Multipath Sparse Coding Using Hierarchical Matching Pursuit

Liefeng Bo, Xiaofeng Ren
ISTC Pervasive Computing, Intel Labs
Seattle WA 98195, USA
{liefeng.bo,xiaofeng.ren}@intel.com

Dieter Fox
University of Washington
Seattle WA 98195, USA
fox@cs.washington.edu

Abstract

Complex real-world signals, such as images, contain discriminative structures that differ in many aspects including scale, invariance, and data channel. While progress in deep learning shows the importance of learning features through multiple layers, it is equally important to learn features through multiple paths. We propose Multipath Hierarchical Matching Pursuit (M-HMP), a novel feature learning architecture that combines a collection of hierarchical sparse features for image classification to capture multiple aspects of discriminative structures. Our building blocks are KSVD and batch orthogonal matching pursuit (OMP), and we apply them recursively at varying layers and scales. The result is a highly discriminative image representation that leads to large improvements to the state-of-the-art on many standard benchmarks, e.g. Caltech-101, Caltech-256, MIT-Scenes and Caltech-UCSD Bird-200.

1 Introduction

Images are high dimensional signals that change dramatically under varying scales, viewpoints, lighting conditions, and scene layouts. How to extract features that are robust to these changes is the key question of computer vision, and traditionally people rely on hand-designed features such as SIFT [18]. While SIFT can be understood and generalized as a way to go from pixels to patch descriptors [3], designing image features is a challenging task that requires deep domain knowledge, and it is often difficult to adapt to new settings.

Feature learning is attractive as it exploits the availability of data and avoids the need of feature engineering, and it has become increasingly popular and effective for visual recognition. A variety of learning and coding techniques have been proposed and evaluated, such as deep belief nets [11], deep Boltzmann machines [26], deep autoencoders [29, 16], convolutional deep belief networks [17], and hierarchical sparse coding [34, 4]. Many are deep learning approaches that learn to push pixels through multiple layers of feature transforms. The recent work on Hierarchical Matching Pursuit [4, 5] is interesting as it is efficient (using Batch Orthogonal Matching Pursuit), recursive (the same computational structure going from pixels to patches, and from patches to images), and outperforms many designed features and algorithms on a variety of recognition benchmarks.

One crucial problem that is often overlooked in image feature learning is the multi-facet nature of visual structures: discriminative structures, which we want to extract, may appear at varying scales with varying amounts of spatial and appearance invariance. While a generic learning model could capture such heterogeneity, it is much easier to build it into the learning architecture. In this work, we propose Multipath Hierarchical Matching Pursuit (M-HMP), which builds on the single-path Hierarchical Matching Pursuit approach to learn and combine recursive sparse coding through many pathways on multiple bags of patches of varying size, and, most importantly, by encoding each patch through multiple paths with a varying number of layers. See Fig. 1 for an illustration of our system.

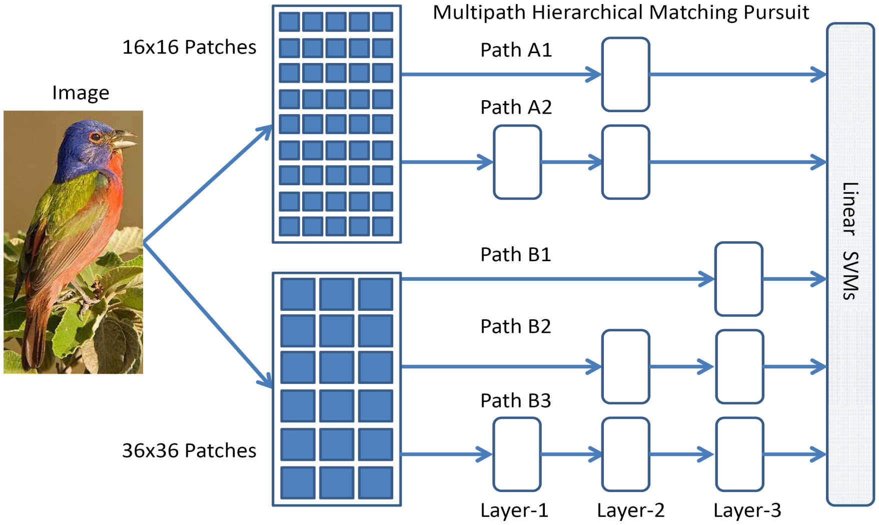

Figure 1: Architecture of multipath sparse coding. Image patches of different sizes (here, 16x16 and 36x36) are encoded via multiple layers of sparse coding. Each path corresponds to a specific patch size and number of layers. The final layer of each path encodes the complete image patches and generates a feature vector for the whole image via spatial pooling. Path features are then concatenated and used by a linear SVM for object recognition.

The multipath architecture is important as it significantly and efficiently expands the richness of image representation and leads to large improvements to the state of the art of image classification, as evaluated on a variety of object and scene recognition benchmarks. Our M-HMP approach is generic and can adapt to new tasks, new sensor data, or new feature learning and coding algorithms.

2 Related Work

In the past few years, a growing amount of research on visual recognition has focused on learning multiple level features using supervised or unsupervised hierarchical architectures.

Deep Networks: Deep belief nets [11] learn a hierarchy of features, layer by layer, using the unsupervised restricted Boltzmann machine. The learned weights are then further adjusted to the current task using supervised information. To make deep belief nets applicable to full-size images, convolutional deep belief nets [17] use a small receptive field and share the weights between the hidden and visible layers among all locations in an image. Invariant predictive sparse decomposition [13] approximates sparse codes from sparse coding approaches using multi-layer feed-forward neural networks and avoid solving computationally expensive optimizations at runtime. Deep autoencoders [29, 16] build deep networks, based on stacking layers of autoencoders that train one-layer neural networks to reconstruct input data. Recursive neural networks [27] discover recursive structure from natural scene images and natural language sentences, by computing a score for merging neighboring regions into a larger region, a new semantic feature representation for this larger region, and its class label. Sum-Product Networks [24] learn deep representations using directed acyclic graphs with variables as leaves, sums and products as internal nodes, and weighted edges.

Sparse Coding: One-layer Sparse coding on top of raw patches or SIFT features has achieved state-of-the-art performance on face recognition, texture segmentation [19], and generic object recognition [30, 7, 8]. Very recently, multi-layer sparse coding networks including hierarchical sparse coding [34, 4] and hierarchical matching pursuit [4] have been proposed for building multiple level features from raw sensor data. Such networks learn codebooks at each layer in an unsupervised way such that image patches or pooled features can be represented by a sparse, linear combination of codebook entries. With learned codebooks, feature hierarchies are built from scratch, layer by layer, using sparse codes and spatial pooling [34, 4, 5]. Unsupervised feature learning approaches have been adapted to depth maps and 3-D point clouds for RGB-D object recognition [2, 5]. The proposed M-HMP has a different architecture, compared with the existing hierarchical sparse coding approaches.

3 Multipath Hierarchical Matching Pursuit

This section provides an overview of our Multipath Hierarchical Matching Pursuit (M-HMP) approach to feature learning. We review the key ideas behind KSVD codebook learning, and discuss how multi-layer sparse coding hierarchies for images can be built from scratch and how multipath sparse coding helps capture features of varying characteristics.

3.1 Codebook Learning via KSVD

The key idea of sparse coding is to learn a codebook/dictionary such that the data can be represented by a sparse, linear combination of codewords. KSVD [1, 4] learns codebooks $D = [d_1, \cdots, d_m, \cdots, d_M] \in R^{H\times M}$ and the associated sparse codes $X = [x_1, \cdots, x_n, \cdots, x_N] \in R^{M\times N}$ from a matrix $Y = [y_1, \cdots, y_n, \cdots, y_N] \in R^{H\times N}$ of observed data by minimizing the reconstruction error

$$\min_{D,X} \|Y - DX\|_F^2 \quad s.t.\ \forall m,\ \|d_m\|_2 = 1 \text{ and } \forall n,\ \|x_n\|_0 \le K \tag{1}$$

where H, M, and N are the dimensionality of codewords, the size of codebook, and the number of training samples, respectively, $\|A\|_F$ denotes the Frobenius norm, the zero-norm $\|\cdot\|_0$ counts non-zero entries in the sparse codes x_n, and K is the sparsity level controlling the number of the non-zero entries. KSVD solves the optimization problem (1) in an alternating manner. During each iteration, the current codebook D is used to encode the data Y by computing the sparse code matrix X. Then, the codewords of the codebook are updated one at a time, resulting in a new codebook. This new codebook is then used in the next iteration to recompute the sparse code matrix followed by another round of codebook update.

3.2 Hierarchical Matching Pursuit

KSVD is used to learn codebooks in three layers where the data matrix Y in the first layer consists of raw patches sampled from images, and Y in the second and third layers are sparse codes pooled from the lower layers. With the learned codebooks D, hierarchical matching pursuit builds a feature hierarchy, layer by layer, by recursively using batch OMP for computing sparse codes, spatial pooling for pooling sparse codes, and contrast normalization for normalizing pooled sparse codes, as shown in Fig. 2.

First Layer: The goal of the first layer in HMP is to extract sparse codes for small patches (e.g. 5x5) and generate pooled codes for mid-level patches (e.g., 16x16). Orthogonal matching pursuit is used to compute the sparse codes x of small patches (e.g. 5x5 pixels). Spatial max pooling is then applied to aggregate the sparse codes. In our terminology, an image patch P is divided spatially into smaller cells. The features of each spatial cell C are the max pooled sparse codes, which are simply the component-wise maxima over all sparse codes within a cell:

$$F(C) = \left[\max_{j\in C} |x_{j1}|, \cdots, \max_{j\in C} |x_{jm}|, \cdots, \max_{j\in C} |x_{jM}|\right] \tag{2}$$

Here, j ranges over all entries in the cell, and x_{jm} is the m-th component of the sparse code vector x_j of entry j. The feature F_P describing an image patch P are the concatenation of aggregated sparse codes in each spatial cell

$$F_P = \left[F(C_1^P), \cdots, F(C_s^P), \cdots, F(C_S^P)\right] \tag{3}$$

where $C_s^P \subseteq P$ is a spatial cell generated by spatial partitions, and S is the total number of spatial cells. We additionally normalize the feature vectors F_P by L_2 norm $\sqrt{\|F_P\|^2 + \varepsilon}$, where ε is a small positive number. Since the magnitude of sparse codes varies over a wide range due to local variations in illumination and occlusion, this operation makes the appearance features robust to such variations, as commonly done in the hand-designed SIFT features. We find that $\varepsilon = 0.1$ works well for all the recognition problems we consider.

Second Layer: The goal of the second layer in HMP is to gather and code mid-level sparse codes and generate pooled codes for large patches (e.g. 36x36). To do so, HMP applies batch OMP and spatial max pooling to features F_P generated in the first layer. The codebook for this level is learned by sampling features F_P over images. The process to extract the feature describing a large image

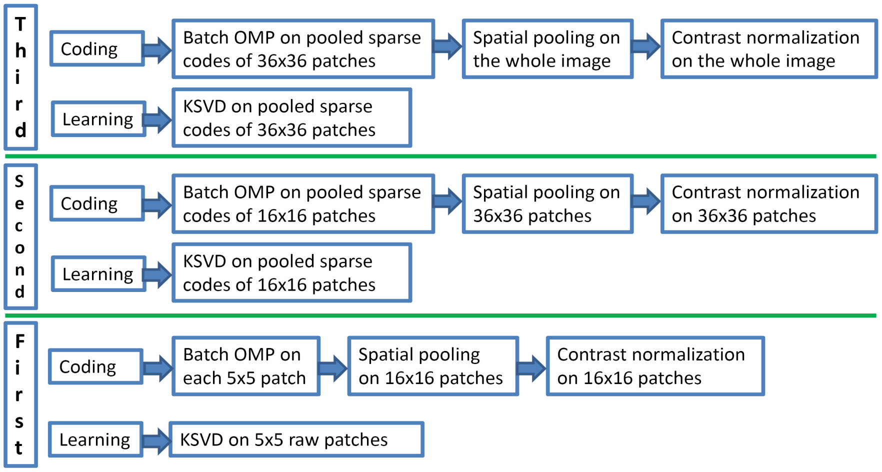
Third
Coding
Batch OMP on pooled sparse codes of 36x36 patches
Spatial pooling on the whole image
Contrast normalization on the whole image
Learning
KSVD on pooled sparse codes of 36x36 patches
Second
Coding
Batch OMP on pooled sparse codes of 16x16 patches
Spatial pooling on 36x36 patches
Contrast normalization on 36x36 patches
Learning
KSVD on pooled sparse codes of 16x16 patches
First
Coding
Batch OMP on each 5x5 patch
Spatial pooling on 16x16 patches
Contrast normalization on 16x16 patches
Learning
KSVD on 5x5 raw patches

Figure 2: A three-layer architecture of Hierarchical Matching Pursuit.

patch is identical to that for the first layer: sparse codes of each image patch are computed using batch orthogonal matching pursuit, followed by spatial max pooling on the large patch. The feature vector is then normalized by its L2 norm.

Third Layer: The goal of the third layer in HMP is to generate pooled sparse codes for the whole image/object. Similar to the second layer, the codebook for this level is learned by sampling these pooled sparse codes in the second layer. With the learned codebook, just as in the second layer, sparse codes of each image patch (for instance, 36x36) are computed using batch OMP, followed by spatial max pooling on the whole images. The features of the whole image/object are the concatenation of the aggregated sparse codes of the spatial cells. The feature vector is then normalized by dividing with its L_2 norm.

A one-layer HMP has the same architecture with that of the final layer of a three-layer HMP, except that KSVD and batch OMP are performed on 36x36 raw image patches instead of pooled sparse codes. A two-layer HMP has the same architecture with that of the second and third layers of a three-layer HMP, except that KSVD and batch OMP in the lower layer are performed on raw image patches.

3.3 Architecture of Multipath Sparse Coding

In visual recognition, images are frequently modeled as unordered collections of local patches, i.e. a bag of patches. Such models are flexible, and the image itself can be considered as a special case (bag with one large patch). Traditional bag-of-patches models introduce invariance by completely ignoring spatial positions of and relationships between patches, generally useful for visual recognition. The spatial pyramid bag-of-patches model [15] overcomes this problem by organizing patches into spatial cells at multiple levels and then concatenating features from spatial cells into one feature vector. Such models effectively balance the importance of invariance and discriminative power, leading to much better performance than simple bags. Spatial pyramid bags are a compelling strategy for unsupervised feature learning because of a number of advantages: (1) bag-of-patches virtually generates a large number of training samples for learning algorithms and decreases the chance of overfitting; (2) the local invariance and stability of the learned features are increased by pooling features in spatial cells; (3) by varying patch sizes, feature learning can capture structures at multiple levels of scale and invariance.

A single-path HMP (See Section 3.2) already have the first two advantages. To exploit the advantage of multiple patch sizes as well as the strength of multi-layer architectures, our Multipath Hierarchical Matching Pursuit (M-HMP) configures matching pursuit encoders in multiple pathways, varying patch sizes and the number of layers (see Fig. 1). Note that in the final layer, sparse coding is always applied to the full image patches (16x16 on the top and 36x36 on the bottom). M-HMP encodes patches of different sizes, such as 16x16 and 36x36, which contain structures of different scales. More importantly, we argue that multiple paths, by varying the number of layers in HMP, is important for a single patch size. For instance, we could learn features for 36x36 image patches

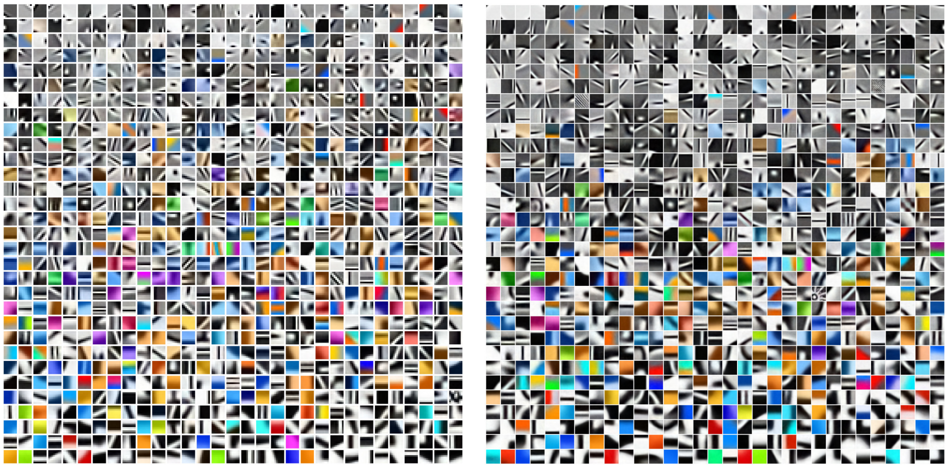

Figure 3: Learned codebooks via KSVD on Caltech-101. *Left*: 1000 16x16x3 codewords. *Right*:1000 36x36x3 codewords.

using a one-layer HMP or a two-layer HMP or a three-layer HMP. These HMP networks with different layers capture different aspects of 36x36 image patches. Intuitively, features of 36x36 image patches learned by a one-layer HMP capture basic structures of patches and are sensitive to spatial displacement. Features of 36x36 image patches learned by a two-layer HMP introduce robustness to local deformations due to the usage of spatial max pooling in the first layer. Features of 36x36 image patches learned by a three-layer HMP are highly abstract and robust due to its ability to recursively eliminate unimportant structures and increase invariance to local deformations by spatial max pooling in the previous layers.

Next, we outline the detailed architecture of the five HMP networks used in the experiments. On image patches of size 16x16 (A), we learn a one-layer HMP on RGB images and a two-layer HMP on grayscale images. For the one-layer HMP (A1), we learn codebooks of size 1000 with sparsity level 5 on a collection of 16x16 raw patches sampled from RGB images. For the two-layer HMP (A2), we first learn first-layer codebooks of size 75 with sparsity level 5 on a collection of 5x5 raw patches sampled from grayscale images. We then generate the pooled sparse codes on 4x4 spatial cells of 16x16 image patches with a pooling size of 4x4 pixels. Finally, we learn the second-layer codebooks of size 1000 with sparsity level 10 on the pooled sparse codes on 16x16 image patches.

On image patches of size 36x36 (B), we learn one-layer HMP on RGB images, and two-layer and three-layer HMP on grayscale images. For the one-layer HMP (B1), we learn the codebooks of size 1000 with sparsity level 5 on a collection of 36x36 raw patches sampled from RGB images. For the two-layer HMP (B2), we first learn the codebooks of size 300 with sparsity level 5 on a collection of 10x10 raw patches sampled from grayscale images. We then generate the pooled sparse codes on 4x4 spatial cells of 36x36 image patches with a pooling size of 9x9 pixels. Finally, we learn the codebooks of size 1000 with sparsity level 10 on the pooled sparse codes on 36x36 image patches. For the three-layer HMP (B3), the first two layers are the same as A2. For the third layer, we first generate the pooled sparse codes on 3x3 spatial cells of the pooled sparse codes in the second layer with a pooling size of 3x3. Finally, we learn codebooks of size 1000 with sparsity level 10 on the pooled sparse codes based 36x36 image patches (36=4x3x3).

In the final layer of A1, A2, B1, B2 and B3, we generate image-level features by computing sparse codes of 36x36 image patches and performing max pooling followed by contrast normalization on spatial pyramids 1x1, 2x2 and 4x4 on the whole images. Note that the above architecture of multi-layer HMP networks leads to fast computation of pooled sparse codes. Note that the accuracy of M-HMP is robust with respect to patch size choice and other reasonable patch sizes such as 20x20 and 40x40 give similar results.

4 Experiments

We evaluate the proposed M-HMP models on four standard vision datasets on object, scene and fine-grained recognition, extensively comparing to state-of-the-art algorithms using designed and

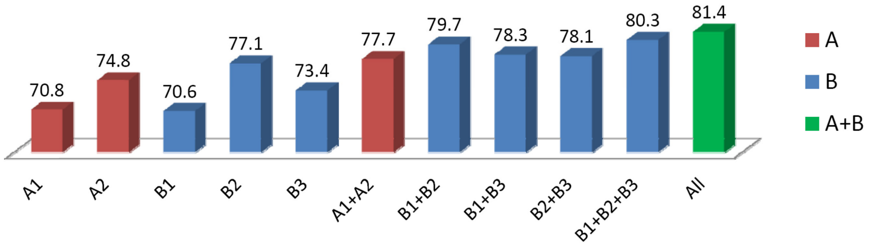

Figure 4: Results by single-path and multipath sparse coding. A1,A2,B1,B2 and B3 denotes the HMP networks of different architectures (See Section 3.3). A1+A2 indicates the combination of A1 and A2. "All" means the combination of all five paths: A1,A2,B1,B2 and B3.

SIFT+T [8]	67.7	LC-KSVD [12]	73.6	Asklocals [7]	77.1
NBNN [6]	73.0	HSC [34]	74.0	LP-β [10]	77.7
LLC [30]	73.4	HMP [4]	76.8	M-HMP	**81.4±0.33**

Table 1: Test accuracy on Caltech-101.

learned features. All images are resized to 300 pixels on the longest side. We remove the zero frequency component from raw patches by subtracting their means in the first layer of the HMP networks. The set of hyperparameters for all five HMP networks are optimized on a small subset of the ImageNet database. We keep this set of hyperparameters in all the experiments, even though per-dataset tweaking through cross validation may further improve accuracy. With the learned M-HMP features, we train linear SVMs for recognition, which are able to match the performance of nonlinear SVMs while being scalable to large datasets [4].

4.1 Object Recognition: Caltech-101

We investigate the behavior of M-HMP for object category recognition on Caltech-101. The dataset contains 9,144 images from 101 object categories and one background category. We use Caltech-101 because a large number of algorithms have been evaluated on this dataset, despite its known limitations. In the following sections, we will demonstrate multipath sparse coding on more standard vision datasets. Following the standard experimental setting, we train models on 30 images and test on no more than 50 images per category [15].

We visualize the learned codebooks in the first layer in Fig 3. As can be seen, the learned codebooks have very rich appearances and include uniform colors of red, green and blue, transition filters between different colors, gray and color edges, gray and color texture filters, double edge gray and color filters, center-surround (dot) filters, and so on. This suggests that a large variety of structures in the raw pixels are captured. In addition, we can see more high-level structures in the codebook learned on 36x36 image patches, such as curved edges, high-frequency patterns, ellipses, or distinctive shapes like half of a car wheel.

We show the results of M-HMP in Fig. 4 (A1, A2, B1, B2 and B3 are defined in Section 3.3). As can bee seen, the one-layer HMP networks (A1 and B1) work surprisingly well and already outperform many existing computer vision approaches, showing the benefits of learning from pixels. The two-layer HMP networks (A2 and B2) achieve the best results among five single pathways. The three-layer HMP networks (B3) is superior to the one-layer networks (A1 and B1), but inferior to the two layer HMP networks (A2 and B2). The combination of HMP networks of different depths (A1+A2 and B1+B2+B3) lead to significant improvements over the best single HMP network (A2 and B2). Multipath coding combining all five HMP networks achieves the best result of 81.4%, suggesting that different paths complement one another and capture different aspects of image structures.

We compare M-HMP with recent published state-of-the-art recognition algorithms in Table 1. SPM [15] is spatial pyramid matching with SIFT features. NBNN [6] is a Naive Bayesian nearest neighbor approach based on multiple types of features. ScSPM [32], LLC [30], LC-KSVD [12], and Asklocals [7] are single layer sparse coding approaches. SIFT+T [8] is the soft threshold coding. All of them are based on SIFT. HSC [34] is a two layer sparse coding network using L1-norm regularization. HSC [34] is a two layer sparse coding network using L1-norm regularization. LP-β [10] is a boosting approach to combine multiple types of hand-designed features. M-HMP achieves the higher test accuracy than all of them by a large margin.

Training Images	15	30	45	60
ScSPM [32]	27.7±0.5	34.0±0.4	37.5±0.6	40.1±0.9
Local NBNN [20]	33.5	40.1	/	/
LLC [30]	34.4	41.2	45.3	47.7
Asklocals [7]	35.2	41.6	/	/
CRBM [28]	35.1±0.3	42.1	45.7	47.9
M-HMP	**40.5**±0.4	**48.0**±0.2	**51.9**±0.2	**55.2**±0.3

Table 2: Test accuracy on Caltech-256.

GIST-color [21]	29.7	SC [4]	36.9	PmSVM-HI [31]	47.2
DPM [9]	30.4	RBoW [22]	37.9	HMP [5]	47.7
SPM [21]	34.4	DPM+Gist+SPM [21]	43.1	M-HMP	**50.5**

Table 3: Test accuracy on MIT-Scenes

4.2 Object Recognition: Caltech-256

To further evaluate the scalability of the proposed approach with respect to the number of categories and the number of images in each category, we perform experiments on Caltech-256. The dataset consists of $30,607$ images from 256 object categories and background, where each category contains at least 80 images. Caltech-256 is much more challenging than Caltech-101 due to the larger number of classes and more diverse lighting conditions, poses, backgrounds, object sizes, etc. Following the standard setup [32], we gradually increase the training set size from 15 to 60 images per category with a step of 15 and test trained models on the rest of the images.

We report the average accuracy over 5 random trials in Table 2. We keep the same architectures with that for Caltech-101 (Section 4.1), with the only exception that the number of codewords in the final layer of HMP is increased to 2000 to accommodate for more categories and more images. As can be seen, our M-HMP approach makes exciting progress on this benchmark and is significantly better than all previously published results. More importantly, the performance gap grows larger with an increasing number of training images. For instance, the gap between M-HMP and CRBM is about 5% for 15 training images per category, but it increases to about 7% for 60 training images. This suggests that (1) rich features are important for large-scale recognition with a large number of categories and a large number of images; (2) M-HMP is well suited for extracting rich features from images, particularly important as we move toward high-resolution images.

4.3 Scene Recognition

Understanding scenes is a fundamental research topic in computer vision. We evaluate our M-HMP approaches on the popular MIT Scene-67 dataset. This dataset contain 15620 images from 67 indoor scene categories. All images have a minimum resolution of 200 pixels in each axis. Indoor scene recognition is very challenging as the intra-class variations are large and some scene categories are similar to one another. Following standard experimental setting [21], we train models on 80 images and test on 20 images per category. We use the same M-HMP architectures and hyperparameters with that for Caltech-101.

We report the accuracy of M-HMP on the training/test split provided on the authors' website in Table 3. First of all, M-HMP achieves much higher accuracy than state-of-the-art recognition algorithms: spatial pyramid matching (SPM) [21], deformable parts models (DPM) [9],Reconfigurable Models (RBoW) [22], Hierarchical Matching Pursuit [5], and even the combination of SPM, DPM, and color GIST [21]. M-HMP also outperforms the recently introduced power mean support vector machines where multiple types of features have been used to boost accuracy. The best single-path M-HMP is an architecture of two layers on 16x16 image patches that obtains 41.8% accuracy. Multipath HMP dramatically increases the accuracy to 50.5%, suggesting that rich but complementary features are essential for achieving good performance on scene recognition.

4.4 Fine-grained Object Recognition

In the last decade, most work has been focused on basic-level recognition tasks: distinguishing different categories of objects, such as table, computer and human. Recently, there is increasing

Approaches	MKL [25]	LLC [33]	Rand-forest [33]	Multi-cue [14]	M-HMP
Accuracy(%)	19.0	18.0	19.2	22.4	**26.5**

Table 4: Test accuracy on Caltech-UCSD Bird-200.

interest and attention on fine-grained (subordinate-level) recognition that classifies similar object categories, such as different species of birds, cats and dogs [25, 23]. Fine-grained recognition differs from basic-level category recognition due to the importance of subtle differences between objects, which may be challenging even for humans to identify.

We evaluate our M-HMP approaches on Caltech-UCSD Bird-200 [25], a standard dataset for fine-grained object recognition. This dataset contains 6033 images from 200 bird species in North America. In each image, the bounding box of a bird is given. Following the standard setting [25], 15 images from each species are used for training and the rest for testing. In Table 4, we compare our M-HMP with four recently published algorithms: multiple kernel learning [25], random forest [33], LLC [30], and multi-cue [14]. Multipath HMP outperforms the state of the art by a large margin and sets a new record for fine-grained object recognition. Note that the previous approaches all use multiple types of features such as SIFT, color SIFT, color histograms and etc. to boost the classification accuracy; the best accuracy using a single type of features is lower than 18%.

Feature learning techniques are particularly suitable for fine-grained recognition since (1) rich appearance and shape features are required for describing subtle differences between categories; and (2) feature learning approaches provide a natural way to capture rich appearance cues by using a large number of codewords (sparse coding) or neurons (deep networks), while traditional computer vision features, designed for basic-level category recognition, may eliminate many useful cues during feature extraction. We believe that this direction is worth pursuing and will gain more attention in the feature learning community.

5 Conclusions

We have proposed Multipath Hierarchical Matching Pursuit for learning expressive features from images. Our approaches combine recursive sparse coding through many pathways, using multiple bags of patches of varying size, and, most importantly, encoding each patch through multiple paths with a varying number of layers. We have performed extensive comparisons on three types of visual recognition tasks: object recognition, scene recognition, and fine-grained object recognition. Our experiments have confirmed that the proposed approach outperforms the state-of-the-art on various popular vision benchmarks. These results are extremely encouraging, indicating that visual recognition systems can be significantly improved by learning features from raw images.

References

[1] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, 2006.

[2] M. Blum, J. Springenberg, J. Wlfing, and M. Riedmiller. A Learned Feature Descriptor for Object Recognition in RGB-D Data. In *ICRA*, 2012.

[3] L. Bo, X. Ren, and D. Fox. Kernel Descriptors for Visual Recognition. In *NIPS*, 2010.

[4] L. Bo, X. Ren, and D. Fox. Hierarchical Matching Pursuit for Image Classification: Architecture and Fast Algorithms. In *NIPS*, 2011.

[5] L. Bo, X. Ren, and D. Fox. Unsupervised Feature Learning for RGB-D Based Object Recognition. In *ISER*, June 2012.

[6] O. Boiman, E. Shechtman, and M. Irani. In Defense of Nearest-Neighbor based Image Classification. In *CVPR*, 2008.

[7] Y. Boureau, N. Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the Locals: Multi-Way Local Pooling for Image Recognition. In *ICCV*, 2011.

[8] A. Coates and A. Ng. The Importance of Encoding versus Training with Sparse Coding and Vector Quantization. In *ICML*, 2011.

[9] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object Detection with Discriminatively Trained Part-Based Models. *IEEE PAMI*, 32:1627–1645, 2010.

[10] P. Gehler and S. Nowozin. On Feature Combination for Multiclass Object Classification. In *ICCV*, 2009.

[11] G. Hinton, S. Osindero, and Y. Teh. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 18(7):1527–1554, 2006.

[12] Z. Jiang, Z. Lin, and L. Davis. Learning a Discriminative Dictionary for Sparse Coding via Label Consistent K-SVD. In *CVPR*, 2011.

[13] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun. Learning Invariant Features through Topographic Filter Maps. In *CVPR*, 2009.

[14] F. Khan, J. van de Weijer, A. Bagdanov, and M. Vanrell. Portmanteau Vocabularies for Multi-cue Image Representations. *NIPS*, 2011.

[15] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *CVPR*, 2006.

[16] Q. Le, M. Ranzato, R. Monga, M. Devin, K. Chen, G. Corrado, J. Dean, and A. Ng. Building High-Level Features Using Large Scale Unsupervised Learning. In *ICML*, 2012.

[17] H. Lee, R. Grosse, R. Ranganath, and A. Ng. Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations. In *ICML*, 2009.

[18] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[19] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Discriminative Learned Dictionaries for Local Image Analysis. In *CVPR*, pages 1–8, 2008.

[20] S. McCann and D. Lowe. Local Naive Bayes Nearest Neighbor for image classification. In *CVPR*, 2012.

[21] M. Pandey and S. Lazebnik. Scene Recognition and Weakly Supervised Object Localization with Deformable Part-Based Models. In *ICCV*, 2011.

[22] S. Naderi Parizi, J. Oberlin, and P. Felzenszwalb. Reconfigurable Models for Scene Recognition. In *CVPR*, 2012.

[23] O. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar. Cats and Dogs. In *CVPR*, 2012.

[24] H. Poon and P. Domingos. Sum-Product Networks: A New Deep Architecture. In *UAI*, 2011.

[25] Branson. S., Wah. C., Babenko. B., F. Schroff, Welinder. P., P. Perona, and S. Belongie. Visual Recognition with Humans in the Loop. In *ECCV*, Sept. 2010.

[26] R. Salakhutdinov and G. Hinton. Deep Boltzmann Machines. In *International Conference on AI and Statistics*, 2009.

[27] R. Socher, C. Lin, A. Ng, and C. Manning. Parsing Natural Scenes and Natural Language with Recursive Neural Networks. In *ICML*, pages 129–136, 2011.

[28] K. Sohn, D. Jung, H. Lee, and A. Hero III. Efficient Learning of Sparse, Distributed, Convolutional Feature Representations for Object Recognition. In *ICCV*, 2011.

[29] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol. Extracting and Composing Robust Features with Denoising Autoencoders. In *ICML*, 2008.

[30] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Guo. Locality-constrained Linear Coding for Image Classification. In *CVPR*, 2010.

[31] J. Wu. Power Mean SVM for Large Scale Visual Classification. In *CVPR*, 2012.

[32] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear Spatial Pyramid Matching using Sparse Coding for Image Classification. In *CVPR*, 2009.

[33] B. Yao, A. Khosla, and L. Fei-Fei. Combining Randomization and Discrimination for Fine-grained Image Categorization. *CVPR*, 2011.

[34] K. Yu, Y. Lin, and J. Lafferty. Learning Image Representations from the Pixel Level via Hierarchical Sparse Coding. In *CVPR*, 2011.