IMAGE CLASSIFICATION USING RBM TO ENCODE LOCAL DESCRIPTORS WITH GROUP SPARSE LEARNING

Jinzhuo Wang[*], *Wenmin Wang*[*], *Ronggang Wang*[*], *Wen Gao*[*,†]

[*]Digital Media R & D Center, Peking University Shenzhen Graduate School
[†]National Engineering Laboratory for Video Technology, Peking University
[*]wangjz@sz.pku.edu.cn, [*]wangwm@ece.pku.edu.cn, [*]rgwang@pkusz.edu.cn, [†]wgao@pku.edu.cn

ABSTRACT

This paper proposes to employ deep learning model to encode local descriptors for image classification. Previous works using deep architectures to obtain higher representations are often operated from pixel level, which lack the power to be generalized to large-size and complex images due to computational burdens and internal essence capture. Our method slips the leash of this limitation by starting from local descriptors to leverage more semantical inputs. We investigate to use two layers of Restricted Boltzmann Machines (RBMs) to encode different local descriptors with a novel group sparse learning (GSL) inspired by the recent success of sparse coding. Besides, unlike the most existing pure unsupervised feature coding strategies, we use another RBM corresponding to semantic labels to perform supervised fine-tuning which makes our model more suitable for classification task. Experimental results on Caltech-256 and Indoor-67 datasets demonstrate the effectiveness of our method.

Index Terms— Image Classification, Feature Coding, Restricted Boltzmann Machine (RBM), Group Sparse Learning (GSL)

1. INTRODUCTION

Bag-of-Feature (BoF) model is one of the most powerful and popular frameworks for image classification which represents image as a histogram of visual words. Standard BoF-based framework is mainly composed of four steps: feature extraction, feature coding, spatial pooling and SVM classification. This pipeline is almost fixed in recent literatures except for the "feature coding" part. To this end, many elegant algorithms have been designed to improve the discriminative power of the learned codes [1–4] among which deep learning based approaches draw a lot of attention due to its representational power of deep transformation compared with other dictionary learning methods in a single step [5–7].

A typical deep learning method for feature coding is to stack layers of Restricted Boltzmann Machines (RBMs) to learn a hierarchical architecture such as [8, 9]. Each layer is regarded as an encoder-decoder functional module. At the bottom of the architecture, pixels of images are used as input with non-linear computations to obtain a slight higher representation, after several layers of transformation, the output of the top layer can be viewed as a higher representation. Although this architecture is often employed as an expert of feature learner, one major limitation is that pixel-level-start scheme contributes less semantical information, thus more layers are needed to construct to bridge the semantic gap, which amplifies the problems such as computational burdens and training errors.

In this paper, we propose to leverage local descriptors as the input of deep architecture, using two layers of RBMs to encode these descriptors. Since the output of the first RBM can be viewed as the learned codes in one layer and the codebook for the whole model, we introduce an approach to constraint the hidden units with group sparse property to limit the size of codebook. To make the learned codes more suitable for classification task, we use another RBM corresponding to semantic labels to perform supervised fine-tuning. With the combination of BoF model and spatial pyramid pooling, we explore to use the learned representations for image classification.

The remainder of this paper is organized as follows. Section 2 reviews the related work, and Section 3 details our approach with focus on employing RBMs to encode local descriptors with group sparse learning (GSL). Experimental results with analysis and comparison are presented in Section 4, and we conclude the paper in Section 5.

2. RELATED WORK

Feature coding has been studied extensively in recent years and readers can refer to [10] for good surveys. Here, we only cover typical deep learning based methods.

Deep learning is a family of learning methods that can provide good representation of data by a multiple-layered structure, where each layer represents different degree of abstraction of data features. Currently it is often used as a

This project was supported by Shenzhen Peacock Plan (20130408-183003656).

powerful model to transform images pixels to obtain higher representations for classification task, such as convolutional neural networks (CNNs) [4] and deep belief networks (DBNs) [11]. More recently, convolutional deep neural networks (CDNNs) have emerged as a powerful model in large-scale dataset [12]. Despite the powerful representations, these methods needed more layers to construct their models which burden the training procedure on both computation and storage.

Our method is based on RBM which has been stacked to form hierarchical representations from pixels [13] with selectivity [8] as the prior to train each layer. However, most previous RBM-based approaches are operated from pixel level. We attempt to leverage local descriptors as the input motivated by the observation that pixel-start transformation fall short of performing image classification within the BoF model. The most recent to ours is [9] which learns Gaussian RBMs from SIFT rather than pixels, but the overall architecture remains relatively heavy. With group sparse constraints on the hidden units of RBM, we exploit to achieve compact and powerful feature codes with only two layers of RBM.

3. METHOD

In this section, we first briefly review RBM along with its training scheme, and then detail our approach with focus on using RBMs to encode local descriptors with group sparse learning (GSL). The proposed framework is shown in Fig.1.

3.1. Restricted Boltzmann Machine

The Restricted Boltzmann Machine (RBM) is a two-layer bipartite structured neural network. The first set of units is defined as the visible layer to represent the observation and the second layer is called the hidden layer which functions as the feature detector. For any given configuration, a uniform energy function is defined as

$$E(v,h) = -\sum_{i,j} v_i h_j w_{ij} - \sum_i b_i v_i - \sum_j c_j h_j \tag{1}$$

where v_i, h_j denote the states of the i_{th} visible and the j_{th} hidden units, w_{ij} represents the weight between them, b and c denote the offsets of the visible and hidden layers. The joint probability distribution is defined as

$$P(v,h) = \frac{\exp(-E(v,h))}{Z} \tag{2}$$

where $Z = \sum_{v,h} \exp(-E(v,h))$ is the Boltzmann partition function. By summing over all the possible hidden units, the marginal distribution over the visible units is generated as

$$P(v) = \frac{1}{Z} \sum_h \exp(-E(v,h)) \tag{3}$$

A probabilistic version of the neuron activation function for inference can be expressed as

$$\begin{aligned} P(v_i|h) &= \sigma(b_i + w_{i\cdot} h) \\ P(h_j|v) &= \sigma(c_i + w_{\cdot j} v) \end{aligned} \tag{4}$$

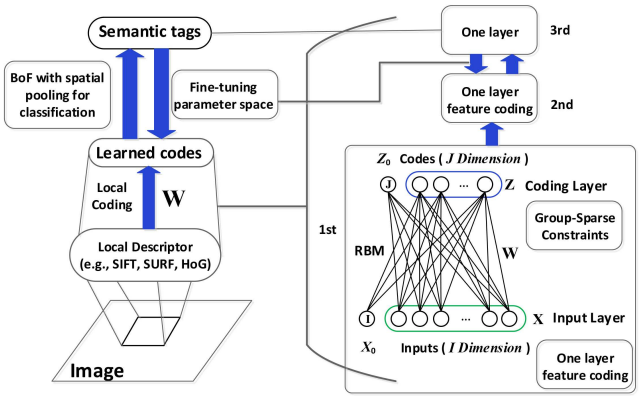


Fig. 1. The proposed framework using RBMs to encode local descriptors for image classification.

The derivative of the log probability of a training data regarding the weights can be formulated as

$$\frac{\partial \log P(v)}{\partial w_{ij}} = \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \tag{5}$$

where $\langle \cdot \rangle$ is the expectation with respect to the subscript specified distribution and the second term is often approximated by contrastive divergenece (CD) algorithm [11]. The weight update rule can then be described as

$$\triangle w_{ij} = \epsilon(\langle v_i h_j \rangle_{\text{data}} - \langle v_i^{\text{neg}} h_j \rangle_{\text{recon}}) \tag{6}$$

where ϵ is the learning rate and v_i^{neg} is sampled from Eq.4. Specifically, given a set of training data $\{v^{(1)}, v^{(2)}, \cdots, v^{(N)}\}$, weight update function can be written as

$$\triangle w_{ij} = \epsilon \left(P(h_j|v_i^{(n)}) v_i^{(n)} - P(h_j|v_i^{(n)\text{neg}}) v_i^{(n)\text{neg}} \right) \tag{7}$$

The corresponding updating rules for both visible layer and hidden layer offsets are

$$\begin{aligned} b_i &= b_i + \epsilon(\langle b_i \rangle_{\text{data}} - \langle b_i^{\text{neg}} \rangle) \\ c_i &= c_i + \epsilon(\langle v_i \rangle_{\text{data}} - \langle v_i^{\text{neg}} \rangle) \end{aligned} \tag{8}$$

The behaviour of RBM can be explained as adjusting the weights and offsets to lower the energy on training data [14].

3.2. Using RBMs to Encode Local Descriptors

To use RBM to encode local descriptors, the visible layer is set to contain N units corresponding to the dimensionality of local descriptor such as 128 for SIFT. The coding layer has J latent units, each representing a visual codeword. The two layers are connected via undirected weights $\mathbf{W} \in \mathbb{R}^{N \times J}$ which is regarded as visual codebook.

Once a layer is trained the parameters $\mathbf{W}$, b, c are frozen and the hidden unit values are inferred. These inferred values serve as the "data" used to train the next higher layer in the network. Note that there is a balance that every RBM-based architecture must trade off, that is, the depth of model and the difficulty of training. Previous works starting from pixels often employ three layers or more to learn feature codes. But

in our approach, local descriptors are used as the input and the dimension is smaller which help us obtain powerful feature codes with fewer layers. So in our implementation, we greedily stack one additional RBM. Note that the third RBM in Fig.1 used for supervised fine-tuning is not regarded as a feature learner.

3.3. Group Sparse Learning

Given $\{\mathbf{v}^1, \mathbf{v}^2, ..., \mathbf{v}^N\}$ representing N local descriptors, the RBM facilitates H hidden units indexed by j where $j = \{1, 2, ..., H\}$. Following the inspiration of group sparse coding [15], we propose to split the hidden units into a sparse group and a redundancy group which consist of s and r hidden units respectively. The objective function is then drawn by the following optimization problem:

$$\min_{w} \left\{ -\sum_{n=1}^{N} \log P(v^{(n)}) + \lambda_1 \sum_{j=1}^{s} \left| \alpha - \frac{1}{N} \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) \right|^2 + \lambda_2 \sum_{j=}^{H} \left| \alpha - \frac{1}{N} \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) \right|^2 \right\} \tag{9}$$

where λ_1 and λ_2 are two regularization coefficients. α and β control the sparseness of the responding hidden units. In practice, we set α a small value to generate the sparse property and β a value close to 1 to keep the hidden units constantly active [8]. If we focus on the sparse regularization term in Eq.9, and compute the gradient for updating the weights given the training data $v^{(n)}$, we can get the learning factor ξ_s as:

$$\begin{aligned} \xi_s &= \frac{\partial}{\partial w_{\cdot j}} \left\{ \sum_{j=1}^{s} \left| \alpha - \frac{1}{N} \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) \right|^2 \right\} \\ &= \sum_{j=1}^{s} \left(\frac{2}{N} \left| \frac{1}{N} \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) - \alpha \right| \cdot \right. \\ &\quad \left. \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) P(h_j{}^{(n)} = 0 | v^{(n)}) v^{(n)} \right) \end{aligned} \tag{10}$$

Similarly, the learning factor ξ_r for the redundancy part is:

$$\begin{aligned} \xi_r = \sum_{j=s+1}^{H} & \left(\frac{2}{N} \left| \frac{1}{N} \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) - \beta \right| \cdot \right. \\ & \left. \sum_{n=1}^{N} P(h_j{}^{(n)} | v^{(n)}) P(h_j{}^{(n)} = 0 | v^{(n)}) v^{(n)} \right) \end{aligned} \tag{11}$$

The regularization terms penalize the selected sparse units and controlled redundancy units until the average active probabilities from these units reaches the controlled value α, β respectively. The learning factors ξ_s and ξ_r are updated iteratively to alleviate the growing tendency of the norm gradient. Algorithm 1 gives an overall description of the proposed group sparse learning (GSL) procedure.

3.4. Supervised Fine-tuning

To conduct supervised fine-tuning, we employ a new classifier RBM connecting the output of the second RBM to an output layer $\mathbf{y} \in \mathbb{R}^C$, with each unit corresponding to a class label c. This RBM is trained by directly associating the output of the second RBM to target outputs $\mathbf{y}$

Input: Random initialization of the weights and offsets $\mathbf{W}, b, c$;
Initialization of learning rate ϵ.
Output: Learned $\mathbf{W}, b, c$

```
for t = 1 to EpochsNumber do
    for n = 1 to SampleNumber do
        Positive phase;
            Compute PosHidProb using Eq.4;
            Compute positive hidden units states;
        Negative phase;
            Reconstruct v^neg using Eq.3;
            Compute NegHidProb using Eq.4;
            Update W by Eq.6 with learning factor ξ_s, ξ_r;
            Update b by Eq.7;
    end
    Update c by Eq.8;
end
return W, b, c;
```

Algorithm 1: Group Sparse Learning (GSL)

$$\triangle w_{ij} = \epsilon(\langle h_i^\star y_c \rangle_{\text{data}} - \langle h_i^\star y_c \rangle_{\text{recon}}) \tag{12}$$

where $h_i^\star$ is the hidden units of the second RBM. All the layers are bound together using top-down sampled signals as targets for bottom-up activations [16]. This technique help us update all RBMs in the architecture concurrently. We use the discriminative softmax cross-entropy loss to penalize feature-based classification errors, which are backpropagated through the parameters consisting of two layers of visual dictionaries and one layer of feature-level classifier.

3.5. Spatial pyramid and linear SVM

By now we have learned feature codes with three RBMs. We then turn to embed them into a standard BoF model with spatial pyramid (SP) to integrate geometric information. We use three level SP with max pooling strategy to generate a vectorial signature for each image. Multi-class linear SVMs are trained with one-against-all strategy for image classification following the instructions in [2].

4. EXPERIMENTS

4.1. Dataset and Setup

In this section, we report the experimental results on Caltech-256 [17] and Indoor-67 [18] dataset. We compare our method mainly with some popular feature coding schemes [2, 9, 19–21], RBM-based approaches [9, 22, 23] and other advanced works [18, 24–31]. For each dataset, we follow the corresponding common experimental settings as in [2, 31].

Our implementation has investigated three types of local descriptors, i.e., SIFT [32], HoG [33] and LBP [34], as the input of our model. Local descriptors are densely extracted

from 16×16 pixel patch on a grid with a step size of 8 pixels. A set of $200,000$ randomly selected descriptors are used to train our model. A three-level spatial pyramid is employed using max pooling grids of 4×4, 2×2 and 1×1 to form the final image representation. The trade-off parameters of SVM regularization term are chosen via 5-fold cross validation on the training data. Following the common benchmarking procedures, we repeat the experimental process by 10 times with different random selected training and testing images to obtain reliable results.

4.2. Results

4.2.1. Caltech-256

Caltech-256 dataset [17] consists of images from 256 object classes containing images from 80 to 827 per class. Fig.2 shows detailed performance comparison with 15, 30, 45 and 60 training images. The compared methods include popular feature coding strategies and a recent RBM-based approach [9] which uses Gaussian RBM from SIFT. As can be seen, our method consistently lead the performance in four types of train number with an average about 4% compared with the best [24] of others on this dataset which validates the power of the learned feature codes with our method.

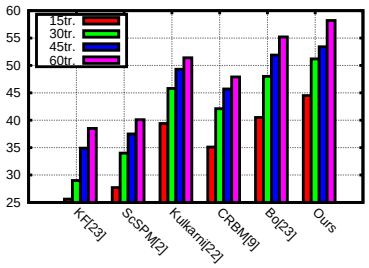


Fig. 2. Performance comparison with other approaches using different training number on Caltech-256 dataset.

4.2.2. Indoor-67

Indoor scene dataset [18] is a scene dataset characterized by 67 indoor classes with high intra-class variations. The average classification rates on Indoor-67 dataset are listed in Table 1. As can be seen from the table, the proposed approach yields a satisfactory performance, especially when compared with the previous methods without deep learning model. Note that two CNN-based models [22, 23] achieve competitive results using CNN starting from raw pixels to learn mid-level feature. However, we still perform better about 3%.

4.3. Analysis and discussion

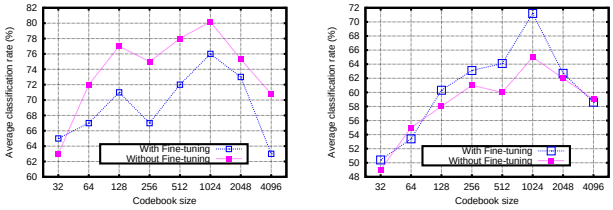
We first investigated the influence of several input local descriptors (SIFT, HoG and LBP). The performance comparison is shown in Table 2. It can be concluded that among these three descriptors, SIFT achieves better performance in most situations except with 15 training images on Caltech-256 dataset. We also explore the influence of different size of the codebook. A larger codebook has more capacity to capture the diversity in the features, but it is also more likely to exhibit codeword redundancy. In our experiments, $1,024$ codewords appear to give a good balance between diversity and conciseness as Fig.3 indicates. Finally, we test the influence of the supervised fine-tuning. The results with or without the third RBM in Fig.1 is shown in Fig.3. The comparison indicates that the supervised fine-tuning procedure is able to improve the power of our model.

Table 1. Average classification rate ($\%$) on Indoor-67 dataset.

SP + HOG [27]	29.8
SP + SIFT [27]	34.4
ROI GIST [18]	26.5
Object Bank [30]	37.6
Scene DPM [31]	43.1
Places-CNN [22]	68.2
CNNaug-SVM [23]	69.0
Ours	**71.2**

Table 2. Performance using Different Local Decriptors

Descriptor	Caltech-256				MIT Scene-67
	15tr.	30tr.	45tr.	60tr.	80tr.
SIFT [32]	44.5	**51.2**	**53.4**	**58.2**	**71.2**
HOG [33]	**46.1**	45.3	46.5	47.8	62.6
LBP [34]	37.9	41.2	46.2	44.5	59.5



(a) Caltech-256 with 60tr. (b) Indoor-67 with 80tr.

Fig. 3. Performance comparison with different size of codebook on Caltech-256 and Indoor-67 datasets

5. CONCLUSION

In this paper, we propose to use two layers of RBMs to encode local descriptors for image classification. We introduce a novel group sparse learning (GSL) procedure for RBM training to control the size of the learned codebook. Another RBM corresponding to semantic labels is employed for supervised fine-tuning which makes our model more suitable for classification task. Experimental results on challenging datasets Caltech-256 and Indoor-67 have shown that our model outperform most of the existing methods.

6. REFERENCES

[1] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*. IEEE, 2006, vol. 2, pp. 2169–2178.

[2] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1794–1801.

[3] Lingqiao Liu, Lei Wang, and Xinwang Liu, "In defense of soft-assignment coding," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2486–2493.

[4] Weiqiang Ren, Yinan Yu, Junge Zhang, and Kaiqi Huang, "Learning convolutional nonlinear features for k nearest neighbor image classification," in *Pattern Recognition (ICPR), 2014 22nd International Conference on*, 2014, pp. 4358–4363.

[5] Qiang Zhang and Baoxin Li, "Discriminative k-svd for dictionary learning in face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2691–2698.

[6] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman, "Discriminative learned dictionaries for local image analysis," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[7] Chih-Fan Chen, Chia-Po Wei, and Y-CF Wang, "Low-rank matrix recovery with structural incoherence for robust face recognition," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2618–2625.

[8] Honglak Lee, Chaitanya Ekanadham, and Andrew Y Ng, "Sparse deep belief net model for visual area v2," in *Advances in neural information processing systems*, 2008, pp. 873–880.

[9] Kihyuk Sohn, Dae Yon Jung, Honglak Lee, and Alfred O Hero, "Efficient learning of sparse, distributed, convolutional feature representations for object recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2643–2650.

[10] Yongzhen Huang, Zifeng Wu, Liang Wang, and Tieniu Tan, "Feature coding in image classification: A comprehensive study," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 3, pp. 493–506, 2014.

[11] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks.," in *Advances in neural information processing systems*, 2012, vol. 1, p. 4.

[13] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew 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.

[14] Geoffrey Hinton, "A practical guide to training restricted boltzmann machines," *Momentum*, vol. 9, no. 1, pp. 926, 2010.

[15] Samy Bengio, Fernando Pereira, Yoram Singer, and Dennis Strelow, "Group sparse coding," in *Advances in Neural Information Processing Systems*, 2009, pp. 82–89.

[16] Hanlin Goh, Nicolas Thome, Matthieu Cord, and Joo-Hwee Lim, "Top-down regularization of deep belief networks," in *Advances in Neural Information Processing Systems*, 2013, pp. 1878–1886.

[17] Gregory Griffin, Alex Holub, and Pietro Perona, "Caltech-256 object category dataset," 2007.

[18] Ariadna Quattoni and Antonio Torralba, "Recognizing indoor scenes," 2009.

[19] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas Huang, and Yihong Gong, "Locality-constrained linear coding for image classification," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3360–3367.

[20] Gabriel L Oliveira, Erickson R Nascimento, Antônio Wilson Vieira, and Mario Fernando Montenegro Campos, "Sparse spatial coding: A novel approach for efficient and accurate object recognition," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 2592–2598.

[21] Zhuolin Jiang, Zhe Lin, and Larry S Davis, "Learning a discriminative dictionary for sparse coding via label consistent k-svd," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1697–1704.

[22] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, "Learning deep features for scene recognition using places database," in *Advances in Neural Information Processing Systems*, 2014, pp. 487–495.

[23] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," *arXiv preprint arXiv:1403.6382*, 2014.

[24] Liefeng Bo, Xiaofeng Ren, and Dieter Fox, "Multipath sparse coding using hierarchical matching pursuit," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 660–667.

[25] Naveen Kulkarni and Baoxin Li, "Discriminative affine sparse codes for image classification," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 1609–1616.

[26] Florent Perronnin and Christopher Dance, "Fisher kernels on visual vocabularies for image categorization," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–8.

[27] Saurabh Singh, Abhinav Gupta, and Alexei A Efros, "Unsupervised discovery of mid-level discriminative patches," in *Computer Vision–ECCV 2012*, pp. 73–86. Springer, 2012.

[28] Jun Zhu, Li-Jia Li, Li Fei-Fei, and Eric P Xing, "Large margin learning of upstream scene understanding models," in *Advances in Neural Information Processing Systems*, 2010, pp. 2586–2594.

[29] Jianxin Wu and Jim M Rehg, "Centrist: A visual descriptor for scene categorization," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 8, pp. 1489–1501, 2011.

[30] Li-Jia Li, Hao Su, Li Fei-Fei, and Eric P Xing, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Advances in neural information processing systems*, 2010, pp. 1378–1386.

[31] Megha Pandey and Svetlana Lazebnik, "Scene recognition and weakly supervised object localization with deformable part-based models," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1307–1314.

[32] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[33] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. IEEE, 2005, vol. 1, pp. 886–893.

[34] Timo Ojala, Matti Pietikainen, and Topi Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.