A DATASETS DETAILS

We choose the following 9 high-level classes.

Class	WordNet ID	Number of sub-classes
Dog	n02084071	116
Bird	n01503061	52
Vehicle	n04576211	42
Reptile	n01661091	36
Carnivore	n02075296	35
Insect	n02159955	27
Instrument	n03800933	26
Primate	n02469914	20
Fish	n02512053	16

Table 4: The 9 classes of ImageNet-9.

All datasets used in the paper are balanced by randomly removing images from classes that are over-represented. We only keep as many images as the smallest post-modification synthetic dataset, so all synthetic datasets (except IN-9L) have the same number of images. We also use a custom GUI to manually process the test set to improve data quality. For IN-9L, the only difference from using the corresponding classes in the original ImageNet dataset is that we balance the dataset.

For all images: we apply the following filters before adding each image to our datasets.

- The image must have bounding box annotations.
- For simplicity, each image must have exactly one bounding box. A large majority of images that have bounding box annotations satisfy this.

For images needing a properly segmented foreground: This includes the 3 MIXED datasets, ONLY-FG, and NO-FG. We filter out images based on the following criteria.

- Because images are cropped before they are fed into models, we require that less than 50% of the bounding box is removed by the crop, to ensure that the foreground still exists. Almost all images pass this filter.
- The OpenCV foreground segmentation function `cv2.grabCut` (used to extract the foreground shape) must work on the image. We remove images where it fails.
- For the test set only, we manually remove images with foreground segmentations that retain a significant portion of the background signal.
- For the test set only, we manually remove foreground segmentations that are very bad (e.g. the segmentation selects part of the image, and that part doesn't contain the foreground object).

For images needing only background signal: This includes ONLY-BG-B and ONLY-BG-T. In this case, we apply the following criteria:

- The bounding box must not be too big (more than 90% of the image). The intent here is to avoid ONLY-BG-B images being just a large black rectangle.
- For the test set only, we manually remove ONLY-BG images that still have an instance of the class even after removing the bounding box. This occurs when the bounding boxes are imperfect or incomplete (e.g. only one of two dogs in an image is labeled with a bounding box).

Creating the ONLY-BG-T dataset: We first make a "tiled" version of the background by finding the largest rectangular strip (horizontal or vertical) outside the bounding box, and tiling the entire

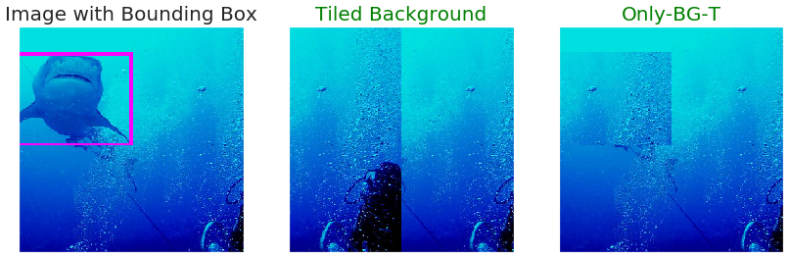

Figure 9: Visualization of how ONLY-BG-T is created.

image with that strip. We then replace the removed foreground with the tiled background. A visual example is provided in Figure 9. We purposefully choose not to use deep-learning-based inpainting techniques such as (Shetty et al., 2018) to replace the removed foreground, as such methods could lead to biases that the inpainting model has learned from the data. For example, an inpainting model may learn that the best way to inpaint a missing chunk of a flower is to place an insect there, which is something we want to avoid.

Motivation for each IN-9 variation: We create ONLY-BG-B and ONLY-BG-T to remove the foreground completely, including the shape of the foreground object. We intend for ONLY-BG-B to be directly comparable to the prior work of Zhu et al. (2017) that uses similar methodology to evaluate older AlexNet models, while ONLY-BG-T is a more natural-looking background that avoids black rectangles introduced in ONLY-BG-B.

The NO-FG dataset is created to retain the foreground shape, but not the texture. We can use it to assess the relative importance of foreground shape compared to foreground texture.

Finally, we create four datasets that have identical foregrounds but each have distinct background signals. ONLY-FG has a pure black background to go with the foreground. MIXED-SAME has background signal from the same class as the foreground. MIXED-RAND has background signal from a random class, so it can be thought of as having neutral background signal. MIXED-NEXT has background signal from the next class, which will always be in conflict with the foreground . Any artifacts in the foreground that result from our image processing pipeline are equally present in all four datasets. Thus, these datasets help to isolate how much backgrounds alone influence model predictions *when the correct foreground exists in the image*.

Full-ImageNet version of each synthetic variation: We also apply the same methodology for disentangling foreground and background signal to the entire ImageNet validation set, creating Full-ImageNet (Full-IN) versions of each of our 7 dataset variations.

We evaluate a pre-trained ResNet-50 on Full-IN for comparison in Table 5, and observe similar trends to ImageNet-9 that lead to similar conclusions on model background reliance. We choose to focus on ImageNet-9 results in the main paper because of the following shortcomings of Full-IN.

1. Individual classes are quite small, as some classes have very few (or even zero) images that make it through our filters due to lack of proper annotated bounding boxes.
2. When bounding boxes do exist, their quality is often lower than those in the IN-9 classes. For example, many images of fruit contain multiple fruit, but only one will be properly annotated with a bounding box.
3. When creating the MIXED-NEXT equivalent for Full-IN, the next class is often similar to the previous one. For example, many dog breeds occur consecutively in ImageNet. Thus, Full-IN's MIXED-NEXT frequently has backgrounds that are similar to backgrounds from the foreground class.

B EXPLAINING THE DECREASED BG-GAP OF PRE-TRAINED IMAGENET MODELS

We investigate two possible explanations for why pre-trained ImageNet models have a smaller BG-GAP than models trained on ImageNet-9. Understanding this phenomenon can help inform how models should be trained to be more background-robust. We find slight improvements to background-robustness from training on more fine-grained classes. We find that training on larger datasets helps only slightly when the training dataset set size is smaller than IN-9L, but larger improvements occur when the training dataset size is bigger. Thus, we encourage training on larger datasets if reduced background robustness is the goal.

B.1 THE EFFECT OF FINE-GRAINEDNESS ON THE BG-GAP

One possible explanation is that training models to distinguish between finer-grained classes forces them to focus more on the foreground, which contains relevant features for making those fine-grained distinctions, than the background, which may be fairly similar across sub-classes of a high-level class. This suggests that asking models to solve more fine-grained tasks could improve model robustness to background changes.

To test the effect of fine-grainedness on ImageNet-9, we make a related dataset called IN-9LB that uses the same 9 high-level classes and can be cleanly modified into more fine-grained versions. Specifically, for IN-9LB we choose exactly 16 sub-classes for each high-level class, for a total of 144 ImageNet classes. To create successively more fine-grained versions of the IN-9LB dataset, we group every n sub-classes together into a higher-level class, for $n \in \{1, 2, 4, 8, 16\}$. Here, $n = 1$ corresponds to keeping all 144 ImageNet classes as they are, while $n = 16$ corresponds to only having 9 high-level classes, like ImageNet-9. Because we keep all images from those original ImageNet classes, this dataset is the same size as IN-9L.

We train models on IN-9LB at different levels of fine-grainedness and evaluate the BG-GAP of those models in Figure 10. We find that fine-grained models have a smaller BG-GAP as well as better performance on MIXED-NEXT, but the improvement is very slight and also comes at the cost of decreased accuracy on ORIGINAL. The BG-GAP of the most fine-grained classifier is 2.3% smaller than the BG-GAP of the most coarse-grained classifier, showing that fine-grainedness does improve background-robustness. However, the improvement is still small compared to the size of the BG-GAP (which is 13.3% for the fine-grained classifier).

B.2 THE EFFECT OF LARGER DATASET SIZE ON THE BG-GAP

A second possible explanation for why pre-trained ImageNet models have a smaller BG-GAP is that training on larger datasets is important for background-robustness. To evaluate this possibility, we train models on different-sized subsets of IN-9LB. The largest dataset we train on is the full IN-9LB dataset, which is 4 times as large as IN-9, and the smallest is 1/4 as large as IN-9. Figure 11 shows that increasing the dataset size does increase overall performance but only slightly decreases the BG-GAP.

Next, we train models on different-sized subsets of ImageNet; we use the pre-trained ResNet-50 ImageNet model for full-sized ImageNet, and we train new ResNet-50 models on subsets that are 1/2, 1/4, 1/8, 1/16, and 1/32 as large as ImageNet. In these cases, we observe in Figure 12 that training on more data does not help significantly when the training dataset sizes are still small, but it does help more noticeably for models trained on 1/2 of ImageNet and all of ImageNet.

It is possible that having both a fine-grained class structure and more training data simultaneously is important for background-robustness. Furthermore, more training data (from other classes that are not in IN-9L) may also be the cause of the increased background-robustness of pre-trained ImageNet models.

B.3 SUMMARY OF METHODS INVESTIGATED TO REDUCE THE BG-GAP

In Figure 13, we compare the BG-GAP of ResNet-50 models trained on different datasets and with different methods to a ResNet-50 pre-trained on ImageNet. We explore ℓ_p-robust training, increasing

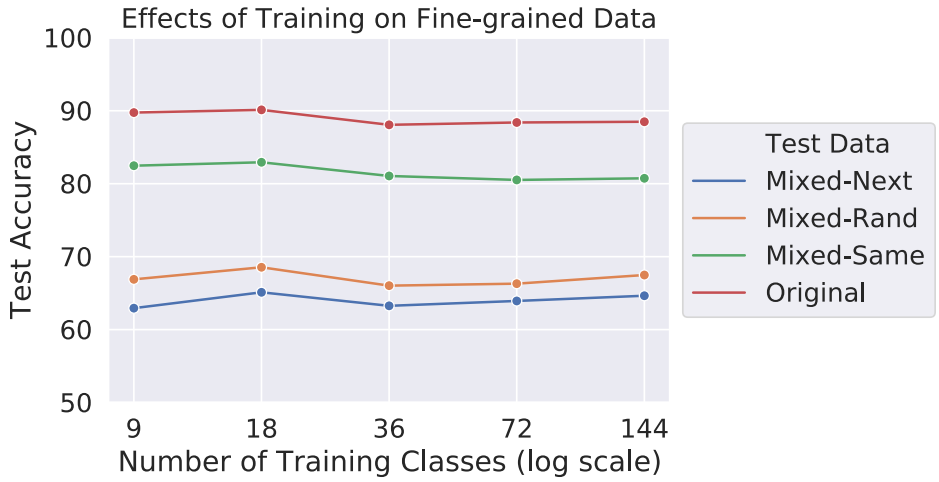

Figure 10: We train models on IN-9LB at different levels of fine-grainedness (more training classes is more fine-grained). The BG-GAP, or the difference between the test accuracies on MIXED-SAME and MIXED-RAND, decreases as we make the classification task more fine-grained, but the decrease is small compared to the size of the BG-GAP.

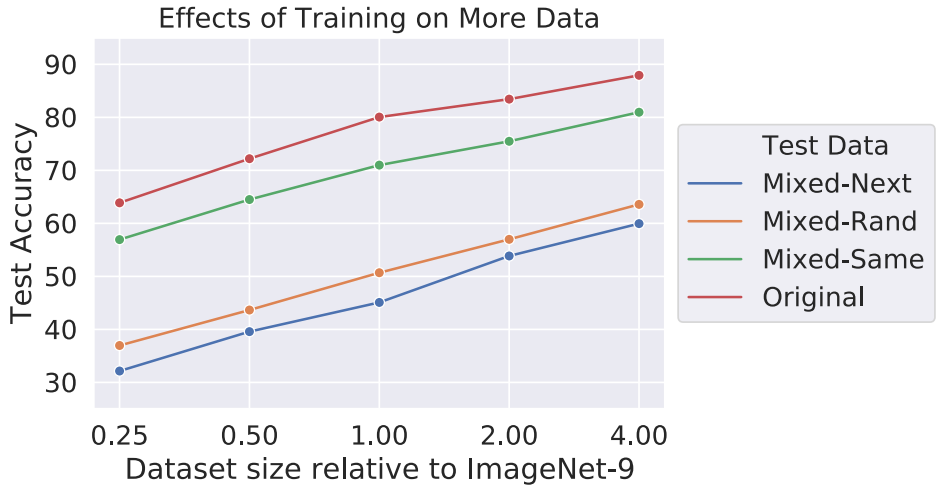

Figure 11: We train models on different-sized subsets of IN-9LB. The largest training set we use is the full IN-9LB dataset, which is 4 times larger than ImageNet-9. While performance on all test datasets improves as the amount of training data increases, the BG-GAP has almost the same size regardless of the amount of training data used.

dataset size, and making the classification task more fine-grained, and find that none of these methods reduces the BG-GAP as much as pre-training on ImageNet. The only method that reduces the BG-GAP significantly more is training on MIXED-RAND. Furthermore, the same trends hold true if we measure the difference between MIXED-SAME and MIXED-NEXT as opposed to the BG-GAP (the difference between MIXED-SAME and MIXED-RAND).

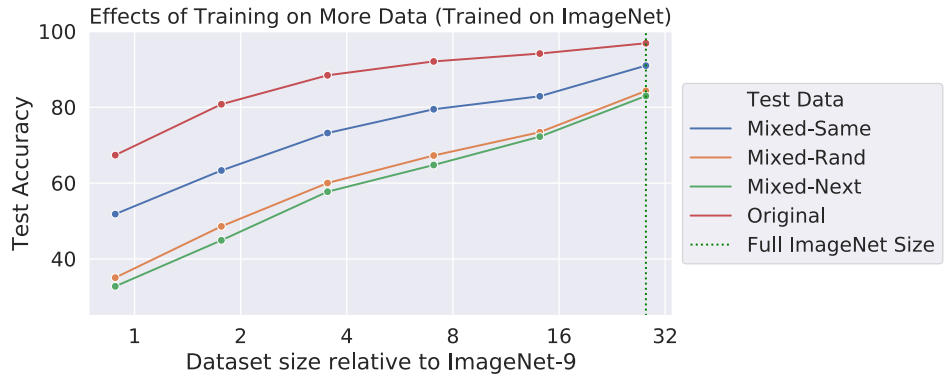

Figure 12: We train models on different-sized subsets of ImageNet. We use a pre-trained ResNet-50 for the rightmost datapoints corresponding to training on the full ImageNet dataset, which is about 30 times larger than ImageNet-9. The BG-GAP begins to decrease when the training dataset set size is sufficiently large.

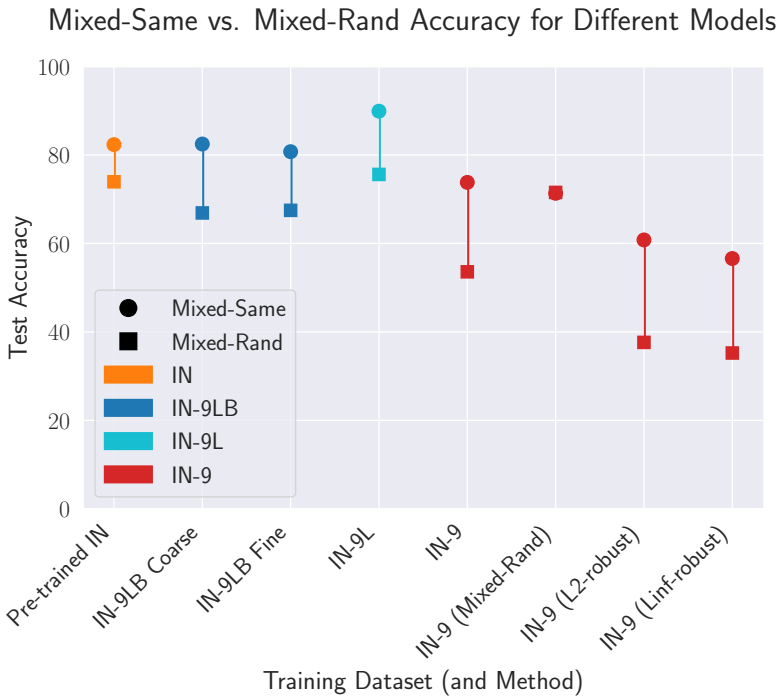

Figure 13: We compare various different methods of training models and measure their BG-GAP, or the difference between MIXED-SAME and MIXED-RAND test accuracy. We find that (1) Pre-trained IN models have surprisingly small BG-GAP. (2) Increasing fine-grainedness (IN-9LB Coarse vs. IN-9LB Fine) and dataset size (IN-9 vs. IN-9L) decreases the BG-GAP only slightly. (3) ℓ_p-robust training does not help. (4) Training on MIXED-RAND (cf. Section 3 appears to be the most effective strategy for reducing the BG-GAP. For such a model, the MIXED-SAME and MIXED-RAND accuracies are nearly identical.

C TRAINING AND EVALUATION DETAILS

For all models, we use fairly standard training settings for ImageNet-style models. We train for 200 epochs using SGD with a batch size of 256, a learning rate of 0.1 (with learning rate drops every 50 epochs), a momentum parameter of 0.9, a weight decay of $1e{-}4$, and data augmentation (random resized crop, random horizontal flip, and color jitter). Unless specified, we always use a standard ResNet-50 architecture (He et al., 2016). For the experiment depicted in Figure 11, we found that using a smaller learning rate of 0.01 was necessary for training to converge on the smallest training sets. Thus, we used that same learning rate for all models in Figure 11.

When evaluating ImageNet classifiers on IN-9, we map all ImageNet predictions to their corresponding coarse-grained class in IN-9. For example, we map both `giant schnauzer` and `Irish terrier` to `dog`, and both `goldfish` and `tiger shark` to FISH. If an ImageNet classifier outputs a class that has no corresponding coarse-grained class in IN-9, we consider the prediction incorrect.

D ADDITIONAL EVALUATION RESULTS

We include full results of training models on every synthetic IN-9 variation and then testing them on every synthetic IN-9 variation in Table 5. In addition to being more comprehensive, this table and these IN-9 variations can help answer a variety of questions, of which we provide three examples here. Finally, we also evaluate a pre-trained model on Full-ImageNet (Full-IN) versions of each synthetic IN-9 variation for comparison.

How does more training data affect model performance with and without object shape?

We already show closely related results on the effect of more training data on the BG-GAP in Figure 11. Here, we compare model test performance on the NO-FG and ONLY-BG-B test sets. Both replace the foreground with black, but only NO-FG retains the foreground shape.

By comparing the models trained on ORIGINAL and IN-9L (4x more training data), we find that

1. The ORIGINAL-trained model performs similarly on NO-FG and ONLY-BG-B, indicating that it does not use object shape effectively.
2. The IN-9L-trained model performs about 13% better on NO-FG than ONLY-BG-B, showing that it uses object shape more effectively.

Thus, this suggests that more training data may allow models to learn to use object shape more effectively. Understanding this phenomena further could help inform model training and dataset collection if the goal is to train models that are able to leverage shape effectively.

How much information is leaked from the size of the foreground bounding box?

The scale of an object already gives signal correlated with the object class (Torralba, 2003). Even though they are designed to avoid having foreground signal, the background-only datasets ONLY-BG-B and ONLY-BG-T may inadvertently leak information about object scale due to the bounding box sizes being recognizable.

To gauge the extent of this leakage, we can measure how models trained on datasets where only the foreground signal has useful correlation (MIXED-RAND or ONLY-FG) perform on the background-only test sets. We find that there is small signal leakage from bounding box size alone—a model trained on ONLY-FG achieves about 23% background-only test accuracy, suggesting that it is able to exploit the signal leakage to some degree. A model trained on MIXED-RAND achieves about 15% background-only test accuracy, just slightly better than random, perhaps because it is harder for models to measure (and thus, make use of) object scale when training on MIXED-RAND.

The existence of a small amount of information leakage in this case shows the importance of comparing MIXED-SAME (as opposed to just ORIGINAL) with MIXED-RAND and MIXED-NEXT when assessing model dependence on backgrounds. Indeed, the MIXED datasets may contain (1) image processing artifacts, such as rough edges from the foreground processing, and (2) small traces of the original background. This makes it important to control for both factors when measuring how models react to varying background signal.

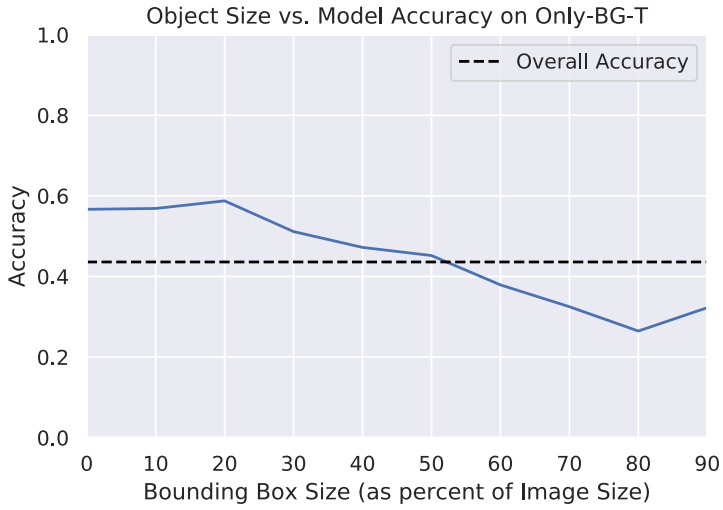

Figure 14: Comparing model accuracy on ONLY-BG-T across different foreground object bounding box sizes. We observe that the model is more likely to succeed when shown only image backgrounds if the removed foreground objects have smaller bounding boxes. The dotted line represents the overall accuracy of the model on ONLY-BG-T (averaged over all bounding box sizes).

How does foreground bounding box size affect accuracy on ONLY-BG-T?

We further find that models are more able to predict accurately using the background signal alone when the foreground object is smaller—this is visualized in Figure 14. Intuitively this result makes sense, as most state-of-the-art models are trained with cropping-based data augmentation, which can remove small foreground objects from training images. Thus, models are actually trained to succeed when small foreground objects are cropped out, and our toolkit confirms that this is indeed the case.

Trained on	**Test Dataset**								
	MIXED-NEXT	MIXED-RAND	MIXED-SAME	NO-FG	ONLY-BG-B	ONLY-BG-T	ONLY-FG	ORIGINAL	IN-9L
MIXED-NEXT	78.07	53.28	48.49	16.20	11.19	8.22	59.60	52.32	46.44
MIXED-RAND	71.09	71.53	71.33	26.72	15.33	14.62	74.89	73.23	67.53
MIXED-SAME	45.41	51.36	74.40	39.85	35.19	41.58	61.65	75.01	69.21
NO-FG	13.70	18.74	42.79	70.91	36.79	42.52	31.48	48.94	47.62
ONLY-BG-B	10.35	15.41	38.37	37.85	54.30	42.54	21.38	42.10	41.01
ONLY-BG-T	11.48	17.09	45.80	40.84	38.49	50.25	19.19	49.06	47.94
ONLY-FG	33.04	35.88	47.63	27.90	23.58	22.59	84.20	54.62	51.50
ORIGINAL	48.77	53.58	73.80	42.22	32.94	40.54	63.23	85.95	80.38
IN-9L	71.21	75.60	89.90	55.78	34.02	43.60	84.12	96.32	94.61
ImageNet	82.99	84.32	90.99	52.69	12.69	17.36	90.17	96.89	95.33
ImageNet (Full-IN)	51.47	48.69	64.34	21.70	7.98	9.51	59.19	76.07	-

Table 5: The test accuracies, in percentages, of ResNet-50 models trained on all variants of ImageNet-9, and a pre-trained ImageNet ResNet-50. The bottom row and the second-to-last-row test the same pre-trained ImageNet model; however, the bottom row tests the model on the Full-IN version of each dataset variation. Testing on Full-IN shows similar trends as testing on ImageNet-9. Note that the MIXED-NEXT test accuracy is actually higher than the MIXED-RAND test accuracy in the bottom row because the next class is often very similar to the previous class in Full-IN.

What about other ways of modifying the background signal?

One can modify the background in various other ways—for example, instead of replacing the background with black as in ONLY-FG, the background can be blurred as in the BG-BLURRED image of Figure 15. As expected, blurred backgrounds are still slightly correlated with the correct class. Thus, test accuracies for standard models on this dataset are higher than on ONLY-FG, but lower than on MIXED-SAME (which has signal from random class-aligned backgrounds that are *not* blurred). While we do not investigate all possible methods of modifying background signal, we believe that the variations we do examine in ImageNet-9 already improve our understanding of

Figure 15: Backgrounds can also be modified in other ways; for example, it can be blurred. Our evaluations on this dataset show similar results.

how background signals matter. Investigating other variations could provide an even more nuanced understanding of what parts of the background are most important.

E Additional Related Works and Explicit Comparisons

There has been prior work on mitigating contextual bias in image classification, the influence of background signals on various datasets, and techniques like foreground segmentation that we leverage.

Mitigating Contextual Bias: (Khosla et al., 2012) focuses on mitigating dataset-specific contextual bias and proposes learning SVMs with both general weights and dataset-specific weights, while (Myung Jin et al., 2012) creates an out-of-context detection task with 209 out-of-context images and suggests using graphical models to solve it. (Shetty et al., 2019) focuses on the role of co-occurring objects as context in the MS-COCO dataset, and uses object removal to show that (a) models can still predict a removed object when only co-occurring objects are shown, and (b) special data-augmentation can mitigate this.

Understanding the influence of backgrounds: For contextual bias from image backgrounds specifically, prior works have observed that the background of an image can influence model decisions to varying degrees. In particular, (Zhang et al., 2007) find that (a) a bag-of-features object detection algorithm depends on image backgrounds in the PASCAL dataset and (b) using this algorithm on a training set with varying backgrounds leads to better generalization. (Beery et al., 2018; Barbu et al., 2019) collect new test datasets of animals and objects, respectively. (Barbu et al., 2019) focus on object classes that also exist in ImageNet, and their new test set contains objects photographed in front of unconventional backgrounds and in unfamiliar orientations. Both works show that computer vision models experience significant accuracy drops when trained on data with one set of backgrounds and tested on data with another. (Sagawa et al., 2020) create a small synthetic dataset of Waterbirds, where waterbirds and landbirds from one dataset are combined with water and land backgrounds from another. They show that a model's reliance on spurious correlations with the background can be harmful for small subgroups of data where those spurious correlations no longer hold (e.g. landbirds on water backgrounds). Furthermore, they propose using distributionally robust optimization to reduce reliance on spurious correlations with the background, but their method assumes that the spurious correlation can be precisely specified in advance. (Rosenfeld et al., 2018) analyzes background dependence for object detection (as opposed to classification) models on the MS-COCO dataset. They transplant an object from one image to another image, and find that object detection models may detect the transplanted object differently depending on its location, and that the transplanted object may also cause mispredictions on other objects in the image.

Explicit Comparison to Prior Works: In comparison to prior works, our work contributes the following.

- We develop a toolkit for analyzing the background dependence of ImageNet classifiers, the most common benchmark for computer vision progress. Only (Zhu et al., 2017), which we compare to in Section 5, also focuses on ImageNet.
- The test datasets we create separate and mix foreground and background signals in various ways (cf. Table 1), allowing us to study the sensitivity of models to these signals in a more fine-grained manner.

- Our toolkit for separating foreground and background can be applied without human-annotated foreground segmentation, which prior works on MS-COCO and Waterbirds rely on. This is important because foreground segmentation annotations are hard to collect and do not exist for ImageNet.
- We study the extent of background dependence in the extreme case of adversarial backgrounds.
- We focus on better vision models, including ResNet (He et al., 2016), Wide ResNet(Zagoruyko & Komodakis, 2016), and EfficientNet (Tan & Le, 2019).
- We evaluate how improvements on the ImageNet benchmark have affected background dependence (cf. Section 4).

Foreground Segmentation and Image Inpainting: In order to create IN-9 and its variants, we rely on OpenCV's implementation of the foreground segmentation algorithm GrabCut (Rother et al., 2004). Foreground segmentation is a branch of computer vision that seeks to automatically extract the foreground from an image (Harville et al., 2001). After finding the foreground, we remove it and simply replace the foreground with copies of parts of the background. Other works solve this problem, called image inpainting, either using exemplar-based methods (Criminisi et al., 2004) or using deep learning Yu et al. (2018); Shetty et al. (2018). (Shetty et al., 2018) both detects the foreground for removal and inpaints the removed region. However, more advanced inpaintings techniques can be slow and inaccurate when the region that must be inpainted is relatively large (Shetty et al., 2018), which is the case for many ImageNet images. Exploring better ways of segementing the foreground and inpainting the removed foreground could improve our analysis toolkit further.

F ADDITIONAL EXAMPLES OF SYNTHETIC DATASETS

We randomly sample an image from each class, and display all synthetic variations of that image, as well as the predictions of a pre-trained ResNet-50 (trained on IN-9L) on each variant.

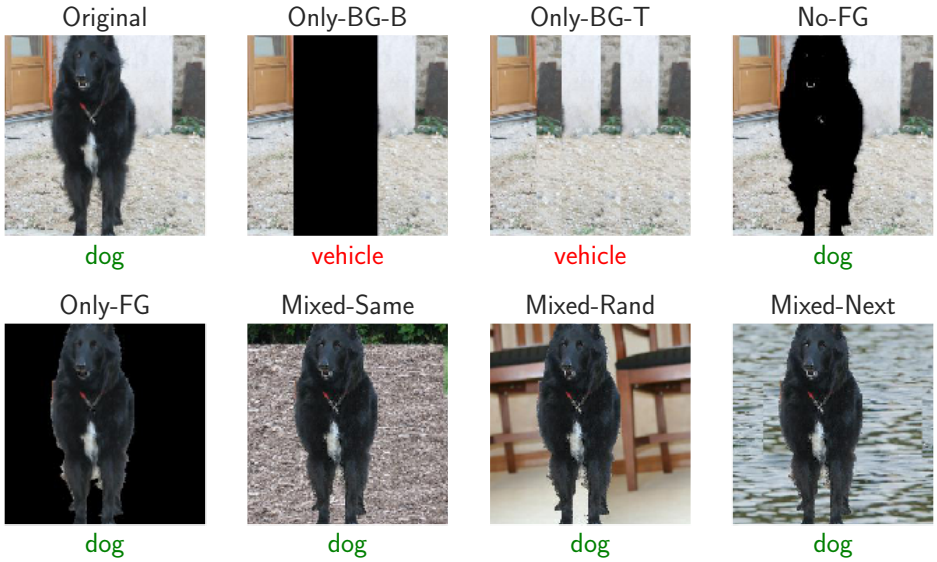

Figure 16: ImageNet-9 variations—Dog.

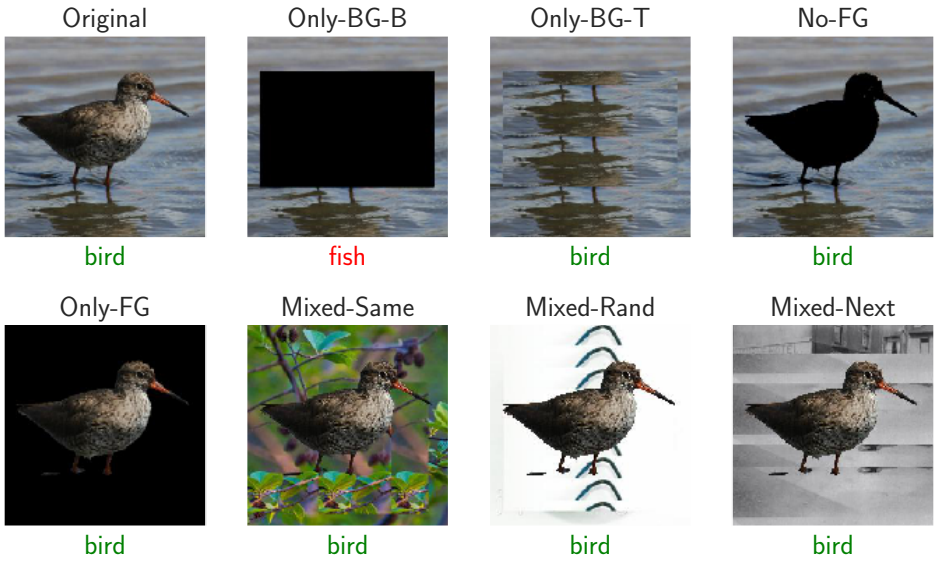

Figure 17: ImageNet-9 variations—Bird.

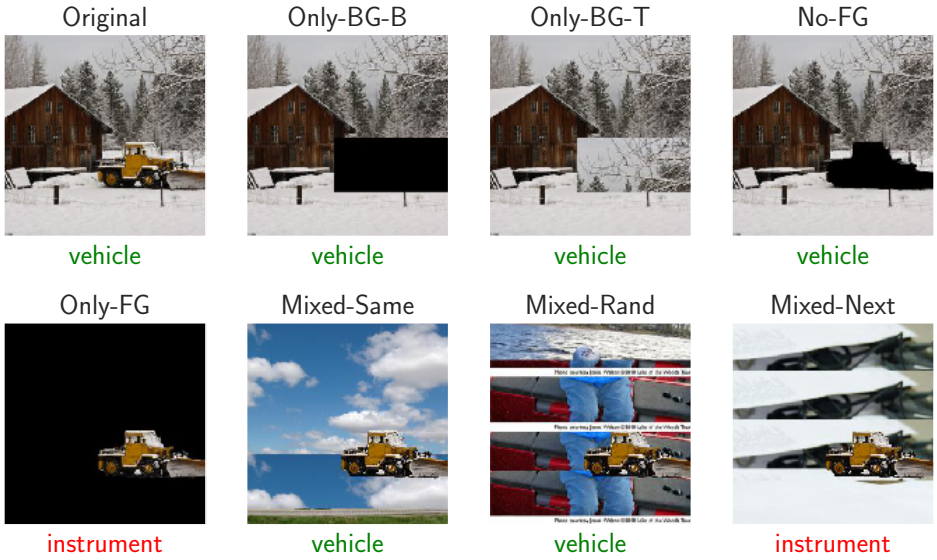

Figure 18: ImageNet-9 variations—Vehicle.

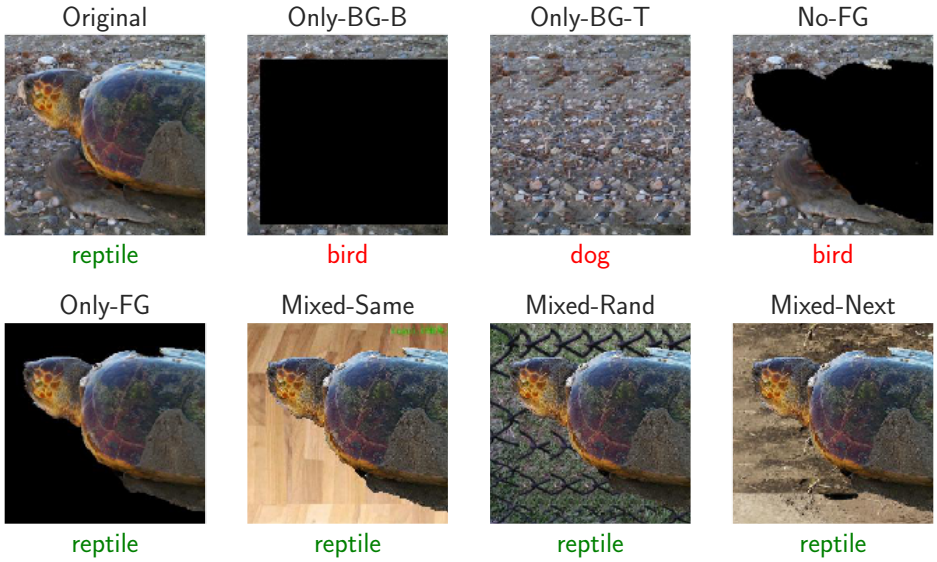

Figure 19: ImageNet-9 variations—Reptile.

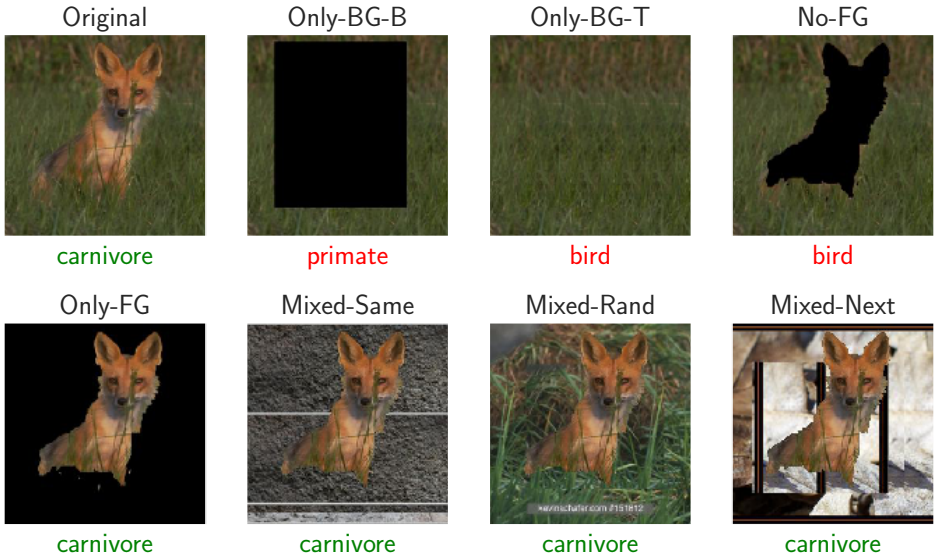

Figure 20: ImageNet-9 variations—Carnivore.

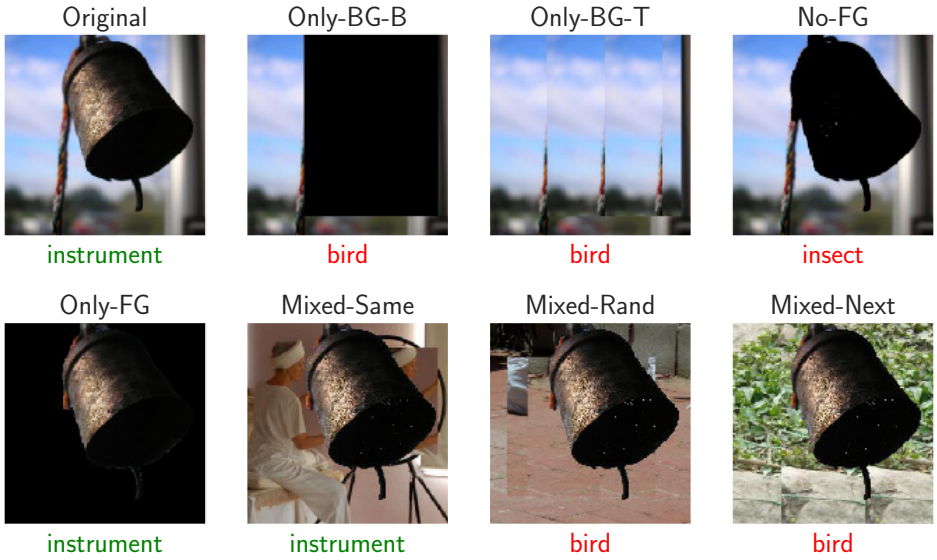

Figure 21: ImageNet-9 variations—Instrument.

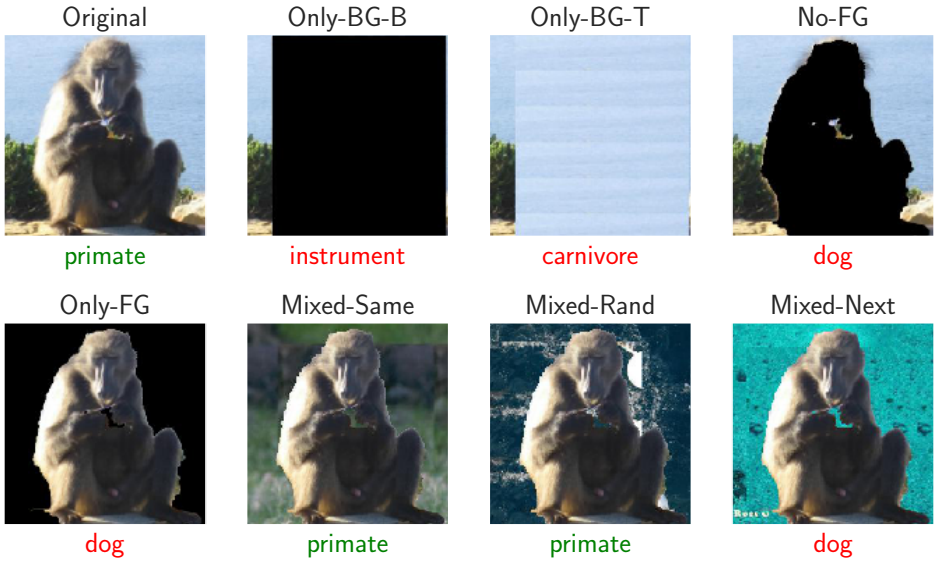

Figure 22: ImageNet-9 variations—Primate.

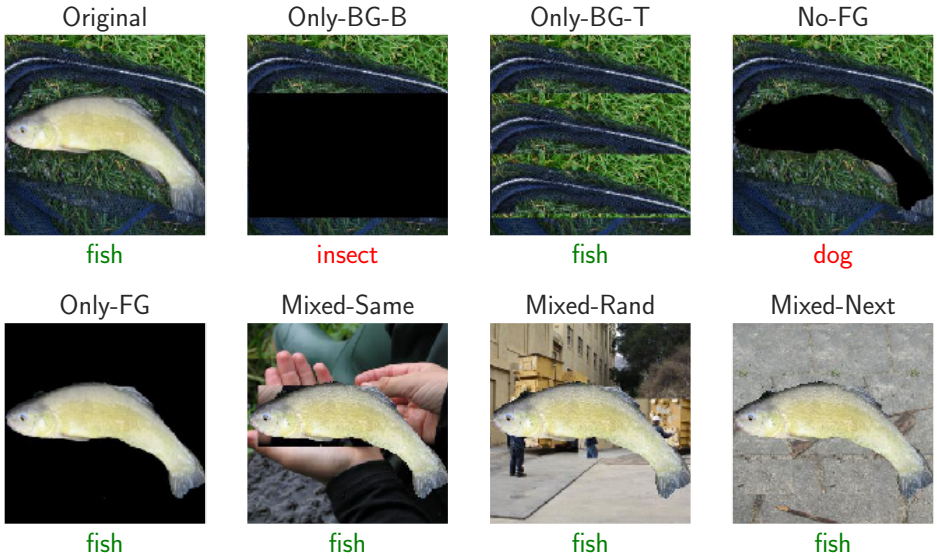

Figure 23: ImageNet-9 variations—Fish.

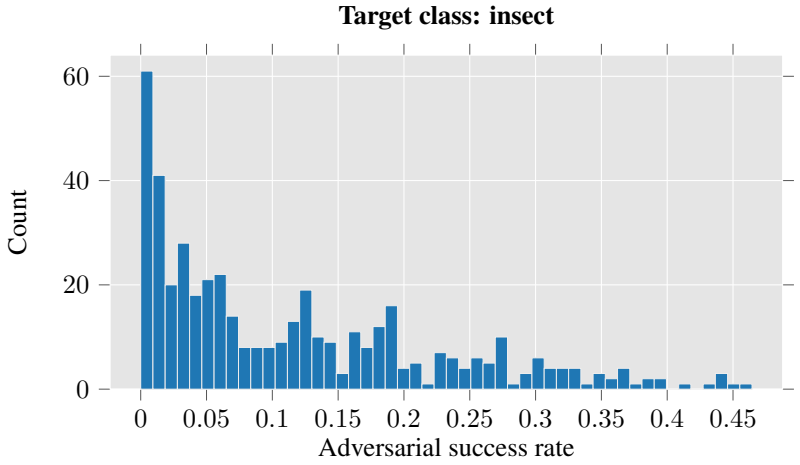

Figure 24: Histogram of insect backgrounds grouped by how often they cause (non-insect) foregrounds to be classified as insect by a IN-9L-trained model. We visualize the five backgrounds that fool the classifier on the largest percentage of images in Figure 4.

G ADVERSARIAL BACKGROUNDS

We compute the adversarial background attack success rate for 4 models in Table 6. While the MIXED-RAND model is more adversarially background robust than the ORIGINAL model, it is less adversarially background robust than the IN-9L model. The model trained on all of ImageNet is the most adversarially background robust of all models. This suggests that increasing training dataset size (IN-9L) has a bigger effect on adversarial background robustness than randomizing backgrounds during training (MIXED-RAND). On the other hand, the MIXED-RAND model has a much lower BG-GAP than the IN-9L model, indicating that models with a smaller BG-GAP are not necessarily robust to adversarial backgrounds, and vice versa.

Training Dataset	ORIGINAL	MIXED-RAND	IN-9L	ImageNet
Attack Success Rate	99.0%	93.5%	88.0%	77.7%

Table 6: Adversarial backgrounds attack success rates for 4 models analyzed in this paper. The ORIGINAL and the MIXED-RAND are trained on equally small datasets, IN-9L is trained on 4x more data, and the ImageNet model is trained on the most data.

Next, we visualize the attack success rate distribution of the different backgrounds from the insect class in Figure 24. The long tail of the distribution indicates that many backgrounds are especially capable of fooling models.

Finally, we include the 5 most fooling backgrounds for all classes, the fool rate for each of those 5 backgrounds, and the total fool rate across all backgrounds from that class (on the left of each row) below.

Figure 25: Most adversarial backgrounds—Dog.

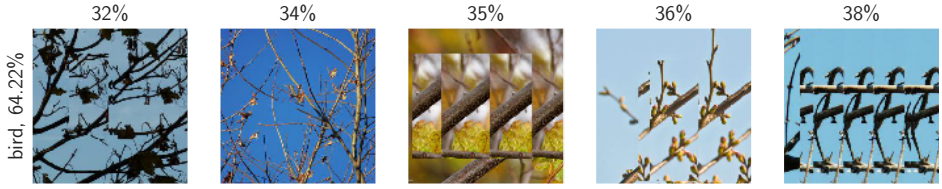

Figure 26: Most adversarial backgrounds—Bird.

Figure 27: Most adversarial backgrounds—Vehicle.

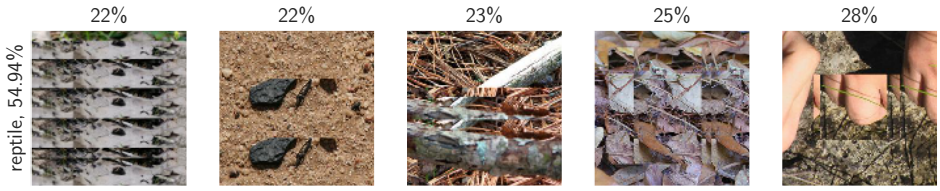

Figure 28: Most adversarial backgrounds—Reptile.

Figure 29: Most adversarial backgrounds—Carnivore.

Figure 30: Most adversarial backgrounds—Instrument.

Figure 31: Most adversarial backgrounds—Primate.

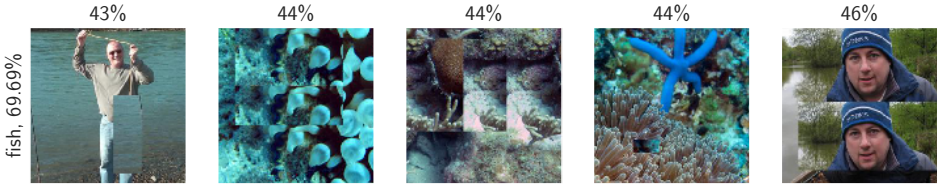

Figure 32: Most adversarial backgrounds—Fish.

H EXAMPLES OF FOOLING BACKGROUNDS IN UNMODIFIED IMAGES

We visualize examples of images where the background of the full original image actually fools models in Figure 33. For these images, models classify the foreground alone correctly, but they predict the same wrong class on the full image and the background. We denote these images as "BG Fools" in Table 3 and Figure 7. While this category is relatively rare (accounting for just 3% of the ORIGINAL-trained model's predictions), they reveal a subset of original images where background signal hurts classifier performance. Qualitatively, we observe that these images all have confusing or misleading backgrounds.

Figure 33: Images that are incorrectly classified (as the class on the top row, which is the same class that their background alone from ONLY-BG-T is classified as), but are correctly classified (as the class on the bottom row) when the background is randomized. Note that these images have confusing backgrounds that could be associated with another class.