Training Set Selection for Image Classification

John Koo

Advised by: David Crandall, Michael Trosset

December 16, 2019

Abstract

Convolutional neural networks (CNNs) are now widely utilized to fit highly accurate image classification models. However, in order to achieve these results, CNNs require vast amounts of training data, especially as the size of these networks grows in an effort to achieve increasingly better performance. In real-world applications, large amounts of training data are often difficult to obtain due to data collection and labelling limitations or difficult to work with due to computational limitations. Expanding upon on previous work by Bambach, Crandall, Smith, and Yu [2], our work explores various methods for subsampling training images under a data budget for fitting an image classification model and compares the results against uniform random sampling. The intuition behind our methods is that we would like to sample a "diverse" training set while controlling for the probability of drawing atypical "outlier" images. While our results fail to outperform random sampling, we demonstrate the effects of fitting CNNs on training subsets drawn with varying degrees of diversity and outlyingness, which are measured using GIST embedding-based distances [18].

1 Introduction

LeNet-5, 1998 [12], was the first widely recognized CNN architecture for image classification. Consisting of only seven layers,three of which are convolutional, training this network on greyscale images of handwritten digits involved fitting 60,000 model parameters. Over time, as larger datasets and more powerful computer hardware became available, CNN architectures grew deeper and more complicated: AlexNet, 2012 [11], consists of 8 layers and 60M parameters, and VGG16, 2015 [15], consists of 41 layers and 138M parameters). Due to the massive number of training parameters, these deep models require large amounts of data to prevent overfitting. In fact, it has been observed that performance gains can continue to scale with the training set size, even into the billions [13].

However, obtaining large datasets for deep models is not always feasible. Manually labelling thousands or even millions of images can be tedious, time-consuming, and expensive [4]. Some proposed and empirically verified solutions to the limited data problem include using a smaller network with fewer parameters, starting with a pre-trained model, and increasing the effective training set size using image augmentation [14]. In this paper, we propose methods for selectively choosing training images under a set data "budget" and discuss how they compare against uniform random sampling.

The bulk of our work involves replicating a previous study [2] using different datasets, checking if our results are consistent with theirs, hypothesizing how these methods work, and expanding upon this work by attempting to come up with new methods based on some of the ideas put forth by the original study. The new methods we propose in this paper fail to outperform uniform random sampling.

2 Previous Work

Bambach et al. [2] demonstrated that given a large pool of images and a fixed training set size, it is possible to tailor the training set for fitting a VGG16 network that results in better or worse performance on a separate test set. They then described two characteristics of the datasets that seemed to correspond to model performance: object size (how much of the image the object takes up) and sample diversity (after embedding a sample of images in $\mathbb{R}^d$, how much space the point cloud takes up). Their study showed that model performance correlated positively with object size in the training set, and training sets consisting of "diverse" images tended to outperform those consisting of "similar" images.

2.1 Replication Study

The data in the above study is as follows:

- Training images were sampled from the frames of first person video feeds, from the point of view of toddlers playing with one of 24 toys. The video was taken with a 70° lens. Bounding boxes of the toys were drawn for each image to determine the size and location of the toy. The images were blurred around the object using the

bounding box information to simulate visual acuity.
- Validation and testing sets consisted of artificially generated images of the 24 toys.

Two experiments were then performed on these data. Both experiments involved fitting VGG16 networks on a particular training set.

The size experiment can be described as follows: Frames were randomly sampled from the video feeds and ranked according to object size (median of around 10%). These were then split into a training set of “big” objects and a training set of “small” objects. It was shown that models fit on the big objects outperformed models fit on the small objects when comparing test accuracies. The images were also cropped into the object to simulate varying focal lengths from the original 70° down to 30° in increments of 10°, and the cropped images outperformed the original images, further supporting this result.

The diversity experiment can be described as follows: Again, frames were randomly sampled from video feeds. These frames were then embedded into Euclidean space using GIST features[1] [18]. Three training subsets were sampled based on the GIST features: a “diverse” subset that maximizes pairwise distances, a “similar” subset that minimizes pairwise distances, and a “random” subset. Models fit on the random subset outperformed the models fit on the diverse subset which outperformed the models fit on the similar subset, using test accuracy to compare models.[2] Images were again cropped to simulate various focal lengths, and lower focal lengths again resulted in better model performance.

We attempted to reproduce[3] these results using the Stanford Dogs dataset[9], which consists of around 20,000 images of 120 dog breeds. Most images contain one dog per image, and images that contain multiple dogs were discarded. For each breed, 100 images were randomly selected for the training set (which were further divided into 50-50 big vs. small and diverse vs. similar training subsets based on the experiment), 25 images were randomly selected for the validation set, and the rest were set aside for testing. The validation set was used to determine when to stop training. No blurring was applied to these images. It is assumed that all images were taken with a 70° lens. Each experiment was repeated 10 times.

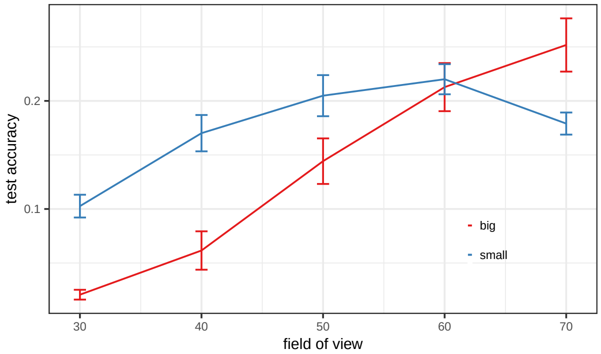

Figure 1: "Size" experiment on the Stanford Dogs dataset. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

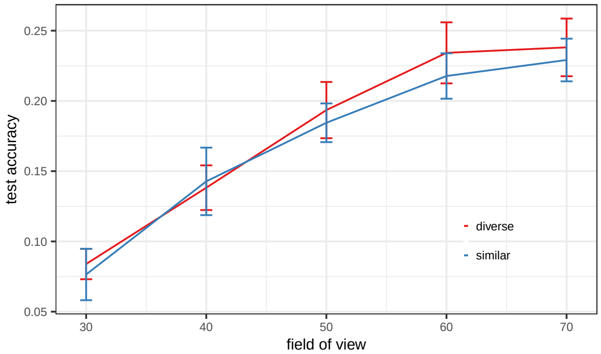

Figure 2: "Diversity" experiment on the Stanford Dogs dataset. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions

Fig. 1 shows that for the original, uncropped images (field of view = 70°), models fit on the “big” subset tend to outperform models fit on the “small” subset. However, instead of increased performance with reduced field of view, we observed the opposite: as the images were cropped into the objects of interest, the resulting models performed worse, and this was especially true of the “big” subset. Closer inspection of the training images reveals that the median bounding box coverage was around 50%, compared to 10% of the toys dataset, and cropping often resulted in cutting off parts of the object of interest (Fig. 3).[4] We can conclude that the optimal training set consists of images in which the objects of interest are prominently featured (i.e., take up most of the image) but are completely contained within the image.

[1] GIST features for our configuration are in $\mathbb{R}^{960}$.

[2] The random subset was twice as large as the diverse and similar subsets.

[3] Models were fit using TensorFlow [1] and Keras [3].

[4] In the paper by Bambach et al. [2], it was similarly shown that in the ImageNet data, on average, the bounding box around the object of interest took up around 50% of the image. The lower object size in the toys dataset may be due to these images taken from first person cameras.

Results of the diversity[5] experiment (Fig. 2) suggest that there may be some differences between the models fitted on "diverse" vs. "similar" training sets at the original focal lengths, but it is not clear whether this is a significant result.

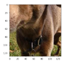

Figure 3: An image from the Stanford Dogs dataset cropped from an assumed 70° field of view to 30°.

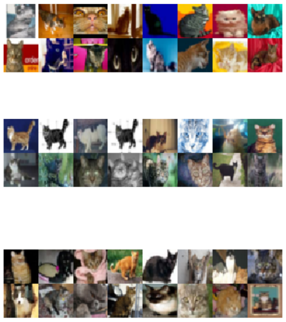

Figure 4: Images of cats from the CIFAR-10 dataset selected by the diverse (top), similar (middle), and random (bottom) sampling methods.

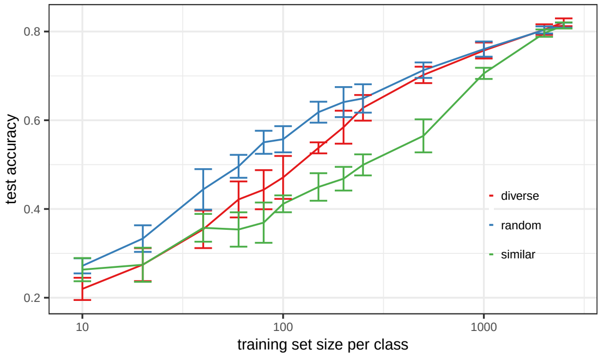

Figure 5: Diversity experiment on CIFAR-10 data. The x-axis is on the log-scale. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

Focusing just on diversity, we tried a similar experiment using the CIFAR-10 [10] dataset, which consists of 10 object classes, each class consisting of 5,000 training images and 1,000 test images. Instead of varying focal lengths (which is not possible with CIFAR-10 images), we varied the training set sizes. For a given training set size n, we sampled $2n$ images from each class and set aside half for validation. The sampling was done according to the "diverse", "similar", and "random" sampling methods (see Fig. 4). Models are compared by their predictive accuracies on the held-out test set. The results show that as n grows, model performance across all three sampling methods converge, which is expected as we approach $n \to$ original training set size. Disappointingly, the random sampling method tends to result in as good or better models than the diverse sampling method (Fig. 5. The similar sampling method tends to result in the worst models of the three. One hypothesis on why the diverse sampling method underperforms is that it tends to select rather unique images that are unlike the others, and the model overfits to those images.

2.2 Other Related Work

Wang et al. [20] proposed a two-round training approach to better fit CNNs on a subset of the training set. In their approach, a CNN was fit on a large training set, then for each image in the training set, an influence measure was computed over all images in a validation set. If the influence was negative, then that training image was discarded. The CNN was then fit again using the reduced training set, and the resulting model was observed to achieve a higher test accuracy than the original model trained prior to subsetting the training data.

Kaushal et al. [8] also proposed an active learning approach to limit the size of training data. Their method is as follows: First, a small subset of a large training set is selected, and a model is fit on the subset. Predictions are then made on the unselected images in the training set based on this model. Images are chosen based on the uncertainty of the model prediction and added to the training subset, and the model is refitted using the larger training subset. This is then repeated over multiple rounds, increasing the size of the training subset each round. The study demonstrates that this method outperforms training on randomly sampled subsets of the same size.

Ferreira [7] proposed a maximum entropy based subset selection method for selecting training data, given that the inputs are of the form $x_i \in \mathbb{R}^d$. Their method starts with a large training set to estimate the density of the feature space.

Wilson [21] demonstrated that edited k-nearest neighbors classifiers outperform regular "unedited" k-nearest neighbors classifiers. The Wilson editing method is described as follows: Given a training set $X_1, ..., X_n \in \mathbb{R}^d$ with corresponding discrete labels $Y_1, ..., Y_n \in \{1, 2, ..., q\}$, use leave-one-out

[5] Diversity was based on pairwise distances of GIST features, which were extracted using a Python package created by Tsuchiya [19].

cross-validated k-nearest neighbors to determine $\hat{Y}_1, ..., \hat{Y}_n$. Discard $i \in \{1, ..., n\}$ where $Y_i \neq \hat{Y}_i$ to construct a reduced, edited training set. Finally, fit a new k-nearest neighbors model on the reduced training set. Wilson's results show that the model fit on the edited data tend to outperform models fit on the entire training set, suggesting that outliers in the training set are detrimental to the resulting model performance.

Based on the literature, training set selection methods can be classified as denoising/filtering methods or as diversification methods. Denoising and filtering methods remove outliers or atypical observations, while diversification methods aim to make training observations as different from each other as possible. These two ideas appear to be at odds with one another. There also doesn't appear to be as much literature on how to apply such methods to image data, as they assume that the data can be naturally represented in Euclidean space (i.e., as a data matrix). However, this does provide some sense of how we can "construct" a good training sample: Given a "good" embedding of images such that the embedding space can be separated by some set of manifolds into regions that correspond to each class, for each class, we should choose a training sample that fills up that class' region (i.e., a "diverse" set) without crossing over into any other regions (i.e., no "outliers").

One thing that is not clear is how we can relate various training subset selection methods to the physical data collection process. Most previous studies sample from a pool of preexisting images, but in a more practical scenario, the data collection process would involve creating new images (e.g., by taking photographs). It is also not clear how we can go from a point in the embedding space (GIST or otherwise) to an actual image.

3 Methods and Results

3.1 Data Editing

We used Wilson editing to remove "outliers" from the training set before subsampling using the diverse sampling method. The random and similar subsamples were drawn in their usual way. The idea behind this method is that Wilson editing will "clean" the training set by remove outliers, and then we will try to construct a diverse training set from the cleaned data—hopefully this will allow us to remove "bad" images while simultaneously spanning as much of the image space as possible. k-nearest neighbors classification was performed using GIST features. We chose $k = 5$ based on cross-validated accuracy[6]. The same VGG16 architecture was fit on these training subsets.

Our results suggest that data editing prior to diverse sampling does improve model performance when training sample sizes are small (Fig. 7). However, randomly sampled training subsets still result in better models (Fig. 5).

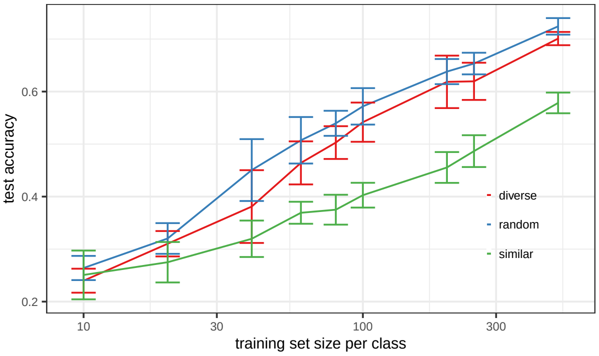

Figure 6: Diversity experiment on CIFAR-10 data, drawing from an edited dataset. The x-axis is on the log-scale. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

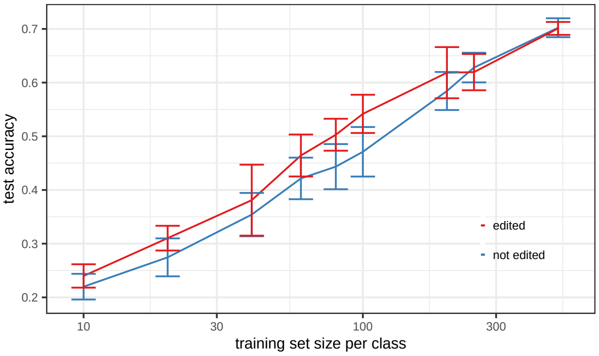

Figure 7: Impact of editing the training data prior to drawing a "diverse" sample on the CIFAR-10 dataset. The x-axis is on the log-scale. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

3.2 Clustering

The motivations behind clustering the training data embeddings are twofold: First, by drawing data uniformly across each cluster, we can draw samples that

[6] To save on computational costs, we used 10-fold cross validation instead of the leave-one-out cross validation prescribed by Wilson. 5-NN achieved approximately 60% accuracy (compared to around 10% accuracy when using just pixels as features).

are relatively dissimilar from each other while reducing the chances of drawing outliers compared to the diverse sampling method. Second, clustering may help us discover subclasses, and drawing balanced samples across these subclasses may result in better model performance (also suggested by Bambach et al. [2]). For each class, we used k-means clustering with $k = 5$ and drew training data maintaining cluster balance for each class. This was compared against random sampling (without using any cluster information). Our results suggest that these training subset selection methods result in equivalent model performance

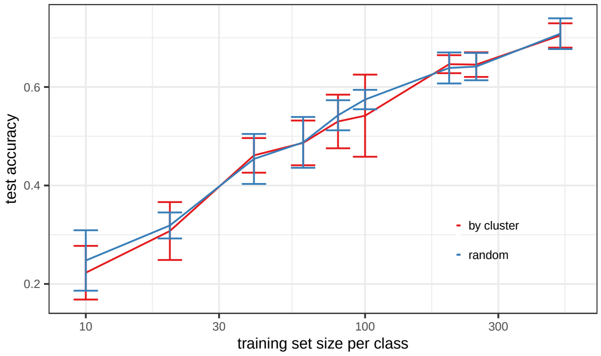

Figure 8: Cluster-based sampling experiment on CIFAR-10. The x-axis is on the log-scale. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

3.3 Transfer Learning

GIST and GIST-based embeddings have been used to fit high-performing image classification models [17] [6]. In our experiments on the CIFAR-10 dataset, we found that a simple 5-nearest neighbors model achieves accuracy that is significantly above that of using pixel-level features despite a threefold reduction in dimensionality[7]. But an alternate embedding may more accurately describe image similarity/dissimilarity for a particular dataset. In this section, we will try using a "best case scenario" embedding short of actually fitting a model to the CIFAR-10 data.

One view of CNNs is as a supervised image embedding algorithm. The second to last layer of an image classification CNN assigns each image a point in $\mathbb{R}^q$ where q is the number of nodes of the layer, then the last layer performs multinomial logistic regression to assign a predicted label to each point in that space. A perfectly accurate CNN model will then produce an embedding such that the classes are linearly separable. Using this as our motivation, we constructed an embedding of CIFAR-10 images based on the Inception V3 [16] network that has been pre-trained[8] on the ImageNet [5] dataset. The Inception V3 embedding was then used to draw training subsamples. This is not a realistic use-case as the CIFAR-10 labels are a subset of the ImageNet labels, but this hopefully serves as an "oracle" embedding[9].

The subsampling methods used here are:

1. Diverse sampling based on maximal interpoint distances
2. Uniform random sampling
3. Wilson editing followed by diverse sampling
4. Stratified by cluster (based on 5-means clustering)
5. Wilson editing followed by cluster-stratified sampling

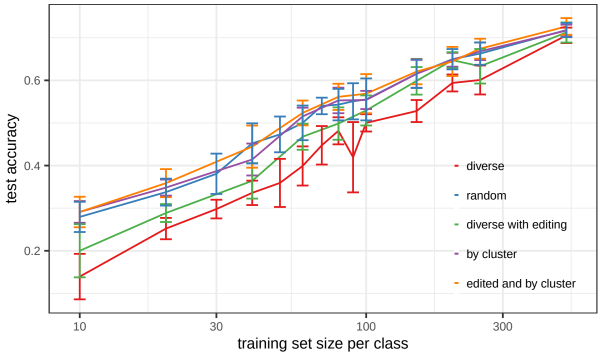

Figure 9: Results from various experiments on drawing training images using the Inception V3 embedding. The x-axis is on the log-scale. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

[7] GIST features are in $\mathbb{R}^{960}$ while pixel features for CIFAR-10 are in $\mathbb{R}^{3072}$.

[8] Pretrained model provided by the Keras [3] package.

[9] A cross-validated 5-nearest neighbors model using this embedding results in approximately 80% accuracy, compared to approximately 60% accuracy using the GIST embedding.

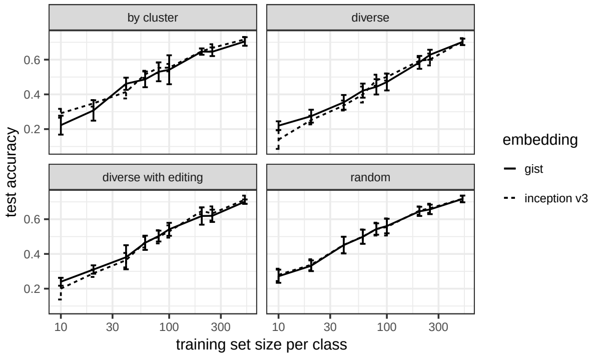

Figure 10: Comparison of sampling CIFAR-10 images using GIST and Inception V3 embeddings. The x-axis is on the log-scale. Errorbars indicate ±1 standard deviation from the mean of 10 repetitions.

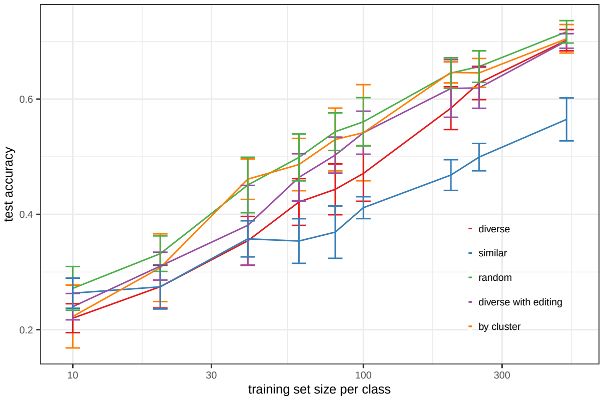

Figure 11: Comparison of the various image sampling methods and the resulting model performance on the CIFAR-10 test set.

The results are largely consistent with those from using GIST embeddings. For a large enough training subsample, the methods converge to the same test accuracy, and for smaller subsamples, random and cluster-based sampling results in equivalent performance while diverse sampling results in reduced performance (Fig. 9). Our results further suggest that there is no significant difference between using the GIST embedding or the Inception V3 embedding for subsampling training data (Fig. 10).

4 Conclusions & Future Work

In our work, we attempted to replicate the results of Bambach et al. [2] on separate datasets. Then we tried to expand upon their methods via data editing and cluster-stratified sampling. These methods were compared against diverse and similar sampling methods based on GIST embeddings as well as uniform random sampling.

We failed to find a sampling method for images that outperforms uniform random sampling. In fact, our attempt at drawing a “diverse” set of images results in reduced performance. We further demonstrated that Wilson data editing on the training set may improve performance and cluster-stratified sampling is on par with random sampling. As expected, “similar” training subsets resulted in significantly poorer performance than the other methods (Fig. 11).

Bambach et al. [2] suggested clustering the embedded images and drawing representative samples from each cluster. Our results show that cluster-straified sampling is not significantly better than random sampling, but we only tried one type of clustering and simply drew cluster-stratified samples. Future work may involve more sophisticated clustering techniques and identifying images that are representative or characteristic of each cluster.

Based on work by Wang et al. [20] and Kaushal et al. [8] as well as our embedding-based approaches, future attempts may involve active learning based methods using iterative CNN-generated embeddings.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL http://tensorflow.org/. Software available from tensorflow.org.

[2] Sven Bambach, David Crandall, Linda Smith, and Chen Yu. Toddler-inspired visual object learning. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 1201–1210. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7396-toddler-inspired-visual-object-learning.pdf.

[3] François Chollet et al. Keras. https://keras.io, 2015.

[4] Per H. Christensen and Wojciech Jarosz. The path to path-traced movies. *Foundations and Trends® in Computer Graphics and Vision*, 10(2):103–175, 2016. ISSN 1572-2759. doi: 10.1561/0600000073. URL http://dx.doi.org/10.1561/0600000073.

[5] 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.

[6] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '09, pages 19:1–19:8, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-480-5. doi: 10.1145/1646396.1646421. URL http://doi.acm.org/10.1145/1646396.1646421.

[7] Pedro M. Ferreira. Unsupervised entropy-based selection of data sets for improved model fitting. *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3330–3337, 2016.

[8] Vishal Kaushal, Rishabh Iyer, Suraj Kothawade, Rohan Mahadev, Khoshrav Doctor, and Ganesh Ramakrishnan. Learning from less data: A unified data subset selection and active learning framework for computer vision, 2019.

[9] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

[10] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1097–1105, USA, 2012. Curran Associates Inc. URL http://dl.acm.org/citation.cfm?id=2999134.2999257.

[12] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[13] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining, 2018.

[14] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning, 2017.

[15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[16] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. URL http://arxiv.org/abs/1409.4842.

[17] A. Torralba, K.P. Murphy, W.T. Freeman, and M.A. Rubin. Context-based vision system for place and object recognition. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 1: 273–280, 13-16 Oct. 2003. doi: 10.1109/ICCV.2003.1238354.

[18] Antonio Torralba. Contextual priming for object detection. *Int. J. Comput. Vision*, 53(2):169–191, July 2003. ISSN 0920-5691. doi: 10.1023/A:1023052124951. URL https://doi.org/10.1023/A:1023052124951.

[19] Yuichiro Tsuchiya. lear-gist-python. https://github.com/tuttieee/lear-gist-python, 2018.

[20] Tianyang Wang, Jun Huan, and Bo Li. Data dropout: Optimizing training data for convolutional neural networks, 2018.

[21] Dennis L. Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Trans. Systems, Man, and Cybernetics*, 2:408–421, 1972.