Embedding Visual Words into Concept Space for Action and Scene Recognition

Behrouz Saghafi
Behr0002@e.ntu.edu.sg
Elahe Farahzadeh
Elah0001@e.ntu.edu.sg
Deepu Rajan
ASDRAJAN@ntu.edu.sg
Andrzej Sluzek
ASSLUZEK@ntu.edu.sg

School of Computer Engineering
Nanyang Technological University
Singapore

Abstract

In this paper we propose a novel approach to introducing semantic relations into the *bag-of-words* framework. We use the latent semantic models, such as LSA and pLSA, in order to define semantically-rich features and embed the visual features into a semantic space. The semantic features used in LSA technique are derived from the low-rank approximation of word-document occurrence matrix by SVD. Similarly, by using the pLSA approach, the topic-specific distributions of words can be considered dimensions of a concept space. In the proposed space, the distances between words represent the semantic distances which are used for constructing a discriminative and semantically meaningful vocabulary. We have tested our approach on the KTH action database and on the *Fifteen Scene* database and have achieved very promising results on both.

1 Introduction

The *bag-of-words*(BOW) framework has been shown to be useful in various computer vision applications like object recognition[2], scene recognition[5] and action recognition[4]. The framework builds a visual vocabulary by vector quantization of raw features extracted from an image or video. The vector quantization essentially involves clustering of the raw features by k-means and choosing a cluster's mean as the codebook or visual word. An unknown image or video is classified by a suitable classifier, according to its histogram of visual words. It is well known that the choice of the number of clusters in k-means clustering determines the discriminative ability of the visual vocabulary that is generated[24]. However, the more important drawback of k-means clustering is that it is based on the appearance of the image or video as represented in the raw features, as opposed to being based on the semantic relations between features. Utilizing the semantics inherent in visual content improves image/video categorization and understanding.

There have been several attempts at to incorporate semantics into the BOW model so that a more discriminative visual vocabulary is realized. Generative methods use latent variable

BMVC 2010 doi:10.5244/C.24.15

models like Probabilistic Latent Semantic Analysis(pLSA)[15, 23] and Latent Dirichlet Allocation (LDA)[5, 15] to obtain models for each category and subsequently to fit the query to one of the models in an unsupervised manner. Although these methods are efficient, their unsupervised nature limits their performance. Moreover, the number of topics in these methods is equal to the number of categories. This too limits their efficiency. Discriminative methods which incorporate label information have also been explored. Among the recent methods is the notable work of Liu and Shah which finds a semantic visual vocabulary via Maximization of Mutual Information (MMI) between visual words and images[10] or videos[11]. The algorithm starts with singleton clusters and in each iteration, merges two clusters which result in the minimum loss in mutual information. This procedure continues until a certain threshold in the information loss or in the number of clusters is achieved. This approach is effective in discovering the optimum number of clusters, but the formed clusters do not necessarily represent topics or synonym words which is required for constructing discriminative histograms. Liu et al.[12] use Diffusion Map (DM) to construct a semantic visual vocabulary. Unlike geodesic distance which is based on the shortest path between points, diffusion distance considers all paths between two points to measure the shortest distance, and, hence, is not sensitive to noise. However, considering connectivity in measuring the semantic distance is not appropriate in the presence of polysemy[1]. For example assume that word B is a polyseme with two distinct meanings: 1 & 2. If word B is connected to word A based on meaning 1 (they both have the same meaning 1) and also word B is connected to word C based on meaning 2, then words A and C will be connected in the diffusion distance framework, but they convey different meanings. So, diffusion distance does not always represent semantic distance.

Considering these drawbacks, we propose a method for action and scene recognition based on a semantic visual vocabulary that uses latent aspect models to embed visual words into a rich semantic space which we call the *concept space*. Using LSA (Latent Semantic Analysis) or its probabilistic version pLSA, the synonym words which convey the same meanings are embedded close to each other so that they can be clustered together into the same semantic cluster. The distance in the proposed concept space is practically based on the meanings of the words and thus, they represent the semantic relations. Consequently, the formed histograms based on these semantic clusters are efficient and discriminative for recognition.

In contrast with generative methods that do not make use of category labels, our method trains a classifier using the histograms from the training set. Moreover, in our method the number of topics can be changed as opposed to the unsupervised framework where this number is fixed to be the number of classes. This will allow us to analyse the semantic relations in more detail and consider as much topics as appropriate. On the other hand, pLSA is able to handle polysemy which is so effective in cases when different categories share the same topics e.g. walking, jogging and running, all have similar movements of the legs. We have tested our proposed method on the KTH human action database [22] and also on the *Fifteen Scene* dataset [8] with promising results.

1.1 Overview of the proposed framework

Figure 1 shows the flowchart for constructing the semantic visual vocabulary via embedding

[1] Polysemy is the existance of words which convey different concepts in different documents. For instance in text domain, the word *table* can either be interpreted as *a piece of furniture* or *an arrangement of data*.

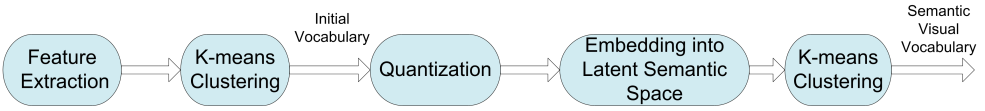


Figure 1: Constructing the semantic visual vocabulary.

into concept space. We first extract features from patches (cuboids) in the images (videos). The initial vocabulary is constructed by performing k-means clustering on the extracted features and choosing the cluster centers as the codewords. The feature vectors are quantized based on the initial codebook to form the word-image (or video) matrix which describes the occurrences of words in images/videos. The codewords are then embedded into the concept space by latent semantic models - we demonstrate the embedding both by LSA as well as by pLSA. Finally, the embedded codewords in the concept space are again clustered using k-means to obtain the desired semantic visual vocabulary.

2 Related Work

In this section, we review some methods that attempt to introduce semantic relations in the BOW framework for object, scene and action recognition. These approaches can be broadly divided into generative and discriminative methods. Generative methods usually involve hidden variables. These methods [6, 15, 23] try to model each image/video as a mixture of hidden concepts using either pLSA or LDA. On the other hand, discriminative methods are only based on observed variables. These approaches usually incorporate a classifer. Among these methods, Vogel and Schiele [25] define a set of concept classes (visual words) like *sand, sky* and *sea* to label image regions. In this method, image regions are represented by a combination of color and texture features and classified into concept classes. Thus for each image, a concept occurence vector is constructed and classified for scene retrieval. In labeling the databases containing ambiguous images, this approach claims that obtaining the ground truth for local semantic concepts is easier than for the whole image. However, this approach suffers from the large amount of manual work needed to annotate local regions. Randomized clustering forests have been used by Moosmann et al. [14] for image classification. Ensembles of decision trees are constructed based on the image class labels. Subsequently, visual words are assigned to each leaf. After building the trees, a bottom-up pruning process is done to reach a threshold of number of leaves to control the codebook size. They use randomized forests for clustering and quantization. Randomized forests have also been used in [7] for object classification and segmentation. In their work decision trees are used directly on image pixels to save time for extracting descriptors. In contrast to [14], which uses forests for clustering only, they use forests for both clustering and classification purposes. Randomized clustering forest is fast, yet discriminative, when compared to conventional k-means clustering, However, it tends to overfit, especially when applied in noisy situations. Moreover the model is complex and it is hard to understand the relation between the predictor variables. Quelhas et al. [19] have used the pLSA model to extract document-specific distributions of topics in order to represent images. This is followed by a SVM classifier in order to classify scenes. Bosch et al. [1] have also used a similar framework. Liu et al. [12] use diffusion distance to build a semantic dictionary. They construct a graph on the visual words in which the weights between points reflect the similarity. Visual words

are represented by pointwise mutual information. By applying the diffusion map, points are embedded into a lower dimensional space in which Euclidean distance is equal to diffusion distance. Our proposed approach is a discriminative one that tries to embed visual words into a concept space in which the dimensions are discovered in an unsupervised manner by latent topic models.

Next, we review some recent methods for action recognition which have not incorporated a semantic vocabulary in BOW framework, but we compare our results with them in the experiments. Schuldt et al. [22] use a sparse feature detector which is actually a 3D counterpart of the Harris corner detector. The detected features are described using spatiotemporal jets. The similarity between two actions is computed by a greedy match of features. A saliency-based feature detector is proposed by Rapantzikos et al. [20]. This feature detector uses color, motion and intensity and is based on multi-scale volumetric representation of the video. Schindler and Van Gool [21] have proposed a method based on snippets of video sequence. Inspired by biological systems, they extract features related to form and motion and compare them separately with learnt templates. The similarities are then concatenated into a single vector and classified using a bank of linear classifiers.

3 Concept Space

We obtain the initial vocabulary by performing k-means clustering on the extracted visual features. This initial codebook forms a reasonable-sized set representing all features, but they are not semantically clustered, i.e., the features in a cluster may convey different concepts. Thus, the formed histograms will not be semantically discriminative for classification. Therefore, we need a space in which semantically related words are adjacent. In order to find such a space, we use latent semantic models that find the underlying latent semantics given the occurrence matrix of word-image (or video). These models are the well known LSA and pLSA, which we briefly review in the following sections. We use *tf-idf* instead of the normal count in the occurrence matrix for a higher efficiency.

3.1 Embedding into concept space using Latent Semantic Analysis

LSA [3] finds a low-rank approximation for the word-document matrix. Here, document refers to the image or video sequence. The word-document matrix itself delivers semantics since synonym words appear in similar documents resulting in similarities among their occurrence vectors. However, the original word-document matrix is noisy, sparse and large. Hence, the low-rank approximation to the original matrix is desirable. The consequence of this dimension reduction is that the dimensions relating to synonym words (e.g. *see* and *look* in text domain) are merged. Let X be the occurrence matrix whose rows correspond to words and columns correspond to documents. Decomposing X using SVD as $X = U\Sigma V^T$ gives the orthogonal matrices U and V and the diagonal matrix Σ. Selecting the L largest singular values and their corresponding singular vectors, we find the rank-L approximation of X by $X \approx U_L \Sigma_L {V_L}^T$. The column vectors of U_L span the concept space of words and the columns of V_L span the concept space of documents. The i^{th} row of X, t_i, describes the i^{th} word. Consequently, the i^{th} row of U_L is the description of the i^{th} word in the concept space with L concepts and we refer to it as $\hat{t}_i$. In fact, each of the L dimensions in the low dimensional vector $\hat{t}_i$ shows the projection of the words along one of the concepts. It is expected that synonym words are close in the concept space.

3.2 Embedding into concept space using Probabilistic Latent Semantic Analysis

pLSA [8] is the statistical version of LSA which defines a generative model on the data. It is assumed that there is a latent topic variable z_l associated with occurrence of the word w_i in the document d_j. The observed variables are w_i and d_j while z_l is latent. The probability of observation pair $P(w_i,d_j)$ is $P(w_i,d_j) = P(w_i|d_j)P(d_j)$. Since the occurrences of w_i and d_j are assumed to be independent, we can marginalize over latent topics z_l in order to find the conditional probability $P(w_i|d_j)$, i.e.,

$$P(w_i|d_j) = \sum_{l=1}^{L} P(w_i|z_l)P(z_l|d_j), \tag{1}$$

where $P(z_l|d_j)$ is the probability of occurrence of topic z_l in the document d_j and $P(w_i|z_l)$ is the probability of occurrence of word w_i given the topic z_l. L is the total number of latent topics. Equation (1) is a decomposition of the word-document matrix, similar to LSA, but with the condition that the values are normalized to be probability distributions. We fit the model by determining $P(z_l|d_j)$ and $P(w_i|z_l)$ given the observation occurrence matrix. Maximum likelihood estimation of the parameters is performed using Expectation Maximization (EM) algorithm. Assuming a vocabulary of M words and N documents, the likelihood function to be maximized is $\prod_{i=1}^{M}\prod_{j=1}^{N} P(w_i|d_j)^{n(w_i,d_j)}$, where $n(w_i,d_j)$ is the number of words w_i in the document d_j and $P(w_i|d_j)$ is obtained by equation (1). The original pLSA algorithm in the unsupervised learning framework tries to categorize the query document given the learned parameters [8]. However, we use the pLSA algorithm only to determine the probabilities $P(w_i|z_l)$. In fact $P(w_i|z_l)$ is equivalent to the l^{th} dimension of $\hat{t_i}$ in the LSA framework. Thus, using pLSA we obtain the concept space embedded vector $\hat{t_i}$ as:

$$\hat{t_i} = \begin{bmatrix} p(w_i|z_1) & p(w_i|z_2) & \dots & p(w_i|z_L) \end{bmatrix}^T. \tag{2}$$

It should be noted here that L, the dimension of the concept space, does not need to be equal to the number of classes; this enables us to define arbitrary latent concepts. In fact the number of semantic topics can be much more than the number of classes. In other words, classes are wider concepts that may include some finer and more detailed concepts which are referred to as topics. For instance "computer" and "pen" are topics related to the class of office. Note that in the unsupervised framework in which pLSA is used (e.g, in [23],[15]), the dimension of the concept space must be equal to the number of classes.

In LSA, each word is projected onto a single point in the concept space so that each word can refer to a single meaning only. Instead pLSA is able to capture polysemy. Thus, given a word w observed in two different documents d_i and d_j, the topics associated with the word in d_i and d_j can be different or, in other words, $argmaxp(z|d_i,w)$ can be different from $argmaxp(z|d_j,w)$ [8],[9]. The advantage of LSA compared to pLSA is the faster and easier implementation. LSA needs a simple SVD, while pLSA uses the iterative EM algorithm which is only guaranteed to find a local maximum of the likelihood function [8].

4 Feature Extraction

For action recognition, we use the interest point detector proposed by Dollar et al.[4] which has shown better performances, compared to the sparse feature detector used by [22] in most

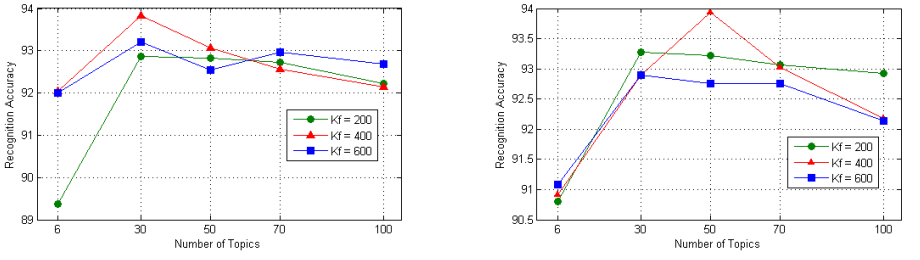


Figure 2: Performance of proposed method on KTH action dataset with different number of topics using (a) LSA (b) pLSA for embedding.

cases. The interest points are detected based on the local maxima of a response function, which incorporates a Gaussian kernel in the spatial domain and a Gabor filter in the temporal domain. Cuboids are extracted around each interest point. The cuboids are described by flattened gradients. The dimension of the descriptors are reduced to 100 using PCA to obtain the final feature vectors.

For scene recognition, dense features are more discriminative than sparse ones [16]. Accordingly, we use dense features sampled using regular grid with space of 8 pixels. SIFT descriptors[13] of 16×16 patches are used on the grid.

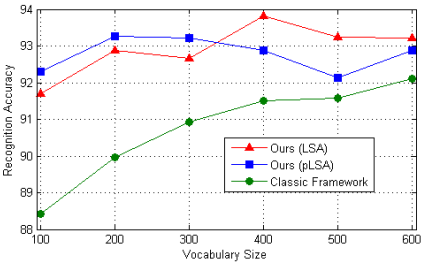
5 Experiments

We perform experiments on the KTH action database [22] and the *Fifteen Scene* database [8]. KTH action database is one of the largest and most challenging datasets for human action recognition. It consists of 6 actions - boxing, hand clapping, hand waving, jogging, running and walking - performed by 25 subjects under 4 different scenarios - outdoors, outdoors with scale changing, outdoors with different cloth and indoors with lighting variations. There are a total of almost 1200 video clips in this database. We use the leave-one-out cross validation technique to test the performance, i.e., each time we train with videos of 24 persons and use the videos pertaining to the remaining person for testing and report the average of the recognition results.

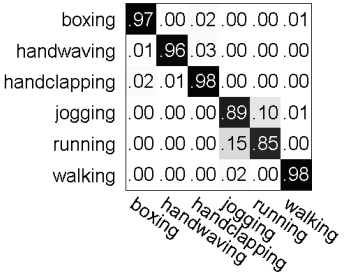
The *Fifteen Scene* database is also one of the most challenging datasets for scene classification. It consists of 13 categories reported in [5] and 2 categories added by Lazebnik et al.[8]. Eight of the categories are the same as the dataset by Oliva and Torralba[17] but in grey scale. To test the proposed algorithm on the scene database, we randomly select 100 images per class for training and the rest is used for test. Support Vector Machine (SVM) with Histogram Intersection kernel is used as the classifier. We choose the size of the initial codebook as 1500 for both action and scene experiments.

5.1 Experiments on KTH database

One of the advantages of the proposed method is that it allows the number of topics to be varied, in contrast to pLSA using unsupervised framework where the number of topics is constrained to be the same as the number of classes. Figure 2 (a) and (b) show the influence of number of topics L on the recognition accuracy with LSA and pLSA as the embedding



(a)



(b)

Figure 3: KTH action dataset. (a)Comparison of results with the classic framework for different sizes of vocabulary. (b)Confusion matrix for the best result achieved.

method. The experiments have been performed using three different semantic vocabulary sizes, *Kf*. As the number of topics is increased from $L = 6$, which is the number of classes, the recognition rate increases since the increased number of topics enables better discrimination between concepts. However, after around $L = 30$ topics, the recognition accuracy decreases. This is mainly because adding more dimensions to the concept space implies further division into semantic units that are not meaningful. This phenomenon occurs at $L = 50$ for pLSA with $Kf = 400$. The recognition accuracy has a variance of about 2%-3% as L varies.

In order to determine the efficiency and discriminative power of the proposed method compared to the classic BOW, which uses the initial vocabulary to build the query histogram, we compared the accuracy of our method with the classic framework with the same size of the final codebook. The results are illustrated in figure 3(a) for $L = 30$. The proposed method outperforms the classic BOW framework for all vocabulary sizes shown, illustrating that our approach is discriminative and effective in recognizing actions. Also the variance of recognition accuracy for our method in different vocabulary sizes is about 2%. Hence, our method is not sensitive to the size of the codebook. For small vocabulary sizes, pLSA outperforms LSA by a small margin due to its ability to handle polysemy. But as the vocabulary size increases, LSA performs better than pLSA probably because the larger codebook size brings in more details and compensates for the polysemy effect. However, pLSA takes into account every possible meaning of a word, even the rare ones, which results in confusion in larger vocabularies, thus reducing the accuracy. Also it should be noted that LSA has always a smaller implementation time compared to pLSA due to the time consuming iterative EM process for pLSA compared to the straightforward SVD in LSA.

The best recognition accuracy of our method is 93.94% which is achieved using pLSA with $L = 50$ and $Kf = 400$. Figure 3(b) shows the related confusion matrix. Most confusions are between jogging and running due to the strong similarities between these actions which are hardly distinguishable even for humans. Moreover, our method has successfully recognized walking despite its similarity to jogging and running.

In order to illustrate the efficiency of the proposed method, we have compared our results with other results in action recognition on the KTH database in table 1. The proposed method outperforms methods using semantic visual vocabulary, like Liu(MMI)[□] and Liu(DM)[□] and also some other important results reported on the KTH dataset. We have obtained the best result of 93.94% accuracy using BOW model without any spatial or temporal information. Hence, we have not compared our results with those that use structural information e.g. section 3 of [□], which reports an accuracy of 94.16%.

Table 1: Comparison with recently reported results for KTH dataset

Method	Schuldt[22]	Dollar[4]	Niebles[15]	Rapantzikos[20]
Accuracy(%)	71.7	81.2	83.3	88.3
Method	Liu(MMI)[11]	Liu(DM)[12]	Shindler[21]	**Ours**
Accuracy(%)	91.3	92.3	92.7	**93.9**

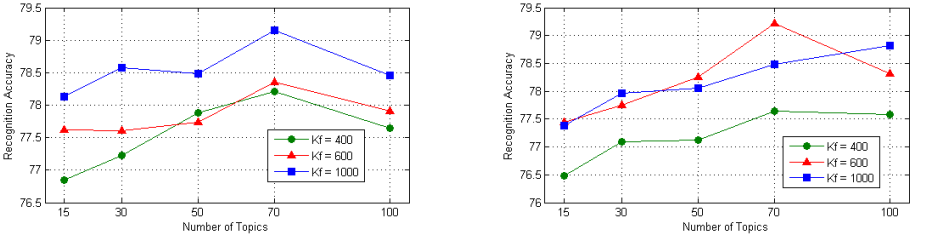


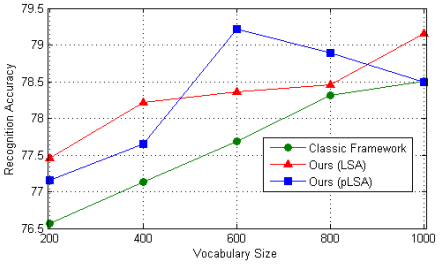
Figure 4: Performance of proposed method on 15 scene dataset with different number of topics using (a)LSA (b)pLSA for embedding.

5.2 Experiments on *Fifteen Scene* database

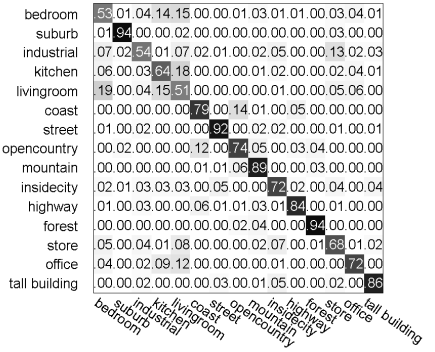
We follow identical experiments for scene recognition as was done for action recognition. As mentioned earlier, one of the advantages of our method compared to unsupervised pLSA is its capability of increasing the number of topics beyond the number of classes. This enables us to consider semantics in more detail. The first experiment studies the effect of the number of topics L on the recognition accuracy for three different sizes of the semantic vocabulary using LSA and pLSA for embedding. The results are illustrated in figure 4. Similar to the results of action recognition, the scene recognition accuracy improves with increasing the number of topics until $L = 70$, after which there is a deterioration in the accuracy. The justification for this trend is similar to that of action recognition.

To verify the efficiency and discriminative ability of the method in scene classification, we have compared it with the classic BOW framework for different vocabulary sizes. The results are shown in figure5(a). The number of topics is chosen to be 70. According to the figure, our method outperforms the classic BOW framework in all cases. This shows the efficiency of the method proposed. Apart from the instabilities in the begining of the curves, we can say that the behaviour of LSA and pLSA in the stable part (final part) is similar to the one in action recogntion with the same explanations, i.e. the performance of LSA tends to increase in larger sizes of the codebook while for pLSA the accuracy gets worse.

The best result achieved on *Fifteen Scene* dataset with our method is **79.22** using pLSA model and the semantic codebook size of 600. The confusion matrix for this accuracy is shown in figure5(b). The recognition rate for outdoor scene classes (e.g. street, suburb, tall building and forest) is much higher than the indoor scenes (e.g. living room ,kitchen, office, store). As mentioned in [18], this is because outdoor scenes can be clearly categorized using global features, but for indoor scenes global features are not enough for exact classification. For these categories of scenes, object identification might help in increasing the accuracy for scene recognition. Table 2 summarizes recognition accuracy of our method and some notable related works. As seen from the table, the proposed method is the best. Just as in action recognition experiments, we have not used any spatial information in this method and



(*a*)



(*b*)

Figure 5: 15 scene dataset. (a)Comparison of results with the classic framework for different sizes of vocabulary. (b)Confusion matrix for the best result achieved.

Table 2: Comparison with recently reported results for *Fifteen Scene* dataset.

Method	Fei-fei [5]	Quelhas [19]	Bosch [1]	Liu(DM) [12]	Liu(MMI) [11]	**Ours**
Accuracy(%)	65.2	71.24	73.4	74.9	75.16	**79.22**

hence, we have not compared the results with approaches that use spatial information e.g. spatial pyramid matching [8]. The work of Fei-fei and Perona [5] which has used LDA in a generative framework, has a lower performance compared to the methods using a semantic vocabulary but incorporating category labels like Liu(DM) [12] and Liu(MMI)[11]. Also the works of Quelhas et al.[19] and similarly Bosch et al.[1] have lower accuracy compared to the works using co-clustering to obtain semantic vocabulary like Liu(MMI)[11], Liu(DM)[12] and ours. This is mainly because in contrast to former methods, which use a histogram of topics equal to the number of categories, latter methods perform the clustering step in the semantic space to further group the semantically related words together and to construct more discriminative histograms with the actual number of topics. Also, comparing the results of DM and MMI in the tables 1 and 2 we realize that DM is performing better in classification of actions compared to scenes. This is probably because the visual words extracted in action videos are noisier than those extracted in scene images, so DM (which aims to be robust to noise) has a better efficiency in action recognition.

6 Conclusion

In this paper, we have proposed a novel approach for using semantic relations in BOW framework. We have used the latent aspect models such as LSA and pLSA to map the visual words into a semantic space. Under the LSA framework, this mapping is done by low rank decomposition of the word-document occurrence matrix using SVD. By using pLSA, the topic-specific distributions of words are considered the projection of the words along different concepts. The distances in the proposed concept space reveal the semantic relations. Clustering is done in the concept space to capture the semantic structures. We tested our method on two complex datasets of human actions and scenes. Our results confirm that the proposed features are efficient and discriminative for scene and action recognition. Also,

our method performs better compared to some similar methods for constructing semantic vocabularies.

References

[1] A. Bosch, A. Zisserman, and X. Muñoz. Scene classification via plsa. In *ECCV*, 2006.

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

[3] S. Deerwester, S.-T. Dumais, G.-W. Furnas, T.-K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[4] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005.

[5] L. Fei-fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, 2005.

[6] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1-2):177–196, 2001.

[7] R. Cipolla J. Shotton, M. Johnson. Semantic texton forests for image categorization and segmentation. In *CVPR*, 2008.

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

[9] D. Liu and T. Chen. Unsupervised image categorization and object localization using topic models and correspondences between images. In *ICCV*, 2007.

[10] J. Liu and M. Shah. Scene modeling using co-clustering. In *ICCV*, 2007.

[11] J. Liu and M. Shah. Learning human action via information maximization. In *CVPR*, 2008.

[12] J. Liu, Y. Yang, and M. Shah. Learning semantic visual vocabularies using diffusion distance. *CVPR*, 2009.

[13] D.-G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2): 91–110, 2004.

[14] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. In *NIPS*, 2006.

[15] J.-C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79(3):299–318, 2008.

[16] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*, 2006.

[17] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001.

[18] A. Quattoni and A. Torralba. Indoor scene recognition. In *CVPR*, 2009.

[19] P. Quelhas, F. Monay, J. m. Odobez, D. Gatica-perez, T. Tuytelaars, and L. Van Gool. Modeling scenes with local descriptors and latent aspects. In *ICCV*, 2005.

[20] K. Rapantzikos, Y. Avrithis, and S. Kollias. Dense saliency-based spatiotemporal feature points for action recognition. In *CVPR*, 2009.

[21] K. Schindler and L. Van Gool. Action snippets: How many frames does human action recognition require. In *CVPR*, 2008.

[22] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local svm approach. In *ICPR*, 2004.

[23] J Sivic, B.-C. Russell, A.-A. Efros, A. Zisserman, and W.-T. Freeman. Discovering objects and their location in images. In *ICCV*, 2005.

[24] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *IJCV*, 62(1):61–81, 2005.

[25] J. Vogel and B. Schiele. Natural scene retrieval based on a semantic modeling step. In *CIVR*, 2004.