FEATURE CODING VIA VECTOR DIFFERENCE FOR IMAGE CLASSIFICATION

Xin Zhao[1,2]*, Yinan Yu*[2]*, Yongzhen Huang*[2]*, Kaiqi Huang*[2]*, and Tieniu Tan*[2]

1. Department of Automation, University of Science and Technology of China, China
2. National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China
Email: {xzhao, ynyu, yzhuang, kqhuang, tnt}@nlpr.ia.ac.cn

ABSTRACT

An effective image representation is important to an image classification task. The most popular image representation framework utilizes a feature coding algorithm to encode the extracted low-level feature descriptors into a vector representation. In this paper, we analyze the recently developed feature coding methods in a general way. According to their common characteristics, we propose a new coding scheme to perform feature coding based on the vector difference in a high-dimensional space which is obtained by explicit feature maps. As we illustrate, our method has promising results with small codebook sizes and generalizes most existing coding methods in a unified form.

Index Terms— Image classification, Feature coding, Vector difference, Additive kernel

1. INTRODUCTION

Image classification is a fundamental problem in computer vision. To do an image classification task, it is necessary to have an effective image representation. The Bag-of-Features (BoF) [1] model is one of the most popular models that are used for image representation. For a given image, the BoF based method extracts a set of local patches by interest point detection or dense sampling, and represents them as local feature descriptors (*e.g.*, SIFT [2]). Then, the descriptors are voted on the codebook which is obtained by clustering. At last, the voting results are represented as a histogram vector for classification. BoF model is a simple and effective model for image representation.

Recently, Fisher kernel (FK) [3] has gained much interest to do image representation. Fisher vector (FV) utilizes FK, where the feature distribution is taken into account in [4, 5]. To simplify the use of FV, the authors in [6] propose an approach that uses the vector of locally aggregated descriptors (VLAD). VLAD approach has the advantages of both BoF and FV. In order to further improve the VLAD approach, Picard *et al.* in [7] combine the second-order information for representation by using the vector of locally aggregated tensors (VLAT). In spite of many low-level feature coding methods, they have a common purpose that is to do a nonlinear feature mapping through the relationships between the extracted descriptors and the codebook [8]. After feature mapping, the similar patches may be close to each other in the transformed feature space and vice versa. In [8], the nonlinear feature mapping which is termed super-vector coding (SVC) has both properties of VLAD and BoF.

In this paper, we propose to do the nonlinear feature mapping by considering the vector differences between descriptors and codebook in a high-dimensional space by explicit feature maps. We represent each low-level feature descriptor by using the vector differences between the descriptor and the codebook as the new feature vector. In order to get a more discriminative representation, we use the explicit feature maps of additive kernels [9] in our method to transform descriptors and codebook into a high-dimensional space. Then, the vector differences between descriptors and codebook are computed in the high-dimensional space. Our final feature coding result has a similar form with the introduced coding methods, especially [6, 8]. Thus, our method is a general image representation method that can capture the properties of the existing coding methods in an intuitive way. At the same time, the proposed coding method has many kinds of representations since we can choose different kinds of additive kernels (*e.g.*, histogram intersection, Hellinger's and χ^2 kernels) for the explicit feature maps. This work makes the following contributions: 1) Our work provides a new analysis to the existing coding methods. 2) We propose a coding method based on vector difference in a high-dimensional space by explicit feature maps. 3) The proposed coding method provides a more unified view on most existing coding methods.

The remainder of the paper is organized as follows. In Section 2, we will analyze the existing methods for image representation. Then, the details of the proposed method will be introduced in Section 3. The experimental results and discussions will be shown in Section 4. At last, we will conclude our work in Section 5.

2. IMAGE REPRESENTATION

In an image classification task, a set of low-level feature descriptors $\mathbf{X} = \{\mathbf{x}_i \in \mathbb{R}^D, i = 1, \cdots, N\}$ (*e.g.*, SIFT [2]) is extracted to represent local patches of an image. Inspired by text categorization, the BoF model [1] votes the extracted descriptors to the codebook $\mathbf{C} = \{\mathbf{c}_j \in \mathbb{R}^D, j = 1, \cdots, K\}$ which

can be generated by a clustering method (*e.g.* K-means). The final image representation is a histogram vector whose elements are the responses on the codebook. We will briefly introduce some extensions of the BoF and analyze their properties in this section. The introduced methods are the state-of-the-art approaches for image classification tasks.

2.1. Fisher vector

In order to utilize the advantages of both generative and discriminative models, Perronnin *et al.* [4] propose to use FK [3] to construct Fishier vector (FV) for a new image representation method. Instead of using BoF, FV [4, 5] includes the high-order statistic information which describes the probability distribution of low-level feature descriptors.

The above ideas are achieved by modeling the whole extracted descriptors of m images $I = \{I_1, \cdots, I_m\}$ using a Gaussian Mixture Model (GMM). Suppose there are K Gaussians, $\Theta = \{\alpha_j, \mu_j, \mathbf{\Sigma}_j, j = 1, \cdots, K\}$ denote the parameters of the GMM. In the parameters, α_j, μ_j and $\mathbf{\Sigma}_j$ denote the weight, the mean vector and the covariance matrix of the j-th Gaussian, respectively. According to the rationale of FK, the log-likelihood of the descriptors in image I_p is $\mathcal{L}(\mathbf{X}_p) = \sum_{i=1}^{N} \log \gamma(\mathbf{x}_i)$. Herein, $\gamma(\mathbf{x}_i) = \sum_{j=1}^{K} \alpha_j \gamma_j(\mathbf{x}_i)$ is a GMM and the descriptors are assumed to be generated by the GMM independently. The goal of FK is that the similar images I_p and I_q should have similar partial derivatives $\mathbf{G}_p = \frac{\partial \mathcal{L}(\mathbf{X}_p)}{\partial \theta}$ and $\mathbf{G}_q = \frac{\partial \mathcal{L}(\mathbf{X}_q)}{\partial \theta}$ where $\forall \theta \in \Theta$. The FK score of the images I_p and I_q can be constructed as

$$K(\mathbf{X}_p, \mathbf{X}_q) = \mathbf{G}_p^T \mathbf{M}^{-1} \mathbf{G}_q, \tag{1}$$

where $\mathbf{M}$ is the Fisher information matrix. Since $\mathbf{M}$ is symmetric and positive definite, $\mathbf{M}^{-1}$ can be represented as $\mathbf{W}^T\mathbf{W}$ where $\mathbf{W}$ is the Cholesky decomposition of $\mathbf{M}^{-1}$. In [5], $\mathcal{G}_p = \mathbf{W}\mathbf{G}_p$ is defined as FV for image representation. FV has a closed-form solution which is the concatenation of the following D-dimensional gradient vectors with respect to the mean and the variance vector of j-th Gaussian,

$$\mathcal{G}_{p,\mu}^j = \frac{1}{N\sqrt{\alpha_j}} \sum_{i=1}^{N} w_{i,j} \left(\frac{\mathbf{x}_i - \mu_j}{\sigma_j}\right), \tag{2}$$

$$\mathcal{G}_{p,\sigma}^j = \frac{1}{N\sqrt{2\alpha_j}} \sum_{i=1}^{N} w_{i,j} \left[\frac{(\mathbf{x}_i - \mu_j)^2}{\sigma_j^2} - 1\right], \tag{3}$$

where $w_{i,j}$ is the soft-assignment of the i-th descriptor $\mathbf{x}_i$ to the j-th Gaussian of image I_p and σ_j is the variance vector of the diagonal covariance matrix $\mathbf{\Sigma}_j$ as we assumed. Thus, the final representation of FV in [5] is a $2DK$-dimensional vector where D is the dimensionality of the low-level feature descriptors and K is the number of Gaussians of the GMM.

2.2. Extensions of Fisher vector

In this subsection, we will introduce two kinds of extensions of Fisher vector. The first one is designed to simplify the use of Fisher vector for large scale problems. The second one is a well-established reconstruction based feature coding method which achieves promising performance on PASCAL VOC datasets.

2.2.1. Aggregated local feature descriptors

Considering the efficiency in large scale problems, Jégou *et al.* [6] propose an approach which is termed vector of locally aggregated descriptors (VLAD) to simplify FV. Instead of using GMM to model the feature distribution, VLAD uses a clustering algorithm (K-means in [6]) to generate a codebook. Then, the descriptors are voted on their nearest codewords. The response on each codeword is the accumulation of the differences between the nearest descriptors of the codeword and itself. The response on the j-th codeword can be denoted as follows:

$$\mathbf{r}_j = \sum\nolimits_{\mathbf{x}_i | \mathbf{NN}(\mathbf{x}_i) = \mathbf{j}} \mathbf{x}_i - \mathbf{c}_j, \tag{4}$$

where $\mathrm{NN}(\mathbf{x}_i)$ is the nearest codeword of $\mathbf{x}_i$. At last, VLAD concatenates the whole responses on the codebook and generates a DK-dimensional vector representation.

Further extension of VLAD has been proposed in [7] by using the combination of VLAD with high-order information. VLAT approach has two sub-terms which represent the first-order and the second-order information respectively. These two sub-terms have closed-form solutions as follows:

$$\mathbf{r}_j^1 = \sum\nolimits_{\mathbf{x}_i | \mathbf{NN}(\mathbf{x}_i) = j} \mathbf{x}_i - \mathbf{c}_j, \tag{5}$$

$$\mathbf{r}_j^2 = \sum\nolimits_{\mathbf{x}_i | \mathbf{NN}(\mathbf{x}_i) = j} (\mathbf{x}_i - \mathbf{c}_j)(\mathbf{x}_i - \mathbf{c}_j)^\top. \tag{6}$$

Concatenating the responses on all codewords, the VLAT approach has a $2DK$-dimensional vector which has the same dimensionality as FV for final representation.

2.2.2. Super-vector coding

Reconstruction based low-level feature coding is another point of view for image representation. In [8], the authors propose a SVC approach to extend Vector Quantization (VQ) by a function approximation scheme. The algorithm uses K-means clustering algorithm to obtain the codebook. The goal is to learn a smooth nonlinear function $\Psi(\mathbf{x}_i)$ of the extracted descriptor $\mathbf{x}_i \in \mathbf{X}$ and this function can be approximated as

$$\Psi(\mathbf{x}_i) \approx \Psi(\mathbf{c}_j) + \nabla\Psi(\mathbf{c}_j)^\top (\mathbf{x}_i - \mathbf{c}_j) \equiv \omega^\top \Phi(\mathbf{x}_i), \tag{7}$$

where $\mathbf{c}_j \in \mathbf{C}$ is the nearest codeword of $\mathbf{x}_i$ and $\Phi(\mathbf{x}_i)$ is a nonlinear feature mapping of $\mathbf{x}_i$.

Therefore, the super-vector coding is equivalent to the nonlinear feature mapping $\Phi(\mathbf{x}_i)$ which can be written as

$$\Phi(\mathbf{x}_i) = [s\lambda_j(\mathbf{x}_i), \lambda_j(\mathbf{x}_i)(\mathbf{x}_i - \mathbf{c}_j)^\top]^\top, \tag{8}$$

in which s is a small constant and $\lambda_j(\mathbf{x}_i) = 1$ when $\mathbf{c}_j$ is the nearest codeword to the descriptor $\mathbf{x}_i$ and $\lambda_j(\mathbf{x}_i) = 0$ otherwise. Eq. 8 can also be considered as the response of the descriptor $\mathbf{x}_i$ on the codeword $\mathbf{c}_j$. Thus, the final representation of $\mathbf{x}_i$ is a $(D+1)K$-dimensional vector which is the concatenation of the responses of $\mathbf{x}_i$ on the codebook.

3. OUR METHOD

Through the introductions of FV, VLAD, VLAT and SVC, we can conclude that all of them depend on the relationships between descriptors and codebook. For example, the codebook in FV can be the means of GMM and we denote $\mathbf{c}_j = \mu_j, j = 1, \cdots, K$. As shown in Eq. 2 and 3, the final representation of FV is the function of $(\mathbf{x}_i - \mathbf{c}_j)$ where $i = 1, \cdots, N; j = 1, \cdots, K$. Therefore, the FV mainly depends on the relationships between descriptors $\mathbf{X}$ and codebook $\mathbf{C}$. The same conclusions can be obtained in the other methods. Moreover, all the methods actually map descriptors to new transformed feature spaces by the nonlinear functions respectively.

The universality of the existing methods motivates us to reconsider the low-level feature coding in a more general way. In this section, we will give the details of our method and analyze its relations to the existing methods. Then, our method will be generalized to a unified form. As we will explain, based on the vector differences between descriptors and codebook, most existing feature coding methods can be considered as the special cases of our method.

3.1. Vector difference based feature coding

The goal of low-level feature coding is to force the similar patches to be close to each other in the transformed feature space after coding and vice versa. As we have analyzed in Section 2, the above idea is also captured by the existing coding methods, and they mainly depend on the relationships between descriptors and codebook. Intuitively, we describe those relationships by utilizing their vector differences. Therefore, it is important to choose an effective method to compute the vector difference.

As shown in the introduced methods, the normal vector difference between the descriptor $\mathbf{x}_i$ and the codeword $\mathbf{c}_j$ is $\mathbf{x}_i - \mathbf{c}_j$. In order to get a more discriminative vector difference, we use a feature map to transform descriptors and codebook into a high-dimensional space. Then, the vector differences of descriptors and codebook are computed in the mapped high-dimensional space. We denote $\mathcal{D}(\mathbf{x}_i, \mathbf{c}_j) \in \mathbb{R}^M$ as the vector difference between the descriptor $\mathbf{x}_i$ and the codeword $\mathbf{c}_j$ in a mapped M-dimensional ($M \geq D$) space.

Considering the purpose of feature coding, we propose to perform a nonlinear feature mapping of descriptors by using the vector differences between descriptors and codebook as new features. For a descriptor $\mathbf{x}_i$, the vector differences between $\mathbf{x}_i$ and K codewords are calculated. Then, the vector difference based coding result is constructed as a MK-dimensional feature vector

$$\Phi(\mathbf{x}_i) = [\lambda_1(\mathbf{x}_i)\mathcal{D}(\mathbf{x}_i, \mathbf{c}_1)^\top, \cdots, \lambda_K(\mathbf{x}_i)\mathcal{D}(\mathbf{x}_i, \mathbf{c}_K)^\top]^\top \quad (9)$$

where $\lambda_j(\mathbf{x}_i) = 1$ if $\mathbf{c}_j$ is the nearest codeword of $\mathbf{x}_i$ and $\lambda_j(\mathbf{x}_i) = 0$ otherwise. Under the nonlinear mapping $\Phi_\mathcal{K}$, the similar patches may be close to each other in the transformed space and vice versa.

3.2. Explicit feature maps and generalization

After the nonlinear feature mapping, the extracted descriptors are mapped to a transformed space. This is achieved by using the vector differences between descriptors and codebook as new features. However, most of the popular visual descriptors are based on histogram (*e.g.*, SIFT). It is proved that additive kernels [9] are effective to map the histogram feature to a high-dimensional space implicitly. In this work, we choose the explicit maps of additive kernels in [9] for feature maps.

An additive kernel is a positive definite kernel. It is designed to measure the similarity between two histogram feature vectors $\mathbf{x}$ and $\mathbf{y}$. It can be formulated as

$$\mathcal{K}(\mathbf{x}, \mathbf{y}) = \sum\nolimits_{k=1}^{D} \kappa(x_k, y_k). \quad (10)$$

The positive semi-definite function $\hat{\kappa}(x_k, y_k)$ is used to compute the similarity between the k-th elements of $\mathbf{x}$ and $\mathbf{y}$. Take χ^2 kernel for instance, it is an additive kernel obviously, since it has the form as $\kappa(x_k, y_k) = \frac{(x_k - y_k)^2}{x_k + y_k}$.

Normally, kernel function is calculated through an implicit feature map $\psi(x)$ of data points from a low-dimensional space to a high-dimensional space. Inspired by [9], we use an explicit feature map to approximate additive kernels as

$$\kappa(x_k, y_k) = < \hat{\psi}_\mathcal{K}(x_k), \hat{\psi}_\mathcal{K}(y_k) >, \quad (11)$$

$$\hat{\psi}_{\chi^2}(x) = e^{i\tau \log x}\sqrt{x \mathrm{sech}(\pi\tau)}, \quad (12)$$

in which $\hat{\psi}_{\chi^2}(x) \in \mathbb{R}^M$ is a M-dimensional explicitly mapped feature vector with χ^2 kernel and τ is considered as the index of the implicitly mapped feature vector $\psi(x)$.

With the explicit feature map $\hat{\psi}_\mathcal{K}(\mathbf{x})$, the vector difference $\mathcal{D}(\mathbf{x}_i, \mathbf{c}_j)$ in Eq. 9 is replaced with the difference of two mapped vectors $\hat{\psi}_\mathcal{K}(\mathbf{x}_i) - \hat{\psi}_\mathcal{K}(\mathbf{c}_j)$. Thus, the j-th response in Eq. 9 is reformulated into our final coding result

$$\Phi_\mathcal{K}(\mathbf{x}_i|\mathbf{c}_j) = \{\lambda_j(\mathbf{x}_i)[\hat{\psi}_\mathcal{K}(\mathbf{x}_i) - \hat{\psi}_\mathcal{K}(\mathbf{c}_j)]\}, \quad (13)$$

which is the response of the descriptor $\mathbf{x}_i$ on the codeword $\mathbf{c}_j$ and $\hat{\psi}_\mathcal{K}$ can be the explicit map of any additive kernel $\mathcal{K}$. $\hat{\psi}_\mathcal{K}(x)$ in Eq. 11 can be approximated by a finite number of samples by Fourier sampling theorem. In this work, we choose the number of samples $n = 1$. Thus, each element of the feature vector in the original space becomes $(2n + 1)$ elements in the transformed space after the explicit feature maps. The final representation of the vector difference based coding is a $(2n + 1)DK$-dimensional concatenated feature vector of the responses on the whole codewords.

Comparing Eq. 13 with the results of other methods, all of them utilize the relationships between descriptors and codebook. Without any operations of explicit maps, our method is similar to VLAD and SVC. If the high-order information is added, our method can be considered as an expansion of FV and VLAT with much more flexible representations. Thus, the proposed vector difference based coding method provides a unified form for understanding the existing coding methods.

4. EXPERIMENTAL RESULTS

In this section, we experimentally analyze our method and other state-of-the-art coding methods on the challenging PASCAL VOC 2007 dataset. We extract SIFT features densely with 3 different scales for the gray channel. The K-means algorithm is used for codebook generation. We use the explicit map of a modified χ^2 kernel function in [9] for feature map. The final representation of our method is a $(2n+1)DK$-dimensional feature vector. Herein, D equals 128, K is the codebook size and $n = 1$ is the number of samples. We aggregate the responses of the descriptors on the codebook by the same method as in [8]. All the experiments use $L2$ normalization for fair comparisons. A standard linear SVM is utilized for the final classification.

We compare our method with the state-of-the-art coding methods on VOC 2007 dataset. The mean Average Precision (mAP) is adopted for evaluation. We use very small codebook sizes in experiments to prove the effectiveness of our method. Our performance can also be further improved by larger codebook sizes and more powerful normalization methods as in [5].

Method	**BoF**	**VLAD**			**VLAT**			**Ours**		
Size	4K	16	32	64	16	32	64	16	32	64
plane	65.3	60.2	61.0	62.0	64.6	65.9	**66.1**	56.2	56.7	57.5
bike	44.5	35.3	38.6	40.3	45.3	47.2	49.2	43.7	47.1	**50.3**
bird	38.7	33.7	34.6	35.1	38.1	**39.6**	39.5	34.6	33.8	37.0
boat	49.4	53.7	55.8	55.3	57.4	59.5	58.7	58.1	61.1	**61.9**
bottle	18.7	13.9	16.0	17.2	17.2	18.5	19.1	16.6	18.9	**19.6**
bus	37.1	41.0	44.8	47.6	46.8	48.3	**48.9**	37.4	43.0	46.3
car	63.3	65.8	68.3	69.2	70.5	**71.7**	71.0	68.2	69.2	70.9
cat	33.9	35.2	37.9	40.9	40.8	41.1	43.1	40.6	43.0	**44.6**
chair	39.0	38.2	40.2	40.4	41.3	**43.1**	42.8	42.4	41.0	38.8
cow	24.8	22.7	22.7	25.3	25.2	26.9	27.3	29.6	25.9	**29.7**
table	22.3	23.3	25.9	28.3	25.7	30.4	30.4	**30.5**	30.3	29.7
dog	26.9	34.1	32.4	34.0	36.6	37.0	37.3	26.4	31.8	**38.0**
horse	58.4	65.8	65.2	66.7	70.7	70.6	**70.8**	64.6	66.9	68.8
motorb	36.2	45.2	47.2	49.2	49.6	51.1	51.3	42.7	50.4	**53.0**
person	75.6	75.6	77.4	78.0	79.4	**80.1**	79.9	73.8	76.3	78.5
plant	11.9	11.9	14.4	17.0	14.9	15.8	16.7	20.0	16.4	**23.3**
sheep	24.2	20.2	26.1	28.6	23.1	28.2	28.2	30.0	36.3	**40.3**
sofa	33.9	34.3	36.2	35.0	39.4	39.7	38.9	39.6	40.2	**41.7**
train	57.7	60.7	64.4	64.3	66.1	66.3	65.3	65.7	65.6	**66.5**
tv	39.8	37.3	34.9	38.3	38.1	42.3	**42.4**	32.0	36.0	40.0
mAP	40.1	40.4	42.2	43.6	44.5	46.1	46.4	42.6	44.5	**46.8**

Table 1. The classification results on VOC 2007 without SPM. Various codebook sizes: 4K (4000), 16, 32, and 64.

Table 1 summaries the results that use various codebook sizes without Spatial Pyramid Matching (SPM) [10] as in [7]. For fair comparisons, the results of BoF [11], VLAD [6] and VLAT [7] are obtained from [7]. We achieve the best mAP result and **12** single class results with small codebook sizes. It is worth to note that we use a linear SVM for the classification while [7] uses a standard SVM with a triangular kernel. Moreover, VLAT approach uses the second-order information and our method only uses the first-order information. It is also interesting to notice that our method has steadily increased results when the codebook size increases.

Method	**FV**	**SVC**	**Ours**
Codebook Size	32	32	32
mAP	45.98	48.02	**49.02**

Table 2. The classification results on VOC 2007 with SPM.

FV and SVC are designed to achieve better performance with SPM [5, 8]. Thus, we compare our method with FV and SVC by choosing 1×1, 2×2 and 3×1 cells for the associated three SPM levels. Table 2 summarizes the results with SPM. FV is implemented by using the components of the means of the GMM with the same settings in [6]. Therefore, all the compared methods use the first-order information for fair comparisons. Our method also outperforms FV and SVC on VOC 2007 with a small codebook size and SPM.

5. CONCLUSIONS

In this paper, we have proposed a generalized feature coding method for image representation. Our analysis on most existing coding methods leads to a new feature coding scheme that conducts coding through the vector differences between descriptors and codebook in a high-dimensional space by explicit feature maps. We have obtained promising results on the popular image classification dataset PASCAL VOC 2007 especially when the codebook size is very small.

6. ACKNOWLEDGMENT

This work is funded by National Natural Science Foundation of China (Grant No. 61175007,61175002), the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No. XDA06030300), the National Basic Research Program of China (Grant No. 2012CB316302), the Tsinghua National Lab for Information Science and Technology Cross-discipline Foundation (Grant No. Y2U1011MC1).

7. REFERENCES

[1] G. Csurka, C. Bray, C. Dance, and L. Fan, "Visual categorization with bags of keypoints," in *ECCV*, 2004.

[2] D. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[3] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in *NIPS*, 1998.

[4] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *CVPR*, 2007.

[5] F. Perronnin, J. Sanchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *ECCV*, 2010.

[6] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *CVPR*, 2010.

[7] D. Picard and P. Gosselin, "Improving image similarity with vector of locally aggregated tensors," in *ICIP*, 2011.

[8] X. Zhou, K. Yu, T. Zhang, and T. S. Huang, "Image classification using super-vector coding of local image descriptors," in *ECCV*, 2010.

[9] A. Vedaldi and A. Zisserman, "Efficient additive kernels via explicit feature maps," in *CVPR*, 2010.

[10] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.

[11] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *CVPR*, 2010.