Corner Cases:
How Size and Position of Objects Challenge ImageNet-Trained Models

Anonymous authors
Paper under double-blind review

Abstract

Backgrounds in images play a major role in contributing to spurious correlations among different data points. Owing to aesthetic preferences of humans capturing the images, datasets can exhibit positional (location of the object within a given frame) and size (region-of-interest to image ratio) biases for different classes. In this paper, we show that these biases can impact how much a model relies on spurious features in the background to make its predictions. To better illustrate our findings, we propose a synthetic dataset derived from ImageNet1k, Hard-Spurious-ImageNet, which contains images with various backgrounds, object positions, and object sizes. By evaluating the dataset on different pretrained models, we find that most models rely heavily on spurious features in the background when the region-of-interest (ROI) to image ratio is small and the object is far from the center of the image. Moreover, we also show that current methods that aim to mitigate harmful spurious features, do not take into account these factors, hence fail to achieve considerable performance gains for worst-group accuracies when the size and location of core features in an image change.

1 Introduction

Spurious features are defined as features that are predictive of the class label without being directly related to it. Such features are usually helpful for object recognition when the object is placed in a *perfect* environment or context. An example of that would be a sea lion near a body of water. This is because most models learn to associate water with sea lions and vice versa. On the contrary, spurious features can be extremely harmful when the object or the "core" features are observed in an unusual environment or against a spurious background. This scenario can happen when the model is deployed in the wild. Deep neural networks can be fooled easily to predict the label from the spurious cues in the background without relying on "object" or "core" features in the image itself. Recently, a plethora of techniques have been proposed to mitigate the reliance on unnecessary cues for image classification. Sagawa et al. (2019) introduced a distributionally robust optimization technique which, coupled with strong regularization, helped in achieving high accuracies for data groups that have strong spurious feature reliance. Similarly, Kirichenko et al. (2022) address this problem by retraining the last layer of a DNN using equal data points from different groups with core and spurious backgrounds. These methods are helpful when the test set exhibits similar biases as the training data, yet they fail to achieve similar performance gains when these biases are explicitly removed.

Biases in datasets can hugely impact a deep neural network's performance. Earlier works have proven that convolutional neural networks are not entirely translation invariant and have the capacity to learn location information about objects Biscione & Bowers (2021). Some studies have found that models perform poorly on untrained locations Biscione & Bowers (2020). Similarly, object size within an input frame can lead to models performing badly when the sizes differ at inference time. The deep learning community has tried to mitigate the effect of these biases by proposing different data augmentation techniques that ensure that models are robust to changes in size and locations of the objects. However, the impact of the aforementioned factors in the presence of spurious features remains less explored.

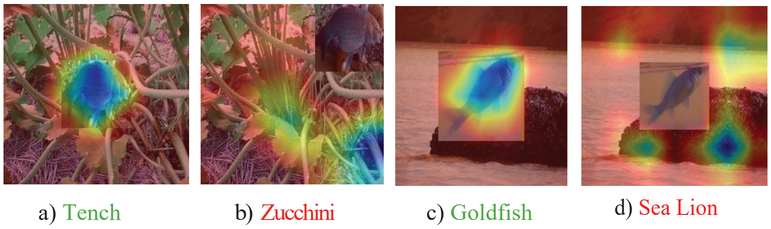
a) Tench b) Zucchini c) Goldfish d) Sea Lion

Figure 1: Gradcam visualizations for Pre-trained ConvNext-Base. a) Model predicts core class "Tench" when the object is located in the center of the image, b) Spurious class "Zucchini" is predicted when the "core" class moves away from the center, c) Class "GoldFish" is predicted when the size of the core object is large (112×112), d) Spurious class "Sea Lion" is predicted when size of core object reduces to 84×84.

In this work, we try to answer the questions: *In the absence of biases mentioned above, namely position and size of objects, how much do pre-trained models rely on spurious backgrounds to make their predictions, and are the current techniques that mitigate harmful spurious features, enough to tackle this problem?* Specifically, the contributions of our work are as follows:

- We calculate centeredness and size scores of different classes in ImageNet Deng et al. (2009), and analyze their relation with the level of spuriousity present in that class.
- We derive a dataset from ImageNet1k, called **Hard-Spurious-ImageNet**, containing objects against spurious backgrounds with varying sizes and positions. The code to generate the dataset will be provided.
- With the help of experimentation and ablation, we conclude that the size and location of the object should be taken into account when trying to mitigate harmful spurious correlations in the dataset.

2 Related Work

2.1 Spurious Features

Moayeri et al. (2022a) show that adversarial training increases model reliance on spurious features. They also show that increased spurious feature reliance occurs when the perturbations added to core features are too small to break spurious correlations. Murali et al. (2023) show that spurious features are related with a model's learning dynamics. Specifically, "easier" features learnt in the start of model training can hurt generalization. Neuhaus et al. (2023) proposed a method to identify spurious features in the ImageNet dataset and introduced a fix to mitigate a model's dependence on these features without requiring additional labels. While the proposed methods to mitigate spurious feature reliance are helpful in many cases, their efficacy is less known when factors such as size and location of core features in an image change.

2.2 Existing Datasets

Xiao et al. (2020) present an analysis of model's performance as a function of varying backgrounds and foregrounds for ImageNet. They conclude that more accurate models have less reliance on backgrounds.They also a propose a dataset called ImageNet-9 with mixed foregrounds and backgrounds. Moayeri et al. (2022b) propose a dataset derived from ImageNet with segmentation masks for a subset of images. These masks label entire objects and various visual attributes. They name this dataset RIVAL10 and also test different models' sensitivity to noise in backgrounds and foregrounds. Moayeri et al. (2022c) propose a dataset with segmentation masks for images in 15 classes of ImageNet1k. These images have high spurious features. They

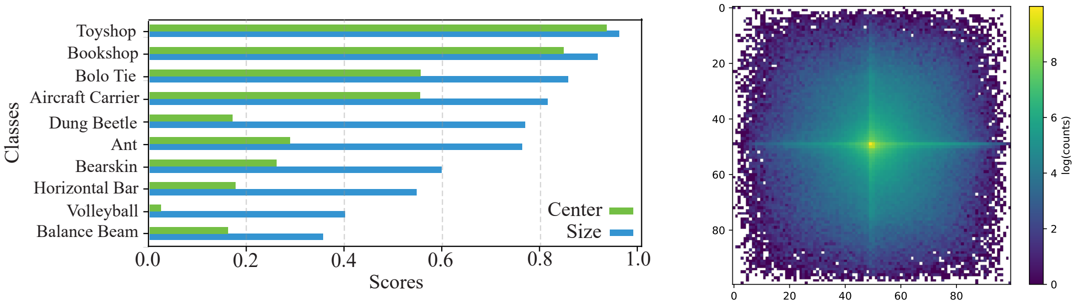

Figure 2: **Left**: ImageNet classes and their center and size scores. *Toyshop* has largest center and size scores, whereas *Volleyball* has smallest center score and *Balance Beam* has smallest size score. Other classes are sampled randomly for visualization. **Right**: Counts in log scale of relative centers of ground truth bounding boxes containing the object corresponding to the image class (ImageNet1k validation set). Most object centers are concentrated around the image center, while some are present along the main axes. Objects of interest are rarely present in image corners.

attribute this to objects being small and less centered in these images. Singla & Feizi (2021) label spurious and core features for ImageNet samples. They achieve this by making use of activation maps as soft masks. Moayeri et al. (2023) rank images in ImageNet dataset based on spurious cues present. They show that spurious feature reliance is influenced more by the data a model is trained on rather than how a model is trained. Lynch et al. (2023) propose a photo-realistic dataset with many-to-many spurious correlations between different groups of spurious attributes and classes. One work closely related to ours is that of Yung et al. (2021). They do a fine-grained analysis of the robustness of different models by varying factors such as object size, location, and rotation. Our technical contributions differ from theirs because we take into account the spuriousity level of backgrounds and correlate it with the above factors as well.

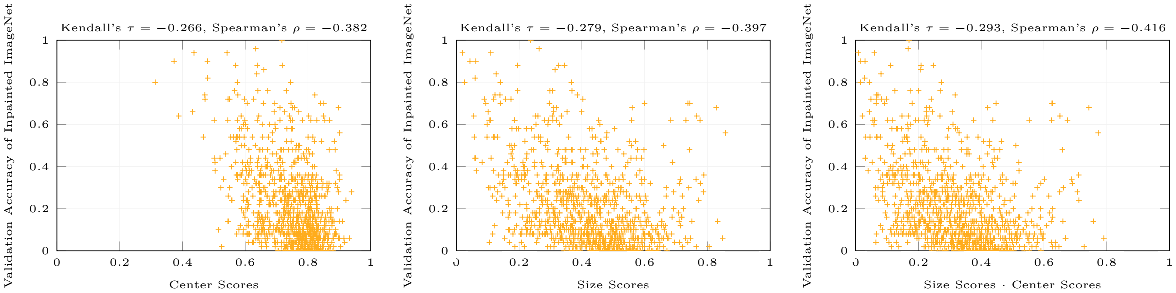

Figure 3: Correlation between the validation accuracy on inpainted ImageNet and, from left to right, center scores, size scores, and their product, respectively. Jointly considering center and size score shows strongest negative correlation with the accuracy.

2.3 Biases in Datasets

While capturing images through a camera, humans often tend to place the region of interest in the center. Due to this, there often exists a bias in classification datasets where objects are mostly located in the center of images and away from the boundary of the image. Exploiting the center bias in ImageNet, resizing and center cropping has been usually used for testing image classification models. Taesiri et al. (2024) show that there exists a strong center bias in out-of-distribution benchmarks such as ImageNet-A and ObjectNet by using resize and center crop operations only. They resize the image to multiple scales and patchify it, followed by a center crop operation at every patch. Doing this, they end up with different zoomed-in versions

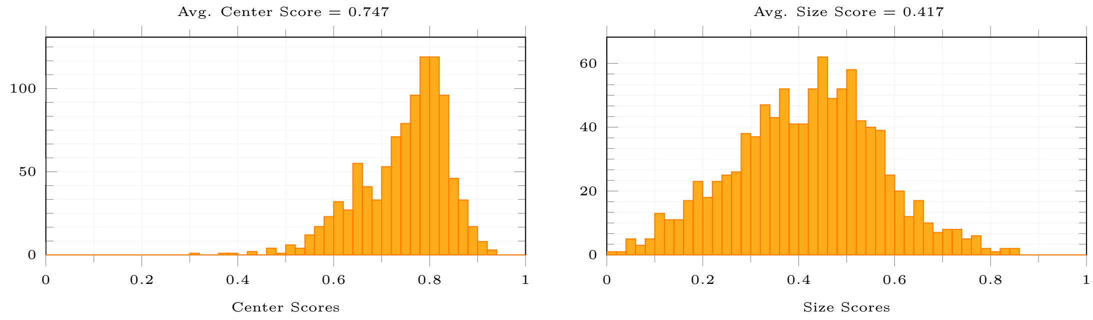

Figure 4: Histograms showing distribution of scores in different classes of ImageNet1k dataset.

of the input images. The computed accuracy of the center crop is maximum showing the presence of a strong center bias in the dataset. In this paper, we do an in-depth analysis of the presence of center and size bias in every class of ImageNet by computing distinct scores, The detailed explanation of these scores are given in following sections.

3 Biases in ImageNet

In this section, we quantitatively analyze positional and size biases present in ImageNet1k. To get a better sense of these biases, we propose *centeredness* and *size* scores.

3.1 Centeredness Score

In the majority of images in ImageNet1k, the objects of interest are located in the image's center (see Figure 1). Hence, in this paper, we use "positional" and "center" as synonyms. To understand the extent of center bias prevalent in ImageNet1k, we propose a *Center Score* defined as

$$C_c = \frac{1}{M}\frac{1}{N}\sum_{i=1}^{M}\sum_{j=1}^{N} 1 - (\|I_{i,c} - O_{i,j,c}\|_\infty), \tag{1}$$

where C_c is the centeredness score for class c, M is total number of images in the class, N is total number of objects within a frame, I is image center, and O is object center. The distance between image center and object center is calculated by the ℓ_∞ norm. It is subtracted from 1 to establish a direct relationship between the score and center bias prevalent in the class c.

3.2 Size Score

To measure the average sizes of objects within images, we define a size score as

$$S_c = \frac{1}{M}\frac{1}{N}\sum_{i=1}^{M}\sum_{j=1}^{N} \frac{h_j w_j}{H_i W_i}, \tag{2}$$

where S_c is the size score for class c, h and w refer to the height and width of object j in image i. H and W are the height and width of the image itself. Figure 2 (left) shows the center and size scores of different classes, with *Toyshop* having the maximum center and size scores. The histograms in Figure 4 show the distribution of center and size scores of all the classes in the ImageNet1k validation data. It can be seen that the majority of the classes in ImageNet1k are highly centered with objects of interest occupying half of the image pixels on average. These scores are calculated by using Ground Truth bounding boxes of ImageNet.

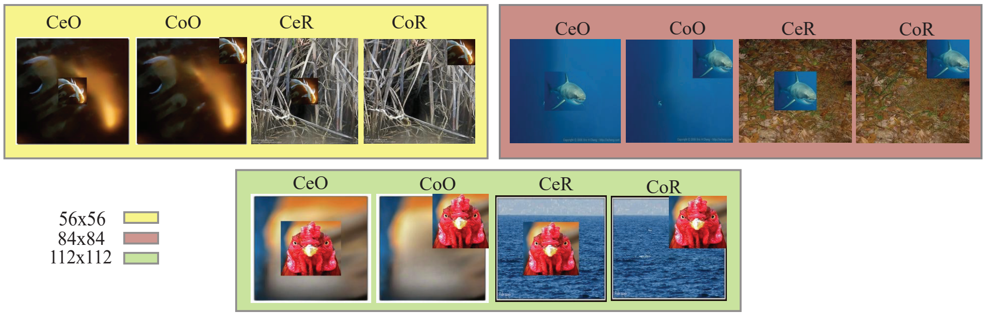

Figure 5: Different samples from Hard-Spurious-ImageNet. Image size remains same in all images i.e. 224 × 224, whereas object size changes. Label of every image is same as foreground object.

3.3 Relationship with the Level of Spuriosity

To establish a correlation between centeredness and size scores of every class to spurious feature reliance in ImageNet, we first calculate the validation accuracies of different classes in ImageNet with object information removed. We achieve this by using Inpaint-Anything Yu et al. (2023) with the goal of creating a more realistic effect when the region of interest is removed from the image. The input to Inpaint Anything are the object bounding boxes and it makes use of Segment Anything Kirillov et al. (2023) to predict masks for objects within these bounding boxes. These predicted masks are then input to the inpainting model LaMa Suvorov et al. (2021) which fills the masked region predicted by SAM. Finally, we resize the inpainted images to 224 × 224. We use ConvNext-Base Liu et al. (2022) pre-trained on ImageNet22k and fine-tuned on ImageNet1k, to compute the validation accuracies for the inpainted dataset. Classes with higher validation accuracies indicate higher spurious feature reliance, since the model has learnt to associate the class label not just with the core object, but also with the background information. In order to assess the correlation present between center and size scores and the level of spuriousity present in different classes of ImageNet, we use Kendall's τ coefficient and Spearman's correlation coefficient. The negative correlation values (see Figure 3) depict that there is an inverse relationship between both inpainted data's accuracy and the different considered scores, which validates the hypothesis that a higher spurious feature reliance is observed in case of non-centered large object sizes. The correlation is overall rather weak, which is to be expected since different classes are differently hard to classify, even from their core features.

4 Dataset

Similar to the waterbirds dataset Sagawa et al. (2019), we say that every datapoint (x, y) has an attribute $a(x) \in A$ which is spuriously correlated with label y. We conjecture that the strength of the correlation between attribute $a(x)$ and label y is controlled by two factors: size s and position p of the core features in the input image. To this end, we propose **Hard-Spurious-ImageNet**, a synthetic dataset to illustrate the problem of spurious feature reliance in the presence of varying object bounding box sizes, locations, and backgrounds. The prime motivation of creating the dataset is to have precise control over these factors and help the community build robust models against stronger spurious cues.

We consider the image content within the provided ground truth object bounding boxes for ImageNet as core features and the features outside the bounding box as the background. In ImageNet, bounding boxes are available for all images in the validation data, yet only a subset of images in training data are annotated. The images are annotated and verified through Amazon Mechanical Turk. We rely on these annotations to provide us an estimate of the location of core features in any image. As a first step, we want to disentangle core features from the rest of the image. We achieve this by cropping out the core objects from the images and inpainting the resulting image, as explained in the previous section. Next, we resize core object bounding boxes to different sizes, and place them in two different locations against inpainted backgrounds. The size

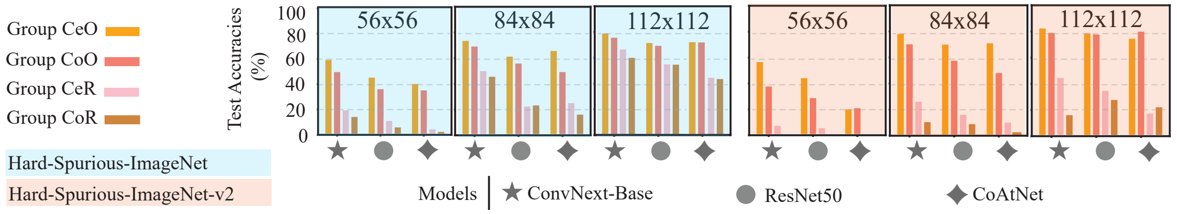

Figure 6: Benchmarking results of different models. Performance for our Hard-Spurious-ImageNet-v2 is the worst across all groups.

and location of core objects and the kind of background chosen, gives rise to different groups in the data. To efficiently gauge the performance of these different groups, we categorize them as follows:

- **Group CeO**: Core object in the ***Ce***nter of image against its ***O***riginal inpainted background.
- **Group CoO**: Core object in the top right ***Co***rner of image against its ***O***riginal inpainted background.
- **Group CeR**: Core object in the ***Ce***nter of image against ***R***andom inpainted background.
- **Group CoR**: Core object in the top right ***Co***rner of image against ***R***andom inpainted background.

We consider three core object sizes: 56 × 56, 84 × 84, and 112 × 112. It is important to note that all the inpainted backgrounds have already been resized to 224 × 224, so the core object sizes mentioned above represent $\frac{4}{64}$th, $\frac{9}{64}$th, and $\frac{16}{64}$th of the whole image. We also experimented with object masks obtained from the Segment Anything Kirillov et al. (2023) model rather than the provided bounding boxes as foreground objects (see Table 7 in supplementary). We observed that the mask quality for some objects was not good enough, hence, we used provided bounding boxes for this work.

4.1 Hard-Spurious-ImageNet-v2

Randomly chosen backgrounds have varying levels of spuriousity based on the classes they are taken from. We derive a variant of the proposed dataset where, instead of choosing backgrounds in a random fashion, they are chosen based on the level of spurious features present in them. To achieve this, we first analyze the level of spuriousity present in every class. We give inpainted images without the core objects, as input to the pretrained ConvNext-Base model , and record the accuracies of every class. The classes where accuracies are high indicate that the model has learnt to predict the class label without the presence of core objects. On the contrary, classes for which the accuracy is low are highly reliant on core features to make predictions. We choose 10 classes that are highly spurious, namely: *snorkel, bobsled, maypole, potter's wheel, gondola, bearskin, volleyball, basketball, canoe, geyser, and yellow lady's slipper* as backgrounds. For foreground objects, we choose 10 classes with high core features such as: *bluetick, box turtle, Chihuahua, Japanese spaniel, Maltese dog, Shih-Tzu, Blenheim spaniel, papillon, Rhodesian ridgeback, and basset.* We combine the above-mentioned foregrounds and backgrounds to create a dataset with 10 classes of foreground objects and highly spurious backgrounds. Similar to before, for every class, the chosen background class remains same for all images belonging to that class, but the backgrounds can differ from one image to another. Finally, we create four groups for the dataset as before and test on pre-trained models.

5 Experimental Results

We test the robustness of different models with the two proposed two variants of Hard-Spurious-ImageNet. The images are already resized to 224 × 224, so no additional resizing is applied to the images when giving as input to the pre-trained models. Images are normalized with mean and standard deviation of the ImageNet dataset. We use HuggingFace PyTorch models to test the dataset.

Model	Clean Accuracy
ConvNext-Base	85.86
ResNet50	80.20
CoAtNet	83.59

Table 1: Clean accuracies of standard ImageNet validation data with different pre-trained models.

Figure 6 shows test accuracies of the proposed data and its variant on three pretrained models. We consider ConvNext-Base trained on ImageNet22k and fine-tuned with ImageNet1k, ResNet50 He et al. (2016) and CoAtNet Dai et al. (2021) pretrained on ImageNet1k to test the performance of proposed dataset. ConvNext Base performs best across all groups and datasets. This can be attributed to the fact that the data augmentation pipeline of ConvNext-Base consists of rigorous steps, which ensures it stays robust to varying object sizes and locations. The difference in accuracy between groups CeR and CoR, when the core object size is 112×112 is less across all the models. This indicates that the core feature size is big enough for the model to ignore changes in location. Moreover, $\frac{1}{4}$th of the number of pixels in the image are occupied by core features in this case, so backgrounds are less exposed as compared to when the core object size is even less. Another interesting observation is that the impact of size change is far stronger on model performance than the location of core features. We also see that Hard-Spurious-ImageNet-v2 has far worse performance on groups CeR and CoR across all architectures and sizes. This indicates that the strength of spurious backgrounds is far greater than that of core features when the size of core features starts to decrease. We also observe that in almost all the groups, there is significant drop in performance compared with clean accuracies on standard validation dataset (see Figure 1).

Based on the above observations, we divide all the 12 groups consisting of different core feature sizes and locations into three distinct categories: **Easy**: This set consists of Groups CeO and CoO for larger core feature sizes, i.e. 84×84 and 112×112, as these groups seem to be doing considerably better than the rest. **Hard**: Groups CeR and CoR are the worst performing across all architecture for core feature sizes 56×56 and 84×84. We categorize them as **Hard** group. The remaining groups, i.e. groups CeO and CoO for size 56×56, and groups CeR and CoR for size 112×112 seem to be performing moderately, we put them in **Medium** category.

Following the analysis done earlier (see Figure 4), we find that most of the images in ImageNet are centered with an estimated size score of ≈ 0.5, indicating that on average, the core features in an image occupy half the number of pixels of the entire image. Keeping this in mind, we create the training data of Hard-Spurious-ImageNet consisting of majority and minority groups, where the number of images belonging to majority groups are far more than in minority groups. This is done to replicate the long-tailed distribution nature of the ImageNet dataset in terms of hardness. For the training data, we consider 80 images per group in the Easy category and 10 images from groups in Medium and Hard categories. This brings the total to 400 images per class in the training data. Out of the 400 images, 320 images belong to the Easy group and 80 to the Medium and Hard groups. For the validation set, we use a balanced dataset having equal data points from every group. We use 20 images per group, resulting in 240 images per class. Both training and validation set of Hard-Spurious-ImageNet are derived from training data of ImageNet, whereas the test set is derived from the validation data. The test set is also balanced, comprising 50 images per group, totaling 600 images in every class.

5.1 Effects of Data Augmentation and Self-Supervised Models

To measure the effect of data augmentations, we compared vanilla ResNet-50 trained without any augmentations on ImageNet1K with an advanced training recipe involving auto-augment, random erase, mixup, and cutmix. The results (shown in Table 2) indicate that while data augmentation increases accuracy across groups CeO, CoO, and CeR, the performance decreases in case of group CoR for all sizes. This indicates that standard data augmentation approaches do not take into account the presence of spurious features in the data while augmenting, hence, may end up highlighting them instead. Moreover, the gap in performance still persists across all four groups for a given core object size. This hints that mere data augmentation strategies are insufficient to deal with this problem. In the supplementary materials provided (see Table 5 and Table 6), we test the model on Hiera-Base with Masked Autoencoder which has been trained in a self-supervised manner. The results follow a similar trend across groups as other methods shown in the paper, although the Group CoR for size shows the worst performance when compared with all the other architectures. Moreover, we also

Model	**Clean Accuracy**	**Object Resolution**	**Group Accuracies**			
			CeO	**CoO**	**CeR**	**CoR**
ResNet-50 (Baseline)	76.13	56^2	38.62	32.53	8.71	7.03
		84^2	56.46	52.47	28.44	27.37
		112^2	65.87	64.16	46.58	46.57
ResNet-50 (Data Augmentations)	80.33	56^2	49.14	38.47	13.19	4.49
		84^2	65.74	58.48	33.30	20.01
		112^2	72.93	68.19	45.12	36.40
ViT Base	81.92	56^2	44.78	39.37	7.15	5.57
		84^2	63.65	58.83	28.56	25.46
		112^2	71.31	70.46	46.43	47.93

Table 2: The first two rows show the impact of data augmentation on the proposed dataset. Performance across group CoR becomes worse, indicating that just augmenting the data might not be enough to deal with spurious correlations. The third row shows the performance on ViT-Base pre-trained using CLIP and fine-tuned on IN-1K, highlighting similar trends observed earlier.

Methods	**Easy**	**Medium**	**Hard**	**Average**
Pretrained	65.39	48.50	16.54	43.48
ERM	74.84	66.67	57.56	65.94
JTT	60.90	53.09	46.49	53.50
DFR	72.47	65.65	59.79	65.97

Table 3: Test Performance of different methods on Easy, Medium, and Hard categories in Hard-Spurious-ImageNet. Average accuracy is the average test performance of all the groups combined.

computed the performance of different groups in the proposed dataset on a ViT pretrained on WIT-400M image-text pairs by OpenAI using CLIP and fine-tuned on ImageNet1k. The results are given in Table 2 and show similar trends as reported earlier.

5.2 Group Robustness Methods

We measure the performance of the proposed dataset using simple fine-tuning and two state-of-the-art group robustness methods. Empirical Risk Minimization or **ERM** Vapnik (1991) is conventional training to optimize average training accuracy without specialized methods for optimizing worst-group accuracy. Deep Feature Reweighting or **DFR** Kirichenko et al. (2022) tackles the problem of spurious correlations by retraining the last layer of a pre-trained model with equal data points from different groups present in the training data. Just Train Twice or **JTT** Liu et al. (2021) upsamples the training images which were wrongly predicted by the ERM trained model by a certain factor λ_{up}, and trains the classifier again. We experiment with different variations of the above methods.

Size	**CeO**	**CoO**	**CeR**	**CoR**
56^2	62.25	60.8	54.56	54.45
84^2	73.35	72.60	69.76	69.96
112^2	77.19	77.13	75.34	75.48

Table 4: Breakdown of test accuracies with ERM^{all} model. The network architecture is ResNet50.

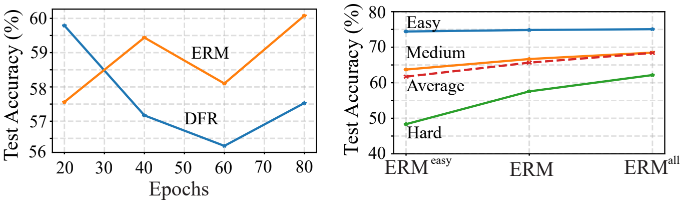

Figure 7: The effect of training epochs of ERM model on the performance of DFR. ERM model trained with 20 epochs gives the highest performance for DFR. (**right**) ERM_{all} narrows the gap between easy, medium, and hard groups.

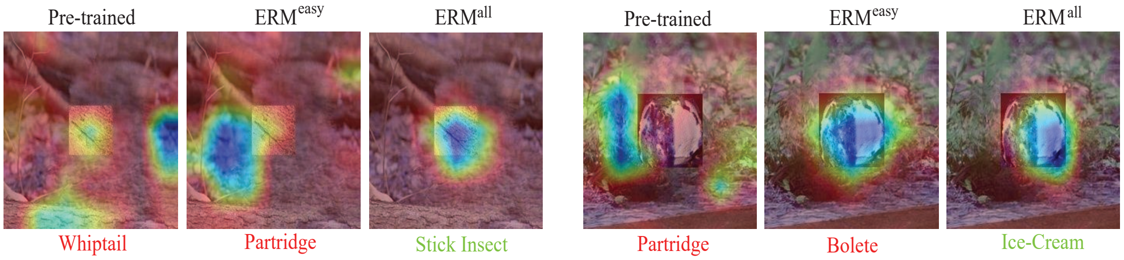

Figure 8: Gradcam visualizations showing regions of the image the model pays attention to in order to make the classification decision. Labels in red show false predictions and labels in green indicate correct prediction.

5.3 Implementation Details

We use pretrained Resnet50 trained on ImageNet1k for our experiments. The Base model is fine-tuned with batch size 256, constant learning rate of 0.001 for 20 epochs. The input images are randomly cropped with an aspect ratio in the bounds (0.75,1.33) and finally resized to 224 × 224. Horizontal flipping is applied afterward. A momentum of 0.9 and weight decay of 0.001 is used. For DFR, we normalize the embeddings using mean and standard deviation of validation data used to train the last layer, and use the same statistics to normalize embeddings of test data. We re-train the last layer for 1000 epochs, learning rate of 1, cosine learning rate scheduler and SGD optimizer with full-batch. We use ℓ_2 regularization with λ set to 100. These hyperparamters are similar to the ones set by Kirichenko et al. (2022) for optimizing the last layer for ImageNet-9 dataset Xiao et al. (2020). Since, the data distribution in the proposed dataset and ImageNet-9 is similar, we assumed the same hyperparameters. In case of JTT, models have the same hyperparameters as the ERM trained model. λ_{up} is set to 50.

5.4 Results

The results in Table 3 show that pretrained ImageNet models perform worst on the hard group. This could be attributed to the fact that the model has very little exposure to small core features against spurious backgrounds in the training data. The ERM model does better across easy, medium and hard groups, but there still exists a disparity in performance among the three groups. DFR is able to perform slightly better in the Hard group by sacrificing some accuracy in Easy and Medium groups. The average test accuracy is similar for ERM and DFR. The performance with JTT also decreases, which hints that the task of learning data has become difficult for the model in the presence of upsampled images. Since the embeddings in DFR are dependent on the ERM-trained model, we also analyze how the number of training epochs the ERM model is trained for, impacts the DFR performance. The epochs for retraining the last layer remain fixed to 1000, all other hyperparameters also remain the same for DFR models trained with different ERM-trained embeddings. The left plot in Figure 7 indicates that, when the base model is fine-tuned for 20 epochs, the performance of DFR on the test set increases. As the training time increases for ERM, performance by DFR decreases, whereas the ERM model continues to improve.

In case of ERM, we also analyze the effect of the percentage of training data in minority groups i.e. easy and hard groups on model's test performance. We refer to ERM_{easy} as the model that has been fine-tuned with data from the majority group only i.e. 0% of data from medium and hard group. Conversely, we refer to ERM_{all} as the model that has been fine-tuned with equal data points from all the groups, and ERM as the standard training data consisting of 20% of data from minority groups. The results are depicted in the right plot in Figure 7. We see that training with the Easy group has worst performance on the Hard group. ERM_{all} seems to narrow the gap between all groups. The accuracy of the Easy group remains similar across the three models.

Table 4 shows the breakdown of accuracies for all the sub-groups for ERM_{all} model. As compared to accuracies shown in Figure 6, there is a considerable improvement in case of CoR and CeR for size 56 × 56 and 84 × 84. The closest to clean accuracy for ResNet50 is observed in case of size 112 × 112 and group CeO.

5.5 Analysing Classifications with Saliency Maps

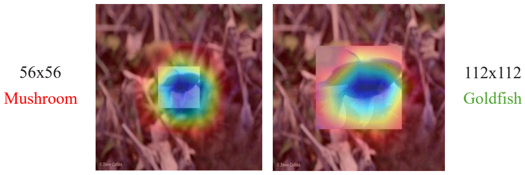

Figure 9: Effect of core feature size on model performance. Both the predictions are for the ERM^{all} model.

We use Gradcam to visualize the predictions on the Resnet50 model. Figure 8 shows the visualizations on the ImageNet pretrained model and two variations of ERM: ERM^{all} which is fine-tuned with equal data points from all the groups and ERM^{easy} which is fine-tuned only with images from the Easy category, consisting of subgroups CeO and CoO for size 54×54 and 112×112 respectively. The images on the left side of Figure 8 show a stick insect of size 56×56 placed in the center against an outdoor environment. The pre-trained and ERM^{easy} model make their predictions by picking up cues from the backgrounds and predicting class *Whiptail* and *Partridge* respectively. Upon inspection, we find that most of the images in these classes are set in similar environments, hence the model has learnt to associate the given outdoor environment with these classes and are ignoring the core features. ERM^{all}, however, is more robust to changes in environment and makes the correct prediction of class *Stick Insect*. The images on the right show that, while the pre-trained model is confused by the spurious cues in the background, ERM^{easy} makes the wrong predictions based on the cues in the core features and the background together. However, ERM^{all} makes the correct prediction by mostly relying on core features. Figure 9 highlights the effect of the size of core features on the ability of the ERM^{all} model to make correct predictions. Having a smaller core feature size results in the model making incorrect prediction of class *Goldfish*.

6 Challenges and Future Work

The dataset variants of Hard-Spurious-ImageNet are proposed to understand the extent of background reliance as a function of size and location of core features. One of the limitations of the datasets is that they rely on ground truth bounding boxes of objects. In case of images where core features are not labeled by bounding boxes, no inpainting is performed on them, subsequently leading to core features in background and foreground occurring simultaneously. Moreover, the presence of secondary objects and clutter in the background makes it difficult for the models to learn small core feature sizes. The lack of segmentation bounding boxes for all images in ImageNet restricted us to using object bounding boxes instead of masks. Currently, we have only experimented with one location per core object. For future work, we plan to experiment with different locations of core objects in the images and analyze the impact of using different network architectures with the dataset. Moreover, it would be interesting to extend this analysis to other datasets and models trained in different ways such as with contrastive learning, and various data augmentation techniques.

7 Conclusion

In this paper, we propose a variant of ImageNet, Hard-Spurious-ImageNet, to help the deep learning community to better understand spurious feature reliance. We show that ImageNet is center-biased and exhibits a bias towards large object sizes. We also provide an analysis showing that there exists a negative correlation between size and location of core features in an image and the strength of spurious cues in the background. We experiment with different group robustness methods and highlight the need for specialized methods to solve this problem.

References

Valerio Biscione and Jeffrey Bowers. Learning translation invariance in cnns. *arXiv preprint arXiv:2011.11757*, 2020.

Valerio Biscione and Jeffrey S Bowers. Convolutional neural networks are not invariant to translation, but they can learn to be. *Journal of Machine Learning Research*, 22(229):1–28, 2021.

Zihang Dai, Hanxiao Liu, Quoc V Le, and Mingxing Tan. Coatnet: Marrying convolution and attention for all data sizes. *Advances in neural information processing systems*, 34:3965–3977, 2021.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4804–4814, 2022.

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

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pp. 38–55. Springer, 2024.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11976–11986, 2022.

Aengus Lynch, Gbètondji JS Dovonon, Jean Kaddour, and Ricardo Silva. Spawrious: A benchmark for fine control of spurious correlation biases. *arXiv preprint arXiv:2303.05470*, 2023.

Mazda Moayeri, Kiarash Banihashem, and Soheil Feizi. Explicit tradeoffs between adversarial and natural distributional robustness. *Advances in Neural Information Processing Systems*, 35:38761–38774, 2022a.

Mazda Moayeri, Phillip Pope, Yogesh Balaji, and Soheil Feizi. A comprehensive study of image classification model sensitivity to foregrounds, backgrounds, and visual attributes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19087–19097, 2022b.

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

Mazda Moayeri, Wenxiao Wang, Sahil Singla, and Soheil Feizi. Spuriosity rankings: sorting data to measure and mitigate biases. *Advances in Neural Information Processing Systems*, 36:41572–41600, 2023.

Nihal Murali, Aahlad Puli, Ke Yu, Rajesh Ranganath, and Kayhan Batmanghelich. Beyond distribution shift: Spurious features through the lens of training dynamics. *Transactions on machine learning research*, 2023, 2023.

Yannic Neuhaus, Maximilian Augustin, Valentyn Boreiko, and Matthias Hein. Spurious features everywhere-large-scale detection of harmful spurious features in imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20235–20246, 2023.

Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, et al. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *International Conference on Machine Learning*, pp. 29441–29454. PMLR, 2023.

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

Sahil Singla and Soheil Feizi. Salient imagenet: How to discover spurious features in deep learning? *arXiv preprint arXiv:2110.04301*, 2021.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. *arXiv preprint arXiv:2109.07161*, 2021.

Mohammad Reza Taesiri, Giang Nguyen, Sarra Habchi, Cor-Paul Bezemer, and Anh Nguyen. Imagenet-hard: The hardest images remaining from a study of the power of zoom and spatial biases in image classification. *Advances in Neural Information Processing Systems*, 36, 2024.

Vladimir Vapnik. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991.

Kai Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. *arXiv preprint arXiv:2006.09994*, 2020.

Tao Yu, Runseng Feng, Ruoyu Feng, Jinming Liu, Xin Jin, Wenjun Zeng, and Zhibo Chen. Inpaint anything: Segment anything meets image inpainting. *arXiv preprint arXiv:2304.06790*, 2023.

Jessica Yung, Rob Romijnders, Alexander Kolesnikov, Lucas Beyer, Josip Djolonga, Neil Houlsby, Sylvain Gelly, Mario Lucic, and Xiaohua Zhai. Si-score: An image dataset for fine-grained analysis of robustness to object location, rotation and size. *arXiv preprint arXiv:2104.04191*, 2021.

A Benchmark Results

The results for Hard-Spurios-ImageNet and its variant are given in Tables 5 and 6 respectively. We test the performance of the datasets on 5 different pre-trained architectures: ConvNext-Base Liu et al. (2022) trained on ImageNet21k and fine-tuned on ImageNet1k, ResNet-50 He et al. (2016), CoATNet Dai et al. (2021), Hiera-Base with MAE Ryali et al. (2023), and MVit2-small Li et al. (2022). Except for ConvNext, these models are pretrained on ImageNet1k only. Across all models, the performance on Group CoR for size 56×56 is the worst. Benchmark results for different groups along with clean accuracies are given in Tables 5 and 6. Clean accuracies in Table 6 are for 10 Hard-Spurious-ImageNet-v2 classes only.

Model	**Clean Accuracy**	**Object Resolution**	**Group Accuracies**			
			CeO	**CoO**	**CeR**	**CoR**
Convnext-Base	85.86	56^2	59.62	50.14	20.15	14.98
		84^2	74.42	70.05	50.81	46.47
		112^2	79.82	76.54	67.43	60.92
ResNet-50	80.20	56^2	45.79	36.74	11.83	6.78
		84^2	62.19	56.80	23.37	24.19
		112^2	72.43	70.22	55.85	55.62
CoATNet	83.59	56^2	40.79	35.78	5.27	3.28
		84^2	66.58	50.14	25.92	17.02
		112^2	72.70	72.51	45.45	44.44
Hiera	84.48	56^2	49.45	34.34	4.61	1.31
		84^2	67.64	55.09	21.81	12.49
		112^2	74.07	69.32	47.26	37.82
MVitv2	83.77	56^2	41.44	31.38	5.41	1.38
		84^2	67.38	51.12	29.53	14.17
		112^2	70.51	64.86	48.00	37.81

Table 5: Test Accuracies on Hard-Spurious-ImageNet.

B Biases in ImageNet

Figure 11 shows the distribution of center and size scores for different classes in the training data of ImageNet. We calculate these scores using the available bounding boxes for ImageNet training data. Figure 4 refers to the distribution for the validation data.

C Inpaint Anything

The predicted masks from Segment Anything are dilated by a kernel size of 15 to avoid edge effects when the "hole" is filled by LaMa. Some examples of the inpainted data are given in Figure 10.

D True Objects in Background

Ensuring that the backgrounds do not contain true objects depends on the fidelity of provided ImageNet annotations. We perform an additional analysis with a foundation model, Grounding DINO Liu et al. (2024), to extract bounding boxes from the images. We consider similarity scores between Grounding DINO predictions and the ImageNet annotations to analyze the correctness of ImageNet annotations. For ImageNet validation data, we get an overall mIOU of 0.8675 across all classes between both sets of bounding boxes with 139 classes having mIOU value less than 0.8 (see Figure 11 for a histogram by mIOU). This shows that

Model	Clean Accuracy	Object Resolution	Group Accuracies CeO	CoO	CeR	CoR
Convnext-Base	85.8	56^2	57.4	38.6	8.2	1.0
		84^2	79.0	71.0	27.0	11.4
		112^2	83.2	79.8	45.4	17.0
ResNet-50	82.2	56^2	45.00	29.6	6.4	0.0
		84^2	70.8	58.4	17.0	9.8
		112^2	79.4	78.6	35.6	28.6
CoATNet	83.4	56^2	20.9	21.8	0.8	0.0
		84^2	71.8	49.0	11.0	4.0
		112^2	75.40	80.8	18.2	23.0
Hiera	85.8	56^2	46.8	22.8	1.0	0.0
		84^2	75.6	55.6	4.0	1.8
		112^2	78.6	74.6	16.0	10.6
MVitv2	86.6	56^2	29.0	15.0	0.4	0.0
		84^2	72.6	47.8	11.2	1.2
		112^2	72.4	66.0	18.4	12.6

Table 6: Test Accuracies on Hard-Spurious-ImageNet-v2 with highly spurious backgrounds.

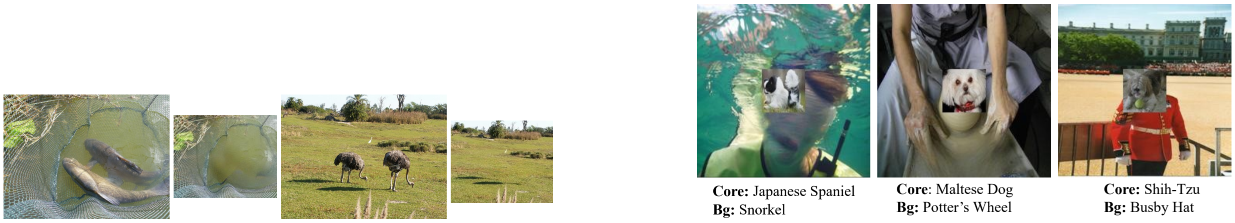

Figure 10: **left**: Original images with their resized inpainted versions. **right**: Despite inpainting, the background (Bg) consists of cues that help the model predict the background label.

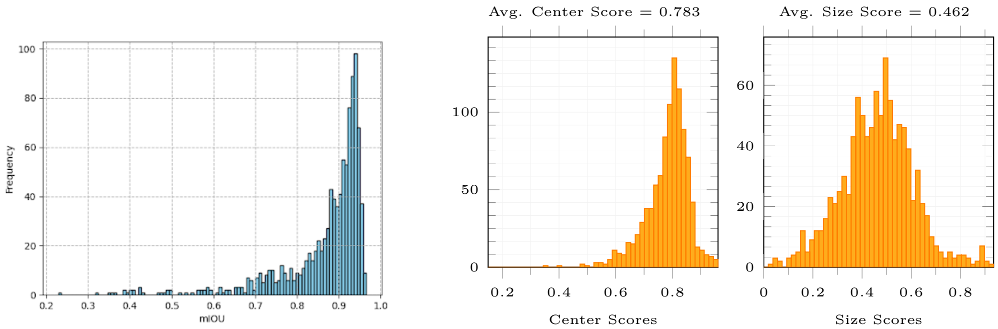

Figure 11: **(left)**: Class-wise mIOU scores between Grounding DINO predictions and ImageNet annotations on the validation set. Averaged mIOU is 0.875. **(right)**: Histograms showing distribution of scores in different classes of train data in ImageNet1k dataset

Model	Clean Accuracy	Object Resolution	Group Accuracies			
			CeO	CoO	CeR	CoR
Convnext-Base	85.8	56^2	46.07	36.07	13.86	6.21
		84^2	61.18	53.92	31.04	22.30
		112^2	67.78	64.69	42.91	13.84
ResNet-50	82.2	56^2	29.33	24.34	6.68	4.36
		84^2	45.17	40.63	19.09	16.24
		112^2	55.24	52.56	31.34	29.87
CoATNet	83.4	56^2	30.57	27.61	7.91	3.93
		84^2	50.94	44.66	21.03	15.63
		112^2	60.60	56.73	33.00	29.30
MVit2	85.8	56^2	37.94	25.88	9.08	2.92
		84^2	54.89	44.73	24.74	15.15
		112^2	63.73	57.94	36.80	30.60
Hiera	86.6	56^2	39.88	27.06	10.34	3.198
		84^2	56.36	46.18	25.13	15.72
		112^2	66.14	60.38	39.15	31.63

Table 7: Test Accuracies on Hard-Spurious-ImageNet with SAM Masks.

the majority of the classes in ImageNet data have correct bounding boxes and the amount of objects from the foreground class in the background is negligible.

E Hard-Spurious-ImageNet with SAM

We also experiment with using the Segment Anything Kirillov et al. (2023) model to obtain masks for the objects inside a bounding box and resize it to 3 different sizes (56, 84, and 112). The resized masks are then placed in the center and corner of the inpainted image, similar to the setting described in the main paper. At the moment, we only consider one object per image. Since we have access to ImageNet-annotated bounding boxes, we use them as prompts to be given to SAM. The results are shown in Table 7. Compared to the results in Table 5, the results with SAM are worse, mainly because the resized SAM object masks are not entirely accurate in cases where objects are small and thin, such as insects, etc. Hence, we preferred human-annotated ImageNet bounding boxes.

Methods	**Easy**	**Medium**	**Hard**	**Average**
Pretrained	71.14	54.93	29.21	51.75
ERM	76.91	70.63	63.48	70.34
ERM$^{\text{easy}}$	77.82	68.33	51.39	65.85
DFR	74.82	68.66	61.68	68.39

Table 8: Test Performance of different methods on Easy, Medium, and Hard categories in Hard-Spurious-ImageNet. Average accuracy is the average test performance of all the groups combined. The model is Convnext-tiny.

F Group Robustness Methods

We use pretrained ResNet-50 trained on ImageNet1k for our experiments. The Base model is fine-tuned with batch size 256, constant learning rate of 0.001 for 20 epochs. The input images are randomly cropped with an aspect ratio in the bounds (0.75,1.33) and finally resized to 224 × 224. Horizontal flipping is applied afterward. A momentum of 0.9 and weight decay of 0.001 is used. For DFR, we normalize the embeddings using mean and standard deviation of validation data used to train the last layer, and use the same statistics to normalize embeddings of test data. We re-train the last layer for 1000 epochs, learning rate of 1, cosine learning rate scheduler and SGD optimizer with full-batch. We use ℓ_2 regularization with λ set to 100. These hyperparamters are similar to the ones set by Kirichenko et al. (2022) for optimizing the last layer for ImageNet-9 dataset Xiao et al. (2020). Since, the data distribution in the proposed dataset and ImageNet-9 is similar, we assumed the same hyperparamteres. In case of JTT, models have the same hyperparameters as the ERM trained model. λ_{up} is set to 50.

After extracting the embeddings from the pre-trained ERM model, the embeddings are normalized using **fit_transform()** and **transform()** functions of **sklearn.preprocessing.StandardScaler** for val and test data, respectively. For the JTT model, the images are applied with random resized cropping followed by horizontal flipping. No additional data augmentation is applied afterward. We also experimented with ConvNext-tiny pre-trained on ImageNet-22k and fine-tuned on ImageNet1k. We fine-tune the pre-trained model on the proposed data under various settings. ERM is trained by replicating the long-tailed distribution of the data, while ERM$^{\text{easy}}$ is trained only with the easy group. ERM$^{\text{all}}$ is trained with equal data points from all groups. DFR is trained by extracting embeddings from ERM, and re-training the last layer only. The number of train and test images is similar to the data setting described in the main paper.