# Investigation of Image Augmentation Techniques and a Modified ResNeXt Network for Fine-Grained Dog Breed Classification

Morgan Freiberg, Christian Gall
University of Virginia, Charlottesville, VA 22904
msf6wk@virginia.edu, cg4ah@virginia.edu

## Abstract

*Classifying dog breeds is a common deep learning task that many use to be introduced to the field given its many solutions and the availability of the large Stanford Dataset. However, many focus on the models themselves and may neglect how data augmentation affects their performance. We focus on how different data augmentations affect performance and what the effect may indicate about the model and the feature extraction process. In our project we explored changes in image orientation and color where each image was then fed into a modified, pre-trained ResNext CNN model. We then compare the top-5 accuracies of these different augmentations. We show that data augmentation focusing on changing color outperform those changing orientation.*

## 1. Introduction

During the last years, Convolutional Neural Networks (CNNs) have become the predominant method in the field of Computer Vision for classification tasks. When it comes to developing and educating CNNs, determining the breed of dogs in the Stanford Dogs Dataset is a popular application. It is used for competitions, tutorials and specifically for investigating fine-grained image classification problems. [6, 1, 3, 8]

While many CNNs are trained to determine a broad category like distinguishing between cats and dogs, fine-grained image classification deals with categorizing images that share many similar features. An example is categorizing animal breeds instead of animal species. A Husky and a German Shepherd, being both dogs, share many features such as a similar body shape. They only differ in details like fur color and the shape of the head. We investigate how a modern CNN like ResNeXt is performing in such tasks.

Comparable investigations have already been performed by other groups. [6, 11, 4] In contrast to them, we also want to investigate how data augmentation, which is used to artificially produce larger data sets by adding modified versions of the given images, affects the fine-grained image classification performance.

## 2. Related Work

Several groups have tested their models on the Stanford Dog Dataset. Early attempts such as [6] focused on a classification based on SIFT and obtained an accuracy of 22%. Using modern CNNs, accuracies of 79.25% and 89.66% were obtained in [11] and [4] respectively.

Regarding Data Augmentation, Taylor and Nitschke investigated in [9] the effect of various geometric and photometric transformations on the performance of CNNs without especially focusing on fine-grained image classification. Their results state that geometric transformation such as flipping, cropping, and rotating are superior to photometric operations like color jittering. Especially cropping is found to be the best image augmentation technique.

In [7] Wang and Perez confirm the effectiveness of geometric image augmentation techniques. Additionally, they investigate more sophisticated techniques such as using Generative Adversarial Networks (GANs) to modify the style of the images. However, they cannot produce major accuracy improvements using these methods compared to geometric transformations. There were also no special investigations for fine-grained image classification problems.

## 3. Model

Our proposed CNN is based on ResNeXt, which was introduced by Xie et al. in [10]. The architecture of ResNeXt is similar to Resnet, but uses filters with less channels and includes blocks that use the split-transform-merge strategy, which was introduced with inception layers in GoogLeNet. Those blocks contain multiple parallel paths of the same topology.

We chose this architecture for two reasons. First, pytorch provides a ResNeXt model that is pre-trained in the ILVSRC task on the Imagenet dataset. We consider this as a major advantage since Imagenet also includes sev-

eral dog categories and using transfer learning promises a short training time. Second, among the provided pre-trained models in pytorch, ResNeXt offers the highest accuracy on Imagenet. [2]

In order to adjust the model to our specific task, we propose to replace the last layer with the following three layers:

1. Linear layer with input size 2048 and output size 1024
2. Linear layer with input size 1024 and output size 512
3. Linear layer with input size 512 and output size 120

All three layers include bias parameters.

## 4. Experiments and Results

We used pytorch to implement our model and used the modified, pre-trained 50-layer ResNext architecture [10] as our base deep convolutional network. We trained our model using stochastic gradient descent with a momentum of 0.9, learning rate of 0.001, and a mini-batch size of 16. Each model was trained for 10 epochs, from which the optimal epoch was analyzed. We show the loss and top-5 accuracies of training and validating over time for this baseline model in Figure 2 and results are summarized in Table 1. For each method images were cropped according to their bounding box in order to minimize other noise in the image [5]. Before training, each image was then resized to 200x200 to standardize image size. This was chosen over cropping after a preliminary exploration of the processed images revealed important features were being removed by cropping. The investigation into orientation changes meant the addition of random flips (both horizontal and vertical), being completed at a probability of $0.5$ and a random rotation of a maximum of $\pm 5^{\circ}$. These changes led to a $0.5\%$ increase in top-5 accuracy [Figure 3]. The second method (color augmentation) was comprised of applying a color jitter transformation to randomly change the brightness, contrast, and saturation of the image. Additionally a random greyscale was applied with a probability of $0.5$. These led to a $1.74\%$ increase in top-5 accuracy [Figure 4]. When these methods were combined into a model that included both orientation and color augmentations the top-5 accuracy was increased by $1.1\%$ [Figure 5].

| Augmentation | Best Epoch | Validation Top-5 |
|---|---|---|
| Baseline | 10 | 98.2% |
| Orientation | 10 | 98.7% |
| Color | 10 | 99.94% |
| Orientation+Color | 5 | 99.3% |

Table 1: Overview of results.

We believe that color augmentations performed best as a result of the nature of convolutional layers. Convolutions innately remove the importance of the placement of features within an image for feature extraction by considering the image in kernel sized batches. This means it can use the same filter to identify some feature no matter where the feature is placed in the image. As a result, the orientation transforms may not have created a meaningful increase in features. This reasoning may be applied to color transformations and their improved results. Since convolutions do not remove any color related dimensions in the images or features, adding new images with random color augmentations added meaningful data to the dataset that may have resulted in better feature extraction. In addition the random greyscale likely forced the model to rely on or extract more hue independent features.

Our initial exploration of model performance revealed that the source of the model's inaccuracy is not evenly distributed across the breeds of dogs. Instead we found that specific breeds perform much worse than others. One example is the dog breed Saluki, where the model consistently performed inaccurately. Future work would include isolating these breeds in the hopes of finding a common factor that the model struggles with so we can better classify those breeds and better understand the model's shortcomings.


Figure 1: Examples of Saluki images that were inaccurately classified.

In the future we would explore visualizing the features and feature maps between these augmentations to confirm or discredit our hypotheses. By understanding what aspects of features models rely on for breed classification we can better augment the data to support our models. These methods could then be extended to similar problems in computer vision as related to the classification of animals.

## References

[1] Competition on kaggle.com: Dog breed identification.
[2] pytorch documentation on pytorch.org: Torchvision.models.
[3] Tutorial on towardsdatascience.com: Dog breed classification using cnns.
[4] A. Ayanzadeh and S. Vahidnia. A modified deep neural networks for dog breeds identification. 2018.
[5] A. Chandra. Crop images using bounding box. 2019.
[6] A. Khosla, N. Jayadevaprakash, B. Yao, and F.-F. Li. Novel dataset for fine-grained image categorization: Stanford dogs.

In *Proc. CVPR Workshop on Fine-Grained Visual Categorization (FGVC)*, volume 2, 2011.

[7] L. Perez and J. Wang. The effectiveness of data augmentation in image classification using deep learning. *CoRR*, abs/1712.04621, 2017.

[8] J. Pu, Y.-G. Jiang, J. Wang, and X. Xue. Which looks like which: Exploring inter-class relationships in fine-grained visual categorization. In *European conference on computer vision*, pages 425–440. Springer, 2014.

[9] L. Taylor and G. Nitschke. Improving deep learning using generic data augmentation. *CoRR*, abs/1708.06020, 2017.

[10] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. *arXiv:1611.05431v2*, 2017.

[11] H. Z. H. S. L. Zhu and Z. Deng. Dog classification into 120 breeds.

## Appendix

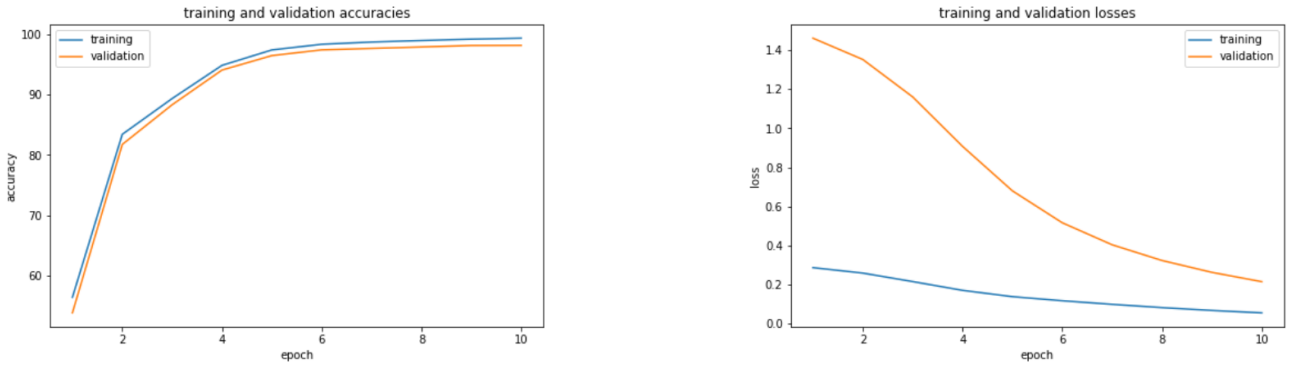


Figure 2: Baseline data augmentation training and validation accuracies and losses.

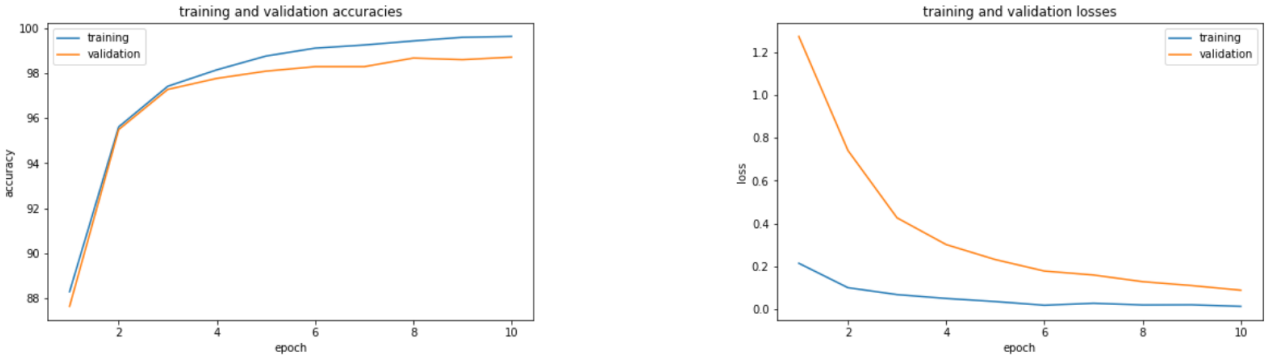


Figure 3: Orientation data augmentation training and validation accuracies and losses.

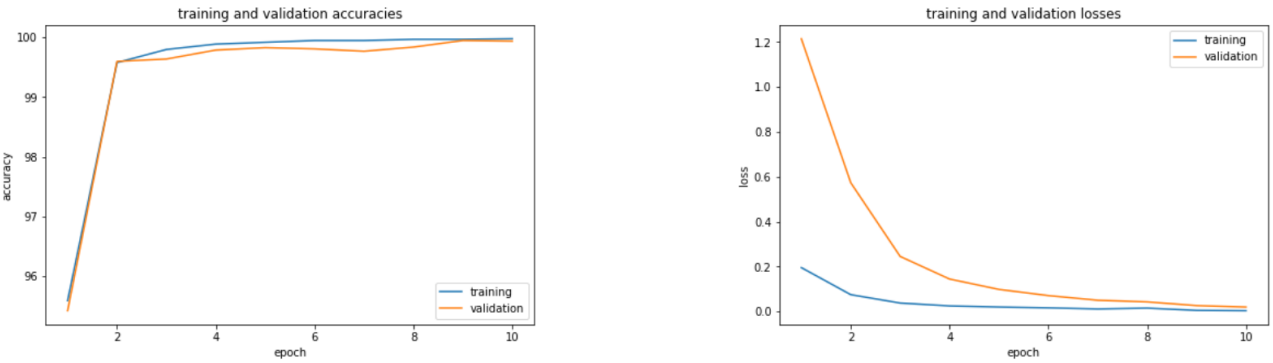


Figure 4: Color data augmentation training and validation accuracies and losses.

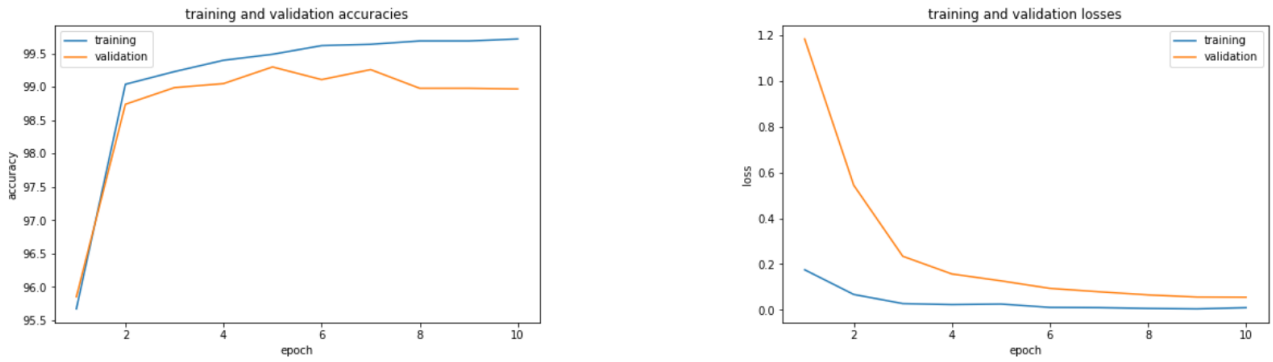


Figure 5: Color and orientation data augmentations training and validation accuracies and losses.