Decoupling Sparse Coding with Fusion of Fisher Vectors and Scalable SVMs for Large-scale Visual Recognition

Zhengping Ji
Advanced Image Research Laboratory
Samsung Semiconductor Inc., Pasadena, CA

Abstract

With the advent of huge collection of images from Internet and emerging mobile devices, large-scale image classification draws amount of research attention in computer vision and AI communities. The advancement of large-scale image classification largely depends on solutions to two problems: how to learn good feature representation from variant scales of pixels, and how to create classification models that can discriminate the feature representation for different semantic meanings of many objects. In this paper, we tackle the first problem by combining different feature representations via sparse coding and Fisher vectors of SIFT and color-based features. To deal with the second problem, we utilize the Averaged Stochastic Gradient Descent (ASGD) algorithm to enable fast and incremental learning of SVMs and further generate confidence values to interpret the likelihood of multiple object categories appearing in the image. We evaluate the proposed learning framework on the ImageNet, a benchmark dataset for large-scale image classification. Our results show favorable performance on a subset of ImageNet containing 196 categories. We also investigate the performance of sparse coding by comparing different combination of algorithms in learning a dictionary and sparse representations. Although there is a natural pair of algorithms to learn a dictionary and sparse representations (e.g., K-SVD with respect to Orthogonal Matching Pursuit), breaking such a pair and rematching are found to result in even better performance. Moreover, detailed comparison indicates that ℓ_1-regularized solver to sparse representation mainly benefit the classification accuracy, regardless of the choice of dictionaries.

1. Introduction

Image classification is one of major focuses in computer vision research. It maps pixel inputs to the semantic meanings of objects. There have been extensive research efforts on developing effective image classification/recognition systems for various benchmark datasets, such as MNIST [18], NORB[19], CIFAR-10[15], Caltech-101[9], Caltech 256 [12], PASCAL VOC [8] etc. Recently, there is an increasing need to build general-purpose learning systems that are able to recognize a large number of object classes, which can be very useful for automatic image tagging and content-based image retrieval.

ImageNet [6] is the first and unique image database to serve the purpose of large-scale recognition, who provides an access to 15M labeled images belonging to 22K object categories. Since 2010, a subset of ImageNet with 1000 categories is extracted to establish an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC), where the progress has been made mainly in following aspects. One is with regards to learn feature representation via coding and pooling of local image descriptors, such as SIFT [23], HoG [5], LBP [35], etc. The well-known coding schemes include vector quantization (VQ), sparse coding (SC) [37], locality-constrained linear coding (LCC) [34, 22] and Fisher vectors (FV) [31]. Examples of effective pooling methods are the Bags-of-Words (BoW) model [10] and its multi-scale alternative called Spatial Pyramid Matching (SPM) [17, 37, 31]. Another major achievement in ILSVRC is to design classification algorithms that scale up the recognition problem without compromising performance. For an example of SVM, learning algorithms of Stochastic Gradient Descent (SGD) [2, 31] and Averaged SGD [22] are developed to train one-against-all SVMs incrementally and in parallel for a large amount of data. More recently, Krizhevsky et al., 2012 [16] extended the convolutional neural networks [19, 33] to a deep fashion with GPU implementation and achieved state-of-the-art performance in ILSVRC-2012.

Aforementioned sparse coding (SC) is one of the most popular approaches to learn the feature representation via an unsupervised generative approach using a linear combination of over-complete bases with the sparse coefficients. Amount of research work assume a known dictionary of bases (also called codebook or weight matrix) and learn sparse representation (also called codes or coefficients) by greedy approximation to ℓ_0-based problem (e.g., Matching Pursuit (MP) [25], Orthogonal Matching Pursuit (OMP) [28]) or using a convex optimization for $\ell 1$-regularized problem (e.g. Basis Pursuit (BP) [3], FOCUSS [11] and Lasso [32]). Another line of sparse coding research aim-

978-0-7695-4990-3/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPRW.2013.74

IEEE computer society

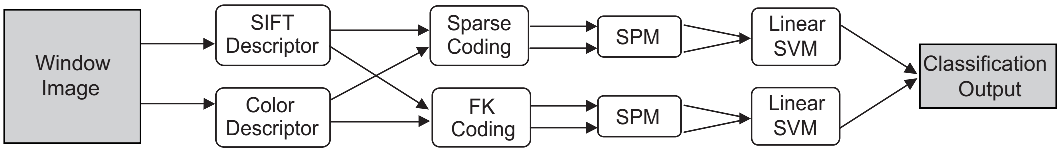


Figure 1. The overview of proposed learning system for image classification.

s to learn a dictionary of bases, rather than use predefined ones. Well-known examples include the pioneering work of Olshausen and Field 1997 [26] to model neuronal responses in the V1 area of the brain, K-SVD of Elad and Aharon 2006 [1], Online Sparse Coding of Mairial et al. 2010 [24] and others [7, 21, 20, 14, 13]. Each dictionary learning algorithm naturally contains a sparse representation solver for an alternating minimization.

To go beyond the sparse coding that is based on a soft quantization of dictionary elements, amount of research work have been proposed to include higher order statistics to model the dictionary distribution. A famous example is the Fisher Kernel framework [29], which adopts Gaussian Mixture Model (GMM) as a generative process of feature elements and shows high accuracy in various tasks when combining linear SVM classifiers [30, 31].

In this paper, we take advantage of state-of-the-art approaches in previous work. We deployed two feature coding schemes (i.e. sparse coding and Fisher vectors) to encode two grid-based dense feature descriptors (i.e., SIFT and color statistics) respectively. The delivered four feature representations are fused at two stages, first in a representation stage via catenation and second in a classification stage using prediction confidence as a fusion weight. The proposed framework showed a boosted classification accuracy on a subset of ImageNet dataset when compared to respective feature representation and classification models without fusion.

As discussed above, a large number of sparse coding algorithms are available and can be split into two streams: one for dictionary/codebook learning and the other for sparse representation/coefficients/codes development. We investigated the performance of sparse coding by comparing the contribution of state-of-the-art algorithms in learning dictionaries and sparse representations. Although there is a natural choice of sparse representation algorithm to pair with dictionary learning (e.g., Orthogonal Matching Pursuit with respect to K-SVD), breaking such a pair and rematching is found to result in even better performance. Such a decoupling scheme was previously conducted by Coates and Ng 2011 [4] to investigate the importance of encoders vs. sparse coding solutions for small and medium-scale object recognition datasets. Here we are focused on exploring optimal solvers within the sparse coding framework to deal with large-scale object recognition problems. The results indicate that: (1) Regardless of choice of dictionaries, learning algorithms for sparse representation mainly result in performance variance. (2) In particular to our problem, $\ell 1$-regularized optimization algorithms in average perform better than greedy approximating algorithms given $\ell 0$-based sparsity. (3) Even using a dictionary with random descriptors (without training), the performance is surprisingly comparable to those using the trained dictionaries.

2. Learning Framework

The proposed system architecture for image classifications is shown in Fig. 1. Given an input image, the system first extracts dense local descriptors, SIFT or color statistics [30]. Then, each local descriptor is coded either using sparse coding or Fisher vectors (FVs), leading to 4 different types of local feature codes. The feature codes are further passed to the pooling with spatial pyramid matching (SPM) to form a single vector for each image. We concatenate two SPM vectors from sparse coding channel and the other two from Fisher vector channel. The concatenated vector of each channel is fed into scalable SVMs for the object classification on large-scale ImageNet dataset. A final decision is made given a fusion of class prediction confidence from the two coding channels.

In what follows, we will describe the learning framework with each computational component at each subsection.

2.1. Feature Extraction

We first use a simpler and faster version of SIFT algorithm, called dense SIFT to extract features, where the location, scale and orientation of each keypoint are predefined rather than extracted from a scale-space extrema. In our experiments, 16×16 pixel patches are densely sampled from each image on a grid, such that the center of each patch is considered as the keypoint. This yields a representation of the image as a set of 128-dimensional (8 orientations× 16 histograms) descriptors, with one descriptor representing each patch in the grid.

We also consider color statistics as another type for feature descriptors, where we subdivide a 16×16 image patch into 4×4 subregions. In each sub-region, the mean and standard deviation is computed for the R, G and B channels respectively. This yields a representation of the image as a set of 96-dimensional (16 subregions × 3 colors × 2 statistical measurements) vectors.

Given a training set containing a number of images $\mathcal{X} = \{\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_N\}$, we have a corresponding training set with extracted feature descriptors (either SIFT or color statistics), i.e., $\mathcal{Y} = \{\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_N\}$. Each $\mathbf{Y}_i = [\mathbf{y}_i^{(1)}, \mathbf{y}_i^{(2)} ..., \mathbf{y}_i^{(P)}]$ represents a matrix containing each feature descriptor as a column vector, where P is the number of feature descriptors for each image.

2.2. Sparse Coding

For the SIFT or the color descriptors among all the images, we randomly choose K descriptors, i.e., $\{\mathbf{y}^{(1)}, \mathbf{y}^{(2)} ..., \mathbf{y}^{(K)}\}$ to learn a dictionary $\mathbf{D}$ via sparse coding, such that

$$\min_{\mathbf{D}, \mathbf{a}^{(k)}} \frac{1}{K} \sum_{k=1}^{K} \left\{ \frac{1}{2} ||\mathbf{y}^{(k)} - \mathbf{D}\mathbf{a}^{(k)}||_2^2 + \lambda\Omega(\mathbf{a}^{(k)}) \right\} \qquad (1)$$

where $\Omega(\mathbf{a})$ is a function to enforce a vector sparsity, with a controlled parameter λ. In this case, the dictionary learning algorithm requires a matching sparse representation solver to minimize the objective function in an alternating manner, first with respect to $\mathbf{a}$ and then to $\mathbf{D}$.

We utilized three different dictionary learning algorithms in this paper:

1. **K-SVD**: K-SVD is a simple but an efficient dictionary learning algorithm developed by Aharon et al. 2006 [1]. K-SVD solves Eq. 1 with regard to the sparse vector first, where aforementioned Orthogonal Matching Pursuit is used to approximate the solution to the non-convex $\ell0$-regularized sparse problem. Second, the dictionary is learned via a batch of input samples, where only one column of $\mathbf{D}$ is updated at a time using the singular value decomposition (SVD).

2. **Lagrange dual**: This is an efficient dictionary learning algorithm proposed by Lee et al, 2006 [20], who developed a sign-search algorithm to solve $\ell1$-regularized least squares problem with respect to a sparse vector $\mathbf{a}$. Then a Lagrange dual method is used to solve the $\ell2$-constrained least squares problem with respect to a dictionary $\mathbf{D}$. Both problems above are known convex with global minima.

3. **SPAMS**: SPAMS is a SPArse Modeling Software containing an optimization toolbox for various sparse estimation problems. We used its dictionary learning solver based on the paper published by Mairial et al, 2010 [24], where a Cholesky-based implementation of the LARS-Lasso algorithm [27] is utilized to solve $\ell1$-regularized sparse coding problem with respect to a sparse vector and a new online optimization algorithm based on stochastic approximations is developed to learn a dictionary.

We use the trained dictionary to code every descriptor and generate the sparse feature representation via an optimization step as below,

$$\forall \mathbf{y} \in \mathcal{Y}, \quad \min_{\mathbf{a}} \frac{1}{2} ||\mathbf{y} - \mathbf{D}\mathbf{a}||_2^2 + \lambda\Omega(\mathbf{a}) \qquad (2)$$

Note that only sparse vector $\mathbf{a}$ is learned here, with a fixed $\mathbf{D}$. Each sparse vector $\mathbf{a} \in \mathbb{R}^{1024 \times 1}$ represents one local descriptor in an image.

We applied three different learning algorithms to compute sparse representation in Eq. 2: (1) Orthogonal Matching Pursuit [28]; (2) Sign-search optimization [20]; (3) a variant of LARS-Lasso algorithm [27]. In fact, each learning algorithm here for sparse representation is used in one of the dictionary learning algorithm above, but we found that the *natural* choice of sparse representation algorithm that matches the dictionary learning (e.g., Orthogonal Matching Pursuit with respect to K-SVD) is not optimal to provide favorable feature representation for classification performance. In other words, when the sparse representation solver (in Eq. 2) mismatches the one in dictionary learning (in Eq. 1), we may surprisingly achieve more favorable results.

2.3. Fisher Vector

Unlike the sparse coding based on a soft quantization (zero-order statistics) of dictionary elements, Fisher vectors encode the image descriptors assigned to each dictionary element via higher-order statistics [31]. Given a set of local descriptors $\mathbf{Y}_i = [\mathbf{y}_i^{(1)}, \mathbf{y}_i^{(2)} ..., \mathbf{y}_i^{(P)}]$ for an image, u_λ is a Gaussian density function with a parameter set λ which models a generative process of local descriptors for an image, such that

$$u_\lambda(\mathbf{y}) = \sum_{m=1}^{M} \mathrm{w}_m u_m(\mathbf{y}) \qquad (3)$$

where λ is a set of parameters $\{\mathrm{w}_m, \ \mu_m, \ \delta_m\}(m = 1, 2..., M)$, respectively denoting the mixture weight, and mean and variance of Gaussian u_m.

Let $\gamma_i(m)$ be the soft assignment of descriptor $\mathbf{y}^{(p)}$ to the m-th Gaussian

$$\gamma_k(m) = \frac{\mathrm{w}_m u_m(\mathbf{y}^{(p)})}{\sum_{m=1}^{M} \mathrm{w}_m u_m(\mathbf{y}^{(p)})}, \qquad (4)$$

such that the Fisher vector $\mathcal{G}_\lambda^{\mathbf{y}}$ is computed as the concatenation following two vectors:

$$\begin{cases} \mathcal{G}_{\mu,m}^{\mathbf{y}} = \dfrac{1}{T\sqrt{\mathrm{w}_i}} \displaystyle\sum_{i=1}^{N} \gamma_i(k) \left(\dfrac{\mathbf{y}_i - \mu_m}{\delta_m} \right) \\ \mathcal{G}_{\delta,m}^{\mathbf{y}} = \dfrac{1}{T\sqrt{\mathrm{w}_i}} \displaystyle\sum_{i=1}^{N} \gamma_i(k) \left[\dfrac{(\mathbf{y}_i - \mu_m)^2}{\delta_m^2} - 1 \right] \end{cases} \qquad (5)$$

The parameter space u_λ can be trained using the maximum likelihood estimation (MLE), and the final Fisher Vector is in dimension $2 \times D \times M$, where D is the dimensionality of the local descriptors, and M is the number of Gaussians.

2.4. Spatial Pyramid Matching

As size of images varies in the training set, the number of coded feature vectors varies as well for each image. Thus, we need to further compute representations with an identical dimension in order to feed into a classification model. Given a set of coded feature vectors for each image, a popular choice is to quantize the feature vectors and then compute a histogram representation. This procedure is called a Bag-of-Words (BoW) model [9], where the spatial order of local feature codes is discarded.

In a more sophisticated SPM approach [17], we partition an image into multiple segments, and max-pool the coded feature vectors within each of the segments. The spatial order of the feature codes is maintained across the segments, and pooled vectors from various segments are then concatenated to form a spatial pyramid representation of an image.

For the sparse coding channel in our paper, each SPM representation (either for SIFT feature or color statistics) has 21504 (=1024 number of dictionary elements × 21 segments) dimensions. We catenated the two SPM representations and fed it into SVM for classification.

For the FV coding channel, each SPM representation has dimension $2 \times D \times M \times 8$ (segments) given Eq. 5. We conducted Principal Component Analysis (PCA) to reduce both SIFT and color features to 64 dimensions, such that $D = 64$ to compute FV. The number of Gaussians is set to 256. Catenation of the two SPM representations leads to a single representation in dimension 524,288, which is further fed into SVM for classification.

These high-dimensional image representations are shown important to deal with large-scale image recognition problems. However, the high memory cost and I/O latency to store/read/write such high-dimensional representations (especially for FVs) make the learning difficult or even infeasible. As suggested by Sánchez et al. 2011 [31], we compress the catenated SPM representation of FVs using the product quantization.

2.5. Classification Model

Given the training data $\{(\mathbf{X}_i, c_i)\}$, $i = 1, 2, \ldots N$, where $\mathbf{X}_i$ is an image input and $c_i \in \mathcal{C} = \{1, 2, \ldots, L\}$ is the corresponding class label of this image. Through each coding channel in Fig. 1, each image $\mathbf{X}_i$ is finally represented as a catenated SPM representation $\mathbf{s}_i$. We used a one-against-all strategy to train L binary linear SVMs, each solving the convex optimization problem as follows

$$\min_{\mathbf{w},\xi,b} \left\{ \frac{1}{2}\|\mathbf{w}_l\|^2 + C\sum_{i=1}^{n} \xi_i \right\} \tag{6}$$

$$\text{s.t. } f(c_i)(\mathbf{w}_l \cdot \mathbf{s}_i - b_l) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

where $f(c_i)$= 1 if $c_i = l$, otherwise $f(c_i) = -1$ ($l = 1, 2, \ldots, L$).

There have been amount of SVM solvers available, but most of them are not feasible for such huge training data. We use an incremental learning algorithm called Averaged Stochastic Gradient Descent (ASGD) [22] to train a decompressed image presentation per time, without a need to load the whole training set. This important property of ASGD makes the learning scalable for such a large-scale problem.

Rather than a maximum operation to predict a class label of a testing sample, we further estimate a confidence score for each SVM prediction [36], such that the output of a testing sample is a distribution of confidence likelihood. A final class prediction is based on an averaging of the confidence outputs respectively from sparse coding and FV channels.

3. Experimental Results

In the experiments, we evaluated the learning framework on the ImageNet dataset [6]. ImageNet is a first and unique image database containing 15M labeled images belonging to 22K object categories, which are organized according to the WordNet hierarchy of meaningful concepts. About 1000 images are included in each concept meaning/label in the ImageNet, and some of them are human-annotated for object detection purpose. A subset of ImageNet with 1000 categorise (most of which are from leaf nodes in the semantic hierarchy) is extracted from the ImageNet to establish an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) since 2010. There are in total of 1.2M training images, 50K validation images, and 150K testing images for the challenge.

We selected 196 categories with their images from ILSVRC-2012 dataset to conduct visual object recognition using the described learning system. We preprocessed all the images such that when a shorter size of the image is larger than 256, we re-scaled the images by a certain ratio such that the shorter size of the image is equivalent to 256.

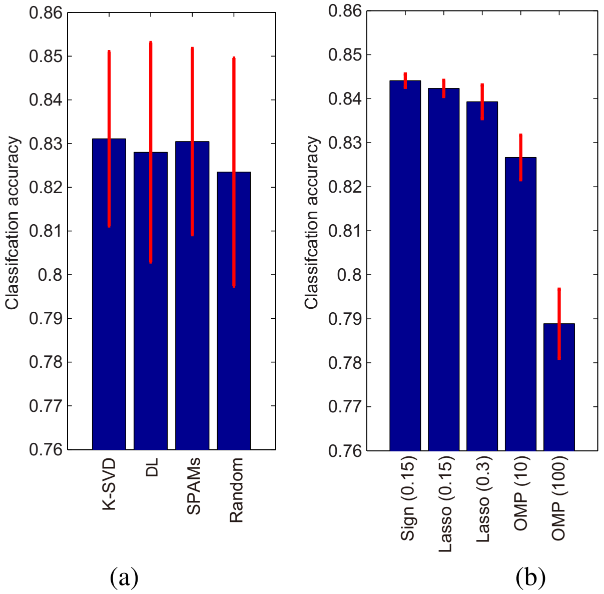
3.1. Sparse Coding Performance

We first invetigate the performance of sparse coding by comparing different pairs of algorithms in learning a dictionary and sparse representations. As discussed in Secs. 2.2 and 2.3, three dictionary learning algorithms, i.e., K-SVD, Lagrange dual (LD) and SPAMS and three sparse representation algorithms, i.e., Sign-search, LARS-Lasso and OMP are included in this experiment, those of which represent state-of-the-art approaches at the current time.

Table 1. Top-5 accuracy rate of sparse coding channel (%).

	K-SVD	LD	SPAMS	Random
Sign Search ($\lambda = 0.15$)	84.62	84.43	84.31	84.28
LARS ($\lambda = 0.15$)	84.24	84.47	84.20	84.02
LARS ($\lambda = 0.3$)	83.86	84.11	84.32	83.43
OMP ($L = 10$)	83.14	82.42	83.03	82.07
OMP ($L = 100$)	79.68	78.57	79.36	77.94

Table 1 shows how the classification accuracy varies given several choice of parameters, as well as different pairs of learning algorithms. The number in each cell presents a top-5 accuracy rate – the fraction of testing images for which the correct label is among the five labels considered most probable by the model. The top-5 rate is a useful measurement for the ImageNet dataset, where each image may contain more than one object (presumingly up to 5). Note that we did not explore the parameter space with all possible values, but referred empirical studies about favorable settings of sparse coding parameters in visual recognition tasks [37][4].



(a) (b)

Figure 2. Mean and standard deviation of top-5 accuracy rates with respect to (a) the dictionary learning algorithms across different sparse representations. (b) the sparse representation algorithms (given various parameters) across different dictionaries.

From Table 1, we can observe that: (1) When learning sparse representations, $\ell 1$-regularized optimization algorithms (the first three rows) in average perform better than greedy approximation algorithms given $\ell 0$-based sparsity (the last two rows). (2) Regardless of a dictionary choice, learning algorithms for sparse representation mainly result in performance variance, as indicated in Fig. 2. (3) Even using a dictionary with random SIFT or color patches (without training), the performance is yet comparable to those using the trained dictionaries (see the last column in Table 1).

The results in Table 1 also indicate that a $natural$ choice of sparse representation algorithm that matches the dictionary learning (e.g., Orthogonal Matching Pursuit with respect to K-SVD) may not be optimal to provide favorable feature representation for classification performance. Finally, we selected the best-performed K-SVD algorithm for dictionary learning and Sign-search algorithm for the sparse representation in the learning system.

3.2. Overall Classification Results

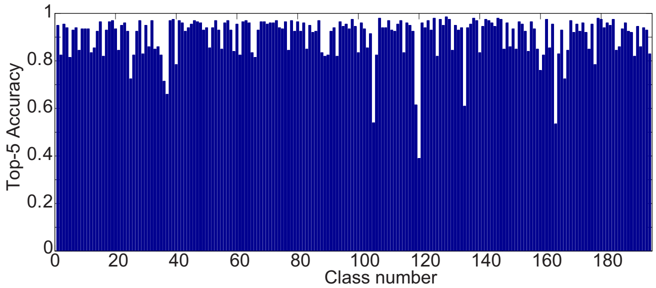
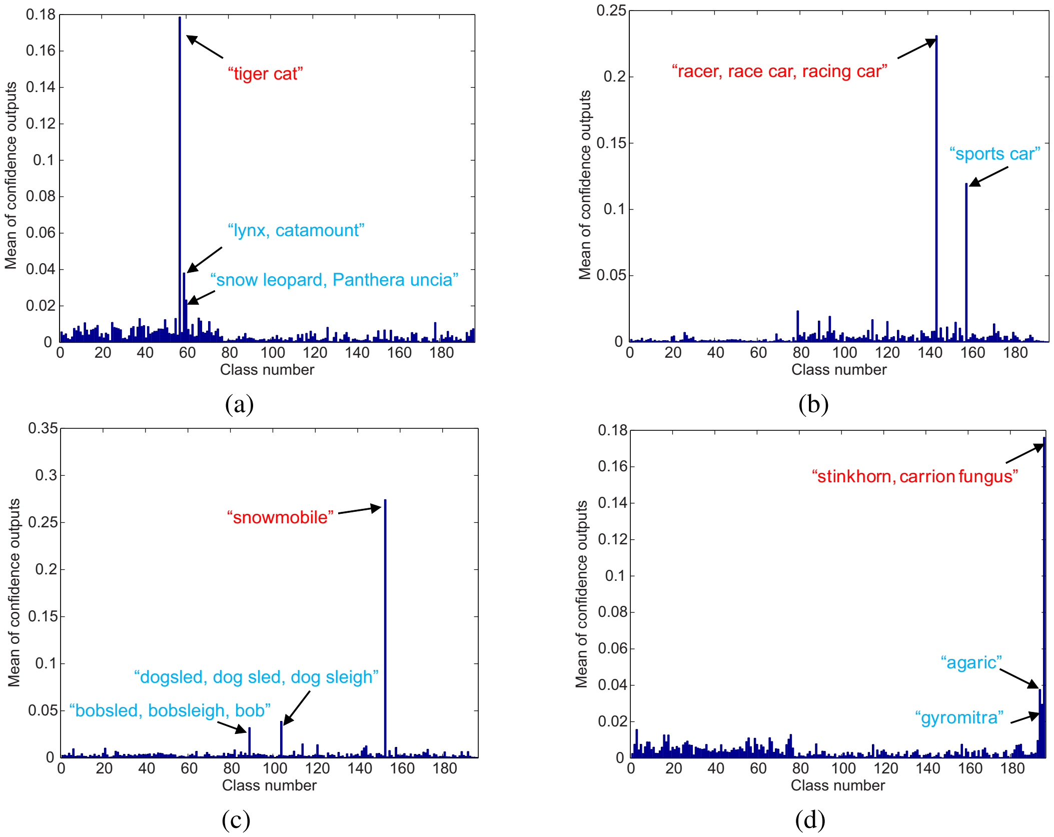


Figure 3. Top-5 accuracy rate per class for ImageNet-196.

Fig. 3 shows the top-5 accuracy rate per class for ImageNet-196, using both sparse and Fisher vectors for feature representation and with fusion of classification results. The proposed learning system reached 90.64% overall accuracy rate for the top-5 prediction.

As discussed in Sec. 2.5, we estimate confidence likelihood for SVM prediction of output classes. In Fig. 5, we plot some examples of testing images, along with their predicted confidence outputs. Each confidence output is normalized with respect to the top-5 classes. The left column illustrates some easy cases (with high confidence regarding a particular class that is correctly predicted) and the right column illustrates some tough cases of the same class, showing very different distribution patterns (with uncertainness and incorrectness about the ground truth). Fig. 4 plots the mean of normalized confidence outputs for testing images within 4 selected classes, i.e., “tiger cat”, “race car”, “snowmobile” and “stinkhorn”. It also shows the most “confusing” classes (with comparatively high probability) for each of the selected classes, e.g., “lynx” and “snow leopard” with respect to “tiger cat”. The interpretation of these correlated classes can help the system target on difficult case for a fine-tuned classification to further boost system performance.

The top-5 accuracy for each class varies in ImageNet-196. We illustrated several examples for object classes that



(a) (b)

(c) (d)

Figure 4. Mean of normalized confidence outputs for testing images within 4 selected classes, whose label names are displayed in red. In the meanwhile, the most "confusing" ("correlated") classes for each of the selected class are displayed in cyan.

delivered the worst performances, as shown in Fig. 6. As we can see, examples within each class are highly variant (e.g., different object forms, backgrounds, poses, sizes, colors, occlusions, and light conditions, etc.), indicating the challenge of these object categories per se.

4. Conclusion

In this paper, we integrated two feature coding schemes (i.e. sparse coding and Fisher vector coding) to respectively encode two grid-based dense feature descriptors (i.e., SIFT and color statistics). The delivered four feature representations are fused in two stages, first in a representation stage and second in an SVM classification stage. We used the Averaged Stochastic Gradient Descent (ASGD) algorithm to enable fast and incremental learning for SVMs and utilized confidence outputs to interpret the likelihood of each object class. The likelihood values are further used as contribution weights to combine classification results. The proposed learning system led to 90.64% top-5 accuracy rate on a subset of ImageNet ($\sim 200,000$ images for 196 classes).

We further investigated the pairs of algorithms for dictionary learning and sparse representation development. The results show that the algorithms for sparse representation mainly determined the classification accuracy, regardless of the choice of dictionaries. Matching the sparse representation algorithm with the one included in each dictionary learning does not guarantee to deliver a better performance.

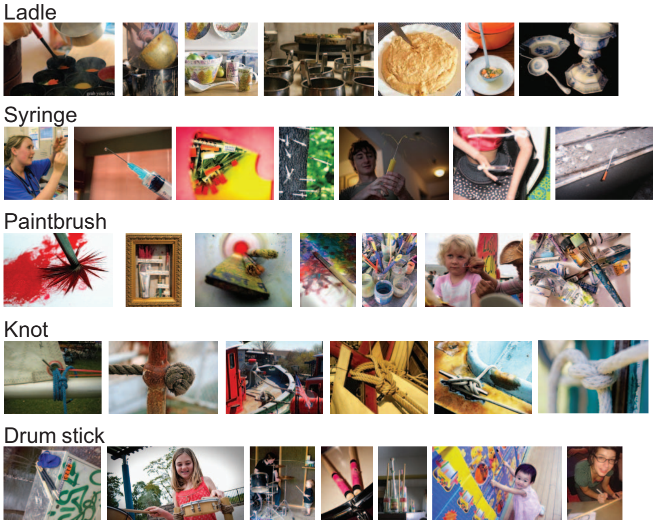
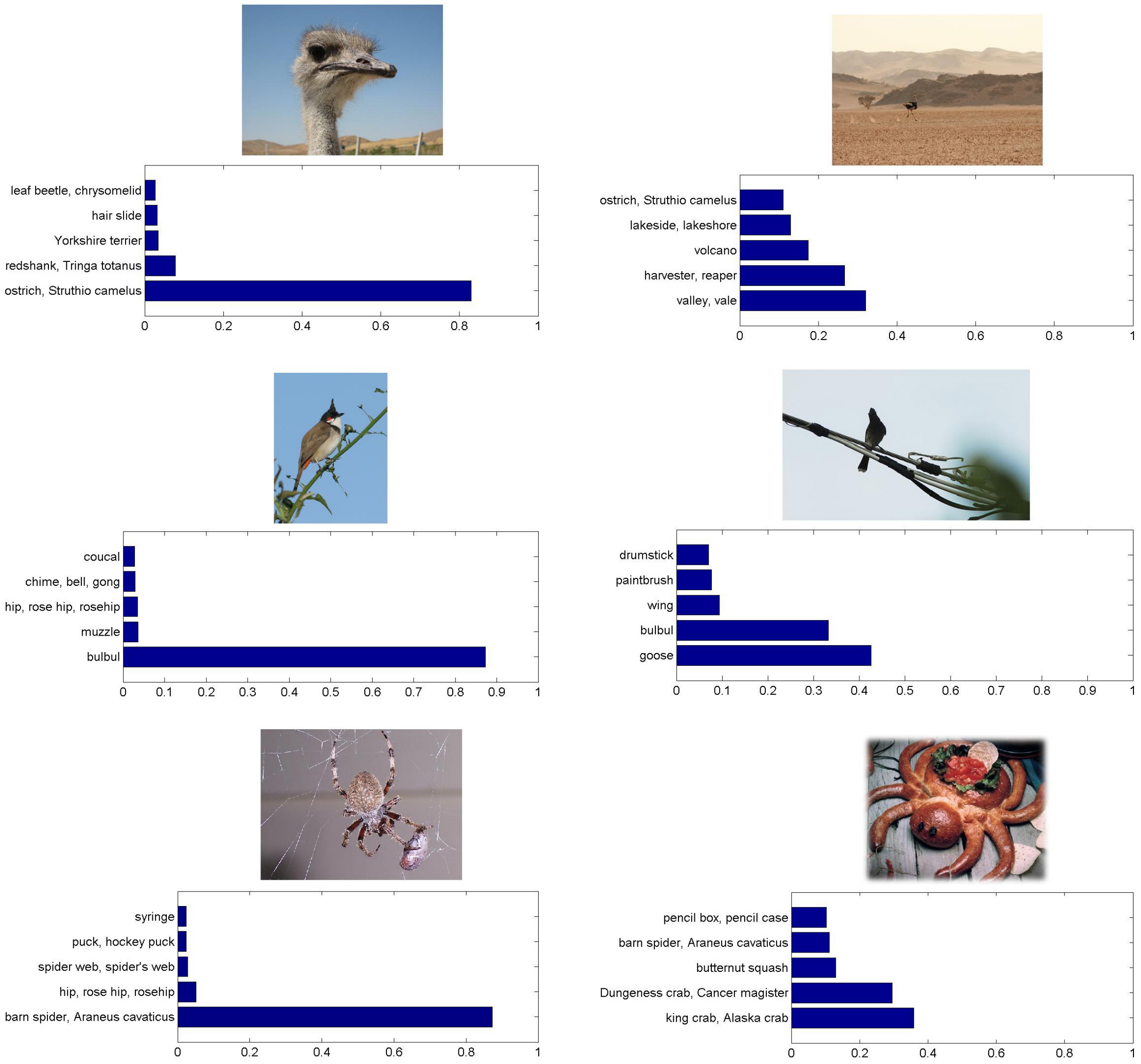


Figure 6. Examples of object classes that deliver the lowest top-5 accuracy in ImageNet-196.



(a)

(b)

Figure 5. Examples of testing images and predicted likelihood outputs for class “ostrich”, “bulbul” and “barn spider”. Each row denotes one class, where the left example shows an easy case and the right one shows a hard case.

Instead, the choice of sparsity itself plays a key role, where optimization of the $\ell 1$-regularized sparse problem in general is superior to greedy approximation to the $\ell 0$-based sparse problem in our task. In fact, even using an unlearned dictionary with imprinted random patches, once we choose suitable algorithms for sparse representation, the performance is still comparable to those with expensive trained dictionaries.

Future work will be focused on scaling up the current framework to handle more classes in the ImageNet. As our learning framework provides confidence likelihood regarding each class, a hierarchical decision model that assesses the distribution of confidence outputs and targets on difficult case for further fine-tuned classification is promising to boost the performance.

Acknowledgement

The author would like to thank James Theiler, Rick Chartrand, Garrett Kenyon, Steven P. Brumby and other DSGM members in Los Alamos National Laboratory for their helpful discussion and comments.

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse repre-

sentation. *IEEE Trans. on Signal Processing*, 54(11):4311–4322, 2006. 2, 3

[2] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *Proc. the 19th International Conference on Computational Statistics*, 2010. 1

[3] S. Chen, D. Donoho, , and M. Saunders. Automatic decomposition by basis pursuit. *SIAM Journal of Scientific Computation*, 1(3):33–61, 1998. 1

[4] A. Coates and A. Ng. The importance of encoding versus training with sparse coding and vector quantization. In *ICML*, 2011. 2, 5

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 1

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 4

[7] K. Engan, S. O. Aase, and J. H. Husoy. Method of optimal directions for frame design. In *IEEE International Confrence on Acoustics, Speech and Signal Processing*, 1999. 2

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 1

[9] 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 *CVPR Workshop on Generative-Model Based Vision*, 2004. 1, 4

[10] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, 2005. 1

[11] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using focuss: a re-weighted minimum norm algorithm. *IEEE Transactions on Signal Processing*, 45(3):600–616, 1997. 1

[12] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. 1

[13] Z. Ji, W. Huang, and S. Brumby. Learning sparse representation via a nonlinear shrinkage encoder and a linear sparse decoder. In *IEEE International Joint Conference on Neural Networks*, 2012. 2

[14] Z. Ji, W. Huang, G. Kenyon, and L. M. A. Bettencourt. Hierarchical discriminative sparse coding via bidirectional connections. In *IJCNN*, 2011. 2

[15] A. Krizhevsky. Learning multiple layers of features from tiny images. In *Master Thesis, Dept. of Comp. Sci.*, 2009. 1

[16] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[17] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006. 1, 4

[18] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of IEEE*, 86(11):2278–2324, 1998. 1

[19] Y. LeCun, F. J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *CVPR*, 2004. 1

[20] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *NIPS*, pages 1137–1144, 2007. 2, 3

[21] M. S. Lewicki and T. J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12(2):337–365, 2000. 2

[22] Y. Lin, L. Cao, F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, and T. Huang. Large-scale image classification: Fast feature extraction and svm training. In *CVPR*, 2011. 1, 4

[23] D. G. Lowe. Distinctive image features from scale invariant keypoints. volume 60, pages 91–110, 2004. 1

[24] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning*, 11:19–60, 2010. 2, 3

[25] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionsaries. *IEEE Trans. Signal Processing*, 41:3397–3415, 1993. 1

[26] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy used by V1? *Vision Research*, 37(23):3311–3325, 1997. 2

[27] M. Osborne, B. Presnell, and B. Turlach. A new approach to variable selection in least squares problems. *IMA Journal of Numerical Analysis*, 20:389–403, 2000. 3

[28] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *The 27th Asilomar Conf. on Signals, Systems, and Computers*, 1993. 1, 3

[29] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007. 2

[30] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *Proc. European Conference on Computer Vision*, 2010. 2

[31] J. Sánchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In *CVPR*, 2011. 1, 2, 3, 4

[32] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996. 1

[33] S. Turaga, J. Murray, F. R. V. Jain, M. Helmstaedter, K. Briggman, W. Denk, and H. Seung. Convolutional networks can learn to generate affinity graphs for image segmentation. *Neural Computation*, 22(2):511–538, 2010. 1

[34] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Guo. Locality-constrained linear coding for image classification. In *CVPR*, 2010. 1

[35] X. Wang, T. X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *ICCV*, 2009. 1

[36] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, 3rd edition, 2011. 4

[37] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 2009. 1, 5