Training with Confusion for Fine-Grained Visual Classification

Abhimanyu Dubey\textsuperscript{1}, Otkrist Gupta\textsuperscript{2}, Pei Guo\textsuperscript{3}, Ramesh Raskar\textsuperscript{2}, Ryan Farrell\textsuperscript{3}, Nikhil Naik\textsuperscript{1}
\textsuperscript{1}Harvard University, Cambridge, MA 02138
\textsuperscript{2}Massachusetts Institute of Technology, Cambridge, MA 02139
\textsuperscript{3}Brigham Young University, Provo, UT 84602

Abstract

Research in Fine-Grained Visual Classification has focused on tackling the variations in pose, lighting, and viewpoint using sophisticated localization and segmentation techniques, and the usage of robust texture features to improve performance. In this work, we look at the fundamental optimization of neural network training for fine-grained classification tasks with minimal inter-class variance, and attempt to learn features with increased generalization to prevent overfitting. We introduce Training-with-Confusion, an optimization procedure for fine-grained classification tasks that regularizes training by introducing confusion in activations. Our method can be generalized to any fine-tuning task; it is robust to the presence of small training sets and label noise; and adds no overhead to the prediction time. We find that Training-with-Confusion improves the state-of-the-art on all major fine-grained classification datasets.

1 Introduction

In the past decade, the advent of large-scale datasets and improvements in training deep neural networks have enabled massive advances in computer vision, especially in image classification [10, 11]. An important computer vision task is Fine-Grained Visual Classification (FGVC), which involves distinguishing between object classes with substantially higher visual similarity compared to those in large-scale image classification. Some examples of FGVC include differentiating between species of birds, flowers and animals; or the makes and models of vehicles. These tasks depart from conventional image classification in that they require expert knowledge, rather than crowdsourcing, for gathering annotations. Additionally for fine-grained wildlife data collection, several species are generally harder to photograph, resulting in long tails in the data distribution. Moreover, FGVC datasets have minute inter-class visual differences in addition to the variations in pose, lighting and viewpoint found in standard image classification [25]. This combination of the effects of small, non-uniform datasets and subtle inter-class differences makes fine-grained visual classification challenging even for powerful deep learning algorithms.

Most of the prior work in FGVC has focused on tackling the variations in pose, lighting, and viewpoint using localization techniques [12, 20, 25, 53, 55], and by augmenting training datasets with additional data from the Web [6, 21]. In this paper, we employ a different method to approach the FGVC problem at the fundamental level of neural network training. We observe that prior work in FGVC does not pay much attention to inter-class visual similarity in the feature extraction pipeline. In large-scale image classification datasets such as ImageNet [7], strongly discriminative learning using the cross-entropy loss is successful in part due to the significant inter-class variation (compared to intra-class variation), which enables deep networks to learn generalized discriminatory features with large amounts of data.

However, for FGVC (which shows smaller inter-class variation), this formulation may not be ideal. For instance, if two samples in the training set have very similar visual content but different class
labels, the cross-entropy loss will force the neural network to learn features that distinguish these two images with a high confidence—potentially forcing the network to learn sample-specific artifacts for visually confusing classes in order to minimize training error. This effect may be pronounced in FGVC tasks, since there are fewer samples for the network to learn general class-specific features from. Based on this hypothesis, we expect the introduction of confusion in output logit activations to enable the network to learn slightly less distinctive features, thereby preventing it from overfitting to sample-specific artifacts.

In this paper, we extend this idea and propose a training procedure entitled “Training-with-Confusion (TWC)”. TWC employs two novel penalty strategies to train convolutional neural networks (CNNs) end-to-end for fine grained visual classification. Using TWC, we obtain state-of-the-art performance across 6 major fine-grained recognition datasets. Moreover, we demonstrate that TWC provides significant improvements over baseline CNNs and is robust to amount of training data and label noise. We experimentally demonstrate that TWC results in greater feature generalization as compared to standard methods. Our method is easy to implement and has no added overhead in training or prediction time.

2 Related Work

Fine-Grained Visual Classification: In recent years, improved localization of the target object in training set images has shown to be very useful for Fine-Grained Visual Classification (FGVC) [8, 25, 49, 51]. Zhang, et al. [51] utilize part-based Region-CNNs [39] to perform finer localization. Lin, et al. [25] propose a novel bilinear pooling operation to combine pairwise local feature sets and improve classification performance, which has been extended in the work of Gao, et al. [8] with improvements in efficiency. Spatial Transformer Networks [12] show that learning a content-based affine transformation layer improves FGVC performance. Pose-normalized CNNs have also been shown to be effective at FGVC [2, 52]. Robust image representations such as CNN filter banks [4], VLAD [14] and Fisher vectors [37] have been prior techniques at tackling fine-grained classification. Model ensembling and boosting has also improved performance on FGVC, as demonstrated by Moghimi et al. [31].

Pairwise Learning: Our work also relates to computer vision methods based on pairwise learning. Parikh and Grauman [34] explore a pairwise ranking scheme for learning attribute rankings. Chopra et al. [3] introduce a discriminative training regime for learning a similarity metric. Pairwise loss functions have also been employed for detection in crowded scenes [44] and online learning [16] with investigations into theoretical guarantees [17].

Regularizing Entropy: Regularization methods that penalize minimum entropy predictions have been explored in the context of semi-supervised learning [9]. Jaynes [13] introduced the maximum entropy principle, which provided an early understanding of the advantage of controlling the value of classifier entropy, leading to the work on deterministic entropy annealing [40]. Entropy-based regularization has also been shown to significantly improve reinforcement learning methods [27, 30].

Learning from Noisy Data: Alternative methods of introducing confusion have been analysed previously in computer vision, such as methods that utilize label noise (e.g., [38]) and data noise (e.g., [50]) in training. Krause et al. [21] utilize noisy training data in the context of fine-grained classification. Neelakantan et al. [32] add noise to the gradient during training to improve generalization performance in very deep networks. Szegedy et al. [46] introduce label-smoothing regularization for training deep Inception models.

In this paper, we examine the utility of two forms of activation confusion methods—Pairwise Confusion and Entropic Confusion—in training neural networks for fine-grained visual classification. We recently became aware of a newly published workshop paper by Pereyra et al. [36] that examines an entropy penalty as a regularizer for classification tasks—which is similar to our Entropic Confusion formulation. For a detailed comparison with this work, see Section 5.

3 Method

We experiment with two related formulations for limiting classifier overconfidence by introducing confusion between class activations for fine-grained visual classification. We call these formulations
“Pairwise Confusion” and “Entropic Confusion”, and we call our training method “Training-with-Confusion (TWC)”.

3.1 Pairwise Confusion

Pairwise loss functions have been explored in the context of metric learning [3, 19] and attribute learning [34]. On similar lines, for a neural network with parameters \( \theta \) that produces the conditional probability distribution \( p_\theta(y|x) \) over \( N \) classes for input image \( x \), we introduce the Pairwise Confusion Loss \( L_p \), where

\[
L_p(x_1, x_2; \theta) = \sum_i \left( p_\theta(y_i|x_1) - p_\theta(y_i|x_2) \right)^2
\]

where \( x_1, x_2 \) are random samples from the training set. \( L_p \) is a conservative estimate for the symmetric KL (Kullback-Leibler) divergence between \( p_\theta(y|x_1) \) and \( p_\theta(y|x_2) \) (see supplement for details). Through this loss function, we aim to directly penalize the distance between the predicted output logits. \( L_p \) can be made sensitive to class labels by only penalizing image pairs from different target classes, however, we do not see significant improvement in performance with class labels included in the formulation. Therefore, we maintain this formulation \( L_p \) for simplicity and applicability as a general regularization scheme.

Through this formulation, we expect the representations for dissimilar classes to be pulled closer in the output manifold. We optimize the objective \( \mathcal{L} \) for a batch \( B = [B_1, B_2] \) of \( b \) samples each:

\[
\mathcal{L}(B) = \mathcal{L}_{ce}(B_1) + \mathcal{L}_{ce}(B_2) + \frac{\lambda}{2b} \cdot L_p(B_1, B_2)
\]

where \( \mathcal{L}_{ce} \) denotes cross-entropy summed over each sample \( b_i \) in the target batch \( B \), and \( L_p \) is calculated between pairs \((b_{ij}, b_{ji})\) of \([B_1, B_2]\). This loss function is interpretable and insensitive to larger ranges of the weighting parameter \( \lambda \). In the next section, we introduce a more general formulation under the assumption of uniform label distributions, motivated by information theory.

3.2 Entropic Confusion

If we assume the distribution of classes in training and prediction phases to be uniform, we can measure the deviation of the output probability from a random classifier as a measure of prediction certainty, and limit it in order to introduce confusion in output activations [38]. To measure this deviation, we consider the KL divergence \( D_{KL}(p_\theta(y|x) \mid \mid p_\theta(y|u)) \), where \( u \) is the uniform vector with norm 1. We see that:

\[
D_{KL}(p_\theta(y|x) \mid \mid p_\theta(y|u)) = \sum_i p_\theta(y_i|x) \cdot \log \left( \frac{p_\theta(y_i|x)}{p_\theta(y_i|u)} \right)
\]

\[
= \log N + \sum_i p_\theta(y_i|x) \cdot \log p_\theta(y_i|x)
\]

\[
= \log N - H(p_\theta(y|x))
\]

where \( H(p_\theta(y|x)) \) is the Shannon entropy of \( p_\theta(y|x) \). Hence, minimizing certainty through \( D_{KL}(p_\theta(y|x) \mid \mid p_\theta(y|u)) \) is equivalent to maximizing the Shannon entropy \( H(p_\theta(y|x)) \). We formulate the Entropic Confusion Loss \( \mathcal{L}_e \) as:

\[
\mathcal{L}_e(x; \theta) = -H(p_\theta(y|x))
\]

We formulate the final objective for a batch of \( b \) samples as:

\[
\mathcal{L} = \mathcal{L}_{ce} + \frac{\lambda}{2} \cdot \mathcal{L}_e
\]

We experiment with both formulations in subsequent sections and find that both forms of confusion significantly benefit generalization abilities in fine-grained visual classification.

4 Experimental Details

We demonstrate the effectiveness of Training-with-Confusion for fine-grained visual classification. We perform all experiments using the Caffe [15] and PyTorch [35] frameworks over a cluster of NVIDIA Titan X, Tesla k40c and 1080 GPUs. Next, we provide brief descriptions of the various datasets used in our paper.
Table 1: Training-with-Confusion (TWC) obtains state-of-the-art performance on six standard fine-grained visual classification datasets. TWC significantly improves the performance of standard networks (e.g., GoogLeNet, ResNet-50) during fine-tuning (A-F).

4.1 Fine-Grained Visual Classification Datasets

We evaluate our method using six standard Fine-grained Visual Classification (FGVC) datasets. The Caltech-UCSD Birds (CUB-2011) dataset [48] has 5,994 training and 5,794 test images across 200 species of birds. The Cars dataset [22] contains 8,144 training and 8,041 test images across 196 car classes. The classes represent variations in the make, model, and year of cars. The Stanford Dogs dataset [18] has 20,580 images across 120 classes (dog breeds). The NABirds dataset [47] contains 23,929 training and 24,633 test images across 550 bird categories. The Flowers-102 dataset [33] consists of 1,020 training, 1,020 validation and 6,149 test images over 102 flower types. Finally, the Aircrafts dataset is a set of 10,000 images across 100 classes denoting a fine-grained set of airplanes of different varieties [29]. For all datasets, we perform training and prediction without using any annotations (where available), and all models are initialized from their publicly available ImageNet-trained weights following standard protocol in FGVC.

4.2 Image Classification Datasets

We also utilize two standard image classification datasets—CIFAR-10 and CIFAR-100—for ablation studies. The CIFAR-10 dataset contains 60,000 32x32 RGB images from 10 different object categories [23], with a train-test split of 50,000 and 10,000 images. The CIFAR-100 dataset contains the same overall number of training and test images as CIFAR-10, but these are split across 100 classes, resulting in a 10× reduction in data points per class.
| Method           | CIFAR-10 | CIFAR-10+ | CIFAR-100 | CIFAR-100+ |
|------------------|----------|-----------|-----------|------------|
| GoogLeNet        | 84.16    | 86.63     | 70.24     | 73.35      |
| \(L_p\) + GoogLeNet | 84.27    | 86.91     | 72.29     | 75.78      |
| \(L_e\) + GoogLeNet | 84.10    | 87.02     | 73.62     | 76.02      |
| DenseNet-121     | 92.19    | 95.04     | 75.01     | 78.60      |
| \(L_p\) + DenseNet-121 | 92.27    | 95.16     | 75.82     | 79.49      |
| \(L_e\) + DenseNet-121 | 91.86    | 94.97     | 76.17     | 79.56      |

Table 2: Accuracies on CIFAR-10 and CIFAR-100 classification when fine-tuning state-of-the-art networks with TWC. Training-with-Confusion on large models provides much larger improvements on CIFAR-100 as compared to CIFAR-10, since introducing confusion is profitable in CIFAR-100 due to presence of classes with small inter-class variations.

| Method          | CIFAR-10 on C10Quick | CIFAR-10 on C10Full | CIFAR-100 on C10Quick |
|-----------------|-----------------------|----------------------|------------------------|
| None            | 100.00 75.54 24.46    | 95.15 81.45 14.65    | 100.00 42.41 57.59     |
| Weight-decay [24] | 100.00 75.61 24.51    | 95.18 81.53 14.73    | 100.00 42.87 57.13     |
| DeCov [5] \(^2\) | 88.78 79.75 8.04      | - - -                | 72.53 45.10 27.43      |
| Dropout [43]    | 99.5 79.41 20.09      | 92.15 82.40 9.81     | 75.00 45.89 29.11      |
| \(L_p\) (Pairwise) | 92.25 80.51 10.74      | 93.88 82.67 11.21    | 72.72 46.72 25.50      |
| \(L_e\) (Entropic) | 91.13 79.81 11.56      | 90.97 81.97 9.02     | 63.02 46.40 16.58      |
| \(L_p\) + Dropout | 93.04 81.13 11.01      | 93.85 83.57 10.28    | 71.15 49.22 21.93      |
| \(L_e\) + Dropout | 91.10 80.41 10.68      | 93.31 83.72 9.61     | 68.04 48.60 19.44      |

Table 3: Image classification performance and train-val gap (\(\Delta\)) for Training-with-Confusion and popular regularization methods.

5 Results

5.1 Fine-Grained Visual Classification

We first describe our results on the six standard FGVC datasets. We find that Training-with-Confusion improves performance across all datasets, with substantial gains in low-performing models. We obtain state-of-the-art results on all six datasets (Table 1-(A-F)).

First, we observe that Training-with-Confusion obtains significant performance gains when fine-tuning from models trained on the ImageNet dataset (e.g., GoogLeNet [45], Resnet-50 [10]), for both forms of the regularization function used. For example, on the CUB-2011 dataset, fine-tuning GoogLeNet without any confusion regularizer gives an accuracy of 68.19%. Fine-tuning with pairwise confusion \(L_p\) achieves 73.65%, and fine-tuning the same model with entropic confusion \(L_e\) gives an accuracy of 74.37%—both significant improvements.

Second, Training-with-Confusion also improves prediction performance for CNN architectures specifically designed for fine-grained visual classification. For instance, confusion improves the performance of the Bilinear CNN [25] on all six datasets and obtains state-of-the-art results. These results demonstrate the utility of the TWC framework for the task of fine-grained visual classification.

Thirdly, it is crucial to note two important aspects of our analysis—we do not compare with ensembling and data augmentation techniques such as Boosted CNNs [31] and Krause et al. [21] since substantial prior evidence indicates that these techniques invariably improve performance, and we evaluate a single-crop, single-model evaluation without any part or object annotations. Additionally, when fine-tuning for FGVC tasks, top image classification models with large number of parameters are known to diverge during training [8] and can observe large oscillations in validation performance during the training process. In contrast, we find that these models converge regularly without oscillations when training with the same learning rate on either form of activation confusion (see Figure 2a) For details on choice of \(\lambda\) used, check Section 6 and the supplement.

\(^1\)Obtained with part annotations.

\(^2\)Due to the lack of publicly available software implementations of DeCov, we are unable to report the performance of DeCov on CIFAR-10 Full.
5.2 Image Classification

We evaluate the performance of Training with Confusion on image classification datasets (CIFAR-10 and CIFAR-100) using several small and large convolutional neural networks. We examine the effect of data augmentation using the scheme followed by Huang et al. [11], denoting the augmented datasets as CIFAR-10+ and CIFAR-100+. The results of this experiment are summarized in Table 2. CIFAR100 has finer category distinction than CIFAR-10, with each “superclass” of 20 containing five finer divisions, and a 100 categories in total. Therefore, we expect TWC to provide stronger gains on CIFAR-100 as compared to CIFAR-10 across models, and our results confirm that. On CIFAR-10, however, we find that the accuracy does not necessarily increase for large models, and sometimes can even decrease due to unwarranted introduction of confusion.

5.3 Comparison with Regularization Methods

We also compare the performance of Training-with-Confusion with commonly used deep-learning regularization methods—weight-decay [24] and Dropout [43]—and recently introduced methods such as DeCov [5]. We experiment with two baseline architectures, “CIFAR10 Quick” and “CIFAR10 Full” on CIFAR10, and “CIFAR10 Quick” on CIFAR100, using their Caffe implementations. For the weight-decay experiment, we use a weight of 0.004 for all layers. Table 3 shows the results of these experiments averaged over 5 trials (please see supplement for a table with standard deviations). Both $L_p$ and $L_e$ obtain better test accuracy than weight-decay and DeCov in all three experiments, without the additional cost of training time. In addition, $L_p$ outperforms Dropout in all experiments and $L_e$ outperforms Dropout in two out of the three experiments. The train-val accuracy gap is also lowered for both $L_p$ and $L_e$, which shows that TWC is effective in preventing overfitting. Finally, we find that a combination of TWC and Dropout has a constructive effect, providing best test accuracy in all three experiments.

6 Analysis

Increase in Feature Generalization: We hypothesize that the introduction of confusion in fine-grained classification is critical to reduce the specificity of features and improve generalization. To evaluate this hypothesis, we perform the eigendecomposition of the covariance matrix (unnormalized PCA) on the penultimate layer features of GoogLeNet trained on CUB-2011, and analyze the trend of sorted eigenvalues (Figure 2b). We examine the features obtained from a network with (i) no fine-tuning (“Basic”), (ii) fine-tuning without confusion (“NoReg”), (iii) fine-tuning with pairwise confusion ($L_p$), and (iv) fine-tuning with entropic confusion ($L_e$).

For a feature matrix with large covariance between the features of different classes, we would expect the first few eigenvalues to be large, and the rest to diminish quickly, since fewer orthogonal components can summarize the data. Conversely, in a completely uncorrelated feature matrix, we would see a larger tail in the decreasing magnitudes of eigenvalues. Figure 2b shows that for the Basic
features (with no fine-tuning), there is a fat tail in both training and test sets due to the presence of a large number of uncorrelated features. After fine-tuning on the training data ("NoReg"), we observe a reduction in the tail of the curve, implying that some generality in features has been introduced in the model through the fine-tuning. The test curve follows a similar decrease, justifying the increase in test accuracy. Finally, for TWC ($\mathcal{L}_p$ and $\mathcal{L}_e$), we observe a substantial decrease in the width of the tail of eigenvalue magnitudes, suggesting a larger increase in generality of features in both training and test sets, which confirms our hypothesis.

**Choice of Parameter $\lambda$:** An integral component of regularization is the choice of weighing parameter. In both our formulations, we observe that the optimization is fairly insensitive to the value of $\lambda$. In our experiments, we observe that the pairwise confusion $\mathcal{L}_p$ is ineffective for $\lambda < 1$, and after our experiments with grid-searching over the hyperparameter value we observe that optimal performance is obtained in the range $0.1N \leq \lambda \leq 0.2N$ ($N$ being the number of classes). For entropic confusion $\mathcal{L}_e$, we find that performance is much more insensitive to the choice of $\lambda$. We describe the variation over a large spectrum of hyperparameter values in Figure 1a, and include experiment-wise details in the supplement.

**Effect on Prediction Probabilities:** For Entropic Confusion, the predicted logit vector is smoother, leading to a higher cross entropy during both training and validation (as also noted by Pereyra et al. [36]). In case of Pairwise Confusion, we also observe a similar effect, although not as pronounced as in the case of the former. Figure 1b shows the average values over all sorted logits over the test set of CUB-2011 to demonstrate this effect.

**t-SNE Visualization:** We also evaluate the 2D t-SNE [28] embeddings to obtain a better understanding of the feature space enforced by Training-with-Confusion. In Figure 3, we examine the embeddings for the CIFAR-10 test set using GoogLeNet, and observe that the class-wise embeddings with confusion have a visually discernible improvement in separation.

**Robustness to Amount of Training Data:** In this experiment, we gradually increase the amount of training data (uniformly sampled) on CUB-2011 and train GoogLeNet with and without TWC. Both forms of TWC provide a consistent improvement in validation set accuracy over the baseline, regardless of the percentage of training data used (Figure 4a).

**Robustness to Label Noise:** In this experiment, we gradually introduce label noise by randomly permuting a fraction of labels for increasing fractions of total data. We follow an identical evaluation protocol as the previous experiment, and observe that TWC is more robust to label noise (Figure 4b).

**Comparison with Pereyra et al. [36]:** The recently published work of Pereyra et al. [36] explores the applicability of regularizing low-entropy outputs in order to introduce generalization, which is similar to our Entropic Confusion formulation ($\mathcal{L}_e$). However, they achieve only marginal gains in the context of image classification on small datasets with large inter-class variation. We extend this...
Figure 3: We visualize 2D t-SNE embeddings for the penultimate layer on GoogLeNet on a randomly selected subset of 10% of the CIFAR10 test set on 3 weights: (left) no confusion, (centre) pairwise confusion and (right) entropic confusion.

Figure 4: Variation of validation performance on CUB-2011 in the presence of (a) fraction of training data and (b) label noise. We observe that both formulations of Training-with-Confusion are effective even in the presence of less training data and label noise.

observation, demonstrating the relative ineffectiveness of $L_e$ on larger models that achieve state-of-the-art performance on the same datasets. We show that $L_e$ is much more useful for fine-grained classification and obtain state-of-the-art results on six standard FGVC datasets. Finally, we provide detailed analysis of the features learnt through TWC and provide experimental evidence to support our initial hypothesis on the benefits of TWC for fine-grained classification tasks.

7 Conclusion

In this work, we introduced two techniques for “Training-with-Confusion” that improve generalizability in fine-grained classification tasks by encouraging confusion in output activations. We performed exhaustive experiments on six major fine-grained visual classification datasets, and improved the state-of-the-art on all of them. Additionally, we displayed significant improvements in fine-tuning performance of a wide class of convolutional architectures for FGVC tasks. Finally, we performed an extensive analysis of our proposed methods and provided experimental evidence in support of our hypothesis for the improvements they provide.

Training-with-Confusion is easy to implement, does not need excessive tuning during training, and does not add any overload during test time. Therefore, our technique should be beneficial to a wide variety of specialized CNN models that are fine-tuned from large scale image classification weights, and even in domains outside of computer vision, in applications that demand for fine-grained classification.

References

[1] Anelia Angelova and Shenghuo Zhu. Efficient object detection and segmentation for fine-grained recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 811–818.
[2] Steve Branson, Grant Van Horn, Serge Belongie, and Pietro Perona. Bird species categorization using pose normalized deep convolutional nets. *arXiv preprint arXiv:1406.2952*, 2014.

[3] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 539–546. IEEE, 2005.

[4] Mircea Cimpoi, Subhransu Maji, and Andrea Vedaldi. Deep filter banks for texture recognition and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3828–3836, 2015.

[5] Michael Cogswell, Faruk Ahmed, Ross Girshick, Larry Zitnick, and Dhruv Batra. Reducing overfitting in deep networks by decorrelating representations. *arXiv preprint arXiv:1511.06068*, 2015.

[6] Yin Cui, Feng Zhou, Yuanqing Lin, and Serge Belongie. Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[8] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. Compact bilinear pooling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 317–326, 2016.

[9] Yves Grandvalet and Yoshua Bengio. Entropy regularization.

[10] 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*, pages 770–778, 2016.

[11] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016.

[12] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015.

[13] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

[14] Herve Jegou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Perez, and Cordelia Schmid. Aggregating local image descriptors into compact codes. *IEEE transactions on pattern analysis and machine intelligence*, 34(9):1704–1716, 2012.

[15] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[16] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM, 2002.

[17] Purushottam Kar, Bharath Sriperumbudur, Prateek Jain, and Harish Karnick. On the generalization ability of online learning algorithms for pairwise loss functions. In *International Conference on Machine Learning*, pages 441–449, 2013.

[18] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs.

[19] Gregory Koch. *Siamese neural networks for one-shot image recognition*. PhD thesis, University of Toronto, 2015.

[20] Jonathan Krause, Hailin Jin, Jianchao Yang, and Li Fei-Fei. Fine-grained recognition without part annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5546–5555, 2015.

[21] Jonathan Krause, Benjamin Sapp, Andrew Howard, Howard Zhou, Alexander Toshev, Tom Duerig, James Philbin, and Li Fei-Fei. The unreasonable effectiveness of noisy data for fine-grained recognition. In *European Conference on Computer Vision*, pages 301–320. Springer, 2016.
[22] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In Proceedings of the IEEE International Conference on Computer Vision Workshops, pages 554–561, 2013.

[23] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset, 2014.

[24] Anders Krogh and John A Hertz. A simple weight decay can improve generalization. In NIPS, volume 4, pages 950–957, 1991.

[25] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In Proceedings of the IEEE International Conference on Computer Vision, pages 1449–1457, 2015.

[26] Maolin Liu, Chengyue Yu, Hefei Ling, and Jie Lei. Hierarchical joint cnn-based models for fine-grained cars recognition. In International Conference on Cloud Computing and Security, pages 337–347. Springer, 2016.

[27] Yuping Luo, Chung-Cheng Chiu, Navdeep Jaitly, and Ilya Sutskever. Learning online alignments with continuous rewards policy gradient. arXiv preprint arXiv:1608.01281, 2016.

[28] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. Journal of Machine Learning Research, 9(Nov):2579–2605, 2008.

[29] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151, 2013.

[30] Mohammad Moghimi, Mohammad Saberian, Jia Li, Shoulong Wang, and Serge Belongie. Boosted convolutional neural networks. In British Machine Vision Conference (BMVC), York, UK, 2016.

[31] Arvind Neelakantan, Luke Vilnis, Quoc V Le, Ilya Sutskever, and Jonathon Shlens. Adding gradient noise improves learning for very deep networks. arXiv preprint arXiv:1511.06807, 2015.

[32] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In Computer Vision, Graphics & Image Processing, 2008. ICVGIP’08. Sixth Indian Conference on, pages 722–729. IEEE, 2008.

[33] Devi Parikh and Kristen Grauman. Relative attributes. In Computer Vision (ICCV), 2011 IEEE International Conference on, pages 503–510. IEEE, 2011.

[34] Adam Paszke and Soumith Chintala. Tensors and Dynamic neural networks in Python with strong GPU acceleration. https://github.com/pytorch. Accessed: [January 1, 2017].

[35] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. arXiv preprint arXiv:1701.06548, 2017.

[36] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. Computer Vision–ECCV 2010, pages 143–156, 2010.

[37] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. arXiv preprint arXiv:1412.6596, 2014.

[38] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.

[39] Kenneth Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. Proceedings of the IEEE, 86(11):2210–2239, 1998.

[40] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, June 2014.

[41] Marcel Simon, Erik Rodner, Yang Gao, Trevor Darrell, and Joachim Denzler. Generalized orderless pooling performs implicit salient matching. arXiv preprint arXiv:1705.00487, 2017.
[43] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[44] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2325–2333, 2016.

[45] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[46] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[47] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 595–604, 2015.

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

[49] Yaming Wang, Jonghyun Choi, Vlad Morariu, and Larry S. Davis. Mining discriminative triplets of patches for fine-grained classification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[50] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2691–2699, 2015.

[51] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *European conference on computer vision*, pages 834–849. Springer, 2014.

[52] Ning Zhang, Ryan Farrell, and Trever Darrell. Pose pooling kernels for sub-category recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3665–3672. IEEE, 2012.

[53] Ning Zhang, Evan Shelhamer, Yang Gao, and Trevor Darrell. Fine-grained pose prediction, normalization, and recognition. *CoRR*, abs/1511.07063, 2015.

[54] Xiaopeng Zhang, Hongkai Xiong, Wengang Zhou, Weiyao Lin, and Qi Tian. Picking deep filter responses for fine-grained image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1134–1142, 2016.

[55] Yu Zhang, Xiu-Shen Wei, Jianxin Wu, Jianfei Cai, Jiangbo Lu, Viet-Anh Nguyen, and Minh N Do. Weakly supervised fine-grained categorization with part-based image representation. *IEEE Transactions on Image Processing*, 25(4):1713–1725, 2016.