Correlating Filter Diversity with Convolutional Neural Network Accuracy

Casey A. Graff
School of Computer Science and Engineering
University of California San Diego
La Jolla, CA 92023
Email: cagraff@ucsd.edu

Jeffrey Ellen
School of Computer Science and Engineering
University of California San Diego
La Jolla, CA 92023
Email: jellen@ucsd.edu

***Abstract*—This paper describes three metrics used to asses the filter diversity learned by convolutional neural networks during supervised classification. As our testbed we use four different data sets, including two subsets of ImageNet and two planktonic data sets collected by scientific instruments. We investigate the correlation between our devised metrics and accuracy, using normalization and regularization to alter filter diversity. We propose that these metrics could be used to improve training CNNs. Three potential applications are determining the best preprocessing method for non-standard data sets, diagnosing training efficacy, and predicting performance in cases where validation data is expensive or impossible to collect.**

***Index Terms*—Convolutional Neural Network, regularization, normalization, preprocessing.**

I. Introduction

Convolutional neural networks have been demonstrated to achieve excellent results on a wide variety of supervised learning tasks. Our goal is to develop useful metrics to understand and enhance results with these networks. In particular, we are seeking to improve performance on planktonic images, which are subjectively more 'subtle' than full-color, ImageNet style images. We use four different, balanced data sets to help explore the generality of the metrics that we developed. The data from ImageNet is well documented and frequently used in CNN research; whereas the other two planktonic data sets represent more specialized images.

Our metrics measure the diversity in the weights of a network's first convolutional layer. Since the filters evolve to minimize training loss, we are unable to directly manipulate the variance of filters, we use normalization and L2 regularization as stimuli since they impact the filter diversity indirectly. Our metrics may be useful for a variety of purposes, including diagnosing network performance, identifying over-fitting, and potentially improving weight initialization for large networks.

II. Experimental Design

A. Data Set Description

We constructed four parallel data sets four our investigation. Each constructed dataset contains twenty-one classes of 1,000 images. We re-sized all images, using center-padding and scaling, to 224x224 pixels with three color channels. Nearly every class in the parent data sets so the 1,000 were selected randomly, as long as they were also larger than 224x224.

For each dataset we created twenty target classes plus one "other" class that contained samples of a variety of other logical classes that occurred infrequently. Our zooplankton parent datasets include this construct, but the ILSVRC dataset does not, so a similar class was artificially generated for the ILSVRC (All) and ILSVRC (Dog) sets to be consistent.

1) ImageNet (All): The second dataset comes from the ImageNet Large Scale Visual Recognition Competition (ILSVRC) 2015 data set [1] for the object localization challenge. Specifically, from the 1,000 classes (called synsets) used in the challenge, twenty classes were randomly selected. For the "other" class, an uneven distribution of fifty other synsets was used to simulate an approximate equivalent of the other category found in the plankton dataset.

2) ImageNet (Dogs): The third dataset comes from the same ILSVRC dataset and is comprised of twenty hand-picked synsets that are closely related. The intent is to construct a data set that mirrors the consistent visual similarity between classes that is present in the plankton data sets. In this case, all of the classes used were dog breeds. For this dataset, the "other" class contained uneven distribution of fifty other dog synsets.

3) Zooplankton: Our zooplankton images are acquired by a technology called Zooscan[2]. In essence, this is a extremely fine-tuned monochromatic flatbed scanner which is used on preserved samples. Example ZooScan images are shown in (Fig. 1).

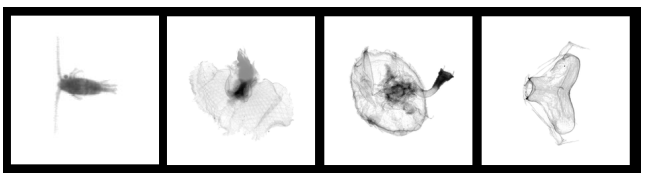

Fig. 1. Example ZooScan images: a copepod, jelly, pteropod, and siphonophore, all preserved and in unnatural postures and various states of completeness.

As shown, the background of these images is white. Prior

Contribution from the National Science Foundationsupported California Current Ecosystem Long Term Ecological Research site. Plankton sample analysis supported by NSF grants to Mark D. Ohman (mohman@ucsd.edu), and by the SIO Pelagic Invertebrate Collection.

to any normalization, the images in the plankton data set were centered by calculating the pixel value center of mass and shifting each sample to place this in the center of the image. This improved plankton validation and testing accuracy across all normalization techniques.

4) Phytoplankton: Our phytoplankton images are acquired by a technology called an Imaging FlowCytobot[3]. This technology images live cells $<$10 micrometers through use of a focused laser. Phytoplankton images are selected from [4], in a manner consistent with 20 of the major classes identified in [3]. As shown, the background of these images is noisier than the zooplankton images which results in the introduction of an artificial edge, not present in the Zooplankton data set, between the original background and the center-padding.

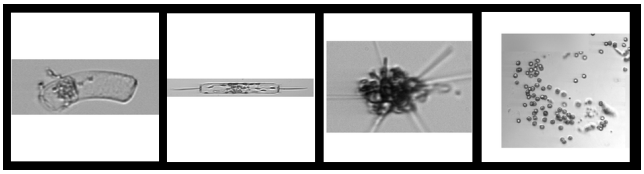

Fig. 2. Example FlowCytobot images: three diatoms (*Guinardia Striata*, *Ditylum* and *Asterionellopsis*, and Flagellate *Phaeocystis*, imaged alive and fully in tact.

B. Architecture

The original network architecture was based on the VGG-11 network architecture described in [5]. However, initial trials and investigation revealed that this architecture vastly over-fit to the training data. This was observed through analysis of validation loss curves, visualization of under-utilized input filters, and empirical testing of smaller network sizes. The final network selected has similar accuracy to the larger network on the datasets, while containing substantially fewer layers and parameters.

The final network architecture selected is as follows:

Input (224x224 RGB image) $\rightarrow$ Conv3-16 $\rightarrow$ MaxPool2 $\rightarrow$ Conv3-32 $\rightarrow$ MaxPool4 $\rightarrow$ Conv3-64 $\rightarrow$ MaxPool-4 $\rightarrow$ FC-1024 $\rightarrow$ FC-21 $\rightarrow$ Softmax.

All convolutional layers listed as "Conv$\langle$receptive field size$\rangle$-$\langle$number of channels$\rangle$" use a stride and padding of one. All fully-connected layers listed as "FC-$\langle$number of nodes$\rangle$" use .50 dropout rate (except for the final fully-connected layer). Max pooling layers are listed as "MaxPool-$\langle$pool size$\rangle$".

During our investigation several training batch sizes were compared. Initially a batch size of 50 was used with the larger network architecture. Lower batch sizes could only be used at the expense of additional training time. After reducing the size of the network substantially, it was found that the batch size could be reduced, yielding improved training accuracy with a negligible increase in training time.

We believe our implementation, although it uses a smaller amount of data and a smaller network than ImageNet provides a roughly equivalent testbed since our implementation provides similar accuracy to support vector machines as reported for both the zooplankton [6] and phytoplankton data [3]. Using less images facilitates more trials.

TABLE I
ACCURACY PER NORMALIZATION TECHNIQUE
(AVERAGED ACROSS ALL TRIALS)

Dataset	Per Image	Per Pixel	ZCA
ImageNet(All)	56.85	63.21	58.26
ImageNet(Dogs)	30.44	36.11	32.53
Zooplankton	71.06	69.18	65.68
Phytoplankton	79.25	79.70	73.93

III. NORMALIZATION INVESTIGATION

Pixel values need to be normalized before used as input to a Convolutional Neural Network. For images where each pixel value is considered to be a feature and not independent from its neighbors, there are many strategies to normalize the input. One option is to normalize all the values within a particular image, this per image normalization is frequently referred to as "Global Contrast Normalization". Another strategy is to normalize each pixel location across the whole stack of images separately, this per pixel normalization is frequently referred to as "Standardization". Another option is to decorrelate features and normalize their variance, whitening and Zero-phase Component Analysis, which is frequently called "ZCA whitening". ZCA whitening is commonly used for images. A fourth option is to normalize pixel values across patches of a single input, rather than the whole set of features, and this is referred to as "Local Contrast Normalization"[7]. We implemented all of these, and we found that per image normalization and per pixel normalization worked the best on our data.

A. Normalization Results

Normalization was applied by first separating the data into a training (80%) and testing (20%) set. Five of these splits were generated for each of the datasets. Once separated, the normalization parameters were fit on the training portion of the split, then applied to the entire split. We applied each of our normalization techniques separately to the three color channels.

$$Loss = E_{train}(W) + \lambda W^2 \tag{1}$$

We used L2 regularization which applies a weight λ to the squared values of the network's weights W and adds it the training error E_{train} to compute the loss value that is used to update the network. For each L2 regularization weight examined five trials were conducted (each using a unique split) for each dataset and normalization pair. As shown in Table I, per pixel normalization worked best for the color images, and per image normalization worked best for zooplankton, with the phytoplankton exhibiting no significant difference between the three types of normalization.

Note that the phytoplankton images, with a moderately noisy background, show very little difference per normalization method, and therefore our dataset may not be diverse

enough or have a sufficient quantity of training images for the normalization strategy to matter.

To investigate further why zooplankton normalization was different, we examined a confusion matrix of the results of the zooplankton data to ascertain why per image normalization worked better than per pixel normalization. Our data shown use a particular split (so the images are exactly the same for each normalization strategy). The results are shown in (Fig. 3).

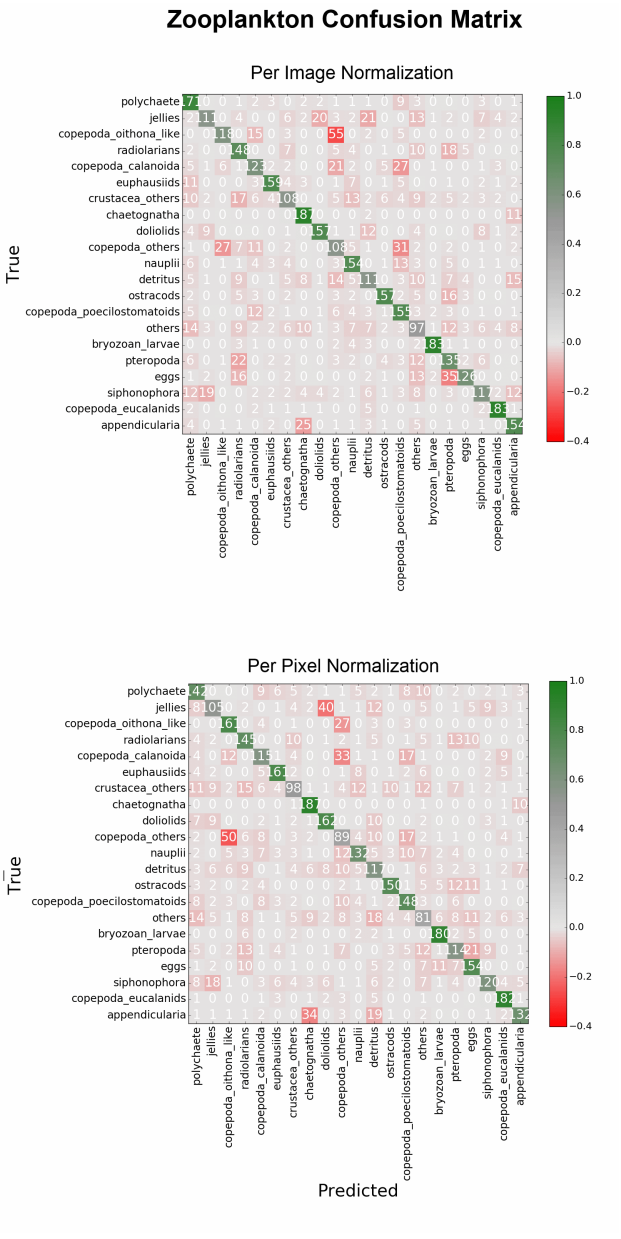

Fig. 3. Histogram illustrating the difference in distribution of filter diversity between normalization techniques.

The three zooplankton classes which improve the most with per image normalization are polychaetes, nauplii, and pteropoda; three classes which have delicate, feathery appendages as their most distinguishing feature between them and their closest neighbor in shape (chaetognath, copepods, and eggs respectively). This property intuitively suggests that a more 'subtle' normalization technique would yield greater accuracy, such as per image normalization. This intuition is supported by the results presented in the confusion matrices. For the purpose of more quantitatively determining the best normalization technique (besides simply examining accuracy) we introduce metrics to quantify filter diversity.

IV. Filter Variance

For the rest of this paper, we will refer to the sets of weights from the first layer of the network as filters. These filters serve as the lowest level detectors, and they often evolve to respond highly to changes in intensity, such as edges.

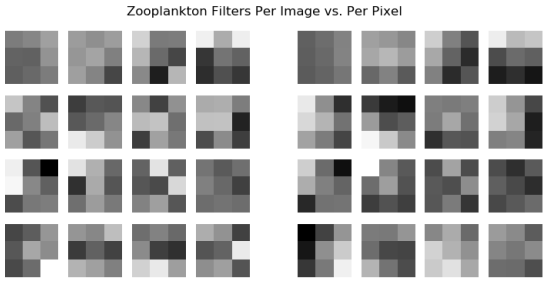

Fig. 4. Normalized Examples of top performing Zooplankton filters; per image normalization on the left, per pixel on the right.

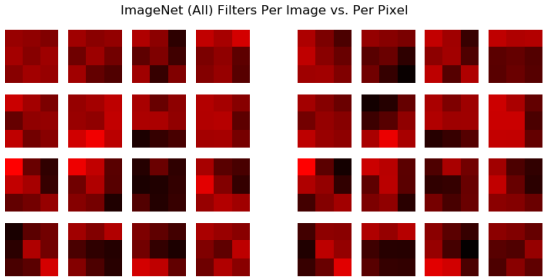

Fig. 5. Normalized Examples of top performing ImageNet (All) filters (red channel only); per image normalization on the left, per pixel normalization on the right.

The pairs of filters presented in (Fig. 4) and (Fig. 5) are similar because they are trained on the same split and ordering of the images. The values were transferred to the $0-255$ range for visualization for each image separately, so generalizations about intensities between images are invalid, however, they illustrate not only the nature of the filters, but also the relative difference between corresponding pairs of filters in the per image vs per pixel normalization strategies.

First we consider the variance within the weights of an individual filter. We hypothesize that this will correlate to how sharply adjacent features vary within a particular image. Second we consider the variance within the weights of a particular set of filters built by a single model. We hypothesize

that this will correlate to how much feature values vary within all regions of all images in a particular dataset. Third we consider the variance between individual filters within a model. We hypothesize that this will also correspond to the variety of the values of features within a particular data set.

All filters are generated for an individual channel. The ImageNet images are standard RGB images, and the plankton data sets are single-channel due to their acquisition mechanisms, so in order to use the same number of parameters per network, we copy their input across all three channels so all networks are operating on 3-channel 224x224 images. While each channel in the ImageNet is marginally different from the other two, overall the pattern of our results holds and for simplicity we present all ImageNet results as the average across all three color channels rather than R, G, and B separately.

A. Variance within Individual Filters

First, we consider the variance within individual filters. Each of our filters has 9 weights (3x3) and we use the standard deviation of these 9 values as a measure of the variance within the filter. Since we have 16 filters learned per model trained, we take a simple arithmetic mean of these values to provide a single number reflecting the variance learned by that particular model, $\overline{\sigma}_F$, which is shown in (Eq. 2)[1], where x_i is an individual weight for filter f.

$$\overline{\sigma}_F = \frac{1}{16}\sum_{f=0}^{16}\sqrt{\frac{1}{8}\sum_{i=1}^{9}(x_{f,i} - \overline{x})^2} \tag{2}$$

Our results are shown in (Fig. 6).

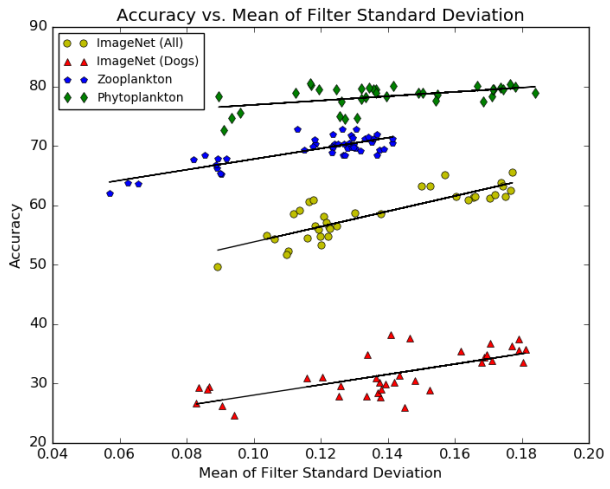

Fig. 6. The regression lines indicate a positive correlation between accuracy and $\overline{\sigma}_F$ across all datasets.

Each datum in (Fig. 6) corresponds to a single model trained as described in our Normalization Experimentation section. Also shown are regression lines for each set of data points. Correlation results are in (Table II), and the last column, Combined PCC, refers to the regression lines in (Fig. 6).

[1]Given our small sample, we use Bessel's correction when calculating our standard deviation

TABLE II
PEARSON CORRELATION COEFFICIENT OF ACCURACY VS $\overline{\sigma}_F$ FOR EACH NORMALIZATION

Dataset	Per Image PCC	Per Pixel PCC	Combined PCC
ImageNet(All)	0.717	0.581	0.844
ImageNet(Dogs)	0.512	0.742	0.700
Zooplankton	0.809	0.874	0.824
Phytoplankton	0.669	0.687	0.491

Given that some of the trials had different data splits, and the noise present in the learning process, we do not expect a strict tolerance in the results, so we interpret values above 0.8 to indicate a very strong correlation, and values above 0.6 to indicate a strong correlation between accuracy and $\overline{\sigma}_F$. Table II also shows that the correlation holds whether considering across both normalizations, as pictured in (Fig. 6) or considering the effects of a single normalization strategy.

If there is a causal relationship between $\overline{\sigma}_F$ and accuracy, then to maximize accuracy, we should try to intentionally increase $\overline{\sigma}_F$.

The data points in (Fig. 6) are on a fixed size network for a particular data set. Many network hyperparameters were held constant, including the initialization of the weights. The only three things creating variation are the type of normalization, the amount of regularization, and the split of the data.

The effect of regularization on $\overline{\sigma}_F$ is straightforward. The filters are the solution to an optimization problem of responding most strongly to the image patches that are most diagnostic of discriminating between classes. An individual filter having a higher $\overline{\sigma}_F$ means that its individual weights are more spread out. Since regularization is designed to reduce the magnitude of the weights, any filter weights with a high standard deviation must be very rewarding to avoid being regularized. This relationship is evident in (Fig. 7).

Individual data points represent separate trials with the same parameters on different splits of the data, and the lines drawn connect the average values of each data set. The bimodal distribution is due to the two different types of normalization. As the regularization increases, $\overline{\sigma}_F$ decreases. Since there is a high correlation between accuracy and $\overline{\sigma}_F$, the same relationship exists between accuracy vs. regularization as shown in (Fig. 8).

Again, the trials are the individual data points and the lines connect the averages. The lines of (Fig. 8) appear to be more flat than the lines in (Fig. 7), but this is because of the scale of the y-axis. But both graphs peak in similar places, as we would expect with them being highly correlated. The shape of this graph is well known, and why the optimal amount of regularization is sought via a search. But our investigation provides insight into the mechanism for this behavior.

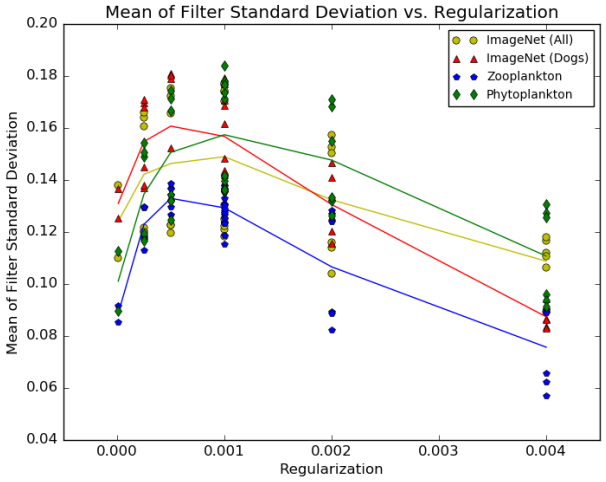

Fig. 7. The relationship between $\overline{\sigma}_F$ and regularization across all trials for all datasets.

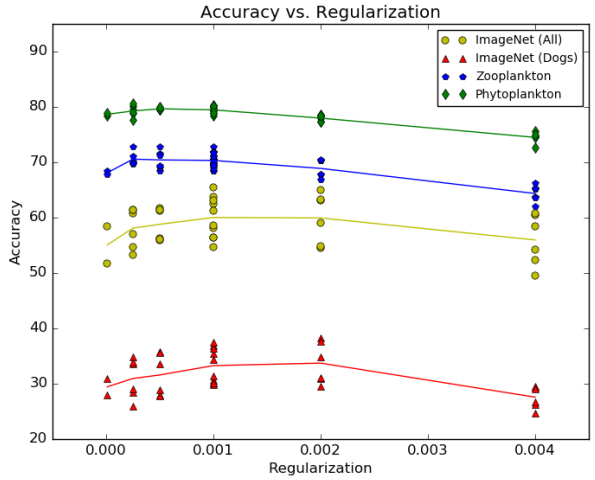

Fig. 8. The relationship between accuracy and regularization across trials for all datasets.

B. Variance Between Filters

As another metric we quantify the distribution of the feature values within a particular image on the weights by calculating the model's global filter standard deviation, specifically the standard deviation of all 144 weights in the first layer of the matrix as shown in (Equation 3).

$$\sigma_{\forall F} = \sqrt{\frac{1}{143}\sum_{f=0}^{16}\sum_{i=1}^{9}(x_{f,i} - \overline{x})^2} \tag{3}$$

We also want to investigate the variance between filters within an individual model. Since the $3x3$ weights comprising our filters in our convolutional neural network are always applied in the same orientation, simple matrix subtraction is appropriate, and will function similar to a Hamming Distance, roughly describing how far apart the two filters are. We calculate this distance in (Equation 4).

$$\overline{\Delta}_F = \frac{\sum_{f=0}^{16}\sum_{g=f}^{16}\sum_{i=1}^{9}|x_{f,i} - x_{g,i}|}{{}_{16}P_2} \tag{4}$$

We then calculate these metrics for our data, as shown in (Fig. 9).

Fig. 9. Bar graph demonstrating increased diversity when using per pixel normalization.

(Fig 9) shows all three metrics, $\overline{\sigma}_F$, $\sigma_{\forall F}$, and $\overline{\Delta}_F$ averaged across all trials, including all regularization strengths. In 11 out of 12 cases, per pixel normalization yields higher values for filter diversity than per image normalization. And this is not just an artifact of considering each image set individually, but also occurs when considering the trials in aggregate as shown in the histogram (Fig 10).

Given the diversity of our image types, we feel this would hold for any types of images. The second pattern is that for both $\sigma_{\forall F}$ and $\overline{\Delta}_F$, the values are larger for the two ImageNet data sets than for the two planktonic data sets. This supports an intuition that the two planktonic data sets are more 'subtle' than the ImageNet data sets. This explains why they their better overall higher accuracy, they have lower filter diversity in two metrics.

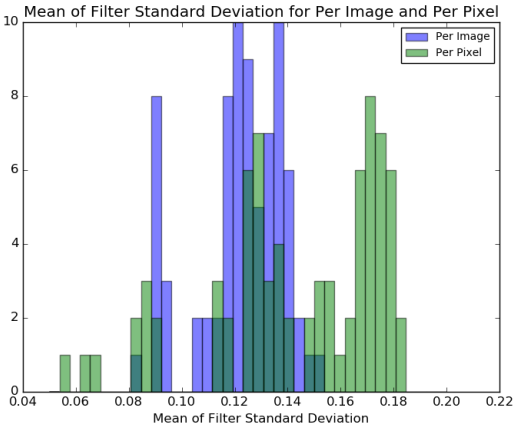

Fig. 10. Histogram illustrating the difference in distribution of filter diversity between normalization techniques.

In summary, regularization strength is known to be a hyperparamter. In (Fig. 8), we see that for each of our data sets, the maximum accuracy occurs when the regularization strengths is around 0.001. (Fig. 7), the relationship between regularization and $\overline{\sigma}_F$ shows that the inflection point for $\overline{\sigma}_F$ is at nearly the same regularization strength around 0.001 for our 4 data sets. It is our hypothesis that this correlation could be used as a gauge for network training: rather than using a holdout set to assess accuracy, the $\overline{\sigma}_F$ of the filters can be examined, and hyperparameter optimization can stop when $\overline{\sigma}_F$ has achieved a local maxium.

If our metric could help reduce the size of the validation/holdout set, then those images could be used for training instead. There is a class of supervised classification problems, particularly in the scientific domain (such as medical imaging), for which many thousands or millions of training examples would be difficult or impossible to obtain. While strategies such as leave-one-out cross validation attempt to overcome this lack of data, they require as many models to be computed as folds of the data. This potentially makes grid search and other operations prohibitively expensive. Instead, we propose that filter diversity could potentially be used to assess the best performing model. More investigation on larger and diverse data sets would be required to fully verify this claim.

V. Implications of Filter Variance on Classification Accuracy

As shown in (Table I), normalization strategy has a clear impact on accuracy. We found that one method of normalization generally outperformed the other regardless of regularization strength and other experiments not included for succinctness, such as network size. The superficial conclusion is therefore that one type of data is better suited to a particular normalization strategy than another. Our investigation sheds light on why this is the case. Figure 9 isolates the effect of normalization on all three of our metrics.

In every case where per pixel normalization results in higher $\overline{\sigma}_F$ than per image normalization, per pixel normalization also results in the highest accuracy. Agreeably, zooplankton is the one data set for which per image normalization had higher $\overline{\sigma}_F$ than per pixel normalization, and it correspondingly achieves better accuracy using per image normalization. Therefore, $\overline{\sigma}_F$ is correlated with accuracy across a variety of conditions.

We also propose that knowledge of this metric could be used to assist in training large networks. Our network, along with many others, uses uniform He initialization[8]. This initialization strategy, along with many others, is designed to speed the convergence of the network. We propose a modified strategy that generates a number of different candidate initializations, calculates the $\overline{\sigma}_F$ for each one, and selects the one with the highest $\overline{\sigma}_F$ as the best candidate. This one-time calculation would trivially add to the network run time. As our network was relatively shallow, and our convergence times fast, we did not assess this with our data.

VI. Conclusion

In this paper, we described three metrics used to asses the filter diversity: $\overline{\sigma}_F$, $\sigma_{\forall F}$, and $\overline{\Delta}_F$. These metrics are intended to measure the diversity within a single filter, as well as across all filters. For all four of our data sets, we found a strong correlation between our devised metrics and accuracy. We feel that these metrics could potentially be used in a variety of ways to improve training models as well as determining the best model for deployment. This include identifying the best preprocessing method for non-standard data sets, potentially improving convergence time, and predicting performance in cases where validation data is valuable because it is expensive or impossible to collect.

Acknowledgment

The authors would like to thank Professor Mark Ohman and Professor Charles Elkan for their support.

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[2] P. Grosjean, M. Picheral, C. Warembourg, and G. Gorsky, "Enumeration, measurement, and identification of net zooplankton samples using the zooscan digital imaging system," *ICES Journal of Marine Science: Journal du Conseil*, vol. 61, no. 4, pp. 518–525, 2004.

[3] H. M. Sosik and R. J. Olson, "Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry," *Limnology and Oceanography: Methods*, vol. 5, no. 6, pp. 204–216, 2007.

[4] H. M. Sosik, E. E. Peacock, and E. F. Brownlee, "Annotated plankton images - data set for developing and evaluating classification methods." [Online]. Available: http://dx.doi.org/10.1575/1912/7341

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[6] J. Ellen, H. Li, and M. D. Ohman, "Quantifying california current plankton samples with efficient machine learning techniques," in *OCEANS 2015 - MTS/IEEE Washington*, Oct 2015, pp. 1–9.

[7] Y. LeCun and M. Ranzato, "Deep learning tutorial," in *Tutorials in International Conference on Machine Learning (ICML13)*. Citeseer, 2013.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *CoRR*, vol. abs/1502.01852, 2015.