Six International Conference on Advanced Computational Intelligence
October 19-21, 2013, Hangzhou, China

Modeling Outer Products of Features for Image Classification

Peng Qi, Shuochen Su, and Xiaolin Hu

Abstract— **Recent studies have shown that sparse coding is an efficient method for feature quantization in image classification tasks. However, sparse coding can only capture linear statistical regularities among the features. In the paper, we show that features can be quantized in a nonlinear way by modeling their outer products. Experiments on some public datasets show that the proposed method can achieve comparable or better results than sparse coding.**

I. Introduction

IMAGE classification is one of the fundamental problems in the field of computer vision. The state-of-the-art image classification models usually follow the same pipeline from input images to classifiers: *feature extraction*, *feature quantization*, and *feature pooling* (see Fig. 1).

Since raw image pixels bear very limited information about the content of an image, informative image features are usually extracted from the input images to reflect local image characteristics. Many feature extraction algorithms for images have been proposed in the literature in the past decades, among which Scale Invariant Feature Transform (SIFT) [1], Histograms of Oriented Gradients (HOG) [2], and their variants are widely used. These features have some invariance to scale, rotation, translation changes of images, which are main difficulties of image classification. These features are usually called "handcrafted features" as they are designed based on empirical observations or engineering considerations. In recent years, a number of unsupervised models have been developed to learn low-level features automatically and achieved promising results on some image classification benchmarks [3], [4], [5].

The image features are descriptive vectors of local image patches. The extracted features are usually vector quantized to provide a higher-level description of images. Usually, a *codebook* is trained on features. A typical approach for codebook training is to run k-means clustering algorithm on the extracted features, and construct the codebook with the K centroids. Then, all features are vector-quantized based on this codebook. Each feature is then represented by a K dimensional vector with one element being 1 and others being 0.

After quantization, the image features are still local. That is, they are informative merely within an image patch considered. Hence the features are usually pooled to provide invariance to small local changes, and introduce spatial interaction between image patches for better describing the entire image. Common techniques used in feature pooling include dimension-wise average, maximum, and histogram, and spatial pyramid matching (SPM) [6]. SPM is commonly used in state-of-the-art image classification models, which divides the image features into different scales of regular grids with regard to their positions, pools image features within each grid, and finally concatenates the poolings to form the final feature (see Fig. 2). Despite its simplicity in operation, SPM introduces valuable spatial information about the input image, thus is especially useful in object recognition tasks. However, due to the simple algorithms used in the feature quantization step, the SPM requires nonlinear classifiers (e.g. kernel SVMs), which hinders it from scaling to larger-scale image classification tasks.

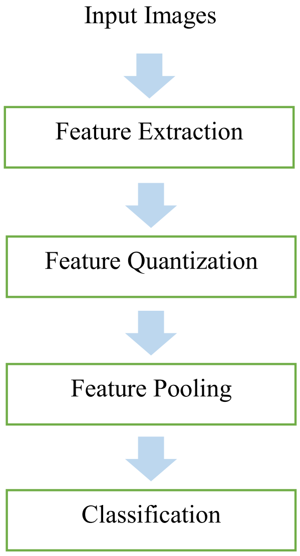


Fig. 1. A general framework for many image classification models

Yang *et al.* proposed to apply sparse coding to the feature quantization step [7], which achieved better results with a linear classifier. The resulting model, ScSPM, reduced the training complexity of SPM from $O(n^3)$ to $O(n)$, and testing complexity from $O(n)$ to $O(1)$, where n is the number of training instances. This renders ScSPM especially effective when large-scale datasets are considered. However, we note that in ScSPM, an absolute value rectification (AVR) is applied to the output of sparse coding, which is important to the effectiveness of this model (see Section IV).

Recently we proposed to model the outer product of the input data in order to capture its higher-order statistics, which has led to higher-level representation of the input [8]. The

The authors are with the State Key Laboratory of Intelligent Systems and Technology, Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China. (Email: xlhu@tsinghua.edu.cn)

This work was supported by the National Basic Research Program (973 Program) of China (Grant Nos. 2013CB329403 and 2012CB316301), National Natural Science Foundation of China (Grant No. 61273023), Beijing Natural Science Foundation (Grant No. 4132046) and TNList Cross-discipline Foundation.

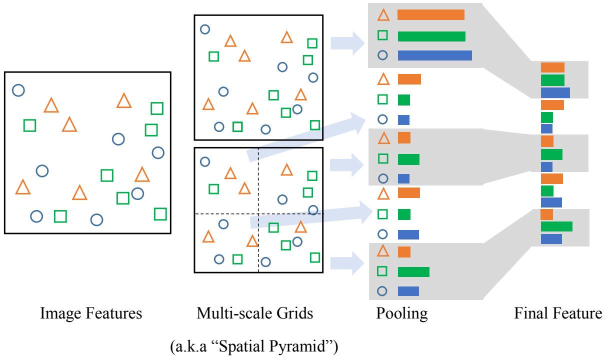


Fig. 2. Illustration of spatial pyramid matching

model units exhibited similar properties of complex cells in the primary visual cortex (V1). In this paper we present an application to image classification. We found that when used for feature quantization, this algorithm achieved comparable results to sparse coding without the need to explicitly perform AVR, which introduced potentially undesirable correlation between feature dimensions. Since the feature quantization step is compatible with the ScSPM framework, the model can gracefully scale up to large datasets.

The rest of this paper is organized as follows. In Section II, we briefly review some typical image classification models. The outer product algorithm is elaborated in Section III, and experiment results on public image classification datasets are presented in Section IV. Finally, Section V concludes the paper.

II. Sparse Coding and ScSPM

We briefly review sparse coding [9], [10] and ScSPM [7] as they are closely related to the our model.

Sparse coding was first introduced by Olshausen and Field to model properties of simple cells in V1 [9], [10]. Given an image patch $\mathbf{x}$, the goal of sparse coding is to find a basis matrix $\mathbf{B}$ that reconstructs the input with a sparse code $\mathbf{y}$, i.e. $\mathbf{x} \approx \mathbf{B}\mathbf{y}$. Formally, it solves the minimization problem

$$\text{minimize}_{\mathbf{B},\mathbf{y}} \; ||\mathbf{x} - \mathbf{B}\mathbf{y}||_{\mathcal{F}}^2 + \lambda \sum_k |y_k| \qquad (1)$$
$$s.t. \; \forall k, ||\mathbf{B}_k||_{\mathcal{F}}^2 \leq 1$$

where $|| \cdot ||_{\mathcal{F}}$ denotes the Frobenius norm, $\mathbf{B}_k$ is the k-th column of the bases matrix, and the second term is a first-norm regularization term to ensure that the codes $\mathbf{y}$ are sparse, i.e. remain zero most of the time and deviate from zero only occasionally. It has been demonstrated that when trained on natural image patches, this algorithm is capable of learning edge detectors that resemble the receptive fields of V1 neurons.

ScSPM is a SPM model well suited for linear classifiers. After densely extracting SIFT features from images, a codebook is learned with sparse coding, and the features are quantized with sparse coding as well. In the subsequent feature pooling step, the features from the spatial pyramid are max-pooled to form a representative feature. Finally, the SPM features are sent to a linear SVM for classification. ScSPM differs from previous SPM models in using sparse coding instead of K-means, max pooling instead of histograms, and linear SVM instead of kernel SVM. Such modifications have been shown to give the model not only the capacity to process larger-scale datasets with reduced computational complexity, but also the capability to achieve higher classification rates on various public datasets than models that used spatial-pyramid histograms and χ^2 kernels.

There are also evidences [11] showing that in such models, codebook training may not be as important as feature quantization in many scenarios.

III. Modeling Outer Products of Features

To be self-contained, we briefly review the outer product model [8] first. Then, we demonstrate the advantage of this model over sparse coding with a toy problem, and discuss how it can be used for image classification.

A. The Outer Product Model

Based on the statistics of outer products of natural image patches, we presented a model to reproduce properties of complex cells in V1 [8], which are known to be selective to orientations but invariant to phase changes. Stemmed from the discovery that natural image patches can be characterized by their covariance in the space of linear filter responses [12], this model has a simpler form considering that the outer product is closely related to the empirical covariance matrix. Specifically, for an input data vector $\mathbf{x}$, we attempt to reconstruct its outer product $\mathbf{x}\mathbf{x}^\top$ with a basis matrix $\mathbf{B}$ and a sparse code $\mathbf{y}$, i.e. $\mathbf{x}\mathbf{x}^\top \approx \mathbf{B}\,\text{diag}(\mathbf{y})\mathbf{B}^\top$. Formally, the problem is defined as

$$\text{minimize}_{\mathbf{B},\mathbf{y}} \; ||\mathbf{x}\mathbf{x}^\top - \mathbf{B}\,\text{diag}(\mathbf{y})\mathbf{B}^\top||_{\mathcal{F}}^2 + \lambda \sum_k |y_k| \qquad (2)$$
$$s.t. \; \forall k, ||\mathbf{B}_k||_{\mathcal{F}}^2 \leq 1 \text{ and } y_k \geq 0$$

where $\text{diag}(\cdot)$ denotes the operation that transforms a vector to a corresponding diagonal matrix. Note that despite the similarity to (1), this model incorporates nonlinear statistical information by modeling the outer product of the input data.

It can be shown that the inference problem is equivalent to

$$\text{minimize}_{\mathbf{y}} \; \frac{1}{2}\mathbf{y}^\top \mathbf{A}\mathbf{y} + \mathbf{b}^\top \mathbf{y} \qquad (3)$$
$$s.t. \; \forall k, y_k \geq 0$$

where

$$\mathbf{A} = (\mathbf{B}^\top\mathbf{B}) \circ (\mathbf{B}^\top\mathbf{B}) \quad \mathbf{b} = -(\mathbf{B}^\top\mathbf{x}) \circ (\mathbf{B}^\top\mathbf{x}) + \lambda\mathbf{1} \qquad (4)$$

where $\circ$ denotes the Hadamard (element-wise) matrix multiplication, and $\mathbf{1}$ denotes the vector with all 1's. Many efficient algorithms have been proposed for problems of such form, including conjugate gradient descent, Lagrangian relaxation, etc. We implemented a nonnegative version of the feature-sign algorithm [13] which is elaborated in Algorithm 1.

Algorithm 1 Feature-sign algorithm for solving (3)

```
1 Initialize $\mathbf{y} := \mathbf{0}$, $\theta := \mathbf{0}$, and $active\ set := \{\}$, where
  $\theta_i \in \{0, 1\}$ denotes $\mathrm{sign}(y_i)$.
2 From zero coefficients of $y_i$, select
  $i = \arg\max_i -\frac{\partial(1/2\mathbf{y}^\top \mathbf{A}\mathbf{y}+\mathbf{b}^\top \mathbf{y})}{\partial y_i}$
  If $-\frac{\partial(1/2\mathbf{y}^\top \mathbf{A}\mathbf{y}+\mathbf{b}^\top \mathbf{y})}{\partial y_i} > 0$, then set $\theta_i := 1$, $active\ set := \{i\} \cup active\ set$
3 Feature-sign step:
    Let $\hat{\mathbf{A}}$ denote a submatrix of $\mathbf{A}$ that contains only the rows and columns corresponding to the $active\ set$.
    Let $\hat{\mathbf{b}}$, $\hat{\mathbf{y}}$ and $\hat{\theta}$ be subvectors of $\mathbf{b}$, $\mathbf{y}$ and $\theta$ corresponding to the $active\ set$.
    Compute the analytical solution to the resulting unconstrained QP ($\mathrm{minimize}_{\hat{\mathbf{y}}}\ \frac{1}{2}\hat{\mathbf{y}}^\top \hat{\mathbf{A}}\hat{\mathbf{y}} + \hat{\mathbf{b}}^\top \hat{\mathbf{y}}$)
        $\hat{\mathbf{y}}_{new} := -\hat{\mathbf{A}}^{-1}\hat{\mathbf{b}}$
    Perform a line search on the closed line segment from $\hat{\mathbf{y}}$ to $\hat{\mathbf{y}}_{new}$:
      Check the objective value at $\hat{\mathbf{y}}_{new}$ and all points where any coefficient changes sign (while the rest remain nonnegative).
      Update $\hat{\mathbf{y}}$ (and the corresponding entries in $\mathbf{y}$) to the point with the lowest objective value.
    Remove zero coefficients of $\hat{\mathbf{y}}$ from the $active\ set$ and update $\theta := \mathrm{sign}(\mathbf{y})$
4 Check Optimality:
  (a) Optimality condition for nonzero coefficients:
      $\frac{\partial(1/2\mathbf{y}^\top \mathbf{A}\mathbf{y}+\mathbf{b}^\top \mathbf{y})}{\partial y_i} = 0, \forall y_j \neq 0$
      If condition (a) is not satisfied, goto Step 3 (without any new activation); else check condition (b)
  (b) Optimality condition for zero coefficients:
      $-\frac{\partial(1/2\mathbf{y}^\top \mathbf{A}\mathbf{y}+\mathbf{b}^\top \mathbf{y})}{\partial y_i} \leq 0, \forall y_j = 0$
      If condition (b) is not satisfied, goto Step 2; else return $\mathbf{y}$ as the solution.
```

Following the proof sketch in [13], and noting that (3) is convex, it is straightforward to prove that this modified feature-sign algorithm converges to the global optimum of the problem, as each feature-sign step strictly lowers the objective function until no improvements can be made. A rigorous proof can be found in [8].

To learn the bases of the outer product model, the Lagrangian dual method proposed in [13] cannot not be used in our case because (2) is non-convex in $\mathbf{B}$. Therefore, we used the stochastic gradient descent algorithm.

B. Comparison with Sparse Coding

To demonstrate the ability of the outer product model for capturing higher-level statistical regularities, two sets of data (1000 points each) were generated from two zero-mean Gaussian distributions with covariance matrices $\mathbf{C}_1 = \mathbf{B}\,\mathrm{diag}(\mathbf{y}_1)\mathbf{B}^\top$ and $\mathbf{C}_2 = \mathbf{B}\,\mathrm{diag}(\mathbf{y}_2)\mathbf{B}^\top$, where

$$\mathbf{y_1} = [1\ 0.1]^\top, \mathbf{y_2} = [0.1\ 1]^\top, \text{ and } \mathbf{B} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}.$$

Fig. 3(a) shows the two set of points with "+" and "x", respectively. On these data, we trained a sparse coding (SC) model and an outer product (OP) model with two bases separately with the same regularization parameter λ. After learning the bases, the features were inferred from the models. Fig. 3 shows the resulting two-dimensional features.

Both algorithms were capable of finding the underlying bases of the data. The inferred feature vectors were aligned with the axes in the feature space (Fig. 3). The features of the OP model were approximately linearly separable (Fig. 3(d)), while those of SC were not (Fig. 3(b)). This suggests that when no auxiliary procedures are present, OP is capable of mapping the input data to a feature space that is more suitable for linear classifiers than SC.

Fig. 3(c) suggests that absolute value rectification (AVR) might be a solution for the linear separability of SC. However, this operation has increased the correlation between the two feature dimensions, which can be verified by explicitly showing the correlation coefficients: $\rho_{\mathrm{SC}} = 0.001$, $\rho_{\mathrm{SC+AVR}} = -0.230$, $\rho_{\mathrm{OP}} = -0.125$. The near-zero correlation between SC feature dimensions is a result of the cancellation between positive and negative correlations; see Fig. 3(b). AVR introduced extra correlation between SC feature dimensions, which was greater than the correlation between OP feature dimensions.

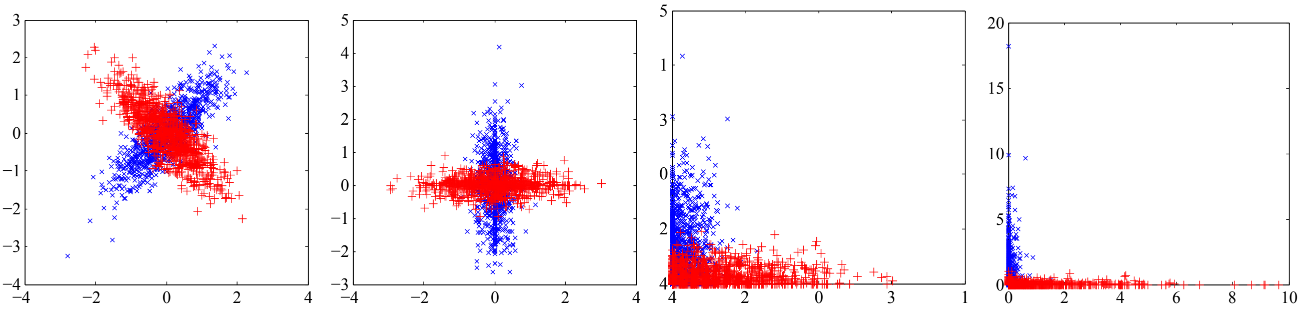
C. Integrating into the Image Classification Framework

Clearly, the OP algorithm can be also used for feature quantization for image classification. The procedure is illustrated in Fig. 1. First, image features (e.g. SIFT) are extracted. Then the OP model is trained on the features to obtain a codebook, i.e., a basis matrix $\mathbf{B}$ in (2). With this codebook, OP inference is performed for each feature, which nonlinearly maps the features from the original space to another space. Hopefully, some higher-level description of images would be resulted in. According to the experiments on the toy data, the resulting representations of images should be less redundant and prone to be more linearly separable than that obtained with sparse coding and AVR. As a consequence, the classification performance should be at least comparable to, if not better than, that of sparse coding.

IV. Experiments

We tested the SPM model with the OP model on two public datasets: Caltech 101 [14] and 15 Scenes [6] (see Fig. 4). Each image was linearly rescaled so that the longer dimension did not exceed 300 pixels before extracting image features. A multi-class linear SVM was used to perform classification, and average results over 10 randomly partitioned training and testing data were reported.

The following algorithms were compared in terms of classification accuracy:



(a) Input data

(b) Results from SC

(c) Results from SC+AVR

(d) Results from OP

Fig. 3. Comparison between sparse coding and outer product models on the toy data. See text for detailed explanations

1) LSPM: linear SPM that uses linear kernel on spatial pyramid histograms [7]
2) KSPM: nonlinear SPM that uses spatial pyramid histograms and χ^2 kernels [6]
3) ScSPM: linear SPM that uses sparse coding for feature quantization [7]
4) OpSPM: linear SPM that uses the outer product model for feature quantization

All models used SIFT features. We cited results from [6] and [7]. To assess the contribution of AVR in ScSPM models, we also reported the results of ScSPM without this operation.

A. Classification Results on Caltech-101

The Caltech-101 dataset includes 9144 images belonging to 102 categories including animals, plants, and various man-made objects. See Fig. 4(a) for some sample images. The images are medium-sized (approximately 300 × 300 pixels). The number of images per class varies from 31 to 800. We follow the common setup on this dataset: randomly pick 15 or 30 images per class for training and the rest for testing. Detailed experiment results are reported in Table I.

On this dataset, OpSPM achieved comparable results to ScSPM especially in the 30 training samples case. If AVR step was not involved, the performance of ScSPM dropped. This is consistent with the observations on the toy data. We would like to point out that it was max pooling used in the model that prevented the performance from dropping too much. This is because max pooling tends to output positive features, and the classifier main operates in the nonnegative quadrant of the feature space. Note that the features in the nonnegative quadrant are more linearly separable than in the entire space, which is clear in Fig. 3(b).

B. Classification Results on Fifteen Scenes

The fifteen scenes dataset contains 15 classes of 4485 medium-sized images. Each class contains 200 to 400 images, with categories varying from living room and kitchen to street and industrial. See Fig. 4(b) for some sample images. Following Lazebnik *et al.*'s [6] experiment procedures, we trained the models on 100 images per class and tested on the rest. The detailed comparison is presented in Table II.

On this dataset, OpSPM surpassed ScSPM by a clear margin, reducing classification error of the latter by about 8%. The success of OpSPM on this dataset was perhaps due to the high variance of the dataset. The SIFT features extracted from the fifteen scenes dataset may require the feature quantization algorithm to capture more higher-level regularities and thus achieve a more efficient representation of images. The formulation of the OP model allowed more higher-level statistical information to be extracted, thus was suitable for the task. In contrast, most images from Caltech-101 are aligned. That might be why OpSPM did not perform better than ScSPM on this dataset.

Again, we found that the absence of AVR deteriorated the performance of ScSPM, though not too much. See Table II.

TABLE I
AVERAGE CLASSIFICATION ACCURACY AND STANDARD ERROR (%) ON CALTECH-101

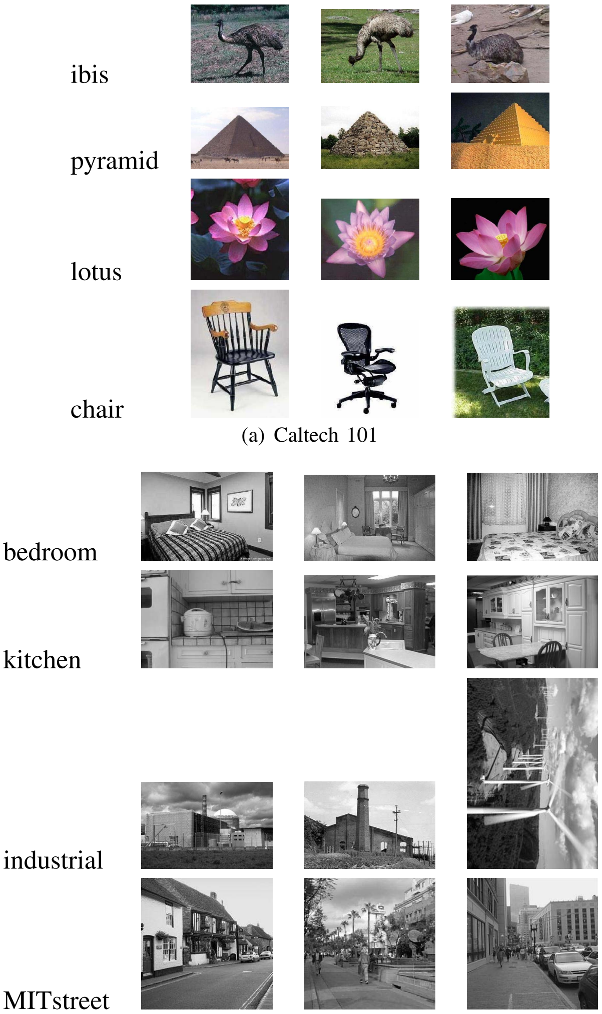
Algorithm	15train	30train
LSPM [7]	53.23(0.65)	58.81(1.51)
KSPM [7]	56.44(0.78)	63.99(0.88)
KSPM [6]	56.40	64.40(0.80)
ScSPM [7]	**67.00(0.45)**	73.20(0.54)
ScSPM (w/o AVR)	65.49(1.26)	71.77(1.18)
OpSPM	65.37(0.60)	**73.23(0.88)**

TABLE II
AVERAGE CLASSIFICATION ACCURACY AND STANDARD ERROR (%) ON 15 SCENES

Algorithm	Classification Rate
LSPM [7]	65.32(1.02)
KSPM [7]	76.73(0.65)
KSPM [6]	81.40(0.50)
ScSPM [7]	80.28(0.93)
ScSPM (w/o AVR)	80.09(0.59)
OpSPM	**81.85(0.66)**

V. CONCLUDING REMARKS

In the paper we show that by modeling the outer product of features, the features can be quantized in a nonlinear way, which has led to better results than sparse coding on the 15 Scenes dataset for image classification. Unlike sparse coding, this method does not need the absolute value rectification on its output. But on the Caltech-101 dataset,



(a) Caltech 101

(b) 15 Scenes

Fig. 4. Sample images from the two datasets

the proposed method achieved comparable, not better, results. Since Caltech-101 images are more aligned than 15 Scenes images, these findings suggest that the proposed method would be more suitable for datasets having more within-category variances.

In this study, the OP model was applied on SIFT features. It is worth investigating the possibility of applying the model on raw image pixels and perform image classification in future.

References

[1] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the 7th IEEE International Conference on Computer Vision*, vol. 2. IEEE, 1999, pp. 1150–1157.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.

[3] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006.

[4] H. Lee, C. Ekanadham, and A. Ng, "Sparse deep belief net model for visual area V2," in *Advances in Neural Information Processing Systems (NIPS)*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds., vol. 20, Vancouver, Canada, Dec. 2007.

[5] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 609–616.

[6] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2006, pp. 2169–2178.

[7] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Computer Vision and Pattern Recognition, 2009*. IEEE, 2009, pp. 1794–1801.

[8] P. Qi and X. Hu, "Learning nonlinear regularities in natural images by modeling the outer product of image intensities," 2013, submitted elsewhere.

[9] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, pp. 607–609, June 1996.

[10] ——, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[11] A. Coates and A. Y. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *International Conference on Machine Learning*, vol. 8, 2011, p. 10.

[12] Y. Karklin and M. S. Lewicki, "Emergence of complex cell properties by learning to generalize in natural scenes," *Nature*, vol. 457, no. 7225, pp. 83–86, 2008.

[13] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," *Advances in Neural Information Processing Systems (NIPS)*, vol. 19, p. 801, 2007.

[14] 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 *Computer Vision and Pattern Recognition Workshop*. IEEE, 2004, pp. 178–178.