LEARNING SEMANTIC KERNELS FOR SCENE CLASSIFICATION

Lei Zhang[1], *Xiantong Zhen*[2], *Jiqing Han* [3], *Xuezhi Xiang*[1]

[1] College of Information and Communication Engineering, Harbin Engineering University, Harbin, PRC

[2] Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, UK

[3] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, PRC

ABSTRACT

In this paper we propose to learn semantic kernels for scene classification. We first decompose the Object Bank representation into subspaces associated with each object, *Anchor Objects* are then created by clustering for each scene class separately. The *Anchor Distances* are computed to measure the distance between objects to scene classes. In order to take the advantage of the discriminative information from different scene classes, we propose semantic kernels based on the anchor distances to different classes for scene classification.

Through extensive experiments on two benchmark datasets: UIUC-Sports dataset and 15-Scene dataset, we prove that the proposed *Semantic Kernels* can significantly improve the original Object Bank and achieve state-of-the-art performance.

Index Terms— Object Bank, Anchor Objects, Semantic Kernels, Scene Classification

1. INTRODUCTION

In scene classification, how to measure the similarity between two scene images based on local features is a key problem. It is nontrivial due to that the cardinality of the image local descriptor set varies with different images and the elements are orderless.

The bag of words (BoW) model and match kernels [1] are two effective approaches to this problem. In BoW model, local feature descriptors for an image are projected into a new representation space spanned by visual words in a vocabulary, obtaining a histogram as a fixed-length vector to present the whole contents in this image. However, for match kernel approach, the kernels over sets of local features are defined to build the relations between two sets of local descriptors. In [1], it is proved that BoW can be viewed as a special case of the match kernels.

In fact, for image processing, the Object Bank approach is another way to describe an image into by a fixed dimension feature vector. The Object Bank represents an image as a response map of a large number of pre-trained object detectors and has achieved dramatic performances for visual recognition [2]. Due to the explicit detection of objects in images, the Object Bank, as a high-level representation, provides an effective avenue to understand scene images. Although the Object Bank approach can fill the semantic gap to some extent, it treats the object filters response equally, which is not satisfied our common sense. For scene classification, in fact, each scene is composed of several objects organized in an unpredictable layout, and objects play different roles for different scene classes. For example, 'bed' is more significant than 'window' for a bedroom scene while 'table' is crucial to an office scene.

In this paper, with the aim to take advantages of the high-level representation and to improve the Object Bank representations, we propose semantic kernels for scene classification. The rest of this paper is organized as follows. We revist the high-level image representation based on object bank in Section 2. In Section 3 we describe the proposed discriminative kernel in details. Section 4 demonstrates our experiments and results and our work is concluded in Section 5.

2. OBJECT BANK

The core of the idea in the object bank representation is to decompose an image according to a pre-refined object filter bank. Specifically, by concatenating the max response of each filter, we can generate the representation with each dimension corresponding to one object filter with certain a configure (scale, location and profile). From a semantic point of view, the obtained representation can give the details about the content of the image on behalf of how likely the corresponding object appears in this image.

In [2], according to the frequency of occurrences of objects in different datasets, 177 of the most frequent objects filter are trained to build object bank.

Given an image Q and a filter F in the object bank, the response of the filter at point (x, y) is the sum of products of the filter coefficients and the corresponding neighborhood points in the area spanned by the filter mask, which can be formulated as:

This work is partly sponsored by National Natural Science Foundation of China #91220301.

$$\sum_{x',y'} F\left[x',y'\right] \cdot Q\left[x+x',y+y'\right] \quad (1)$$

Moving the center point (x, y) to go through all the pixels in the image, responses for all the pixels are obtained. Since the filter can reflect the outline of each object, the sum operation in (1) essentially calculates the similarity of the object in the filter and the according image patch. If normalized, the maximum value can be viewed as the possibility of the object occurring in the image.

An image is finally represented by a feature vector concatenating the maximum responses of all the object filters. If the scales and the location information by spatial pyramid are further considered, the feature vector could be in a high-dimensional space. In our experiments, 177 object filters (each with front and profile models), 6 scales and 3-level spatial pyramid (with $1 + 4 + 16$ location's information) will lead to $177 \times 2 \times 6 \times 21 = 44604$ dimensions. Then for one image, this vector with 44604 dimensions reflects the contents of this image.

3. LEARNING SEMANTIC KERNELS

Given an image Q, its representation by the Object Bank is $\mathbf{I}^Q = \{\mathbf{I}_1^Q, \mathbf{I}_2^Q, ..., \mathbf{I}_N^Q\}$, where $N = 177$ is the number of objects with different profiles, scales and from differen levels of the spatial pyramid. In the Object Bank approach, it is assumed that each object is independent from others. Therefore, in the Object Bank representations, responses from all the object filters can be treated separately. For instance, $\mathbf{I}_n^Q$ is the corresponding subspace from the object n.

In the following sections, we divide the whole space into 177 subspaces according to different objects, and each subspace corresponds to responses from different objects with 2 profiles, 6 scales and 21 different pyramid levels. Thus the dimension of each subspace should be $2 \times 6 \times 21$.

3.1. Anchor Objects

As is known, the success of the Object Bank is the explicit modeling of objects which could incorporate more semantic meaning. In this paper, we aim to take the advantages of high-level representation of the Object Bank, and propose to deal with the distances between objects and classes, which is inspired by the naive Bayes nearest neighbor (NBNN) classifier. However, the nearest neighbor search in NBNN is extremely time-consuming.

To alleviate the computational burden of the NN search in NBNN, instead of using NN objects we propose to usee fewer representative objects called *Anchor Objects* for each subspace with certain object categories. This can significantly speed up the algorithm while being more robust and insensitive. The Anchor Objects could be obtained by either random sampling or clustering. We use the k-means clustering algorithm to create the Anchor Objects due to its effectiveness and efficiency. In addition, to be more discriminative, we propose to generate the Anchor Objects for each class separately.

In Fig. 1, it can be seen that the anchor points (objects) possess better generalization properties than the NN points. The red stars represent test samples, and the length of dashed line means the NN distance while the length of real line means the distance to the anchor point. As we can see that for the points in the misleading region (intersection) between two classes, the distances based on anchor points are more meaningful.

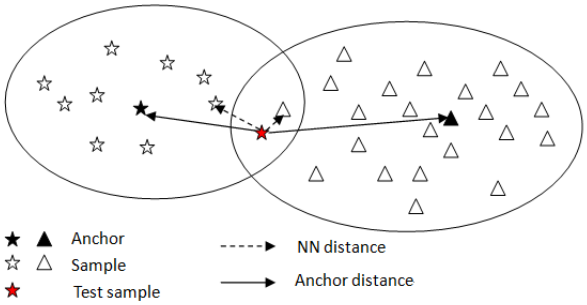

Fig. 1. Illustration of anchor objects.

3.2. Anchor Distances

Once we obtain the K Anchor Objects for each scene class by k-means, a complete new space spanned by the Anchor Objects can be obtained. Thus for each test sample, the distances of the object n to all the Anchor Objects for a scene class c in the n^{th} subspace will compromise a vector with each element corresponding the distance to an anchor [3], which we call *Anchor Distance*.

In fact, for each test sample, only the distance to the nearest anchor in one scene class is crucial to represent the distance to this class. It expresses the concept of object-to-class distance, which is the minimum among the distance of the test sample to anchors in scene class of n^{th} subspace. The whole algorithm to compute the distance of subspace n in test image Q, named $\mathbf{I}_n$, to scene class c is shown in Algorithm 1.

Algorithm 1 Computation of the distance from Object n to Class c

Input: Training dataset as $Q_n = \{Q_n^1, ...Q_n^c, ..., Q_n^C\}$
K as the cluster number in each class c
Output: The nearest anchor distance $d_A(\mathbf{I}_n, c)$
for all $c \in \mathbf{C}$ **do**
obtain K anchors $\{\mathbf{q}_{n1}^c, ...\mathbf{q}_{nK}^c\}$ for the object subspace samples in each class c by k-means on object subspace training set Q_n^c.
end for
$d_A(\mathbf{I}_n, c) = \min_{i=1,...K} dis_*(\mathbf{I}_n, \mathbf{q}_{ni}^c)$

The basic distance in Algorithm 1 noted as $dis_*(.)$ can be chosen as either the Manhattan distance in Equation 2 or the Euclidean distance in Equation 3.

$$d_M(\mathbf{I}_n^i, \mathbf{I}_n^j) = \sum_{d=1}^{D} \left| \mathbf{I}_n^i(d) - \mathbf{I}_n^j(d) \right| \tag{2}$$

$$d_E(\mathbf{I}_n^i, \mathbf{I}_n^j) = \sum_{d=1}^{D} \sqrt{(\mathbf{I}_n^i(d) - \mathbf{I}_n^j(d))^2} \tag{3}$$

3.3. Semantic Kernels

Inspired by the NBNN kernel [4], an extension of NBNN [5], which kernleizes the image-to-class (I2C) distances for support vector machines (SVM), we propose the kernerlization of the Anchor Distances.

Having $d_A(\mathbf{I}_n, c)$, we can predict a test sample by choosing the class with smallest sum of the d_A as $\sum_n d_A(\mathbf{I}_n, c)$. In this summation, each object may play different roles to different scene class. A smaller value of distance $d_A(\mathbf{I}_n, c)$, means this object is more significant than other objects to this scene class.

In the NBNN classifier, only the shortest distance as Eq.(4) is kept to predict the class labels, and the distances to other scene classes are ignored, which in practice tends to be less discriminative.

$$c^* = \arg\min_c \sum_n d_A(\mathbf{I}_n, c). \tag{4}$$

In fact, these distances carry discriminative information which could be useful for classification. In order to effectively utilize those information, as the NBNN kernels, we propose the kernerlization on the anchor distances to all the classes.

For two sets of features as $X = \{I_1^x, ...I_n^x, ...I_N^x\}$ and $Y = \{I_1^y, ...I_n^y, ...I_N^y\}$, the normalized sum match kernel as [6] are selected to satisfy the mercer condition in Eq. (5):

$$\begin{aligned} K(X,Y) &= \sum_{c \in \mathbf{C}} K^c(X,Y) \\ &= \frac{1}{|X||Y|} \sum_{c \in \mathbf{C}} \sum_n \sum_n k^c(I_n^x, I_n^y) \end{aligned} \tag{5}$$

where $\mathbf{C} = \{1, ..., c, ..., C\}$ is the set of all classes. The local kernel $k^c(I_*^x, I_*^y)$ is selected as:

$$\begin{aligned} k^c(I_n^x, I_n^y) &= \phi^c(I_n^x)^T \phi^c(I_n^y) \\ &= f^c(d_A(I_n^x, 1), ..., d_A(I_n^x, C))^T \\ &\times f^c(d_A(I_n^y, 1), ..., d_A(I_n^y, C)) \end{aligned} \tag{6}$$

$d_A(I_*^x, c)$ denotes the anchor distance from I_n^x to Class c. By using the local kernel function, I_*^x and I_*^y are not compared directly. Even if the two features I_n^x and I_n^y are far apart in the original feature space, they are considered to be similar if they have close distance to the same class. In practice, the kernel function $f^c(d_A(I_n^x, 1), ..., d_A(I_n^x, C))$ are chosen as:

$$f^c(d_A(I_n^x, 1), ..., d_A(I_n^x, C)) = d_A(I_n^x, c) \tag{7}$$

As mentioned above, the Anchor Distances explicitly incorporates the semantic information, namely the probability of an object occurring in a scene, the obtained kernels can therefore carry the semantic meanings and can be regarded as *Semantic Kernels*.

15-Scene			
bedroom	inside of city	industry	kitchen
mountain	living room	highway	suburb
coast	open country	store	office
street	tall building	forest	
UIUC-Sports			
rowing	snow boarding	badminton	polo
sailing	rock climbing	croquet	bocce

Table 1. Scene categories in 15-Scene and UIUC-Sports

4. EXPERIMENTS

To validate the effectiveness of the proposed semantic kernels for scene representation and classification, we have conducted extensive experiments on two benchmark scene datasets, i.e., the 15-Scene and UIUC-Sports datasets shown in Table 1.

4.1. Experiments Setting

We follow the experimental settings in [2]. For the **15-Scene dataset** [7], 100 images are randomly selected as training data and the rest for testing. For the **UIUC-Sports dataset** [8], 70 images are randomly drawn for training and 60 for testing. A linear SVM classifier [9] is employed for the final scene classification. 44604 dimensions (with 177 object filters, 2 profiles, 6 scales and 3 spatial pyramid)is considered in our experiments. As for K, the number of anchors in each scene class, it is fixed as 10. The Object Bank approach is chosen as baseline here. In Table 2, the discriminative representation in [3] is also incorporated for comparison.

4.2. Results on 15-Scene

The results on the 15-Scene dataset are shown in Table 2. We observe that the proposed semantic kernels with either the Manhattan or Euclidean distance can significantly improve the baseline Object Bank approach. Furthermore, the semantic kernels can also outperform the discriminative representations in [3]. The reason is probably due to that distances

	d_M	d_E
Discriminative Representation [3]	84.79%	84.52%
Semantic Kernels	85.07%	84.88%
Object Bank	82.03%	

Table 2. Performance of semantic kernels on 15-scene dataset.

	d_M	d_E
Discriminative Representation [3]	82.85%	82.73%
Semantic Kernels	81.79%	82.04%
Object Bank	77.5%	

Table 3. Performance of semantic kernels on UIUC-Sports dataset.

to all scene classes are exhaustively utilized by the kernels. In addition, it shows the similar performance for the Manhattan and Euclidean distances both for discriminative representation and the semantic kernels. The best performance is 85.07% with the Manhattan distance by the proposed semantic kernels.

4.3. Results on UIUC-Sports

Results on the UIUC-Sports dataset are reported in Table 3. The proposed semantic kernels significantly improves the Object Bank approach with a large margin, from 77.5% to 81.79% and 82.04% with the Manhattan and Euclidean distances, respectively. However, compared with the discriminative representations, the performance of the semantic kernels is a little lower. The reason is that in the semantic kernels, the dimension of mapped space is just the number of classes, while the dimension in the discriminative representations is K times of number of classes. For the challenging complex dataset e.g., UIUC-Sports, higher dimension is helpful for better performance.

4.4. Comparison with State of the Arts

We have also compared the proposed approach with the state-of-the-art methods in Table 4. We can see that out semantic kernels has achieved comparable even much better results than most of recently proposed methods in [10, 11, 12, 13, 14], which validate the effectiveness of the proposed semantic kernels.

5. CONCLUSION

In this paper, we have proposed semantic kernels for scene classification. By considering different object effects during classification and taking advantage of the discriminative information in the distance space, the proposed approach can provide more effective high-level representations than the original Object Bank (OB) approach. Extensive experiments on two benchmark 15-Scene and UIUC-Sports datasets have demonstrated that the proposed semantic kernels can significantly improve the original OB and achieve the state-of-the-art performances.

Methods	15-Scene	UIUC-Sports
Semantic Kernels	**85.07%**	**82.04%**
Shabou[10]	82.67%	87.23%
Niu[11]	82.5%	78%
Liu[12]	82.7%	82.29%
Gao[13]	83.68%	84.92%
Dixit[14]	83.2%	82.5%

Table 4. Performance comparison of the proposed approach with state-of-the-art.

6. REFERENCES

[1] Liefeng Bo and Cristian Sminchisescu, "Efficient match kernel between sets of features for visual recognition," in *NIPS*, 2009, pp. 135–143.

[2] L.J. Li, H. Su, E.P. Xing, and L. Fei-Fei, "Object bank: A high-level image representation for scene classification and semantic feature sparsification," *NIPS*, vol. 24, 2010.

[3] L Zhang, S Xie, L Shao, and X Zhen, "Discriminative high-level representations for scene classification," in *ICIP*, 2013.

[4] T Tuytelaars, M Fritz, K Saenko, and T Darrell, "The nbnn kernel," in *ICCV*, 2011.

[5] O Boiman, E Shechtman, and M Irani, "In defense of nearest-neighbor based image classification," in *CVPR*, 2008.

[6] Siwei Lyu, "Mercer kernels for object recognition with local features," in *CVPR*, 2005.

[7] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006.

[8] L.J. Li and L. Fei-Fei, "What, where and who? classifying events by scene and object recognition," in *ICCV*, 2007.

[9] Chih-Chung Chang and Chih-Jen Lin, "LIBSVM: A library for support vector machines," *ACM TIST*, vol. 2, pp. 27:1–27:27, 2011.

[10] A. Shabou and H. LeBorgne, "Locality-constrained and spatially regularized coding for scene categorization," in *CVPR*, 2012, pp. 3618–3625.

[11] Z. Niu, G. Hua, X. Gao, and Q. Tian, "Context aware topic model for scene recognition," in *CVPR*, 2012, pp. 2743–2750.

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

[13] S. Gao, I. Tsang, and L.T. Chia, "Kernel sparse representation for image classification and face recognition," in *ECCV*, 2010.

[14] M. Dixit, N. Rasiwasia, and N. Vasconcelos, "Adapted gaussian models for image classification," in *CVPR*, 2011.