in line with prior art and our primary motivation of improving robustness against spurious correlations.* We notice the highest gains in Hard ImageNet, a fundamentally more difficult dataset with no minority group images in the training dataset and multiple spurious correlations per class. Our standard ERM model also witnesses the highest gains among all other baselines. On average, our 2-stage training baselines improve by a higher margin on average than 1-stage baselines due to improved explicit generalization over ASPIRE augmentations. The method proposed by Gowal et al. (2021) consistently underperforms ASPIRE, thereby highlighting that explicitly removing spurious features in the generated dataset improves robustness. On the other hand, the method proposed by Azizi et al. (2023) significantly underperforms ASPIRE in worst group accuracy but outperforms in average accuracy in some settings. This is due to the fact that standard data augmentation amplifies spurious correlations already present in the training set as it generates images with similar features to those on which it is conditioned.

Ablations. Table 2 removes certain key components in the ASPIRE pipeline to prove their efficacy. As we see, the ASPIRE performance decreases significantly when the image generation step is removed (and only edited images are used for training the robust classifier). Additionally, ASPIRE undergoes a sharper drop in performance when foreground identification is removed than the background, which we attribute to the design of the test set minority groups of existing datasets.

4.2 Qualitative Analysis

Fig. 3 illustrates the GradCAM visualizations of the features used by the standard ERM model trained with and w/o ASPIRE augmentations for two classes from the Hard ImageNet dataset, *Volleyball* and *Horizontal Bar*. When trained with ASPIRE augmentations, the model tends to focus better on core features corresponding to the actual class than spurious ones. Fig. 4 illustrates examples of original images, edited images (edited by the ASPIRE pipeline), and ASPIRE-generated augmentations. ASPIRE successfully captures the major spurious cues learned by a model (shown in top-k) and generates diverse images *without* them. We show more examples in Appendix B and C and illustrate some failure cases in Appendix D.



(a) Without augmentations. (b) With augmentations.

(c) Without augmentations. (d) With augmentations.

Figure 3: GradCAM visualizations of the features used by the standard ERM model trained with and w/o ASPIRE augmentations on the Hard ImageNet dataset (*Balance beam* top and *Volleyball* bottom). As clearly visible, when trained with ASPIRE augmentations, the model tends to focus better on core features than spurious ones (more in Appendix B).

5 Literature Review

Geirhos et al. (2020) provides a detailed survey on how image classification models perform poorly when trained on datasets with spurious correlations. Following this, a lot of works explore SGD training dynamics and inductive biases of such models in the presence of spurious correlations (Nagarajan et al., 2021; Pezeshki et al., 2021; Rahaman et al., 2019). Shah et al. (2020) shows how deep neural networks, trained using ERM, can take shortcuts and learn to rely on spurious features rather than core features for a class. They call this phenomenon the *extreme simplicity bias*. Hermann and Lampinen (2020) and Jacobsen et al. (2019)

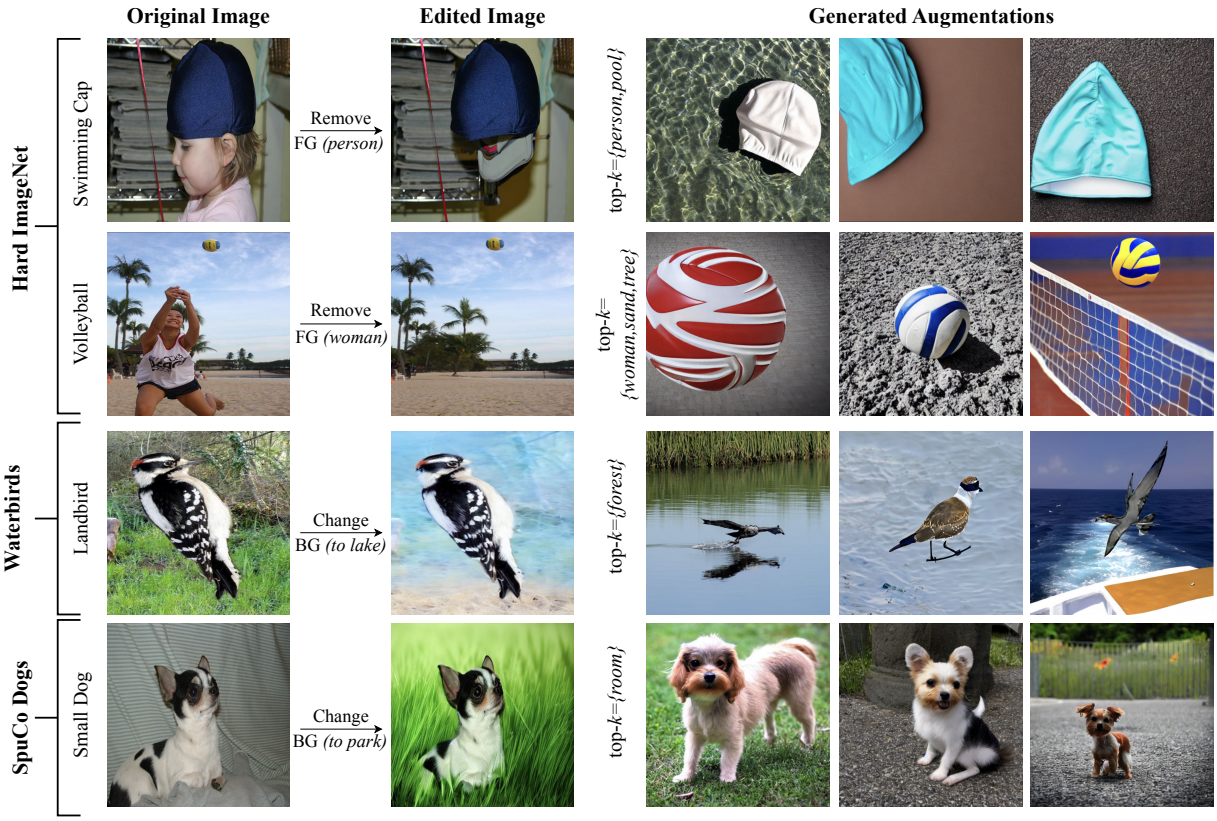


Figure 4: Examples of **Original Images**, **Edited Images** from the ASPIRE pipeline and **Generated Augmentations**. To the left of the **Generated Augmentations**, we also mention the top-k spurious correlations discovered by ASPIRE for the particular class. ASPIRE generates diverse augmentations with the desired non-spurious features that can be used to train robust models.

further present examples with both natural and synthetic images, highlighting instances where these networks overlook core features. Shinoda et al. (2023) explore the types of shortcuts that are more likely to be learned.

A plethora of methods in literature propose novel training strategies for improving robustness against spurious correlations (Ben-Tal et al., 2011; Hu et al., 2018; Sagawa* et al., 2020; Oren et al., 2019; Zhang et al., 2021). A detailed explanation of all these methods can be found in Section 3 and Appendix A.4.

The use of synthetic data for improving the performance of downstream CV tasks has been explored extensively in the past. For data-driven generative models, GANs have remained the predominant approach to date (Brock et al., 2018; Li et al., 2022). Very recently, He *et al.* (He et al., 2022) employ large-scale text-to-image models like GLIDE (Nichol et al., 2021) to augment training data with synthetic images and show improvement in image classification performance.

Prior work explores language-guidance for image generation for varied objectives. For example, Prabhu et al. (2023) proposes to generate counterfactual images for stress-testing image classification models. On similar lines, Wiles et al. (2022) and Vendrow et al. (2023) propose to identify failure cases and spurious correlations using augmented data generated using language-guidance. Finally, Dunlap et al. (2023) proposes to adapt a model to new domains using augmented data. To the best of our knowledge, generative data augmentation with or without language-guidance for improving robustness against spurious correlations has not yet been explored.

6 Conclusion

In this paper, we present ASPIRE, a novel data augmentation methodology to augment existing datasets with non-spurious minority group images to build robust and de-biased image classifiers. We evaluate ASPIRE on 4 benchmark datasets with 9 baselines and show that ASPIRE augmentations improve the worst-group accuracy of all baselines while maintaining average accuracies.

Limitations and Future Work

As part of future work, we would like to address the current limitations of ASPIRE, which include:

1. ASPIRE is limited to how well image captioning models can describe the image. Though captioning models improve over time, we would like to explore novel ways to resolve this bottleneck. For example, Large Multi-Modal Language Models like LLaVa (Liu et al., 2023b) have been shown to perform exceptionally well at generating detailed captions of input images.

2. The edited images used to personalize text-to-image generation may sometimes be of low quality, leading to poor augmentations in more complex datasets, and we would like to explore ways to resolve this bottleneck. We also acknowledge that the advancement of text-to-image diffusion models to better follow text prompts will eventually lead to performance improvement of ASPIRE.

3. The different components of ASPIRE add computational overhead to the ASPIRE pipeline (over just the ERM classifier). However, it should be noted that a wealth of literature in offline data augmentation for NLP and CV tasks (through synthetic data generation) almost always employs computationally expensive foundation models for additional data generation. Textual-inversion fine-tuning of diffusion models used in our experiments is also computationally cheaper than full fine-tuning. Lastly, as part of future work, we would like to explore computationally cheaper alternatives to an LLM for information extraction from captions. Additionally, the augmentation process is completely offline and needs to be done just once for each dataset.

4. We also illustrate some failure cases of ASPIRE in Appendix D.

Ethics Statement

Image generation models are prone to generating harmful, obscene and offensive context for certain classes pf objects, we prevent this from happening in ASPIRE by using a safety checker for the Stable Diffusion model which estimates whether a generated images could be considered offensive or harmful. For the CelebA dataset, ASPIRE performs modification where genders of people are swapped to debias the model towards certain attributes related a class. This approach is used only to improve the fairness and debiasing of the model.

References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. 2017. A closer look at memorization in deep networks. In *International conference on machine learning*, pages 233–242. PMLR.

Saeid Asgari, Aliasghar Khani, Fereshte Khani, Ali Gholami, Linh Tran, Ali Mahdavi Amiri, and Ghassan Hamarneh. 2022. Masktune: Mitigating spurious correlations by forcing to explore. *Advances in Neural Information Processing Systems*, 35:23284–23296.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. 2023. Synthetic data from diffusion models improves imagenet classification. *Transactions on Machine Learning Research*.

Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. 2011. Robust solutions of optimization problems affected by uncertain probabilities. *Advanced Risk & Portfolio Management® Research Paper Series*.

Wieland Brendel and Matthias Bethge. 2019. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *arXiv preprint arXiv:1904.00760*.

Andrew Brock, Jeff Donahue, and Karen Simonyan. 2018. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402.

Joan Bruna and Stéphane Mallat. 2013. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886.

Joan Bruna, Stéphane Mallat, Emmanuel Bacry, and Jean-François Muzy. 2015. Intermittent process analysis with scattering moments.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.

Lisa Dunlap, Clara Mohri, Devin Guillory, Han Zhang, Trevor Darrell, Joseph E. Gonzalez, Aditi Raghunathan, and Anna Rohrbach. 2023. Using language to extend to unseen domains. In *The Eleventh International Conference on Learning Representations*.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. 2023. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Sven Gowal, Sylvestre-Alvise Rebuffi, Olivia Wiles, Florian Stimberg, Dan Andrei Calian, and Timothy A Mann. 2021. Improving robustness using generated data. *Advances in Neural Information Processing Systems*, 34:4218–4233.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. 2022. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*.

Katherine L. Hermann and Andrew K. Lampinen. 2020. What shapes feature representations? exploring datasets, architectures, and training. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Weihua Hu, Gang Niu, Issei Sato, and Masashi Sugiyama. 2018. Does distributionally robust supervised learning give robust classifiers? In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *JMLR Workshop and Conference Proceedings*, pages 2034–2042. JMLR.org.

Badr Youbi Idrissi, Martin Arjovsky, Mohammad Pezeshki, and David Lopez-Paz. 2022. Simple data balancing achieves competitive worst-group-accuracy. In *Conference on Causal Learning and Reasoning*, pages 336–351. PMLR.

Allan Jabri, Armand Joulin, and Laurens Van Der Maaten. 2016. Revisiting visual question answering baselines. In *European conference on computer vision*, pages 727–739. Springer.

Joern-Henrik Jacobsen, Jens Behrmann, Richard Zemel, and Matthias Bethge. 2019. Excessive invariance causes adversarial vulnerability. In *International Conference on Learning Representations*.

Siddharth Joshi, Yu Yang, Yihao Xue, Wenhan Yang, and Baharan Mirzasoleiman. 2023. Towards mitigating spurious correlations in the wild: A benchmark & a more realistic dataset. *arXiv preprint arXiv:2306.11957*.

Dimitris Kalimeris, Gal Kaplun, Preetum Nakkiran, Benjamin Edelman, Tristan Yang, Boaz Barak, and Haofeng Zhang. 2019. Sgd on neural networks learns functions of increasing complexity. *Advances in neural information processing systems*, 32.

Fereshte Khani and Percy Liang. 2021. Removing spurious features can hurt accuracy and affect groups disproportionately. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 196–205.

Jae Myung Kim, A Koepke, Cordelia Schmid, and Zeynep Akata. 2023. Exposing and mitigating spurious correlations for cross-modal retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2584–2594.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. 2023. Last layer re-training is sufficient for robustness to spurious correlations. In *The Eleventh International Conference on Learning Representations*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. *arXiv preprint arXiv:2304.02643*.

Fanjie Kong, Shuai Yuan, Weituo Hao, and Ricardo Henao. 2023. Mitigating test-time bias for fair image retrieval. *arXiv preprint arXiv:2305.19329*.

Yoonho Lee, Huaxiu Yao, and Chelsea Finn. 2022. Diversify and disambiguate: Learning from underspecified data. *CoRR*, abs/2202.03418.

Daiqing Li, Huan Ling, Seung Wook Kim, Karsten Kreis, Sanja Fidler, and Antonio Torralba. 2022. Bigdatasetgan: Synthesizing imagenet with pixel-wise annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21330–21340.

Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. 2021a. Just train twice: Improving group robustness without training group information. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6781–6792. PMLR.

Evan Zheran Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. 2021b. Just train twice: Improving group robustness without training group information.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023b. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. 2023c. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.

Yang Liu, Guanbin Li, and Liang Lin. 2023d. Cross-modal causal relational reasoning for event-level visual question answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738.

Mazda Moayeri, Sahil Singla, and Soheil Feizi. 2022. Hard imagenet: Segmentations for objects with strong spurious cues. *Advances in Neural Information Processing Systems*, 35:10068–10077.

Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. 2021. Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. 2020. Learning from failure: Debiasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

OpenAI. 2023. Gpt-4 technical report.

Yonatan Oren, Shiori Sagawa, Tatsunori B. Hashimoto, and Percy Liang. 2019. Distributionally robust language modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4227–4237, Hong Kong, China. Association for Computational Linguistics.

Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. 2021. Gradient starvation: A learning proclivity in neural networks. In *Advances in Neural Information Processing Systems*.

Viraj Uday Prabhu, Sriram Yenamandra, Prithvijit Chattopadhyay, and Judy Hoffman. 2023. LANCE: Stress-testing visual models by generating language-guided counterfactual images. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. 2019. On the spectral bias of neural networks.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*.

Shiori Sagawa*, Pang Wei Koh*, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks. In *International Conference on Learning Representations*.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization.

Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. 2020. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33.

Kazutoshi Shinoda, Saku Sugawara, and Akiko Aizawa. 2023. Which shortcut solution do question answering models prefer to learn? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13564–13572.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. 2022. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2149–2159.

Saeid Asgari Taghanaki, Aliasghar Khani, Fereshte Khani, Ali Gholami, Linh Tran, Ali Mahdavi-Amiri, and Ghassan Hamarneh. 2022. Masktune: Mitigating spurious correlations by forcing to explore.

Antonio Torralba and Alexei A Efros. 2011. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. 2023. Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*.

Guillermo Valle-Perez, Chico Q Camargo, and Ard A Louis. 2018. Deep learning generalizes because the parameter-function map is biased towards simple functions. *arXiv preprint arXiv:1805.08522*.

Joshua Vendrow, Saachi Jain, Logan Engstrom, and Aleksander Madry. 2023. Dataset interfaces: Diagnosing model failures using controllable counterfactual generation.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset.

Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. 2022. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*.

Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. 2010. Caltech-ucsd birds 200. Technical Report CNS-TR-201, Caltech.

Olivia Wiles, Isabela Albuquerque, and Sven Gowal. 2022. Discovering bugs in vision models using off-the-shelf image generation and captioning. In *NeurIPS ML Safety Workshop*.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. 2017. The marginal value of adaptive gradient methods in machine learning. *Advances in neural information processing systems*, 30.

Yu Yang, Besmira Nushi, Hamid Palangi, and Baharan Mirzasoleiman. 2023. Mitigating spurious correlations in multi-modal models during fine-tuning. *arXiv preprint arXiv:2304.03916*.

Jingzhao Zhang, Aditya Krishna Menon, Andreas Veit, Srinadh Bhojanapalli, Sanjiv Kumar, and Suvrit Sra. 2021. Coping with label shift via distributionally robust optimisation. In *International Conference on Learning Representations*.

Michael Zhang, Nimit S Sohoni, Hongyang R Zhang, Chelsea Finn, and Christopher Ré. 2022. Correct-n-contrast: A contrastive approach for improving robustness to spurious correlations. *arXiv preprint arXiv:2203.01517*.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464.

A Appendix

A.1 Dataset Details

Table A.8 shows dataset details for all 4 datasets used in our experiments. As clearly visible, there is a notable disparity between the number of images representing minority groups (non-spurious images) and those representing majority groups (images with spuriously correlated features). In contrast, the test set for each dataset maintains a balanced representation between the minority and majority groups. This can lead classifiers to quickly adopt spurious correlations, resulting in sub-optimal performance on the test set.

Dataset	Train		Test	
	Majority Group	Minority Group	Majority Group	Minority Group
Hard ImageNet	19097	0	375	375
Waterbirds	4555	240	2897	2897
CelebA	161383	1387	19782	180
Spuco Dogs	17000	1000	1000	1000

Table 3: Dataset details

A.2 Algorithm

Algorithm 1 describes algorithmically the ASPIRE pipeline. Readers can refer to the algorithm for a detailed step-by-step understanding of the workings of ASPIRE.

A.3 Traditional NLP algorithm details

Introduction Extracting foreground objects and the background from a caption using traditional Natural Language Processing (NLP) techniques and libraries like SpaCy involves several steps. Here's a general approach:

Text Preprocessing First, preprocess the text to ensure it's in a suitable format for analysis. This might include:

- Lowercasing all words.
- Removing punctuation and special characters.
- Tokenization: Breaking the text into individual words (tokens).

Part-of-Speech Tagging Use SpaCy to perform part-of-speech (POS) tagging, which identifies the grammatical parts of speech for each word (e.g., noun, verb, adjective). This is crucial for identifying potential objects and elements of the scene.

Named Entity Recognition (NER) Employ Named Entity Recognition to identify named entities in the text, which can include names of people, places, organizations, or other proper nouns. These entities can be part of the foreground or background.

Dependency Parsing Dependency parsing helps understand the grammatical structure of the sentence, showing how words relate to each other. This is useful to distinguish between main subjects

(likely foreground objects) and contextual elements (possibly background).

Chunking or Phrase Detection Use chunking or phrase detection to group together contiguous sequences of tokens that form meaningful phrases. Noun phrases, in particular, are often key in identifying objects and scene elements.

Identifying Foreground and Background Foreground Objects Typically, these are nouns or noun phrases that are the main subjects or objects of the sentence. They often appear with adjectives and are part of active clauses.

Background Information This can include descriptions of settings, locations, or contexts. Adverbial phrases and clauses, as well as descriptive language, can signal background details.

SpaCy Implementation Here's a simple implementation using SpaCy:

```
import spacy

# Load the SpaCy model
nlp = spacy.load("en_core_web_sm")

def extract_foreground_background(text):
    doc = nlp(text)

    foreground = []
    background = []

    for token in doc:
       # Check for nouns and proper nouns for foreground
      if token.pos_ in ["NOUN", "PROPN"]:
           foreground.append(token.text)

          # Background might be set by adverbial phrases or adjectives
        if token.pos_ in ["ADJ", "ADV"]:
           background.append(token.text)

        # Check for named entities
        if token.ent_type_:
         if token.ent_type_ in ["PERSON", "ORG", "GPE"]:
           foreground.append(token.text)
            else:
           background.append(token.text)

    return foreground, background

# Example Usage
text = "The cat sat on the mat in the sunny room."
foreground, background = extract_foreground_background(text)
print("Foreground:", foreground)
print("Background:", background)
```

A.4 Details on Baselines

To maintain training efficiency, for training each baseline with ASPIRE augmentations, we add only 1× more augmentations to the original dataset for CelebA, Waterbirds, and SPUCO Dogs, or effectively or effectively double the number of non-spurious minority group images in each dataset. These 3 datasets have labeled minority groups, and thus, the number of augmentations to be added amounted to the total minority group images in each class of the original dataset. For Hard ImageNet, we add as many more augmentations as the total number of original training samples in each class of the original dataset. We elaborate on the rationale behind the choice of our baseline setup in Appendix A.5, where we also describe why we choose not to compare ASPIRE with large multi-modal models. We next describe how we add ASPIRE augmentations to the original training pipeline for different baselines.

Emperical Risk Minimization (ERM) For this baseline, we compare a ResNet-50 model trained using ERM (with SGD) on the original dataset with a ResNet-50 model trained on the original dataset augmented with ASPIRE augmentations. For ERM, we just add ASPIRE augmentations to the initial training set.

1-stage training baselines. Group DRO (Sagawa et al., 2019) is a state-of-the-art method that uses group information on train and adaptively upweights worst-group examples during training. **SUBG** (Idrissi et al., 2022) is ERM applied to a random subset of the data where the groups are equally represented, which was recently shown to be a strong baseline. We also add ASPIRE augmentations to the initial training set for both baselines.

2-stage training baselines. Just Train Twice (JTT). JTT follows a 2-stage training process wherein they first identify training examples that are misclassified by a standard ERM model, and then train the final model by upweighting the examples identified in the first stage. **Learning from Failure** (LfF). (Nam et al., 2020) Similar to JTT, LfF follows a 2-stage training process wherein

Algorithm 1 ASPIRE Data Augmentation Algorithm

Data: Image Classification Dataset $\mathcal{D}_{train} \rightarrow \{x_i\ (Image), y_i\ (Label)\}$;
$\mathcal{E} = Classifier(x_i, y_i)$ // Image classification model
$\mathcal{C} = \text{Captioning}(x_i)$ // Image captioning model
$\mathcal{L} = \text{LLM}(Prompt, Captions, y)$ // LLM to extract foreground and background objects.
$\mathcal{BG} = \text{InstructPix2Pix}(b, \tilde{b}, x_i)$ // InstructPix2Pix to convert background of an image.
$\mathcal{G} = \text{GroundingDino}(f, x_i)$ // Creates bounding box around objects.
$\mathcal{S} = \text{SegmentAnything}(bb)$ // Extracts image segmentation maps from bounding boxes.
$\mathcal{I} = \text{InpaintAnything}(\mathcal{M}, x_i)$ // Removal of objects corresponding to segmentation maps.
$\mathcal{D}_{correct} \leftarrow \varnothing$ **for** x_i *in* $\mathcal{D}_{train}$ **do**
  // Consider only the images which are predicted correctly.
  if $\mathcal{E}(x_i) == y_i$ **then**
    $\mathcal{D}_{correct} \leftarrow \mathcal{D}_{correct} \cup \{(x_i, y_i)\}$;
  end
end
Sample $p\%$ of $\mathcal{D}_{correct}$ to create $\mathcal{D}_{hold}$.
$\mathcal{D}_{captions} \leftarrow \mathcal{C}(\mathcal{D}_{hold})$ // Caption the images in the holdout set.
$\mathcal{F}_i, \mathcal{B}_i \leftarrow \mathcal{L}(Prompt, \mathcal{D}_{captions}, \mathcal{D}^{y}_{hold})$ // Extract foreground and background objects by prompting the LLM.
$\mathcal{D}_{synth} \leftarrow \varnothing, \mathcal{T}_{synth} \leftarrow \varnothing$;
for $\{f\}$ *in* $\mathcal{F}_i$ **do**
  $bb \leftarrow \mathcal{G}(f, x_i)$; // Create the bounding boxes.
  $\mathcal{M} \leftarrow \mathcal{S}(bb)$; // Extract the segmentation maps.
  $x_i^{mod} \leftarrow \mathcal{I}(\mathcal{M})$; // Modify image by removing the foreground object.
  // Consider only the images which are predicted wrong after modification.
  if $y_i^{correct} \neq \mathcal{E}(x_i^{mod})$ **then**
    $\mathcal{D}_{synth} \leftarrow \mathcal{D}_{synth} \cup \{x_i^{mod}\}$;
    $\mathcal{T}_{synth} \leftarrow \mathcal{T}_{synth} \cup \{f\}$;
  end
end
for $\{b, \tilde{b}\}$ *in* $\mathcal{B}_i$ **do**
  $x_i^{mod} \leftarrow \mathcal{BG}(b, \tilde{b}, x_i)$;// Change image background as suggested by the LLM.
  // Consider only the images which are predicted wrong after modification.
  if $y_i^{correct} \neq \mathcal{E}(x_i^{mod})$ **then**
    $\mathcal{D}_{synth} \leftarrow \mathcal{D}_{synth} \cup \{x_i^{mod}\}$;
    $\mathcal{T}_{synth} \leftarrow \mathcal{T}_{synth} \cup \{b\}$;
  end
end
// Collapse synthetic dataset based on text phrases that are similar to each other. Select top-k items that have the highest count per image class in the dataset.
$\mathcal{T}^k_{synth}, \mathcal{D}^k_{synth} \leftarrow \text{TopK}(\text{Col}(\mathcal{T}_{synth}, \mathcal{D}_{synth}))$
Train the Stable Diffusion model $\mathcal{SD}$ using $\mathcal{D}^k_{synth}$.
Generate $\mathcal{D}_{aug}$ from $\mathcal{SD}$.
// Creating a new training dataset by combining the augmentations with the original training data.
$\mathcal{D}^{new}_{train} \leftarrow \mathcal{D}_{train} \cup \mathcal{D}_{aug}$;
// Retrain the original image captioning model on the new training data.
Retrain $\mathcal{E}$ on $\mathcal{D}^{new}_{train}$.

they first identify training examples that are misclassified by a biased ERM model, and then train the final model by re-weight training samples using the relative difficulty score based on the loss of the biased model. **Correct-n-Contrast** (CnC) (Zhang et al., 2022) detects the minority group examples similarly to JTT and uses a contrastive objective to learn representations robust to spurious correlations. **Deep Feature Reweighting** (DFR). (Kirichenko et al., 2023) DFR follows a 2-stage training process wherein they first fine-tune a pre-trained ResNet model (pre-trained on the entire ImageNet dataset) using ERM on the entire train split followed by re-training the last layer using a small set from the train with an equal number of instances for both majority and minority groups. **MaskTune.** (Asgari et al., 2022) follows a 2-stage training process, wherein they first fine-tune a ResNet model on a dataset using ERM on the entire train split followed by re-training the model with new masked data for one full epoch. For all these baselines, we add ASPIRE augmentations to the set used in the second stage of training.

A.5 Choice of Baselines

To the best of our knowledge, there exists no prior method in literature that generates minority group images to expand the training set. Most work has focused on devising novel training methods for robust classification, all of which are complementary to ASPIRE and compared to our method in this paper. As also mentioned in Section 5 of our paper, generative data augmentation for improving overall accuracy has been explored but is unrelated to our method. Additionally, the primary aim of ASPIRE is to improve the downstream performance of *image classification models*. We acknowledge that other types of models, like instruction-tuned Vision-Language Models (Liu et al., 2023b), might identify and classify the image correctly into a predefined class given specific prompts (again, this is an underexplored area in CV), but comparing this is beyond the scope of this paper and experimental setting. Our setting is consistent with most prior art in (methods listed in Table 1).

A.6 Prompts

GPT-4. The general-purpose prompt we use for GPT-4 is listed as follows: *I will provide you with a list of tuples. Each tuple in the list has 2 items: the first is a caption of an image and the second is the label of the image. For each, you will have to return a JSON with 3 lists. One list should be the list of all phrases from the caption that are objects that appear in the foreground of the image but ignore objects that correspond to the actual label (the label for the phrase might not be present exactly in the caption) (named 'foreground'). The second list should have the single predominant background of the image to the foreground objects (named 'background'). If you do not find a phrase that corresponds to the background, return an empty list for the background. The third is an alternative background for the image, an alternative to the background you suggested earlier (named 'alt'). Here are some examples which also show the format in which you need to return the output. Please just return the JSON in the following format:* ***Exemplars*** *⋯ and here is the caption:*. We will provide the exemplars on our GitHub.

InstructPix2Pix. The prompt we use for InstructPix2Pix is: *turn the background from original background to alternative background.*

A.7 Examples of top-k identified by ASPIRE

Table 4 shows the top-k spuriously correlated features (or groups of features) for each class and for each dataset. As mentioned earlier, due to diversity in captions, the same kind of foreground object or background may be expressed with different phrases. ASPIRE thus returns groups of top-k items rather than a single top-k item for each k.

A.8 Collection of Test-Set for Hard ImageNet

Our institution's Institutional Review Board(IRB) has granted approval for the data collection. We followed the following steps for collecting a test set of the Hard ImageNet dataset:

1. We first identified spurious features in the Hard ImageNet and verbalized them. These features were identified from annotations in the original proposed dataset by Moayeri et al. (2022).

2. 3 annotators with extensive vision and language experience collected 1/3rd of the total 750 images. The annotators were not hired from any crowdsourcing platform and, in fact, were volunteers from our organization. The only instruction that was provided was that the image should have the primary target label of the image, and while majority group images

should have the identified spurious features, minority group images should not.

3. Post this step, each annotator validated the images collected by the other annotators.

4. We filter the images for offensive content and replace them with non-offensive images, if any.

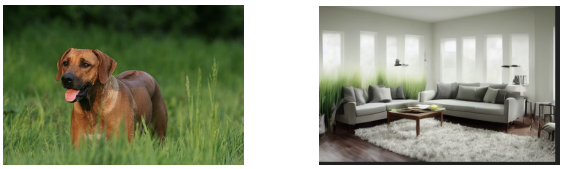
B GradCam Visualizations

Figure 5 and 6 illustrates the GradCAM visualizations of the features used by the last layer of a standard ERM model (ResNet-50), for prediction on the test set images, trained with and w/o ASPIRE augmentations on all 4 datasets used in our experiments. For a fair comparison, and to clearly show the benefits of ASPIRE, we show GradCAM visualizations only for the standard ERM model, as all other baselines perform explicit steps to reduce reliance on such features. Standard ERM training is also still the most widely used methodology for training image classifiers.

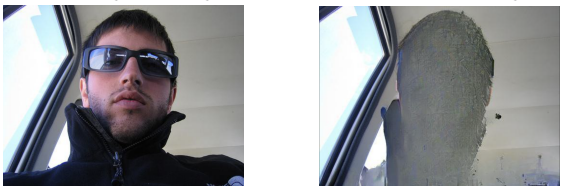
While Fig. 5 shows examples of the majority group images from the test set with spurious features, Figure 6 shows examples of minority group images without any spurious features. For Fig. 6 we show examples where the ERM classifier predicted the class of the image incorrectly (due to the absence of spurious features) while the one trained with ASPIRE predicted the class correctly. As we clearly see, in both cases, when trained with ASPIRE augmentations, the model learns to focus on core features than spurious ones while making predictions.

C Generation Examples

Table 8 shows examples of original images from the train set, the edited images from the ASPIRE pipeline and augmentations generated using ASPIRE. To the left of the generated augmentations, we also mention the top-k spurious correlations discovered by ASPIRE for the particular class. ASPIRE generates diverse augmentations with the desired non-spurious features that can be used to train robust models



(a) Original Image (b) Edited Image



(c) Original Image (d) Edited Image

Figure 7: Images illustrating cases of ASPIRE failures.

D ASPIRE Failure Cases

This section lists some failure cases of our proposed ASPIRE framework. As APSIRE leverages external models in its pipeline, the success of ASPIRE at times depends on the capabilities of these models. For generating augmentations, we notice the following failure cases:

1. **Superimposition of other foreground objects on the foreground object of interest.** Recall that ASPIRE detects foreground objects to remove (for spurious correlation detection) by parsing captions. These objects are then removed to detect if the object is spurious or not. In cases where another foreground object in the image is superimposing the foreground object, though the language-grounded pipeline is able to detect it properly, the inpainting model is at times unable to precisely remove just that object without not removing the superimposing foreground object and removes both the original object and the object superimposing it. An example is a *human* wearing *spectacles*, where we only want to remove the human, but the inpainting model removes both the human and the spectacles it is wearing. We provide an example of this case in Figure 7 (bottom row).

2. **Foreground objects change on changing background.** InstructPix2Pix, at times, tends to change the foreground object when prompted to change the background significantly, for example, changing *outdoor background* → *indoor background*. We provide an example of this case in Figure 7 (top row).

3. **Bias in Stable Diffusion.** Although our Stable Diffusion fine-tuning step, with textual

Dataset	Class	Top-*k* groups
Hard ImageNet	Volleyball	{volleyball player, female volleyball player, two volleyball players} {woman, young girl, girl, women} {beach, sandy beach}
	Spacebar	{keyboard, computer keyboard} {number pad} {mouse}
	Horizontal bar	{girl, little girl, young girl, two girls} {ballet, ballet barre} {olympic games}
	Snorkel	{boy, little boy, two boys} {blue swimsuit, swimsuit} {water, ocean}
	Balance Beam	{child, children, child's feet} {female gymnast, gymnast, gymnasts} {split, leg split}
	Seatbelt	{back of the car, back of a car} {handle, door handle} {seat, car seat, back seat}
	Dog sled	{two dogs, dogs, dog, husky dogs, dog team} {snowy hill, snowy landscape, snowy slope} {snow}
	Miniskirt	{woman, young woman, women} -
	Sunglasses	{pink hat, hats} {woman, blonde woman, women} {man, man's face}
	Howler monkey	{tree, tree branch, branch} {log, wooden bench, wooden beam} {leaves}
	Puck	{hockey player, hockey players, ice hockey players} {ice, ice rink} {hockey logo, hockey stick}
	Swimming cap	{boy, young boy, little boy} {pool, swimming pool, pool edge} {swimmer, swimmers}
	Patio	{chairs, chair, lawn chair} {building, buildings} {deck, new deck}
	Ski	{mountain, mountain top, snowy mountain, snowy mountain side} {ski poles, ski slope, ski lift} {person, group of people}
	Baseball player	{baseball field, field} {baseball game, game} {stadium}
Waterbirds	Waterbird	{ lake, stream, pond} {beach, sand} {water, river bank}
	Landbird	{forest, bamboo forest} {woods, trees} {branch, branches}
Spuco Dogs	Big dog	{field, grass field, green field, grassy field, green grass covered field, lush green field} {ground, playground} {floor, concrete floor}
	Small dog	{blanket, blue blanket, green blanket, red blanket, white blanket} {bed, dog bed, blue dog bed, small bed} {couch, red couch, gray couch}
CelebA	Blonde	{woman, lady}
	Not blonde	-

Table 4: Details of Top-*k* (*k*=3 for our experiments) spuriously correlated features per dataset and per class identified by ASPIRE. As discussed in the main paper, due to the fact that our captioning model generates diverse and variable phrases for the same type of object, we collapse these phrases into groups and instead work with groups (using the algorithm explained in Section 3 of the main paper) of spurious features. Groups in { } show spurious foreground objects while groups in { } are spurious backgrounds.

inversion, helps overcome its current biases, limited samples present for fine-tuning sometimes hurt this adaptation. For example, even after fine-tuning, Stable Diffusion might generate images of *dog sled* with dogs in it.

E Additional Details

E.1 Model Parameters

Git-Large-Coco has ≈ 300M parameters with a CLIP/ViT-L/14 image encoder and a 6 layer transformer decoder with 12 attention heads and 768

hidden-state. Stable Diffusion is a ≈ 860M parameter UNet and ≈ 123M parameter text encoder model. ResNet-50 has ≈ 25M parameters with 50 layers.

E.2 Compute Infrastructure

All our experiments are conducted on NVIDIA A100 GPUs. We batch prompted LLaMa-2 13B, with a BS of 16, where LLaMa-2 performed distributed inference on 4 A100 GPUs. That translates to 52.51 TFLOPs per batch. Fine-tuning SD with textual inversion with a BS of 8 takes and an avg. of ≈1 hour. For generating 1× augmentation, we use 1 A100 GPU for an average ≈1.2 hours in total.

E.3 Implementation Software and Packages

We implement all our models in PyTorch[1] and use the HuggingFace[2] implementations of ERM, Git (Wang et al., 2022), LLaMa-2 13B (Touvron et al., 2023) and InstructPix2Pix (Brooks et al., 2023). We also use the following code/components in our pipeline Grounding DINO[3] (Liu et al., 2023c), Segment Anything[4] (Kirillov et al., 2023) and Stable Diffusion using textual-inversion[5] (Gal et al., 2023).

We also use the following repositories for running the baselines: Group DRO[6] (Sagawa et al., 2019), SUBG[7] (Idrissi et al., 2022), JTT[8] (Liu et al., 2021b), Learning from Failure [9] (Nam et al., 2020), Correct-n-Contrast[10] (Zhang et al., 2022), Deep Feature Reweighting[11] (Kirichenko et al., 2023), MaskTune[12] (Asgari et al., 2022) and DivDis[13] (Lee et al., 2022).

All the above GitHub code has been released under an MIT license, free for research use.

E.4 Dataset Links

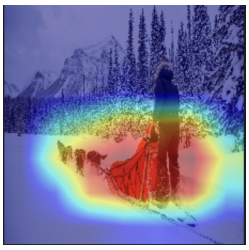
We use the Waterbirds [14], SPUCO Dogs [15], Hard ImageNet [16] and the CelebA [17] dataset. All the datasets are free for research use.

E.5 Potential Risks

Generative models are prone to hallucinate and potentially generate nonsensical, unfaithful or harmful content to the provided source input that it is conditioned on.

[1] https://pytorch.org/
[2] https://huggingface.co/
[3] https://github.com/IDEA-Research/GroundingDINO
[4] https://github.com/facebookresearch/segment-anything
[5] https://github.com/rinongal/textual_inversion
[6] https://github.com/kohpangwei/group_DRO
[7] https://github.com/facebookresearch/BalancingGroups
[8] https://github.com/anniesch/jtt
[9] https://github.com/alinlab/LfF
[10] https://github.com/HazyResearch/correct-n-contrast
[11] https://github.com/PolinaKirichenko/deep_feature_reweighting
[12] https://github.com/aliasgharkhani/masktune
[13] https://github.com/yoonholee/DivDis
[14] https://www.vision.caltech.edu/visipedia/CUB-200.html
[15] https://github.com/BigML-CS-UCLA/SpuCo
[16] https://openreview.net/forum?id=76w7bsdViZf
[17] https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

(a) Dog sled w/o ASPIRE

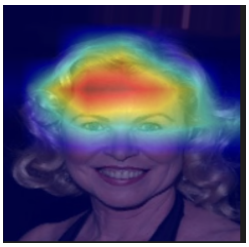


(b) Dog sled w/ ASPIRE

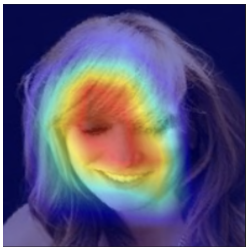
(c) Puck w/o ASPIRE

(d) Puck w/ ASPIRE

(e) Blonde female w/o ASPIRE

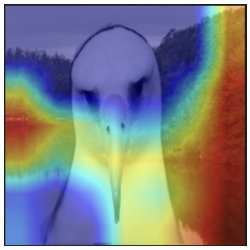


(f) Blonde female w/ ASPIRE

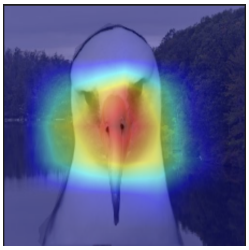


(g) Blonde female w/o ASPIRE

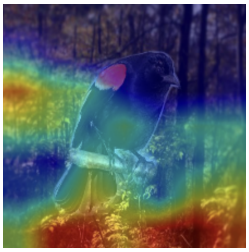
(h) Blonde female w/ ASPIRE



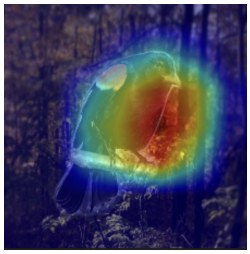
(i) Waterbird w/o ASPIRE



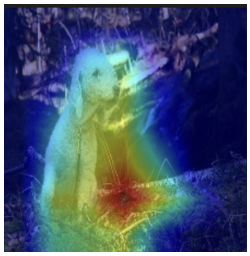
(j) Waterbird w/ ASPIRE



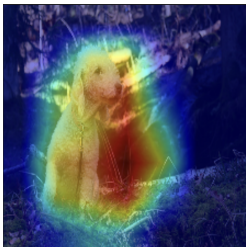
(k) Landbird w/o ASPIRE



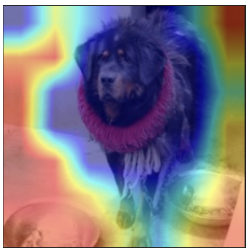
(l) Landbird w/ ASPIRE



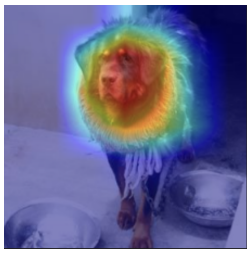
(m) Bigdog w/o ASPIRE



(n) Bigdog w/ ASPIRE

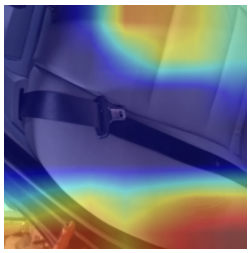


(o) Smalldog w/o ASPIRE

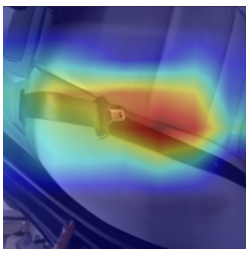


(p) Smalldog w/ ASPIRE

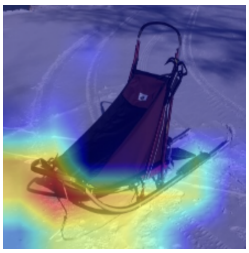
Figure 5: GradCam visualizations of features used by the last layer of a standard ERM model to predict majority group images (with spuriously correlated features) from the test set of 4 datasets.



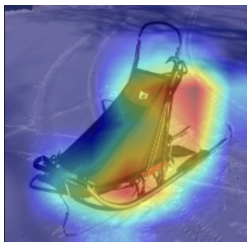
(a) Seatbelt w/o ASPIRE



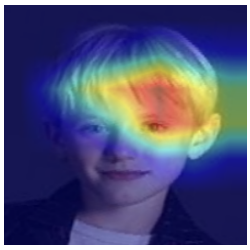
(b) Seatbelt w/ ASPIRE



(c) Dogsled w/o ASPIRE



(d) Dogsled w/ ASPIRE

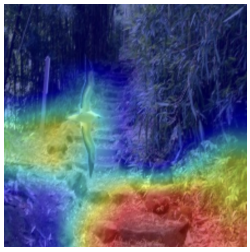


(e) Male blonde w/o ASPIRE

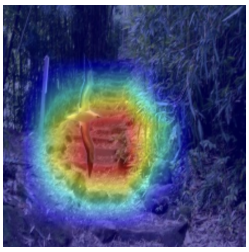
(f) Male blonde w/ ASPIRE

(g) Male blonde w/o ASPIRE

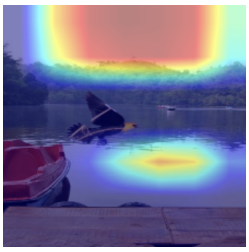
(h) Male blonde w/ ASPIRE



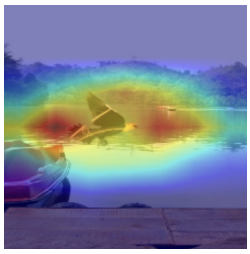
(i) Waterbird w/o ASPIRE



(j) Waterbird w/ ASPIRE

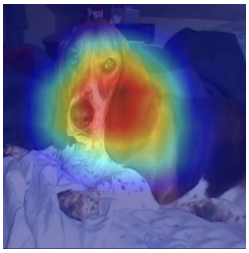


(k) Landbird w/ ASPIRE

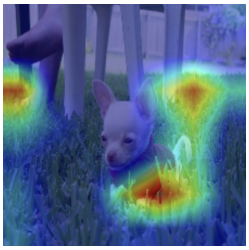


(l) Landbird w/o ASPIRE

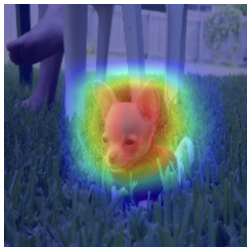
(m) Bigdog w/o ASPIRE



(n) Bigdog w/ ASPIRE



(o) Smalldog w/o ASPIRE



(p) Smalldog w/ ASPIRE

Figure 6: GradCam visualizations of features used by the last layer of a standard ERM model to predict majority group images (*without* spuriously correlated features) from the test set of 4 datasets.

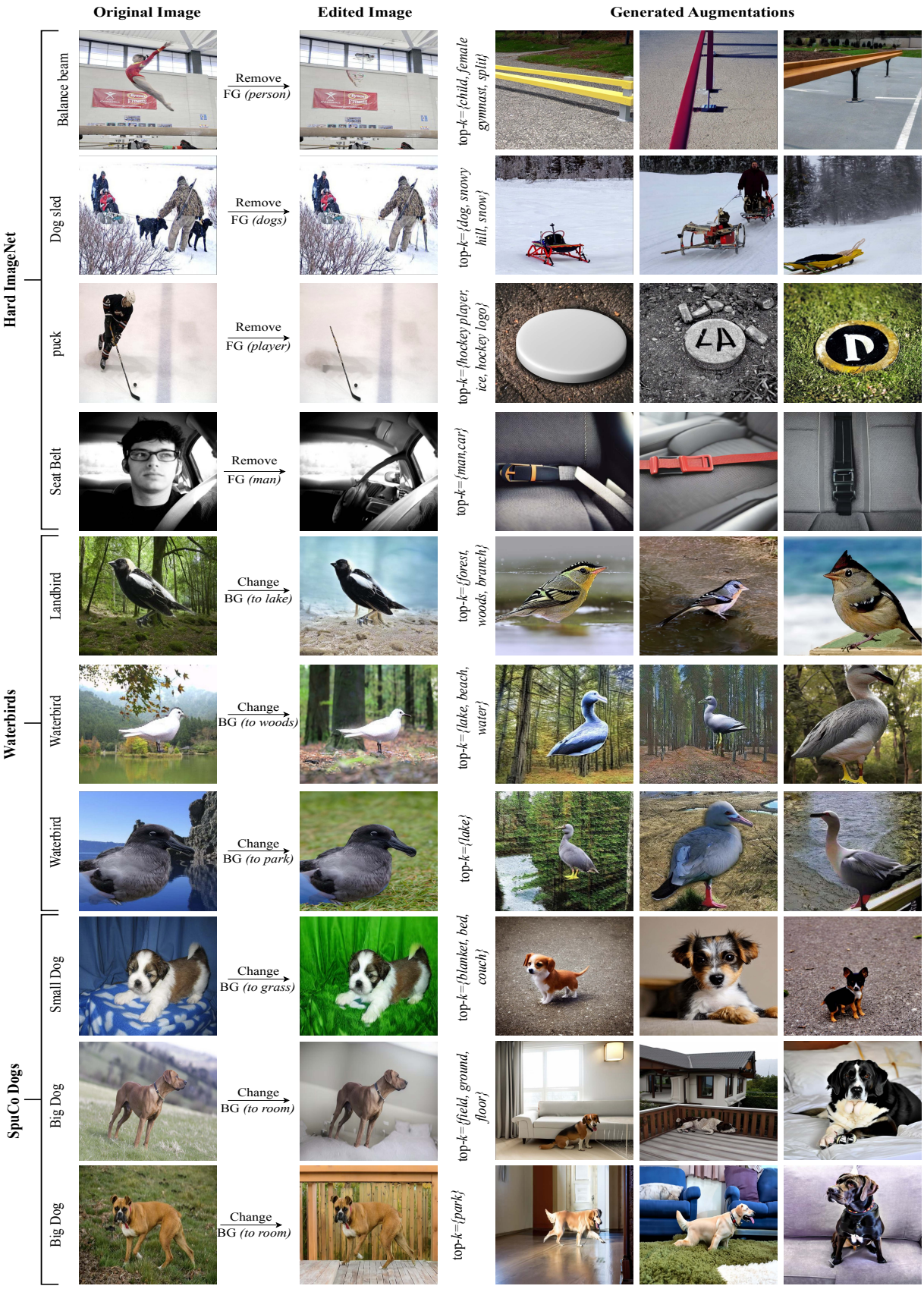


Figure 8: Examples of **Original Images**, **Edited Images** from the ASPIRE pipeline and **Generated Augmentations**. To the left of the **Generated Augmentations**, we also mention the top-k spurious correlations discovered by ASPIRE for the particular class. ASPIRE generates diverse augmentations with the desired non-spurious features that can be used to train robust models.