Sparse Coding for Object Recognition

Stefan Lee

September 2nd, 2013

Abstract

In recent years, the application of sparse coding techniques has led to frameworks that match or set the state-of-the-art in object recognition tasks. Despite such success, applying sparse coding to vision tasks presents unique challenges and many papers addressing these concerns appear in top conferences annually. This paper acts as an introduction to the subject of sparse coding, identifies the key research areas improving its applicability to recognition tasks, and surveys recent approaches to treating these challenges.

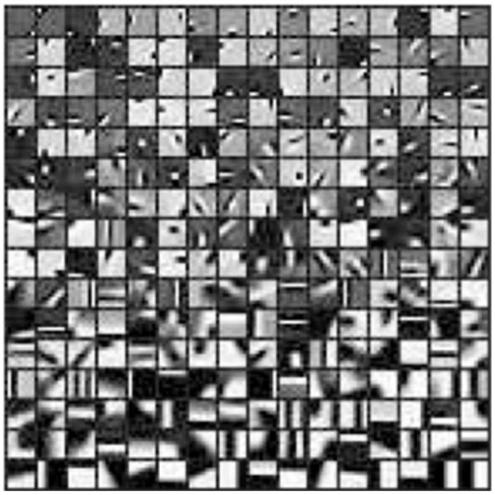

Figure 1: Example learned dictionary taken from [14]. Note how many of the bases appear to be Gabor filters.

1 Introduction

The study of sparse coding techniques has a complex history with roots in statistics [41, 37, 8, 12], neuroscience [39, 40] , and signal processing [7, 6] (where it is often known as compressed sensing). The general goal of sparse coding is to find a "good" reconstruction of an input signal using a linear combination of only a "few" elements taken from some dictionary. In early work with signal compression, the dictionary was often manually designed as a set of wavelets or Gabor filters.

In their 1997 seminal work, Olshausen and Field [40] extend their earlier work [39] by applying sparse coding to learn an overcomplete dictionary from natural image patches. The resulting dictionary shared properties thought to be at work in the human visual cortex (i.e. the basis are localized, oriented, and band-pass). Building on this formulation, works by Elad et al. [14] and Raina et al. [42] in 2006 helped to introduce sparse coding techniques to common problem frameworks in vision like hand-written character recognition, image denoising, and object recognition.

Since then, sparse coding techniques have been applied with great success to both low and high level vision tasks, including face recognition [53], image classification [45, 47, 50, 25], image denoising and inpainting [14, 2, 36], and anomaly detection in video [58]. Much of this success has been attributed to learning the dictionary from unlabeled task specific training data instead of using predesigned basis sets [2, 24, 42, 38]. This theme of transfer learning has been seen before in computer vision as the popular bag-of-words (BoW) models [10] and in fact, sparse coding can fit quite well as a substitute to BoW models in modern classification architectures [50, 4, 25, 42].

Despite their success, applying sparse coding techniques to recognition problems in vision gives rise to unique challenges, which are separate from the concerns over reconstruction fidelity which dominated their early development. This paper covers recent work on three of the most prominent challenges being treated by researches:

Invariance and Robustness
Sparse coding is unstable with respect to translation and rotation when applied to images. Even small changes can result in completely different sets of optimal reconstruction basis, such that similar image patches may end up distant in the encoding. This can have negative effects when sparse coding is used as a feature extraction step in classification or recognition tasks [52, 47, 26, 21]. Additionally, the most common form of the sparse coding problem implicitly assumes Gaussian noise in the reconstruction error and is prone to overfit to poorly reconstructed patches [53].

Supervised Discriminative Dictionary Learning
Although it performs quite well in classification frameworks, sparse coding is designed to minimize the reconstruction error under sparsity constraints and not to be used as a discriminative feature encoding. Further improvements can be made by incorporating class labels and differentiable classifiers into the sparse coding formulations to explicitly encourage discriminative dictionaries [35, 25, 52, 5].

Efficient Sparse Coding for Large Datasets
Many classification problems in vision are burdened by tremendous quantities of data. This is due to both the fact that images are implicitly high dimensional and also that the best performing techniques require dense sampling of image regions. Specialized algorithms for learning and encoding can help improve applicability of sparse coding techniques to large scale or real time vision problems [34, 45, 58, 51, 22, 51].

The structure of the paper from this point will be as follows. Section 2 covers the general theory of sparse coding. Sections 3, 4, and 5 each discuss recent work in one of the identified research areas. Section 6 concludes by discussing future research opportunities to improve the applicability of sparse coding to vision problems and specifically object recognition.

2 Terminology and Common Formulations

As would be expected of a subject with such diverse source and application domains, the terminology and formulation varies between authors and fields. One goal of this paper, in addition to familiarizing the reader with the history and current research questions in sparse coding, is to help disambiguate the differing terminology. The most common terminology in the vision community will be used, but footnotes will provide alternative syntax seen during the literature review.

In modern vision frameworks, sparse coding is divided into two sub-problems; learning a dictionary that well represents the training data and encoding input signals sparsely. In the following two subsections these problems are defined and discussed in their most common formulations.

2.1 Sparse Reconstruction

Concretely, the goal of sparse coding, given a signal vector $x_i \in \mathbb{R}^k$ and a highly overcomplete[1] dictionary $D \in \mathbb{R}^{k \times m}$ where each column d_i is called a basis [2], is to produce the sparsest linear coefficients

[1] $m >> k$

[2] The term basis is an abuse of notation here in that the dictionary is overcomplete. Other terms used include receptive fields, prototypes, atoms, basis vectors, basis elements, visual words, and codewords.

over the columns of D that best reconstruct x_i. This optimal encoding is given as the solution to the constrained optimization in equation (1).

$$\begin{aligned} \alpha^* &= \underset{\alpha \in \mathbb{R}^m}{\arg\min} \overbrace{\Psi(\alpha)}^{\text{Sparsity}} \\ &\quad \text{s.t. } \underbrace{f(x, D\alpha)}_{\text{Coding Error}} = 0 \end{aligned} \tag{1}$$

This optimization is usually presented as its Lagrangian dual as in equation (2). [3]

$$\alpha^* = \underset{\alpha \in \mathbb{R}^m}{\arg\min} f(x, D\alpha) + \lambda \Psi(\alpha) \tag{2}$$

For most applications, the coding error $f(x, D\alpha)$ takes the form of the sum of squared error, $||x - D\alpha||_2^2$, which makes implicit assumptions about the Gaussian distribution of reconstruction error.[4] The choice for the sparsity function $\Psi(\alpha)$ divides the domain into hard and soft sparseness.

When a non-differentiable function is used, the encoding is considered a hard sparseness problem. The most commonly used hard sparsity constraint is the ℓ_0-quasi-norm, denoted $||\alpha||_0$, which counts the nonzero elements of a vector. When this is used, the optimization becomes a combinatorial problem and producing exact solutions is NP-hard [45, 44]. Many greedy and approximate algorithms have been developed to provide good solutions in practice, such as pursuit algorithms [41, 37, 8] and active set methods [29].

When $\Psi(\alpha)$ is selected as a differentiable function, this formulation is considered a soft sparseness problem and traditional continuous optimization algorithms like gradient descent can be applied. In particular, when the ℓ_1 norm is used in conjunction with the sum of squares reconstruction error, the problem reduces to ℓ_1-regularized linear regression for which many specialized algorithms have been developed. Some of the more popular in the literature are the LASSO method [46], least angle regression [13], pathwise coordinate optimization [20], the iterative shrinkage and thresholding algorithm [11, 3], and coordinate descent [32].

For most sparse coding applications in vision, soft sparseness is the preferred formulation as it provides more freedom in selecting optimization approaches and because it has been shown that for most large underdetermined systems, the ℓ_1-norm constrained solution is also the sparsest [12].

2.2 Dictionary Learning

The solution to the overcomplete dictionary learning problem given a dataset $X = \{x_i\}_{i=1}^n$ for sparse coding can be expressed as the dictionary with the minimum reconstruction error and sparsity summed over the whole of X. The two most common formulations are show in equations (3) and (4) corresponding to soft and hard sparseness respectively.

Soft Sparseness

$$\begin{aligned} D^* &= \underset{D}{\arg\min} \sum_{x_i \in X} \min_{\alpha} ||x_i - D\alpha||_2^2 + \lambda ||\alpha||_1 \\ &\quad \text{s.t. } ||d_i||_2^2 = 1, \forall i = 1, ..., m \end{aligned} \tag{3}$$

The regularization of the columns of D is to guard against trivial solutions where the basis become large to reduce the magnitude of the corresponding α's.

Hard Sparseness

$$\begin{aligned} D^* &= \underset{D}{\arg\min} \sum_{x_i \in X} \min_{\alpha} ||x_i - D\alpha||_2^2 \\ &\quad \text{s.t. } \quad ||\alpha||_0 < T \end{aligned} \tag{4}$$

These are difficult non-convex optimization problems; however, each can be decomposed into two sub-problems. When the encodings are fixed, the

[3] Note that the λ here is the reciprocal of the Lagrange multiplier that would be present if the Lagrange method were applied directly to (1).

[4] See Lewicki and Sejnowski [30] for in-depth discussion of the underlying probabilistic framework.

optimization reduces to a convex least squares problem, and when D is fixed the problem of encoding is solved as in the above section. Many algorithms take advantage of this and alternate between the two subproblems until convergence [2, 29, 38, 26, 55, 51]. Other approaches build on the probabilistic framework of Olshausen and Field [40] and explicitly model distribution assumptions to arrive at analytic solutions [5, 30].

One of the most influential approaches to solving both the hard and soft sparseness problems is the K-SVD algorithm presented by Aharon, Elad, and Bruckstein [2]. K-SVD acts as a generalization of the well-known k-mean's algorithm and, as the name implies, makes use of k singular value decompositions to update the dictionary. The algorithm alternates between encoding and dictionary update steps; moreover, the dictionary updates are done sequentially for each column d_k of D by fixing all other columns. This update is done by finding the singular value decomposition of the matrix of reconstruction errors when d_k would have been used, but is withheld. When a column is updated, corresponding alterations to codes using that column are made. This has been shown to greatly improving the convergence rate as further columns updates within one dictionary update step benefit from the new information. Aside from its quick convergence, another advantage of K-SVD is that it is flexible enough to utilize any encoding technique during the update phase. Many newer approaches have made use of the K-SVD algorithm and multiple enhancements have been proposed [24, 38, 25] .

3 Inducing Invariance and Robustness

Advancements in feature pooling techniques, such as spatial pyramid approaches [27], applied to sparse coding based classification and recognition frameworks have helped to alleviate translation variance in feature extraction [55, 50, 4]; however, sparse coding techniques suffer from another source of instability. The general reconstructive sparse coding framework only weakly encourages similar input signals to be encoded by similar sparse codes and in practice it has been shown that only a minor correlation exists between the two similarities. This can have negative consequences in a classification framework, where minor warpings or rotations could result in very different encodings [21]. Two of the recent approaches addressing this problem are presented here.

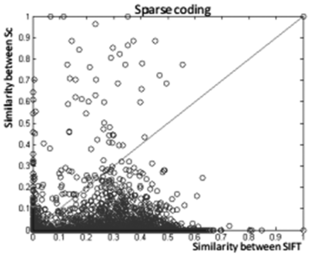

Figure 2: Correlation between similarity of SIFT features and corresponding sparse codes in [21].

Wang et al. [47] presented a sparse coding framework for recognition based on locality-constrained linear coding (LLC) [56]. This scheme encourages similar encodings for similar input signal by weighting the sparsity constraint based on signal and basis similarity. The usual sparsity constraint is replace by $||z_i \odot c_i||^2$ where $\odot$ is the element wise multiplication operator and z_i is a vector of weights based on the distance between the input signal and each basis. This term penalizes large coefficients for basis vectors that are distant from the input in the original signal space. The authors show this modified objective function has a closed form analytic solution; however, computing it can be expensive for large datasets.

The authors also suggest an even more efficient framework that provides a further approximation to the LLC theory. In this setting, a vector is encoded by finding its k-nearest neighbor basis vectors and then solving the much smaller encoding problem of taking just these bases as the dictionary. An iterative algorithm is also presented to optimize the codebook utilizing this encoding scheme. This approximate method was applied to the Caltech 101 [18] and Pascal VOC 2007 [16] datasets. In both they compared favorably to the other contemporary approaches and occasionally yielded higher average accuracy. A comparison with the current leading algorithms can be seen in Figure 3.

A similar framework was proposed by Gao et al. [21]. The approach augments the usual soft sparsity objective function with a term that penalizes the distance between all pairs of encodings in the training set. The individual terms of this summation are weighted by the similarity of the corresponding input signals. As it is impractical to compute pairwise differences between every signal for large training sets, the authors make use of a smaller subset consisting of *"template"* features.

The learning algorithm is initialized by subsampling the training set into a much smaller set of exemplars. A dictionary is learned over this set with the proposed objective function, utilizing all pairwise distances. When a new feature is to be encoded, a weight vector is computed over the set of template features. Specifically the weights for the new signal's k-nearest neighbors are computed by some distance function (in this case histogram intersection) and the other entries are set to zero. In this way features are encouraged to have the same set of encoding basis as similar features in the template space. This encoding method can be used as a subroutine for K-SVD based approaches.

The authors apply this approach, coupled with a spatial pyramid pooling method, to a number of object recognition and scene recognition tasks with competitive results. Again a comparison with other leading results can be seen in Figure 3.

The added accuracy of these methods over similar sparse coding approaches that do not encourage invariance shows that there is room to tune the general reconstructive formulation to better suit its role as a feature extractor in a classification setting.

4 Supervised Discriminative Dictionary Learning

Mairal et al. [35] extended the K-SVD algorithm to learn dictionaries optimized for classification problems. Given some labeled data points $\{x_j, y_j\}_{j=1}^{N}$ from Y classes the approach simultaneously trains a set of dictionaries $\mathbf{D} = \{D_i\}_{i=1}^{Y}$ using a modified objective function defined as

$$D^* = \arg\min_{D} \sum_{i=1}^{N} \sum_{j=1}^{Y} \qquad C_j^{\lambda}\left(\{\mathcal{R}^*(x_i, D_y)\}_{y=1}^{Y}\right) \\ + \quad \mathcal{R}^*(x_i, D_j), \qquad (5)$$

where C_i^{λ} is the softmax cost function and $\mathcal{R}^*(x_i, D_j)$ is the coding error when the optimal α is used. The softmax cost function C_i^{λ} nears its minimum when the ith input element is the smallest values of the set. This formulation produces dictionaries that are good at reconstructing members of their corresponding class, but poor at reconstructing signals from other classes. The authors modify the K-SVD algorithm to optimize this new objective function.

At classification time, the reconstruction error from the dictionaries is directly used to select the labeling (i.e. the assigned label is the label of the dictionary that best reconstructs the input). The inclusion of the C_i^{λ} term can be thought of as a differentiable approximation of the loss function for the min-classifier being used, although the authors do not frame it in this manner. This method can then be understood as directly optimizing the dictionaries for greater performance with a specific classifier - an approach that is popular in more recent work on discriminative dictionaries.

Yang et al. [52] made this idea explicit by developing a framework which optimizes the dictionary along with a set of regularized classifier parameters, w, to minimize a classification loss function. The error function is given in equation (6) where X_i is a training image.

$$E(w, D) = \sum_{i=1}^{N} \ell(y_k, f(\psi(X_i, D), w)) + \lambda ||w||_2^2 \tag{6}$$

In the above equation $\ell(\cdot)$ is the loss function, $f(\cdot)$ is the classifier, and $\psi(X_i, D)$ is the image level feature. $\psi(X_i, D)$ is constructed by combining all of the encoded features extracted from the image. The image level feature could be formed by spatial pyramid pooling for example. The authors present a simple stochastic gradient descent based approach to optimize w and D - utilizing work on back propagation for sparse coding via implicit differentiation by Bradely and Bagnell [5].

The authors present results of this method on multiple face recognition and OCR datasets using a linear SVM and global max pooling of encoded raw image patches. In these tests, the supervised model always outperformed the unsupervised version and often made improvements upwards of 40%. The framework was also compared favorable to other techniques including local coordinate coding, convolutional neural networks, and deep belief networks.

A similar framework using a simple linear predictor with a squared error loss function was developed by Zhang et al. [57]. The K-SVD algorithm is directly used to perform the optimization. This is accomplished by transforming the input and dictionary variables such that

$$\hat{x_i} = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \hat{D} = \begin{bmatrix} D \\ W \end{bmatrix}$$

where y_i is the class label of the ith datapoint and the matrix W is the parameter for the linear predictor, $y_i = Wx_i$. Using the K-SVD framework, both the dictionary and W are optimized to produce encodings that recover the input well and lend themselves more easily to class prediction.

This approach was further refined in 2011 by Jiang, Lin, and Davis [25], producing the Label Consistent K-SVD (LC-KSVD) algorithm. In addition to the squared classification error term, Jiang et al. add a label consistency term that encourages individual dictionary basis to contribute to the reconstruction of training vectors from as few classes as possible - ideally just one. Again the K-SVD algorithm is used to perform the optimization and the authors apply the same concatenation idea presented above to include the label consistency terms. The authors compare the LC-KSVD algorithm to twelve other methods on the Caltech101 object recognition dataset [18] including the approaches by Yang et al. and Zhang et al. presented here. The results show marked improvement over all compared methods.

The success of these approaches in object recognition underpins the benefits of taking holistic approaches, where the encoding framework and classifier provide feedback rather than being trained sequentially.

5 Improving Efficiency for Large Scale Tasks

5.1 Online Learning

Mairal et al. [34] present an online dictionary learning algorithm for sparse coding, which shows impressive gains in efficiency compared to the contemporary batch approaches. The authors also compared computation time with the stochastic gradient descent based solution presented by Aharon and Elad [1]. Both algorithms reconstructed the input signals well; however, this was only after significant tuning of the learning parameter for the stochastic gradient descent method. The online algorithm presented does not have these sensitive tuning parameters. The algorithm allows efficient computation of dictionaries from large corpuses of training data and can in some cases converge to better solutions than batch algorithms.

The proposed algorithm alternates between encoding and dictionary updates to minimize the expectation of the cost function defined in equation (3). In the encoding phase, a random input vector is sampled from the training set and the encoding problem is solved via least angle regression [13] based on the previous iteration's dictionary. The algorithm then optimizes the dictionary using coordinate descent [32] taking the previous dictionary as a warm restart. The algorithm is proven to converge to a stationary point given some reasonable and enforceable assumptions.

Zhao et al. [58] made use of this technique and an augmented reconstruction error function to dynamically learn and detect unusual events in video, without the need for large training example sets or direct supervision. Their approach out-performed another leading approach in multiple test video sequences. Perhaps more importantly, it performed better than a static dictionary learned on the first five minutes of each video, demonstrating the advantage of adapting the dictionary in on-going tasks.

5.2 Approximate Encoding

Gregor and LeCun [22] modify two existing iterative encoding techniques to provide better approximate optimal codes after a fixed number of iterations by learning dictionary and input distribution specific parameters. The experiments demonstrate that the learning process reduces the approximation error of the base models when the number of iterations are fixed, showing that sparse encoding in fixed time can be accomplished with low additional error. This is a matter of great importance to time sensitive vision applications.

The authors modify two popular iterative encoding algorithms; the iterative shrinkage and thresholding algorithm (ISTA) and coordinate descent presented by Daubechies et al. [11] and Li and Osher [32] respectively. ISTA operates on a given input vector X by recursive application of equation (7) until convergence.

$$\alpha^{(i+1)} = h_\theta(W_e x + S\alpha^{(i)}) \tag{7}$$

In the above equation, S is a mutual inhibition matrix, W_e is a scaling of the dictionary, and h_θ is a component-wise non-linear shrinkage function with thresholds θ. In this work the default values for these parameters are replaced by a set learned through predicting the codes of the training set and then back-propagating the error. The coordinate descent algorithm works similarly but only updates one entry of α at a time; selected as the component that would have the greatest effect.

For a fixed number of iterations (i.e. a constant run-time) the trained approaches, referred to as LISTA and LCoD, outperform their corresponding base algorithms. This demonstrates that these methods are very useful for time sensitive tasks; however, there are no guarantees of convergence. Performance may in fact decrease if they are run for more iterations than used to train the encoders.

5.3 Exploiting Structure

Inducing a structure within sparse coding formulations is a common approach to encourage similar input vectors are reconstructed with similar coefficients; however, structure can also be exploited to improve the efficiency of sparse coding approaches. Two recent papers exemplifying this approach will be presented. In both papers the input space is partitioned and differing dictionaries are used to encode each partition.

The first by Yang, Yu, and Haung [51] introduces an efficient methodology to produce and encode from massive dictionaries ($>$ 250,000 basis). The authors use a maximum likelihood approach to simultaneously learn a M component mixture model and a set of M dictionaries $D_m \in \mathbb{R}^{k\times d}$ each assigned to one mixture component. Reconstruction error is modeled as a zero mean, isotropic Gaussian and sparsity is induced by a Laplacian prior similar to work in [30]. An expectation maximization algorithm is used to optimize the model by alternating between optimizing the mixture parameters, modifying the mixture dictionaries to better reconstruct training vectors assigned to the corresponding mixture component, and updating the mixture weights. At encoding time a feature is assigned to multiple mixtures based on the mixture posteriors and the sparse codes are extracted from each dictionary and concatenated. As the number of mixtures is usually much larger than the associated dictionary sizes, the individual computation of the encodings is swift while the effective dictionary size Md is quite large.

The authors claim this method is a good approximation of local coordinate coding theory [56] in which a non-linear function is approximated by locally linear subspaces. The theory provides an upper limit to the approximation error based on the number of subspaces (called anchor points) and their local approximation accuracy. Additionally, by employing spatial pyramid matching and a dictionary of effective size 262,144, the approach resulted in state-of-the-art accuracy on the VOC 2007 [16] and VOC 2009 [17] datasets.

The second work, presented by Szlam et al. [45], also partitions the space and modifies the dictionaries used by each subspace; however, it does so in an even less computationally intensive framework, enabling real time object recognition. When applied to the Caltech 101 dataset [18], the approach performed only slightly worse than a full orthogonal matching pursuit framework at a fraction of the time cost.

The approach is concerned with learning a set of L groups, where each group is linked to a set of m dictionary basis, and some hash function $h(x)$, which maps an input vector to the group which can best reconstruct it. The authors accomplish this efficiently by letting the hash function $h(x)$ be a 2-means tree in the input space, with each leaf assigned to a group. The optimization iterates over three stages: each leaf is assigned to the group that best reconstructs the set of training vectors in that leaf, each group selects the set of indices to best represent the training vectors assigned to it via a modified version of orthogonal matching pursuit, and then the dictionary is updated holding the encodings fixed.

At encoding time the new vector is passed through the tree until it reaches a leaf node where the encoding problem is solved on the group bases. This is a much smaller problem than over the whole dictionary and can be solved quickly. Using this method with a efficient pseudo-SIFT implementation and the spatial pyramid matching technique, the authors achieved comparable accuracy on the Caltech 101 dataset[18] at a stunning speed of 22 frames per second on a quad-core computer.

These results show the capabilities of sparse coding approaches to achieve reasonable efficiency even as the size of the dictionary grows.

6 Future Work

It is clear that adaptions to the classic sparse coding framework can be applied to better suit the approach to object recognition and other classification tasks in vision. Further methods should be explored in all three presented topics to improve the applicability of sparse coding techniques to classification problems.

One suggestion for future work is combining supervised dictionaries (such as that in [25]) with fast approximate encoding approaches (like [22]) under the same optimization framework. This approach seems capable of producing better results than the individual algorithms applied sequentially (i.e. a discriminative dictionary and associated classifier are trained, and then new input signals are encoded approximately). It should stand to reason that combining the learning of the dictionary, the approximate encoding parameters, and the classifier in the same framework has certain advantages. The exchange of information between the three elements should allow for approximations that better fit the dictionary and are more likely to properly encode important discriminative bases to reduce the classifier loss. Additionally, the dictionary would adjust to improve the reconstruction error of the approximate encodings. The result should be a framework that can accurately make predictions in constant time.

References

[1] M. Aharon and M. Elad. Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM Journal on Imaging Sciences*, 1(3):228–247, 2008.

	Caltech-101[18]	Caltech-256[23]	15-Scene[27]	VOC 2007[16]	VOC 2009[17]	Corel10 [31]	MNIST[28]	Brodatz[43]
Wang[47]	73.44	47.68	-	41.19	-	-	-	-
Gao[21]	-	40.43	**89.75**	-	-	**92.0**	-	-
Mairal[35]	-	-	-	-	-	-	-	95.5
Szlam[45]	75.1	-	81.5	-	-	-	-	-
Yang[52]	-	-	-	-	-	-	99.16	-
Jiang[25]	73.6	-	-	-	-	-	-	-
Yang[51]	-	-	-	**59.6**	64.6	-	-	-
Best Other	**84.3**[49]	**50.7**[4]	84.0[48]	59.39[15]	**66.47**[54]	90.0[33]	**99.73**[9]	**99.45**[19]

Figure 3: A comparison of the presented algorithms and the best other approach found in the literature. Note that many of the leading techniques are very computationally intensive compared to these sparse coding approaches. Results for the VOC datasets are in average precision and all others are in accuracy.

[2] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcmplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, 2006.

[3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[4] L. Bo, X. Ren, and D. Fox. Multipath sparse coding using hierarchical matching pursuit. In *NIPS workshop on deep learning*, 2012.

[5] D. M. Bradley and J. A. Bagnell. Differential sparse coding. *Carnegie Mellon Robotics Institute*, 2008.

[6] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *Information Theory, IEEE Transactions on*, 52(2):489–509, 2006.

[7] E. J. Candes, J. K. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2006.

[8] S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

[9] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber. Flexible, high performance convolutional neural networks for image classification. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pages 1237–1242. AAAI Press, 2011.

[10] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, page 22, 2004.

[11] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11):1413–1457, 2004.

[12] D. L. Donoho. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. *Communications on Pure and Applied Mathematics*, 59(6):797–829, 2006.

[13] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2):407–499, 2004.

[14] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15(12):3736–3745, 2006.

[15] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[16] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results. http://www.pascal-network.org/challenges/VOC/voc2009/workshop/index.html.

[18] 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. *Computer Vision and Image Understanding*, 106(1):59–70, 2007.

[19] J. B. Florindo and O. M. Bruno. Multiscale fractal descriptors applied to texture classification. In *Journal of Physics: Conference Series*, volume 410, pages 12–22. IOP Publishing, 2013.

[20] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302–332, 2007.

[21] S. Gao, I. W. Tsang, L.-T. Chia, and P. Zhao. Local features are not lonely–laplacian sparse coding for image classification. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 3555–3561. IEEE, 2010.

[22] K. Gregor and Y. LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on Machine Learning*, pages 399–406, 2010.

[23] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007.

[24] K. S. Gurumoorthy, A. Rajwade, A. Banerjee, and A. Rangarajan. Beyond SVD: Sparse projections onto exemplar orthonormal bases for compact image representation. In *ICPR*, pages 1–4, 2008.

[25] Z. Jiang, Z. Lin, and L. S. Davis. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 1697–1704. IEEE, 2011.

[26] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. Le-Cun. Learning invariant features through topographic filter maps. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1605–1612. IEEE, 2009.

[27] 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, IEEE Computer Society Conference on*, volume 2, pages 2169–2178. IEEE, 2006.

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

[29] H. Lee, A. Battle, R. Raina, and A. Ng. Efficient sparse coding algorithms. In *Advances in Neural Information Processing Systems*, pages 801–808, 2006.

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

[31] F. F. Li, R. VanRullen, C. Koch, and P. Perona. Rapid natural scene categorization in the near absence of attention. *Proceedings of the National Academy of Sciences*, 99(14):9596–9601, 2002.

[32] Y. Li and S. Osher. Coordinate descent optimization for l1 minimization with application to compressed sensing; a greedy algorithm. *Inverse Probl. Imaging*, 3(3):487–503, 2009.

[33] Z. Lu and H. H.-S. Ip. Image categorization with spatial mismatch kernels. In *Computer Vision and Pattern Recognition,IEEE Conference on*, pages 397–404. IEEE, 2009.

[34] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 689–696. ACM, 2009.

[35] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative learned dictionaries for local image analysis. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8. IEEE, 2008.

[36] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *Image Processing, IEEE Transactions on*, 17(1):53–69, 2008.

[37] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397–3415, 1993.

[38] R. Mazhar and P. D. Gader. EK-SVD: optimized dictionary design for sparse representations. In *Pattern Recognition, 19th International Conference on*, pages 1–4. IEEE, 2008.

[39] B. A. Olshausen et al. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

[40] 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.

[41] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993.

[42] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th International Conference on Machine Learning*, pages 759–766. ACM, 2007.

[43] T. Randen and J. H. Husoy. Filtering for texture classification: A comparative study. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(4):291–310, 1999.

[44] M. Rehn and F. T. Sommer. A network that uses few active neurones to code visual input predicts the diverse shapes of cortical receptive fields. *Journal of Computational Neuroscience*, 22(2):135–146, 2007.

[45] A. Szlam, K. Gregor, and Y. LeCun. Fast approximations to structured sparse coding and applications to object classification. In *European Conference on Computer Vision*, pages 200–213. Springer, 2012.

[46] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[47] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 3360–3367. IEEE, 2010.

[48] J. Wu and J. M. Rehg. Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. In *Computer Vision, IEEE 12th International Conference on*, pages 630–637. IEEE, 2009.

[49] J. Yang, Y. Li, Y. Tian, L. Duan, and W. Gao. Group-sensitive multiple kernel learning for object categorization. In *Computer Vision, IEEE 12th International Conference on*, pages 436–443. IEEE, 2009.

[50] 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, IEEE Conference on*, pages 1794–1801. IEEE, 2009.

[51] J. Yang, K. Yu, and T. Huang. Efficient highly over-complete sparse coding using a mixture model. In *European Conference on Computer Vision*, pages 113–126. Springer, 2010.

[52] J. Yang, K. Yu, and T. Huang. Supervised translation-invariant sparse coding. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3517–3524. IEEE, 2010.

[53] M. Yang, L. Zhang, J. Yang, and D. Zhang. Robust sparse coding for face recognition. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 625–632. IEEE, 2011.

[54] K. Yu. Image Classification Using Gaussian Mixture and Local Coordinate Coding. *Visual Recognition Challange workshop*, 2009.

[55] K. Yu, Y. Lin, and J. Lafferty. Learning image representations from the pixel level via hierarchical sparse coding. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 1713–1720. IEEE, 2011.

[56] K. Yu, T. Zhang, and Y. Gong. Nonlinear learning using local coordinate coding. In *Advances in Neural Information Processing Systems*, pages 2223–2231, 2009.

[57] Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 2691–2698. IEEE, 2010.

[58] B. Zhao, L. Fei-Fei, and E. P. Xing. Online detection of unusual events in videos via dynamic sparse coding. In *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pages 3313–3320. IEEE, 2011.