A Deconvolutional Strategy for Implementing Large Patch Sizes Supports Improved Image Classification

Xinhua Zhang
University of New Mexico
xinhua@unm.edu

Garrett Kenyon
Los Alamos National Laboratory
garkenyon@gmail.com

ABSTRACT

Sparse coding is a widely-used technique for learning an overcomplete basis set from unlabeled image data. We hypothesize that as the size of the image patch spanned by each basis vector increases, the resulting dictionary should encompass a broader range of spatial scales, including more features that better discriminate between object classes. Previous efforts to measure the effects of patch size on image classification performance were confounded by the difficulty of maintaining a given level of overcompleteness as the patch size is increased. Here, we employ a type of deconvolutional network in which overcompleteness is independent of patch size. Based on image classification results on the CIFAR10 database, we find that optimizing our deconvolutional network for sparse reconstruction leads to improved classification performance as a function of the number of training epochs. Different from previous reports, we find that enforcing a certain degree of sparsity improves classification performance. We also find that classification performance improves as both the number of learned features (dictionary size) and the size of the image patch spanned by each feature (patch size) are increased, ultimately the best published results for sparse autoencoders.

Keywords

Sparse coding, unsupervised feature learning, whole image classification

1. INTRODUCTION

Unsupervised feature learning plays an important role in computer vision, as the supply of labeled images is limited. Sparse coding with an over-complete basis set provides one approach to unsupervised feature learning [8]. It has been shown that increasing the the degree of overcompleteness of a dictionary results in better image classification performance, even for a dictionary trained in an unsupervised manner for optimal sparse reconstruction[2]. Presumably, the greater the number of elements in the dictionary, the more likely some of the those elements will be discriminative between different image classes. It has been much more difficult, however, to study how the size of the image patch spanned by each dictionary element (patch size) affects performance on the same tasks. This difficulty arises because for many sparse coding methods, the overcompleteness of a basis set is inversely proportional to the number of pixels, or area, comprising the image patch spanned by each dictionary element. In most implementations, the number of dictionary elements needed to achieve a given level of overcompleteness grows as the square of the length of the patch spanned by each feature. As a result, in most implementations the size of the image patch spanned by each dictionary element is typically much smaller than the image as a whole, since learning an overcomplete basis set consisting of dictionary elements with very large patch sizes becomes computationally prohibitive.

In this paper, we get around the computational constraints on patch size by using a deconvolutional approach based on replicating kernels to reduce the dimensionality of the learned feature maps such that the degree of overcompleteness is independent of patch size [12, 13]. As a result, we are able to explore how patch size affects image classification performance with the degree of overcompleteness held fixed. The main conclusions of our present study are: 1) patch size strongly influences the types of features that can be learned; 2) non-Gabor-like features, which emerge most often in dictionaries employing large patch sizes, are more likely to be discriminative between different image classes; 3) by using replicating kernels, we are able to achieve the state-of-the-art classification performance on the CIFAR10 [6] dataset using relatively small feature maps.

1.1 Related Work

Several works have examined the properties of over-complete basis sets and their application to image classification tasks.

Olshausen et al. [7] found that the degree of over-completeness influences the diversity of basis functions. In particular, more interesting basis functions emerge, such as blob- and ridge-like features, as the overcompleteness of the dictionary increases. Compared to Gabor-like features, these more interesting basis functions are likely to be more specifically matched to particular features that are present in natural images. In fact, the results presented here suggest that such features are indeed more discriminative between different image categories.

BICT 2015, December 03-05, New York City, United States

DOI 10.4108/eai.3-12-2015.2262498

Coates et al. [2] showed that increasing the number of basis functions improves image classification performance. Increasing the number of basis functions is equivalent to increasing the degree of over-completeness as long as the patch size is held fixed. Also, they showed that if the number of basis functions is fixed, increasing the patch size results in a performance decrease. Increases in patch size with the number of basis functions held fixed is equivalent to decreasing the degree of over-completeness. These results suggest that over-completeness is directly related to classification performance.

Rigamonti et al. [9] explored the relationship between image classification performance and the degree of sparsity. On one hand, sparsity is not helpful during classification, i.e. as the degree of sparsity increases, classification performance drops significantly. On the other hand, sparsity is helpful in the basis learning phase. Our results indicate that the relationship between sparsity and classification performance is described by an inverted U-shape: classification performance improves as sparsity is reduced both during training and testing up to a minimum value, below which classification performance sharply degrades.

Zeiler et al. [13] used a deconvolutional approach to encode the whole image using a relatively small number of learned basis vectors replicated with a stride of 1. In their deconvolutional approach, over-completeness, defined as the total size of the dictionary divided by the number of pixels in the image, only depends on the number of basis functions in a single replica and is independent of patch size.

The main contribution of this paper is to exploit the ability of deconvolutional neural networks to break the dependence of overcompleteness on patch sizes, and thus directly explore the relationship between over-completeness, sparsity and patch sizes in the acquisition of more discriminative features and thus improved image classification performance.

2. ALGORITHMS AND METHODS

2.1 Notation

To improve clarity of notation without loss of generality, the following description is limited to real-valued gray-scale images, as the generalization to color images is straightforward. We use $Ub(x)$ to stand for the upper-bound over-completeness of dictionary x and $Lb(x)$ to stand for the lower-bound over-completeness of dictionary x.

2.2 Convolutional and Deconvolutional Networks

Unsupervised feature learning is a process of minimizing an energy function. Given an image I, which is a $N \times M$ matrix, sparse coding (SC) minimizes the following energy function[8]:

$$E = \sum_{j=1}^{J} \left[\frac{1}{2} ||\hat{I}_j - \Phi^T a_j||_2^2 + \lambda ||a_j||_1 \right] \quad (1)$$

where $\hat{I}_j$ is a $H \times H$ image patch, j denotes the image patch, J is the total number of patches, $\Phi = [\phi_1, \phi_2, \cdots, \phi_K]^T$ is a feature vector, ϕ is a $H \times H$ feature patch, a_j is a $K \times 1$ coefficient vector and λ is a parameter that balances reconstruction error and sparsity. A typical way to apply Eq. 1 to an image is the convolution with Φ. Instead of encoding a single image patch, deconvolutional networks (DN) compute the $N' \times M' \times K$ coefficient matrix z, i.e. feature maps, for the whole image simultaneously [13].

$$E = \frac{1}{2} ||I - \sum_{j=1}^{J} \Phi^T * z_j||_2^2 + \lambda \sum_{j=1}^{J} ||z_j||_1 \quad (2)$$

where $*$ is the convolution operation and $J = N'M'$. If the stride is σ then $N' = \frac{1}{\sigma}N, M' = \frac{1}{\sigma}M$.

2.3 Dictionary Overcompleteness

Achieving a desired level of over-completeness is a major computational obstacle when learning a dictionary consisting of elements with large patch sizes. The fact that patches are overlapped is not considered in Eq. 1. Hence the over-completeness of the basis set is inversely proportional to the size of a feature patch, i.e. $\frac{K}{H^2}$. Eq. 2 adds competition between overlapped patches. In this case, $Lb(\Phi) = \frac{K}{H^2}$ and $Ub(\Phi) = \frac{N'M'K}{NM}$. If the stride between patches equals the patch size, Eq. 2 dealing with non-overlapped patches is equivalent to Eq. 1 and the over-completeness is $\frac{K}{H^2}$.

The locally competitive algorithm (LCA) is a neurally plausible sparse solver that minimizes Eq. 1 [11]. In this paper, we use a modified LCA, which uses a replicating kernels method (RKM) with a soft-threshold [12], to minimize Eq. 2. RKM sets restrictions on both the stride and the dimension of the feature maps. In this case, $Ub(\Phi) = \frac{K}{\sigma^2}$, the overcompleteness is no longer affected by the patch size.

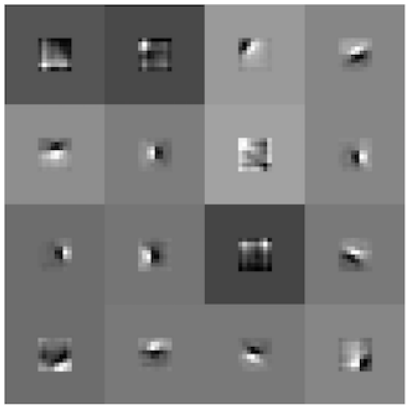
2.4 Image Classification

Many factors can affect image classification accuracy when using RKM. We design several experiments to identify the relationships between these factors and classification accuracy. We use the original color images in the CIFAR10 dataset. All our dictionaries are trained in an unsupervised manner to minimize sparse reconstruction error on the training images with $\lambda = 0.025$. During classification, we first encode both the training and testing set images using a learned dictionary with $\lambda = 0.005$. Then we apply average-pooling on 5×5 patches with a stride of 1 [5]. We use LIBLINEAR (L2-regularized, L2-loss support vector classifier) for classification [10]. All simulations use the PetaVision open-course, high-performance neural simulation toolbox [1].

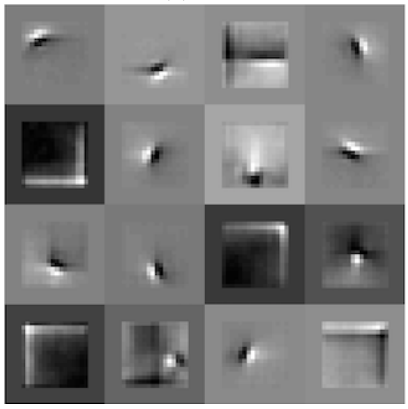
3. EXPERIMENTS AND RESULTS

3.1 Patch Size and Discriminative Feature

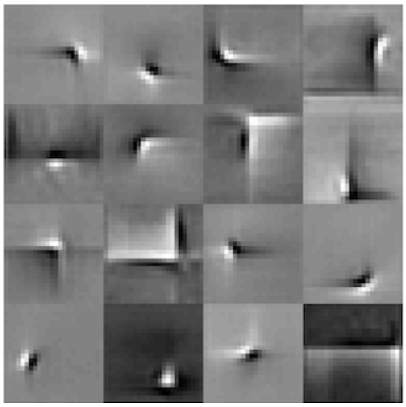
Learning discriminative features is important to object recognition. Since Olshausen et al. [7] pointed out that the over-completeness can affect the diversity of the learned basis functions, and in order to show that large patch size is a factor that influences the learning of discriminative features, we control for the effect of over-completeness, training three $Ub(\Phi) = 2$ over-complete dictionaries with different patch sizes. We apply rectified RKM, i.e. only preserving positive values, to gray CIFAR10 training set images. The lower-bound and upper-bound are divided by 2 because rectification effectively doubles the number of basis elements needed to achieve a given level of over-completeness, i.e. $Ub(\Phi) =$



(a) 8×8



(b) 16×16



(c) 24×24

Figure 1: Three dictionaries with different patch sizes. Small patches are zero-padded to 24×24.

$\frac{K}{2\sigma^2}, Lb(\Phi) = \frac{K}{2H^2}$. We use $\lambda = 0.025, \sigma = 2, K = 16$. Table 1 shows the degree of over-completeness for each set of parameters.

The three dictionaries are shown in Fig. 1. Although each dictionary has the same overcompleteness, their features are obviously different. For the dictionary whose elements have the smallest patches, individual features are mainly Gabor-like. As the patch size increases, the diversity of features increases. Some non-Gabor-like functions, such as corner-like and curve-like functions, begin to emerge. With small patch sizes, features are located mainly in the center of the patch, whereas with big patch sizes, features appear in different areas within the patch (note that all features are replicated with a given stride so that location within a patch is often not important). Finally, and most obviously, the features in a dictionary with small patch sizes are limited in size. With large patch sizes, however, the range of feature sizes varies more widely. In other words, large patch sizes allow the dictionary to learn features that best fit the image statistics regardless of size.

Table 1: Dictionary over-completeness

Dictionary	H	Lb	Ub
Φ_1	8	0.125	2
Φ_2	16	0.03125	2
Φ_3	24	0.01389	2

3.2 Effect of Unsupervised Training

For the sake of saving training time, we need to know how much training is sufficient to achieve an asymptotic level of classification performance. We defined training on 5000 images as one epoch, learning 10 dictionaries $\Phi_1, \cdots, \Phi_{10}$, where Φ_i is trained in the ith epoch and Φ_{i+1} is initialized on Φ_i. We use $K = 192, \sigma = 2$, which is $Ub(\Phi_i) = \frac{K}{3*2*\sigma^2} = 8$ (the factor 3 comes from the 3 color bands) times over-complete, and $H = 20$.

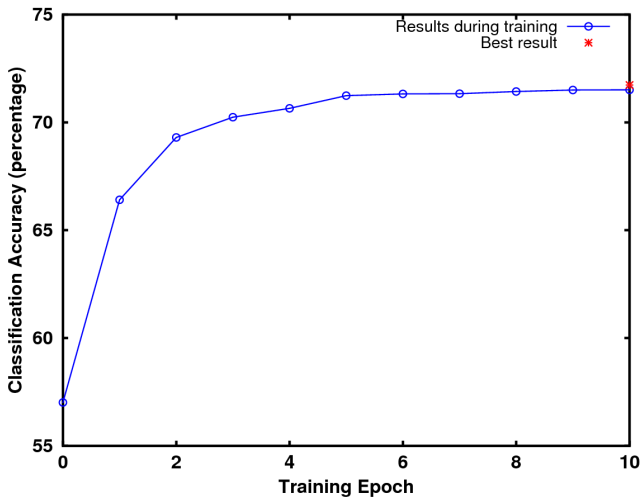


Figure 2: Effect of unsupervised training. Epoch 0 means a dictionary initialized with random values. Epoch 10 is equivalent to training through the CIFAR10 training set once. The best result is obtained after 100 epochs, or ten complete pases through the training set.

Figure 2 shows that most of the improvement in classification performance happens in the first few epochs. Even with 9 times more training, the improvement is less than 1%. However, it takes more time to train a dictionary with a large K. This is because the sparsity of the representa-

tion increases with K, so that only a relatively small fraction of the dictionary elements are updated per image. In the following we will use either of the two training strategies: training two times through the training set, using high learning rate in the first round and low learning rate in the second one; training through training set once with a low learning rate.

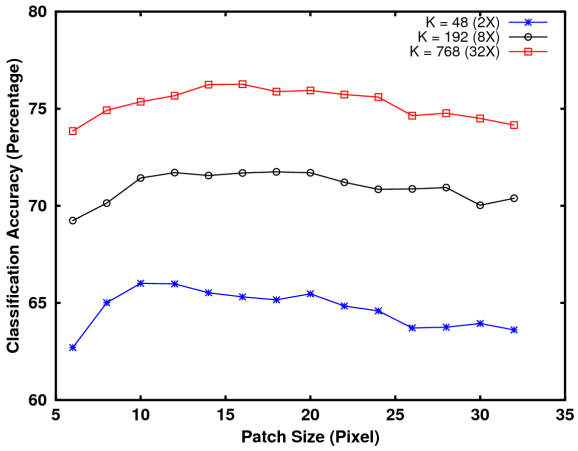
3.3 Effect of Patch size

The primary objective of this paper is to investigate the effect of patch size on image classification performance.

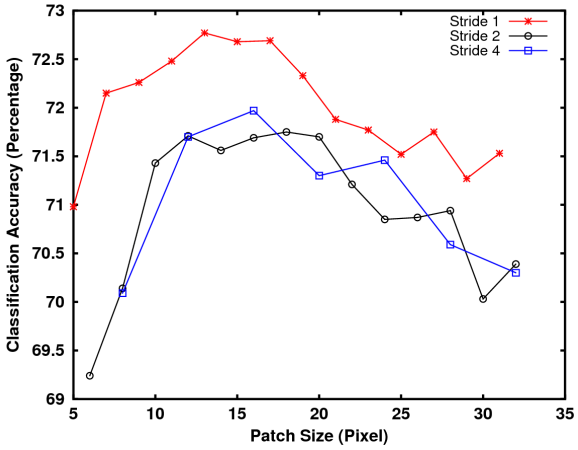
We trained dictionaries for three levels of over-completeness i.e. $2\times, 8\times, 32\times$, and $\sigma = 2$. As shown in Fig. 3a, the observation that increasing the number of features results in increased classification accuracy is consistent with previous findings [2]. What is novel here is that increasing the patch size does not decrease classification accuracy. Classification performance increases as a function of patch size up to a peak value, at which point classification performance stabilizes or declines slightly. The observation that classification performance remains approximately constant for a wide range of large patch sizes is probably related to the fact that for an RKM, the over-completeness is independent of patch size. This insensitivity suggests that the RKM is able to maintain a given level of overcompleteness despite the use of large patch sizes. There is also a tendency for the peak accuracy to occur at larger patch sizes as the overcompleteness increases. The peak accuracy occurs at a patch size of approximately 16×16 (the CIFAR10 image is a 32×32 color image). It is possible that edge effects dominate as the patch size becomes too close to the image size. We did a similar experiment on Caltech101 [3] dataset with a $2\times$ overcomplete dictionary and a stride of 2. Every image was re-sized to 256×256. Fig. 3c shows a clear trend of monotonically increasing in accuracy for patch sizes up to 56. Above the patch size 24, the increasing rate drops dramatically, suggesting that size 24 is best choice with respect of the trade-off between accuracy and computational complexity. Because of the large image size of Caltech101 images, edge effects are less likely to affect the result.

3.4 Effect of Stride

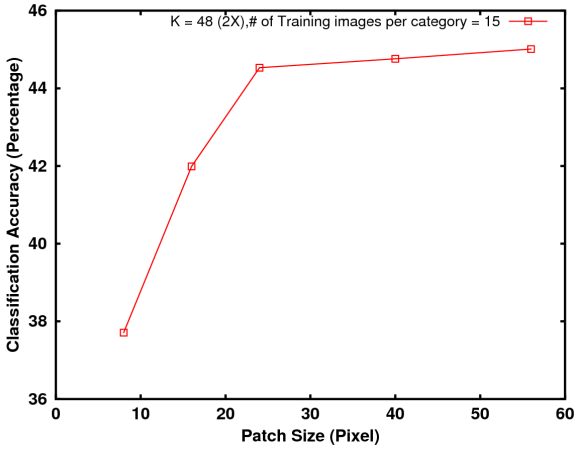
The choice of the stride used in RKM has two consequences. A low stride preserves more information, but can be quite computationally expensive. For example, the number of feature maps with a stride of 1 is 4 times larger than with a stride of 2. In this experiment, we use $K = 192$, and $\sigma = 1, 2, 4$, Fig. 3b. The stride 1 case exhibits the best performance, but the stride 2 case exhibits a similar performance. The performance gap tends to be narrower for large patch sizes than for small ones . This is explained by the fact that large features do not have to be replicated with as fine a resolution, reducing the importance of smaller strides. Clearly, the improvement produced by increasing K is much larger than the improvement produced by decreasing the stride. In Fig. 3a, feature maps $N' = M' = 16, K = 768$ have the same size as the feature $N' = M' = 32, K = 192$ in Fig. 3b, yet the former achieves approximately 5% improvement whereas the latter only has about 1% improvement. It also validates the principle adopted by RKM, that increasing K is better than increasing N' and M'.



(a)



(b)

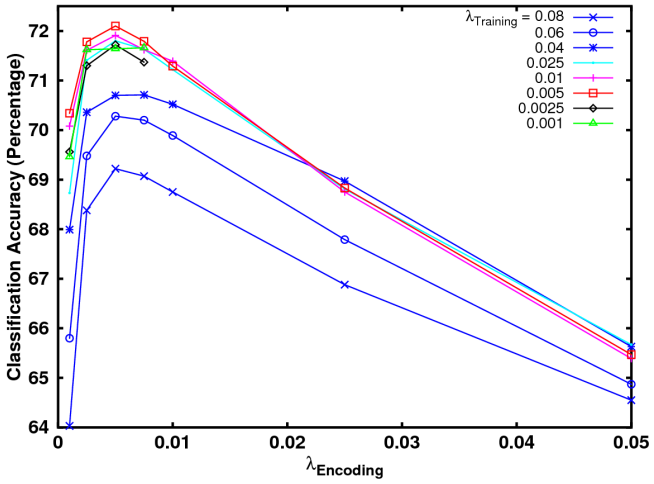


(c)

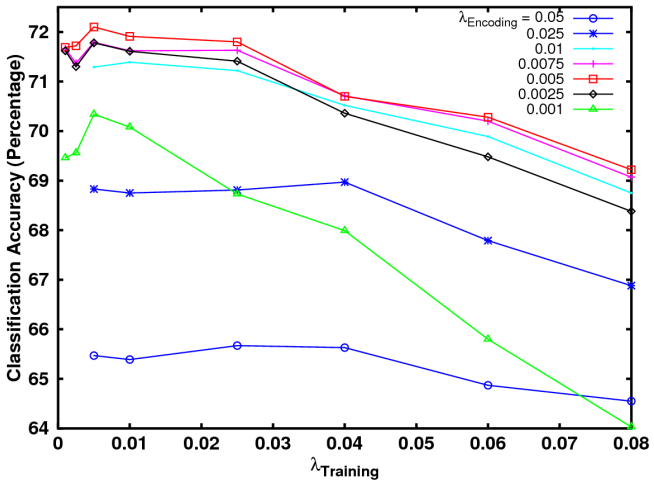
3.5 Effect of λ

The parameter λ controls the balance between reconstruction error and sparsity. Intuitively λ represents sparsity, as smaller values of λ produce less sparse representations and visa versa. λ can be set separately during two different phases, corresponding to image encoding and dictionary learning, respectively. In our experiment, we used

$\lambda_{learning} = 0.025$ to train an initial set of dictionary elements, where were then used to test a variety of $\lambda_{training}$ and $\lambda_{encoding}$. Fig. 4 shows that as a function of both $\lambda_{training}$ and $\lambda_{encoding}$, there is a classification accuracy peak at 0.005. Other authors [9] report that classification accuracy decreases monotonically as $\lambda_{encoding}$ increases. It is possible that our results differ because our dictionary is more overcomplete, thus amplifying the importance of competition between dictionary elements. Our results futher suggest that sparsity is important in both dictionary learning and encoding phases.



(a)



(b)

Figure 4: Effect of λ. (a) Each line is trained by the same $\lambda_{Training}$; (b) Each line is encoded by the same $\lambda_{Encoding}$.

3.6 Discussion and Future Work

In this paper, we use a deconvolutional strategy for holding the overcompleteness of a dictionary fixed while increasing the patch size of the individual elements. Our best results exceed the previously established state-of-the-art performance on the CIFAR10 dataset. Specifically, the best published result for a sparse auto-encoder with a single hidden layer on the CIFAR10 dataset was 73.4% [2], obtained using a dictionary with 1,600 elements, a patch size of 6×6 and a stride of 1. Using a smaller configuration, namely, a stride of 2 and 768 elements, we obtained a similar classification accuracy of 73.85%. However, with a larger patch size of 16×16, our classification accuracy increased to 76.26%. Thus, even with fewer elements and a larger stride, we obtained better than state-of-the-art performance by increasing the patch size of the individual dictionary elements.

In this study, key factors supporting improved image classification were as follows: sufficient unsupervised training to optimize a dictionary for sparse reconstruction, using dictionaries that were more overcomplete, employing dictionary elements with large patch sizes (up to an optimum size), and optimum λ that subtly balanced between reconstruction error and sparsity .

Our results show that even randomly-generated dictionary elements support image-classification performance well above chance. Further, unsupervised training improves image classification performance substantially. It is not obvious a priori that optimizing a dictionary for sparse reconstruction should produce features that are optimal for image classification. Nonetheless, our results suggest that this is indeed the case. Even relatively simple Gabor-like features are likely to be more discriminative than random features. It has been shown in the context of a restricted Boltzman machine that training features in a supervised fashion leads to performance improvements on image classification tasks, and that the problem of learning optimal features in a supervised manner is effectively convex [4]. Whether the unsupervised learning of sparse coding features is likewise effectively convex is a question for further research. Here, we demonstrate that optimizing a dictionary for sparse reconstruction, under the constraints of an RKM, leads to corollary improvements in performance on an image classification task, even though at no point was classification performance used as a criteria for dictionary learning.

As in previously studies, increasing the number of dictionary elements, or overcompleteness, also improved classification performance. Intuitively, a dictionary with more elements is more likely to include elements that are non-Gabor-like, and thus potentially more discriminative. Indeed, increasing the number of randomly-generated features also led to improved classification performance, presumably for the same reason. The more randomly-generated features available in the dictionary that can be used to encode a given image, the more likely that some of the those features will enable better discriminations between image categories.

We find that dictionaries with larger patch sizes support improved classification performance, at least compared to dictionaries with very small patch sizes. Again, our results suggest that larger patch sizes allow for the possibility of learning less Gabor-like, more discriminative features. Larger patch sizes open the possibility of capturing structure at larger spatial scales, including non-Gabor-like structures. Assuming that some of these "large" features enable better discrimination between image categories, it is not surprising

that larger patches support better image classification performance. Perhaps more surprising is the fact that image classification performance peaks at a given patch size and then begins to decline, albeit slowly, for even larger patch sizes. The cause of this effect is unclear, but could be related to the difficulty of learning spatially extended features.

Also consistent with previous studies, our results show that image classification performance can be improved by increasing sparsity during training. Again, this effect is likely related to the relative probability of learning non-Gabor-like, more discriminative features as sparsity is increased. Since a non-Gabor-like feature may in principle be represented as a combination of several simpler Gabor-like features, high-sparsity may discourage such solutions, forcing the dictionary to instead learn only a single non-Gabor-like feature that may in turn be more discriminative. In fact, the best performing algorithm for training a single hidden layer in an unsupervised manner on CIFAR10 is K-means [2], which can be thought of as an extreme limit of sparsity in that only one dictionary element is updated for each image.

Finally, as has been previously reported, we find that using a reduced level of sparsity during classification leads to improved performance.

Our results point to the possibility that a combination of multiple patch sizes might further improve the image classification, as small patches are more likely to catch local information and large features are more is likely to catch global information. This dynamic suggests an interesting avenue for future investigation.

4. ACKNOWLEDGMENTS

This research was supported by Defense Advanced Research Projects Agency (DARPA) Unconventional Processing of Signals for Intelligent Data Exploitation (UPSIDE) program.

5. REFERENCES

[1] PetaVision. `github.com/PetaVision/OpenPV`.

[2] A. Coates, H. Lee, and A. Y. Ng. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. *International Conference on Artificial Intelligence and Statistics*, pages 215–223, 2011.

[3] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In *IEEE. CVPR, Workshop on Generative-Model Based Vision*. Ieee, Apr. 2004.

[4] I. J. Goodfellow, O. Vinyals, and A. M. Saxe. Qualitatively characterizing neural network optimization problems. *International Conference on Learning Representations*, 2015.

[5] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? *Proceedings of the IEEE International Conference on Computer Vision*, pages 2146–2153, 2009.

[6] A. Krizhevsky. Learning Multiple Layers of Features from Tiny Images. Technical report, Computer Science Department, University of Toronto, 2009.

[7] B. A. Olshausen. Highly overcomplete sparse coding. In *SPIE 8651*, 2013.

[8] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997.

[9] R. Rigamonti, M. A. Brown, and V. Lepetit. Are sparse representations really relevant for image classification? In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1545–1552, 2011.

[10] F. Rong-En, C. Kai-Wei, H. Cho-Jui, W. Xiang-Rui, and L. Chih-Jen. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 2008.

[11] C. J. Rozell, D. H. Johnson, R. G. Baraniuk, and B. A. Olshausen. Sparse coding via thresholding and local competition in neural circuits. *Neural computation*, 20(10):2526–2563, Oct. 2008.

[12] P. F. Schultz, D. M. Paiton, W. Lu, and G. T. Kenyon. Replicating Kernels with a Short Stride Allows Sparse Reconstructions with Fewer Independent Kernels. 2014.

[13] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus. Deconvolutional networks. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2528–2535, 2010.