BARYCENTRIC COORDINATES BASED SOFT ASSIGNMENT FOR OBJECT CLASSIFICATION

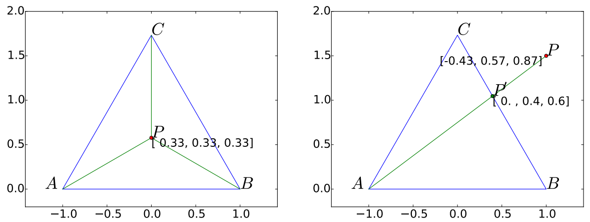
Tao Wei[1]*, Chang Wen Chen*[1]*, Changhu Wang*[2]

Department of Computer Science and Engineering, University at Buffalo, Buffalo, NY, 14260[1]
Microsoft Research, Beijing, China, 100080[2]
{taowei, chencw}@buffalo.edu, chw@microsoft.com

ABSTRACT

For object classification, soft assignment (SA) is capable of improving the bag-of-visual-words (BoVW) model and has the advantages in conceptual simplicity. However, the performance of soft assignment is inferior to those recently developed encoding schemes. In this paper, we propose a novel scheme called barycentric coordinates based soft assignment (BCSA) for the classification of object images. While maintaining conceptual simplicity, this scheme will be shown to outperform most of the existing encoding schemes, including sparse and local coding schemes. Furthermore, *with only single-scale features,* it is able to achieve comparable or even better performance to current state-of-the-art Fisher kernel (FK) encoding scheme. In particular, the proposed BCSA scheme enjoys the following properties: 1) preservation of linear order precision for encoding which makes BCSA robust to linear transform distortions; 2) inheriting *naturally* the visual word uncertainty which leads to a more expressive model; 3) generating linear classifiable codes that can be learned with significant less computational cost and storage. Extensive experiments based on widely used Caltech-101 and Caltech-256 datasets have been carried out to show its effectiveness of the proposed BCSA scheme in both performance and simplicity.

Index Terms— Object Classification, Barycentric Coordinates, Soft Assignment



(a) (b)

Fig. 1. Barycentric coordinates and truncation. (a) Barycentric coordinates are area coordinates in the context of triangles. (b) Truncation of the barycentric coordinates if a point that is outside of the polytope.

1. INTRODUCTION

Bag-of-words is a classic model in information retrieval for the processing of text documents (web pages) [1]. It has also been successfully applied to the field of computer vision [2], including content based image retrieval [3], image classification [2], and fine-grained object categorization [4]. Bag-of-words model in computer vision treats local image features as words, hence it is typically referred as bag-of-visual-words (BoVW) model. BoVW enjoys its success because it elegantly tackles the significant variability among object images. This variability is a challenging issue and is caused by different scales, different illuminating conditions, occlusions of background clutters, and variations of the pose for the objects within images.

The BoVW model [2] first collocates local image descriptors extracted from image patches, then quantizes the descriptors into a single vector, and finally classifies images according to their signature vectors. The BoVW model relies on the vector quantization (VQ) to allow a descriptor to be encoded into only one of its neighbors. However, hard assignment in VQ will incur large quantization errors and hence degrade the performance of image classification. An intuitive solution to this problem will be to design a soft assignment which allows the encoding to distribute over multiple neighbors. Such a soft assignment is expected to substantially reduce the quantization errors caused by VQ and improve the classification accuracy.

In [5], a soft assignment coding scheme called kernel codebook encoding (KCB) was proposed to assign the weights according to a kernel function that is inversely proportional to the distance of descriptors. Despite of its improvement over BoVW, sparse and local coding schemes [6, 7] gained popularity as they had shown better performance.

Researches on these newly developed encoding schemes indicate that max-pooling is able to achieve better classification performance because it selects the largest response rather than the averaged response. In [7], the authors have performed a comprehensive cross evaluation of different coding and pooling schemes, and conclude that max-pooling almost always improves average-pooling. Based on these observations, localized soft assignment (LSA) coding scheme is proposed in [8]. This scheme employs locality and max-pooling, and has shown to achieve comparable or even better performance over existing sparse or local coding schemes.

In this research, we propose a novel soft assignment coding scheme based on barycentric coordinates. This barycentric coordinate based soft assignment (BCSA) scheme is able to improve LSA with a margin of more than 3% classification accuracy, and achieve comparable as or better performance than the current state-of-the-art Fisher kernel (FK) [9] encoding scheme. Moreover, the proposed BCSA scheme is much simpler to compute than the FK scheme, and thus is useful for diverse scenarios when more extensive computing cannot be afforded.

Why this proposed BCSA scheme performs very well, especially comparing against LSA? The underlining reason is that BCSA preserves linear order precision. By linear order precision, we mean that, in the descriptor space, if a descriptor is a linear combination of its local neighbors, then in the encoding space, its code also has the same linear combination of the codes of its local neighbors. This property has the potential to make the encoding process more precise and more robust to coding transform distortions. Another reason shall be that this BCSA scheme *naturally* employs visual word uncertainty. Visual word uncertainty is also called probabilistic interpretation in LSA [8], and has been shown to be a key attribute for the good performance of image classification. Although both KCB and LSA enjoy visual word uncertainty, we notice that these visual word uncertainties are “artificial”, as the denominator normalizer is forced to apply by adding over the numerators. While in the proposed BCSA scheme, visual word uncertainty is “natural”, as it is a geometric property of barycentric coordinates as area coordinates and can be physically interpreted as the centroid of the masses. Finally, it is also worth noting that the proposed BCSA scheme has potential to be combined with convolutional neural networks (CNNs) [10] to further improve its performance, as the proposed BCSA algorithm steps described in Section 3.3 match with a CNN operation unit which is typically composed of a convolutional layer, a non-linear activation layer and a normalization layer.

To summarize, the proposed BCSA scheme has the following advantages:

1. Preserving linear order precision encoding, resulting from a basic property of barycentric coordinate system, making BCSA scheme more robust to linear transform distortions.
2. Inheriting *naturally* the probabilistic interpretation if we adopt barycentric coordinates to be affine, leading to a more expressive BCSA model.
3. Generating linear classifiable codes that can be learned with linear classifiers rather than non-linear classifiers, requiring only $O(n)$ training time, $O(1)$ testing time, and no kernel matrix storage.
4. Achieving comparable or better performance than the state-of-the-art Fisher kernel (FK) [9] encoding scheme with much simpler implementation with single scale features and compatible SPM pooling. It also has potential to be combined with CNN to further improve its performance.

2. BACKGROUND

BoVW is initially derived from the bag-of-words (BoW) model in information retrieval. It was introduced by Sivic and Zisserman [11] in the context of content based image retrieval, and by Csurka *et al.* [2] with application in image classification. The original BoVW is based on VQ for the descriptor encoding, which is to minimize the following function:

$$\min_{U,C} \sum_{n=1}^{N} \|d_n - u_n C\|^2 \tag{1}$$

$$\text{subject to } \|u_n\|_{l^0} = \|u_n\|_{l^1} = 1, u_n \geq 0, \forall n \tag{2}$$

where C is the codebook of cluster centers, U is VQ encoding of descriptors, d_n is the n-th descriptor and N is the total number of descriptors, u_n is the VQ encoding of descriptor d_n. It is easy to see that

$$u_{nk}^{VQ} = VQ(d_n)_k = \begin{cases} 1 & \text{if } k = argmin_j \|d_n - c_j\|^2 \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

In order to reduce the large quantization error introduced by hard assignment via VQ, an efficient way of using soft assignment for descriptor encoding has been proposed in [5]. This encoding scheme is defined as

$$u_{nk}^{KCB} = \frac{G_\sigma(dist(d_n, c_k))}{\sum_{k=1}^{K} G_\sigma(dist(d_n, c_k))}, \tag{4}$$

where $dist(d_n, c_k) = \|d_n - c_k\|^2$ is the l^2 distance between the n-th descriptor d_n and k-th codeword c_k, G_σ is a suitable kernel (e.g. Gaussian kernel) for soft assignment and σ is the smoothing parameter for the kernel, and K is the size of the codebook.

Inspired by the work in [7], a localized soft-assignment (LSA) coding scheme is proposed in [8] to remedy KCB's neglect of the underlying manifold structure of local features. LSA has been shown to be able to achieve comparable or even better accuracies comparing against sparse and local coding schemes. This scheme is proposed as

$$u_{nk}^{LSA} = \begin{cases} \frac{G_\sigma(dist(d_n,c_k))}{\sum_{k=1}^{K} G_\sigma(dist(d_n,c_k))} & \text{if } c_k \in Neighbor(d_n) \\ 0 & \text{other} \end{cases} \tag{5}$$

Besides soft assignment, several advanced descriptor encoding schemes have also been developed, including sparse-coded spatial pyramid matching (ScSPM) [6], local similarity global coding (LSGC) [12], vector of locally aggregated descriptors (VLAD) [13], and Fisher kernel encoding (FK) [9].

3. BARYCENTRIC COORDINATES BASED SOFT ASSIGNMENT

3.1. Barycentric Coordinates

Barycentric coordinates were introduced by Möbius in 1827 [14]. They describe the geometric centroid of three masses placed at the vertices of a reference triangle. Within proper representation of triangles, barycentric coordinates are also called area coordinates, because the barycentric coordinates of an interior point P of a triangle are proportional to the areas of the sub-triangles formed by this point and the corresponding vertices of the outer triangle (shown as $\triangle PBC, \triangle PAC$, and $\triangle PAB$ respectively, in Figure 1a).

Formally, let $\mathcal{A}$ be an affine space, and $p_1, \cdots, p_K$ be vertices in $\mathcal{A}$ that can form a simplex. For some point p in $\mathcal{A}$, if

$$(u_1 + \cdots + u_K)p = u_1 p_1 + \cdots + u_K p_K \tag{6}$$

is satisfied and at least one of $u_1, \cdots, u_K$ does not equal to zero, then we call the coefficients $(u_1, \cdots, u_K)$ as the barycentric coordinates of point p with respect to $p_1, \cdots, p_K$ in affine space $\mathcal{A}$. The barycentric coordinates defined above are homogeneous, and are equivalent up to a constant. To make the barycentric coordinates unique, we further employ the summation-to-unity condition

$$\textstyle\sum_{k=1}^{K} u_k = 1, \tag{7}$$

then the barycentric coordinates will be affine coordinates.

The definition of barycentric coordinates requires that $p_1, \cdots, p_K$ be vertices of a simplex, which means that the vectors $p_2 - p_1, \cdots, p_K - p_1$ are linear independent. However, this requirement is not always satisfied. Therefore, we need to define generalized barycentric coordinates with respect to convex polytopes.

Let Ω be a convex polytope in affine space $\mathcal{A}$ with vertices of $p_1, \cdots, p_K$. Any set of functions $u_k : \Omega \to \mathbb{R}, k = 1, \cdots, K$ is called generalized barycentric coordinates if they satisfy the following three properties

$$u_k(p) \geq 0, \forall k \quad \text{(non-negativity)} \tag{8}$$

$$\sum_{k=1}^{K} u_k(p) = 1 \quad \text{(summation to unity)} \tag{9}$$

$$\sum_{k=1}^{K} u_k(p) p_k = p \quad \text{(reproduction).} \tag{10}$$

Note that the non-negativity condition (8) is only applied to point $p \in \Omega$. If a point is outside the region of the convex polytope Ω, the generalized barycentric coordinates can be negative.

3.2. Linear Order Precision of Generalized Barycentric Coordinates

One particularly interesting property derived from the proposed BCSA is that the generalized barycentric coordinates are able to preserve the linear order precision: let $u_k : \Omega \to \mathbb{R}, k = 1, \cdots, K$ be the generalized barycentric coordinates defined in a convex polytope Ω, and $\phi : \Omega \to \Omega$ be any linear transform, we have

$$\sum_{k=1}^{K} u_k(p) \phi(p_i) = \phi(p). \tag{11}$$

This is actually a direct derivation from the definition of generalized barycentric coordinates, by combining the reproduction condition and linearity

$$\phi(p) = \phi\left(\sum_{k=1}^{K} u_k(p) p_k\right) = \sum_{k=1}^{K} u_k(p) \phi(p_i). \tag{12}$$

This property theoretically guarantees that the proposed barycentric coordinates based soft assignment is robust to linear transformations, which is common in various image classification schemes.

3.3. Barycentric Coordinates based Soft Assignment (BCSA)

The basic idea of BCSA is to assign a descriptor to several local nearest neighbors, instead of one single neighbor, according to the barycentric coordinates.

To be more precise, let $c_1, \cdots, c_K$ be the K cluster centers of the codebook. Rather than computing the barycentric coordinates in the affine space spanned by all $c_1, \cdots, c_K$, they are computed only within M-nearest neighbors of the descriptor. If $M = 1$, BCSA reduces to traditional VQ. There are two reasons to adopt the proposed approach. First, the computation of M-nearest neighbors in BCSA is much more efficient without tackling the whole data. Second, we assume that the nearest neighbors of one particular descriptor can form a simplex in most cases. Only few singular cases that may need to be calculated by the generalized barycentric coordinates.

Note that the proposed BCSA scheme is different from replacing the kernel in equation (4) with a linear kernel. In fact, BCSA is defined with respect to the "volumes" of simplexes and the encodings are positively correlated to these "volumes" while KCB is defined with respect to the "distances" and the encodings must be negatively correlated to the "distances."

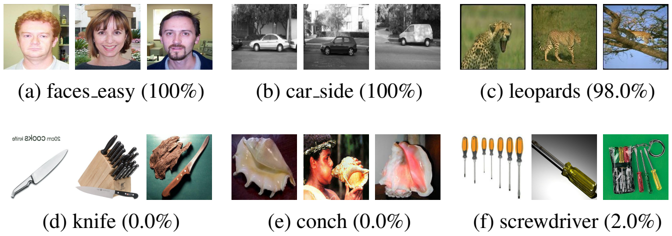
Let $C = (c_{i_1}, \cdots, c_{i_M}) \in \mathbb{R}^{D \times M}$ be the M nearest neighbors of a local descriptor $d \in \mathbb{R}^D$, where D is the descriptor dimension, and let u be the barycentric coordinates for the descriptor d. According to the definition of barycentric coordinates, the three properties (8), (9) and (10) will lead to the following formulation:

$$C \cdot u = d \tag{13}$$

(a) accordion (100%) (b) dollar_bill (100%) (c) snoopy (100%)

(d) inline_skate (0.0%) (e) wild_cat (0.0%) (f) octopus(20.0%)

Fig. 2. Caltech-101 dataset results. Top: samples of object classes achieve 100% accuracy. Bottom: samples of object classes perform poorly on BCSA.


(a) faces_easy (100%) (b) car_side (100%) (c) leopards (98.0%)

(d) knife (0.0%) (e) conch (0.0%) (f) screwdriver (2.0%)

Fig. 3. Caltech-256 dataset results. Top: samples of object classes perform well on BCSA. Bottom: samples of object classes perform poorly on BCSA.

$$\text{subject to } \|u\|_{l^1} = 1, u_m \geq 0, \ \forall m. \tag{14}$$

We need to pad equation (14) into (13). Let

$$\tilde{C} = \begin{bmatrix} \lambda \\ C \end{bmatrix}, \quad \tilde{d} = \begin{bmatrix} \lambda \\ d \end{bmatrix}, \tag{15}$$

where λ is a large value (e.g. $\lambda = 10^4$) introduced to enforce the summation to unity property. Then, we can get a single equation

$$\tilde{C} \cdot u = \tilde{d}. \tag{16}$$

And the original problem can now be reformulated as:

$$\min_u \|\tilde{C} \cdot u - \tilde{d}\|^2 \tag{17}$$

$$\text{subject to } u_m \geq 0, \ \forall m. \tag{18}$$

The above formulation is a non-negative least square problem, which can be solved by computing iteratively a set of basic vectors and the associated dual vectors [15]. However, this computation is not efficient enough for the encoding of large amount of descriptors. In order to resolve this problem, we apply a simple truncation technique as illustrated in Figure 1b. The point $P = (1.0, 1.5)$ with barycentric coordinates $[-0.43, 0.57, 0.87]$ is first truncated to $[0, 0.57, 0.87]$, and then re-normalized to $[0, 0.4, 0.6]$ with l_1 norm, which represents the approximate point $P' = (0.40, 1.05)$. An interesting fact is that this truncation operation preserves "linearity", which means that points A, P', P are co-linear in Figure 1b.

In summary, the proposed BCSA scheme takes the following steps:

1. Solve $\min_u \|\tilde{C} \cdot u - \tilde{d}\|^2$ without the non-negativity constraint.
2. Truncate the negative u components to 0: $\forall m \ u_m = \max(0, u_m)$.
3. Re-normalize u with l_1 norm.

It can be seen that the proposed BCSA scheme has a similar operation flow as a CNN [10]: the first step is actually a least square problem, which can be considered as a fully connected layer in a neural network; the second step is an ReLU layer; and the third step is a local response normalization layer. Hence, it shall have potential to be combined with CNN to further improve its performance.

3.4. Advantages of the Proposed BCSA Scheme

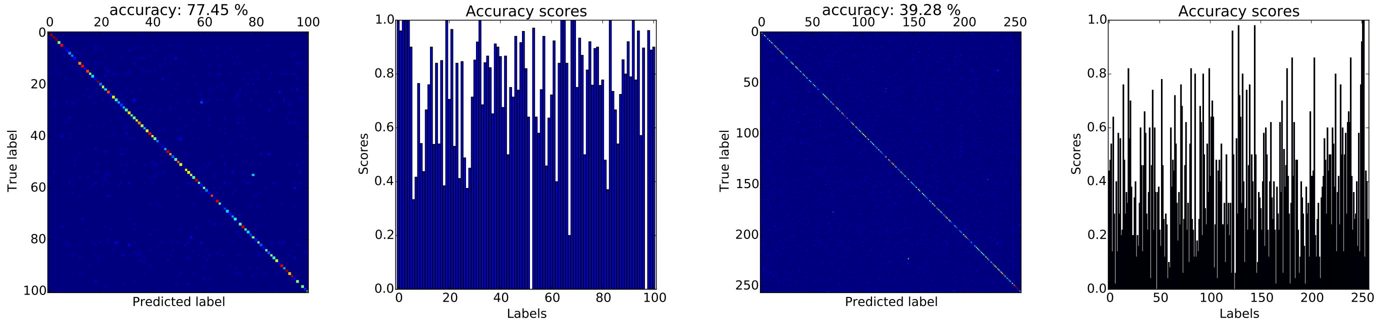
The proposed BCSA scheme has the following advantages.

First of all, the BCSA scheme preserves linear order precision for encoding. This was explained in detail in section 3.2, which is a property of generalized barycentric coordinates. In the context of local generalized barycentric coordinates system, this property is demonstrated as local linear order precision. This actually means that if the descriptors are encoded within the same local coordinates system, the encoding preserves linear order precision.

Second, this BCSA scheme *naturally* inherits the uncertainty of visual word ambiguity for local descriptors. Visual word ambiguity [5] is the action of assigning the same image descriptors to multiple visual words continuously which leads the model to be more expressive. Uncertainty is a probabilistic view of visual word ambiguity. Visual word uncertainty is also called probabilistic interpretation in LSA [8], and it has been shown to be a key attribute for the good performance in image classification [8]. Although visual word uncertainty also applies for KCB and LSA, it is considered "artificial" as the denominator normalizer is forced by adding over the numerators. While for the BCSA scheme, visual word uncertainty is considered "natural", as it can be interpreted both geometrically as area coordinates and physically as the centroid of the masses.

Third, the codes generated by the proposed BCSA scheme are highly linear classifiable. They can be learned efficiently by linear classifiers rather than non-linear ones with significant less computational cost and storage. In this research, we train the classifiers with linear SVMs, which have a complexity of $O(n)$ in training and $O(1)$ in testing. Whereas for non-linear kernel SVMs, they have a complexity of $O(n^2 \sim n^3)$ in training and $O(n)$ in testing. Besides, the linear SVMs do not require the storage of the kernel matrices. The size of those kernel matrices are $N \times N$, where N is the number of training images, which is typically very large.

Fourth, comparing against the current state-of-the-art FK encoding scheme, the proposed BCSA scheme is much simpler to implement with single scale features and SPM pooling. One significant drawback of FK is that the resulting signature



(a) Caltech-101 confusion matrix. (b) Caltech-101 accuracy scores. (c) Caltech-256 confusion matrix. (d) Caltech-256 accuracy scores.

Fig. 4. Caltech-101, Caltech-256 datasets with confusion matrices and accuracy scores of BCSA on them.

vector dimension is too high, typically $256 \times 80 = 20480$ [9]. This will lead to two potential problems: 1) In order to achieve good performance, FK requires denser and more scales of SIFT descriptors to be extracted from the original images. However, SIFT descriptors are expensive to compute [16]; 2) It has been shown [17] that spatial layout information is critical for object classification systems, whereas when FK is combined with SPM, it will result in $20480 \times 21 \approx 430K$ image signature vector, which demands unrealistic memory requirement for the training process.

4. EXPERIMENTAL RESULTS

4.1. Experiments Setup

In the proposed experiments, we first convert all the images into gray-scale ones and extract dense SIFT [16] descriptors from 16×16 batches with a step size of 4. Then, the descriptors are encoded by the proposed BCSA scheme to a codebook learnt by K-means clustering, where $K = 2048$. The number of neighbors M for each descriptor of BCSA coding is set to 5. Next, the descriptor codes are pooled by SPM [6] max-pooling and concatenated to generate the image signature vectors. Finally, the image signature vectors are classified with one-vs-rest linear SVMs. For training and testing, in this research, we follow the same setup as other researchers [5, 6, 12]. For both Caltech-101 dataset and Caltech-256 dataset, we use 30 images per category for training and up to 50 images per category for testing.

4.2. Caltech-101 Dataset

We first report our results on the Caltech-101 dataset [18]. This dataset consists of pictures of 101 different object categories, and contains 9146 medium resolution images (about 300×300). The number of images per category ranges from 31 to 800. Figure 2 shows some of the samples, and also illustrates three of the most and least successful categories that are classified by the proposed BCSA scheme. And Figure 4 (a) and (b) show the confusion matrix and accuracy scores of the proposed BCSA scheme on the Caltech-101 dataset. Table 1 compares the performance of the proposed BCSA scheme against several existing schemes. In this table, KCB, LSA, ScSPM, LSGC are the algorithms proposed by the original authors, and KSPM is a nonlinear χ^2 kernel SPM reported in [6]. The results in [7] that provides a comprehensive comparison for VQ, SA, and sparse coding are also included. For the FK encoding scheme, due to that the authors of [9] did not report results on the Caltech-101 dataset, we adopt the results from the well-known open source VLFeat library[1].

As can be seen the results in Table 1, the proposed BCSA performs much better than the KCB and KSPM schemes, resulting more than 13.0% improvements on the Caltech-101 dataset. The advantages of the proposed BCSA scheme over these two schemes are the linear classifiable codes generated by max pooling. The proposed BCSA scheme also outperforms ScSPM by a margin of 4.3%. This should be benefited from BCSA's robust encoding to linear distortions, as for sparse coding, a small perturbation could result in a totally different unit being activated. One interesting observation is that the proposed BCSA scheme is able to outperform LSA by a margin of 3.2%. Since both schemes employ locality and max-pooling, we believe that this performance gain contributes from BCSA's robust linear order precision encoding and *natural* probabilistic interpretation.

The table also shows that the proposed BCSA scheme using *single scale* SIFT features over-performs current state-of-the-art FK encoding scheme using seven scale SIFT features by up to 4.4% on this dataset. Note that the multi-scale SIFT features are critical for the performance of FK. When we adopt exactly the same setup as BCSA of single scale SIFT descriptors, replacing only the encoding scheme with the VLFeat implementation for FK, the result shows a much lower 61.20% accuracy.

[1] `http://www.vlfeat.org/applications/apps.html`

Table 1. BCSA classification accuracies on the Caltech-101 datasets.

Datasets Schemes	Caltech-101
VQ [17]	64.6±0.80
VQ [7]	64.3±0.9
SA (KCB) [5]	64.1
SA [7]	69.0±0.8
SA [8]	72.56±0.65
LSA [8]	74.21±0.81
KSPM [6]	63.99±0.88
Sparse (ScSPM) [6]	73.2±0.54
Sparse [7]	71.5±1.1
LSGC [12]	75.07
FK (muti-scale)	73.02
BCSA (single-scale)	**77.45**±0.90

4.3. Caltech-256 Dataset

The second experiment we carry out is on the Caltech-256 dataset [19]. This dataset is an extension of Caltech-101. It holds 30607 images which are split into 257 categories. The number of images in each category ranges from 80 to 827 and an average image size is 300 × 300 pixels. Caltech-256 is much harder than Caltech-101 for image classification due to its larger number of categories and more diverse poses and lighting conditions. Figure 3 illustrates three of the most and least successful categories that are classified by the proposed BCSA scheme. Figure 4 (c) and (d) show the confusion matrix and accuracy scores of the proposed BCSA scheme on the Caltech-256 dataset. And the experimental results are shown in Table 2.

As shown in Table 2, the proposed BCSA scheme achieves 39.28% accuracy on the Caltech-256 dataset. This accuracy is more than 12.0% higher than the KCB scheme, more than 5.0% higher than the ScSPM scheme, and is comparable to the FK encoding scheme. Note again that a significant advantage of FK is that the features are extracted at five scales in [9], whereas for the proposed BCSA scheme and the other schemes in Table 2, only single scale features are used for the encoding process.

5. CONCLUSIONS

In this paper, we have presented a novel scheme called BCSA that assigns the descriptors to a vocabulary codebook by using the barycentric coordinates. As a soft assignment encoding scheme, BCSA is able to outperform LSA by a margin of 3.2% in classification performance. We have also demonstrated in detail desired advantages of the proposed BCSA scheme: this scheme is able to preserve linear order precision during the encoding step that will benefit, inherit naturally uncertainty of visual word ambiguity, generate linear classifiable codes that can be learned with significant less computational cost and storage, and more friendly to single scale features and SPM pooling compared against current state-of-the-art Fisher kernel encoding scheme. Finally, extensive experiments on the Caltech-101 and Caltech-256 datasets have demonstrated that the proposed BCSA scheme is effective in both performance and simplicity.

Table 2. BCSA classification accuracies on the Caltech-256 datasets.

Datasets Schemes	Caltech-256
SA (KCB) [5]	27.2
KSPM [6]	29.51±0.52
Sparse (ScSPM) [6]	34.02±0.35
FK (multi-scale) [9]	40.8±0.10
BCSA (single-scale)	**39.28**±0.13

6. REFERENCES

[1] C. D. Manning, P. Raghavan, H. Schütze, *et al.*, *Introduction to information retrieval*, vol. 1. Cambridge university press Cambridge, 2008.

[2] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *ECCV Workshop on Statistical Learning in Computer Vision*, 2004.

[3] T. Ge, Q. Ke, and J. Sun, "Sparse-coded features for image retrieval," in *Proc. BMVC*, British Machine Vision Conference (BMVC), September 2013.

[4] O. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar, "Cats and dogs," in *Proc. CVPR*, June 2012.

[5] J. van Gemert, C. Veenman, A. Smeulders, and J.-M. Geusebroek, "Visual word ambiguity," *IEEE Trans. PAMI*, 2010.

[6] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. CVPR*, June 2009.

[7] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. CVPR*, IEEE, 2010.

[8] L. Liu, L. Wang, and X. Liu, "In defense of soft-assignment coding," in *Proc. ICCV*, IEEE, 2011.

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

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[11] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *Proc. ICCV*, Oct 2003.

[12] A. Shaban, H. Rabiee, M. Farajtabar, and M. Ghazvininejad, "From local similarity to global coding: An application to image classification," in *Proc. CVPR*, June 2013.

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

[14] H. Coxeter, "Introduction to geometry," 1969.

[15] C. L. Lawson and R. J. Hanson, *Solving least squares problems*. SIAM, 1974.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, 2004.

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

[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," in *Proc. CVPR Workshop*, June 2004.

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