Evaluating Effectiveness of Latent Dirichlet Allocation Model for Scene Classification

Shizhi Chen and YingLi Tian
Media Lab, Department of Electrical Engineering
The City College, City University of New York
New York, USA
{schen21, ytian}@ccny.cuny.edu

***Abstract*— Scene classification from images is a challenging problem in computer vision due to its significant variability of scale, illumination, and view. Recently, Latent Dirichlet Allocation (LDA) model has grown popular in computer vision field, especially in scene labeling and classification. However, the effectiveness of the LDA model for the scene classification has not yet been addressed thoroughly. Especially, there is little experimental evaluation on the model's performance for different types of features. Fusion of multiple types of features is usually necessary in the scene classification due to the complexity of scene images. In this paper, we investigate the effectiveness of the LDA model in scene classification by using 7 types of features (i.e. uniform grid based interest points, Harris corner based interest points, scale invariant feature transform (SIFT), texture, shape, color, and location) and their various combinations. Furthermore, we compare the performance of the LDA model with Support Vector Machine (SVM) classifier. All experiments are performed on the UIUC Sport Scene database. The experiments demonstrate that the performance of the LDA model 1) is significantly lower than the SVM classifier for the scene classification over different types of features; and 2) decreases by fusing multiple features while improvement shown in SVM classifier.**

Keywords: scene classification; LDA; SVM; feature fusion; bag of words;

I. Introduction

Given an image of a complicated scene, can a computer immediately recognize the scene category? Scene classification is very challenging. As shown in Figure 1, even for the same scene category, there are significant variations of backgrounds, lighting, scale, rotation, etc. For example, Figure 1(a) shows two images in Badminton category. However, it is very difficult to extract effective common features to classify them as the same category.

Despite the challenges in the scene classification, recently, several approaches have been suggested [2, 6, 14, 15, 17]. Among them, topic discovery models, such as probabilistic Latent Semantic Analysis (pLSA) [9] and Latent Dirichlet Allocation (LDA) models [1] have grown popular [2, 6, 11, 17]. Bosch *et al.* [2] classify scene images using four different types of visual features under the framework of pLSA. Fei-Fei and Perona [6] adopted the LDA model in the scene classification with features of pixel intensity and Scale Invariant Feature Transform (SIFT) descriptor [13].

Due to the complexity of scene images, single type of feature is not able to handle large intra-class variations. Multiple types of features, which provide complementary information of images, are usually necessary in the scene classification. Li *et al.* [11] combines four types of region features and one type of interest point features under the LDA framework for scene classification. The four types of region features are texture, shape, color, and location features of scene images. The interest point feature is uniform grid interest points with SIFT descriptor. These features describe different perspective of scene images and provide complementary information.

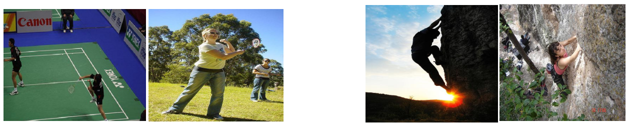

(a) Badminton (b) Rock-Climbing

Figure 1: Sample scene images from the UIUC Sport Scene database [12]. (a) Badminton scene with significantly different backgrounds; (b) Rock-Climbing scene with different viewing angles and illumination.

However, the effectiveness of the LDA model for the scene classification has not yet been addressed thoroughly. Especially, there is little experimental evaluation on the model's performance with fusion of multiple types of features. Li *et al.* [11] simply employ the LDA model to combine several commonly used features in scene classification without considering the contribution of each individual type of features. Lazebnik *et al.* [10] compare the LDA model with the Support Vector Machine (SVM) classifier [4] only over the Spatial Pyramid Matching features.

In this paper, we attempt to address these issues by experimentally evaluating several types of common features for scene classification under the LDA framework, as compared to the benchmark classifier, i.e., the SVM model with the bag of words representation [5]. The SVM has been a very popular classifier in object classification, which usually achieves the state of the art performance. We also evaluate the effectiveness of each type of features as well as the fusion of multiple types of features.

II. METHOD

A. Overview

The overview of our method for the scene classification is summarized as a four-step process shown in Figure 2. The first step is to extract visual features for all images. Then we employ a k-mean clustering algorithm to form a codebook of visual vocabulary, followed by a vector quantization process based on the constructed codebook. Finally a LDA Model or SVM classifier is used to classify the scene images.

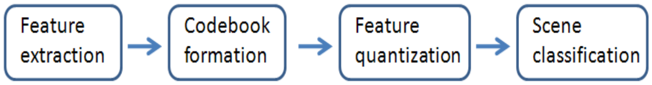
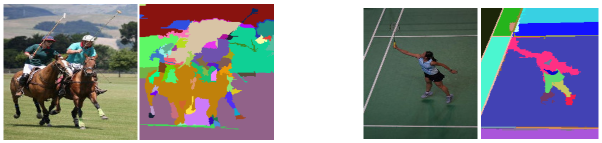


Figure 2. Overview of the approach for scene classification. (1) visual feature extraction; (2) k-mean clustering to form a codebook of visual vocabulary; (3) vector-quantized visual features based on the constructed codebook; (4) train and test using LDA model or SVM classifier.

B. Feature Extraction

Feature extraction is a critical step in scene classification. Distinctive features can make classification much easier. In this section, seven commonly used features in the scene classification are described in details, including both region features and interest point features.



(a) Polo (b) Badminton

Figure 3. Segmentation examples of two scene images. Pixels with the same color belong to the same segment.

B1. Region Features: Texture, Shape, Location, and Color

Following the approach in paper [11], four types of region features are extracted: texture, shape, location, and color. Before generating any region features, we first perform segmentation on images using the algorithm proposed by Felzenszwalb [7]. As shown in Figure 3, pixels with same color belong to one segmented region. Then at each segmented region, four types of region features are extracted.

Texture feature is generated by passing the original image with S filter bank [18]. S filter bank is rotationally invariant with 13 isotropic. Therefore, there are 13 responses for each image. The mean and standard deviation of each response are calculated for individual segmented regions in the image. In other words, each segmented region has 13 means and standard deviations of the filter responses. These means and standard deviations are combined together as the texture feature of each segmented region.

A simple shape feature is calculated in this experiment. The size of each segmented region are determined based on the maximum length of the segmented region in x and y directions. Then each segment's shape feature is formed by combining the size and the number of pixels in it.

Location feature of each segment can be generated using the following two steps. First, a binary image is formed by setting all pixels in a segmented region to 1, while the remaining area in the image is set to 0. Then the binary image is resized down to an 8 by 8 image through the standard bi-linear interpolation. Second, the normalized top position is formed by dividing the minimum *y* position of a segment over the image height. Normalized bottom position of a segment is formed by dividing the maximum *y* position of a segment over the image height. Since the ratio of a segment's height over width has been captured by the 8 by 8 location image, the normalized left and right position of a segment can be inferred from the top and bottom positions. The location feature of a segment is then formed by combining the 8 by 8 resized binary image with the normalized top and bottom positions of the segment.

Color features are formed by calculating each segmented region's color histogram over RGB color space. Each color space is divided into 10 bins. Therefore the color feature vector of a segment has 1000 dimensions.

B2. Interest Point Features: Uniform Grid, Harris Corner, and SIFT Interest Points

In addition to the region features above, three types of interest point features are evaluated in our experiments: Uniform Grid interest point, Harris corners and SIFT interest points (corners detected by the SIFT detector). Regardless the type of interest point detector used, SIFT descriptor is used to describe all interest points [13].

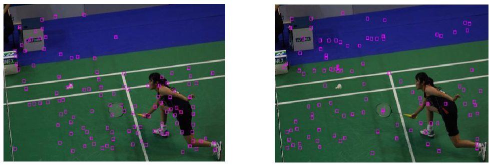
Interest Point Extraction: Uniform Grid method samples interest points uniformly along the *x* and *y* directions in the image. The distance between the adjacent interest points is set as 10 pixels in order to have dense sampling points to represent a scene image. Number of interest points generated for a typical image (resolution of 300 by 500) is around 1500.

Unlike the Uniform Grid method, Harris corner detector utilizes gradient information to detect more stable interest points in an image [8]. The average number of Harris corners in an image is approximately 100 in our experiments, which is significantly less than the number of Uniform Grid interest points. As we can see from Figure 4(a), Harris corners have captured most objects which are important in Badminton category, such as shuttlecock, badminton racket and human.

SIFT detector [13] finds corners by detecting local maxima or minima in both spatial space and scale space. Therefore, SIFT interest points achieve scale invariant. The corners are shown in Figure 4(b). However, SIFT interest points missed the shuttlecock object in the same image. Furthermore, as compared with the Harris corners, the SIFT interest points capture few interest points on badminton racket and human objects. We also limit the number of corners detected by SIFT detector to around 100 in order to compare with the Harris corners fairly.

Interest Point Descriptor: SIFT descriptor [13] is used to describe all interest points regardless their detection methods. A square descriptor window with each interest point at its center is extracted. The descriptor window size is 24x24 pixels with 4x4 uniformly sampled sub-regions over the descriptor window. For each sub-region, an 8-Bin orientation histogram of gradients within the descriptor window is constructed. The gradient magnitudes are furthered weighted by a Gaussian function with its mean corresponding to the center of each sub-region. Then all histograms are concatenated together to form the SIFT descriptor, which has 128 feature dimensions.

At this point, all features including four types of region features and three types of interest point features have been extracted. The next step is to form a codebook of visual vocabulary and to vector-quantize the extracted features.



(a) Harris Corners (b) SIFT interest points

Figure 4. An example of feature extraction of (a) Harris corners; and (b) SIFT interest points.

C. *Codebook Formation and Feature Quantization*

After extracting feature vectors from the training images, k-mean clustering algorithm is applied to group the feature vectors together based on Euclidean distance. As a result, a set of center feature vectors are formed as representatives of all features. The resulted center feature vectors are the codebook vocabulary [5].

Features in each image are vector-quantized to one of the center feature vectors in the codebook. Vector quantization process of a visual feature is to find a center feature vector in the codebook with the smallest Euclidean distance. Then the visual feature is represented by the closest center feature vector. The purpose of vector quantization is to reduce the feature space complexities by using a small set of representative feature vectors.

The number of codebook size for each feature type is summarized in Table 1.

At this point, feature processing is complete. We are ready to train a classifier from the extracted features. The classifiers used in the experiments are the LDA model [1] and SVM classifier [4].

Table 1. (a) Codebook size for each type of region features; (b) Codebook size for each type of interest point features;

Feature Type	Shape	Color	Texture	Location
Codebook Size	100	30	120	50

(a)

Feature Type	Harris	Uniform Grid	SIFT
Codebook Size	500	500	500

(b)

D. *Classifiers*

In order to evaluate the effectiveness of the LDA model for the scene classification, we employ the SVM classifier as the benchmark. Therefore, two classifiers are compared in our experiments, i.e., the LDA model and the SVM classifier.

D1. LDA Model

LDA Model: LDA model is one of the most successful topic discovery models used in the statistical text analysis literatures. It uses bag of words approach to automatically find topic for documents [1].

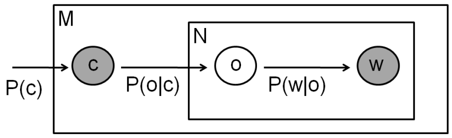


Figure 5. Modified LDA graph model in the scene classification. There are "*M*" images and "*N*" visual features in each image. "*c*" is the category of an image. Each feature "*w*" is assigned to an object "*o*".

In order to apply the LDA model in scene classification, the model has been modified as shown in Figure 5 [6, 11]. In the modified model, the node "*c*" stands for a scene category. Category "*c*" is a shaded node since it is an observable quantity. The node "*o*" stands for object in the image. Object is not observable. Therefore it is a hidden or latent variable in the model. Word "*w*" is interpreted as a visual feature in the image. Obviously, visual feature is an observable quantity. The outer plate and inner plate represent image and visual feature respectively. That means that there are "*M*" images and "*N*" features in each image.

Object "*o*" is first sampled from the category specific object distribution *P(o|c)*. Then visual feature "*w*" is sampled from the feature distribution given the sampled object *P(w|o)*. The relationship between the category and the feature can then be established using the following equation.

$$P(w|c) = \sum_{o=1}^{K} P(w|o) * P(o|c) \quad , \qquad (1)$$

where K is the total number of different objects. In our experiments, we set K to 30.

Training with LDA: As shown in equation (1), the conditional probabilities *P(w|o)* and *P(o|c)* have to be known in order to calculate *P(w|c)*, which is feature distribution given a category. *P(w|o)* and *P(o|c)* can be obtained during training phase through two concurrent matrices (feature-object concurrent matrix M_{wo} and object-category concurrent matrix M_{oc}) as shown in Figure 6. The conditional probability *P(o=k|c=m)* can be calculated using the following three steps. The first step is to find the number of feature tokens belonging to both object k and category m, which is also the element value of M_{oc} at k^{th} row and m^{th} column denoted by $M_{\text{oc}}(k,m)$. The second step is to find the total number of feature tokens belonging to the category m, which is the sum of all element values at m^{th} column of $M_{\text{oc},}$ denoted by $\sum_{o=1}^{K} M_{\text{oc}}(o,m)$. The third step is then dividing $M_{\text{oc}}(k,m)$ by the sum of all elements at m^{th} column of M_{oc}, as shown in equation (2) below. Similarly, *P(w=v|o=k)* can also be calculated as shown in equation (3).

$$P(o = k|c = m) = \frac{M_{\text{oc}}(k,m)}{\sum_{o=1}^{K} M_{\text{oc}}(o,m)} \qquad (2)$$

$$P(w = v|o = k) = \frac{M_{\text{wo}}(v,k)}{\sum_{w=1}^{V} M_{\text{wo}}(w,k)} \ , \qquad (3)$$

where V is the total number of vocabulary in the codebook.

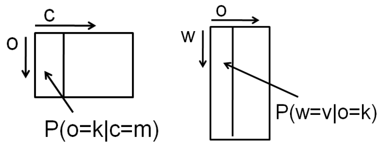


Figure 6. Object-Category concurrent matrix M_{oc} and feature-object concurrent matrix M_{wo}. Through the concurrent matrices, conditional probabilities *P(o=k|c=m)* and *P(w=v|o=k)* can be calculated.

In order to construct concurrent matrices M_{oc} and M_{wo} through training, we have to know which object each feature token belong to. However, the object is a hidden variable. That means we do not know the object assignment of each feature token before training starts. This problem can be solved by randomly assigning an object to each feature token. Therefore, the initial M_{oc} and M_{wo} can be constructed for each feature type. Following the approach in [11], M_{oc} and M_{wo} of each feature type can be updated iteratively using the collapsed Gibbs sampling inference [19] with the assumption of independence among different feature types.

Testing with LDA: Given an unknown test image, all visual features are extracted. For each feature type, *P(w|o)* can be calculated from the corresponding feature-object concurrent matrix M_{wo} as shown in equation (3). Since an interesting region or point is described by multiple types of features, we use the notation of $P(w_i|o)$ to indicate the conditional probability of visual feature for i^{th} feature type. With the assumption of independence among different feature types, the probability of the interesting region or point have the specific visual feature w_i for each feature type can be calculated using equation 4 below.

$$P(W|o) = \prod_{i=1}^{N_f} P(w_i|o) \quad , \qquad (4)$$

where *P(W|o)* is the joint conditional probability of visual features over all feature types for an interesting region or point. N_f is the number of feature types. Similar to equation (1), *P(W|c)* of each interesting region or point in the test image can be obtained by integrating out the objects "*o*" using equation (5).

$$P(W|c) = \sum_{o=1}^{K} P(W|o) * P(o|c) \quad , \qquad (5)$$

Assuming visual features of different interesting regions or points are independent, the probability of the test image belonging to each category *P(image|c)* can be obtained by equation (6). Then the category which has the maximum *P(image|c)* is selected as the classified category of the test image.

$$P(image|c=m) = \prod_{\substack{\text{W=all interesting} \\ \text{region or points} \\ \text{in image}}} P(W|c=m) \qquad (6)$$

D2. SVM Classifier

We employ the SVM classifier with the RBF kernel as our benchmark classifier [4]. The SVM is to find a set of hyperplanes which separate each pair classes of data with the maximum margin. That is to assign a visual scene category to an unknown image based on a feature representation. The bag of words [5] feature representation is chosen due to its simplicity and excellent performance in the scene classification. When fusing multiple types of features, simple concatenation of each feature type's bag of word histogram is used as the final feature representation of an image.

III. EXPERIMENTS

A. Experimental Setup

All experiments are performed on the UIUC Sport Scene database [11]. Images in the original database [12] are very noisy since they were downloaded from the internet. We clean up the database by removing unrelated images in each category. Sample images of each category are shown in Figure 7. The total number of categories is 8, which includes Badminton, Bocce, Croquet, Polo, Rock-Climbing, Rowing, Sailing and Snowboarding.

From each category, 400 training images and 100 testing images are randomly chosen. Therefore, the total number of training and testing images are 3200 and 800 respectively for all categories. Due to the randomness of object initialization during the training phase of the LDA model, we repeat experiment of the LDA model 5 times under each condition. Average classification performances are reported in this paper.

(a) Badminton
(b) Bocce
(c) Croquet
(d) Polo
(e) Rock-Climbing
(f) Rowing
(g) Sailing
(h) Snowboarding

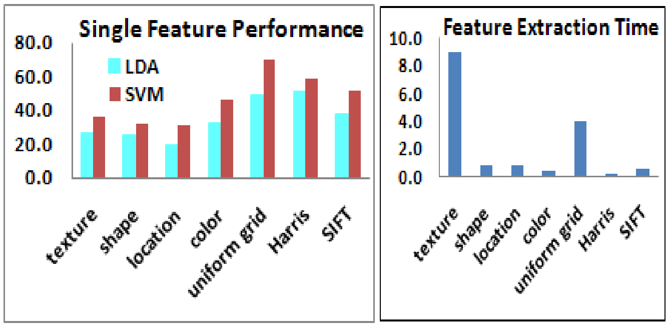
Figure 7: Sample images of the 8 scene categories from the UIUC Sport Scene Database [12].

B. Results of Single Type of Features

We first evaluate both LDA and SVM models for each single type of feature. The detailed comparisons between these two models are shown in Figure 8(a). SVM outperforms LDA model over every type of features by the average of 12%. Over the Uniform Grid features, SVM exceeds LDA model by more than 21%.

The significant performance degradation of the LDA model can be due to the fact that the LDA model is an unsupervised model, which automatically assigns an object to a visual feature. The similar conclusion is also reported in paper [10].

The best performance of the LDA model is around 51% using the Harris corner features. The accuracy is calculated using the ratio between the number of correctly classified images and the total number of images used in the testing. A sample confusion matrix is shown in Table 2(a). The best performance of the SVM classifier achieves 70% with the Uniform Grid feature. Table 2(b) shows the corresponding sample confusion matrix.



(a) (b)

Figure 8. (a) Comparing classification performance of LDA and SVM classifiers over single type of features. (b) Comparing feature extraction time in second for one image using different feature types. All feature extractions are performed over 300 images. Then the average time for one image is reported here. Note that all region feature extraction time include segmentation, which is 0.3 second each image.

Table 2. (a) Confusion matrix of the LDA model using the Harris corner features; (b) confusion matrix of SVM using the Uniform Grid features. Rows are the ground truth category while columns are the classified category.

True\Model								
Badminton	66	4	5	8	3	4	6	4
Bocce	7	38	13	8	10	9	7	8
Croquet	13	10	51	5	11	2	1	7
Polo	4	11	5	62	6	6	2	4
Rock-Climb	1	8	11	8	59	4	1	8
Rowing	6	5	4	2	8	61	8	6
Sailing	3	3	2	3	9	21	55	4
Snowboard	5	17	7	2	7	15	7	40

(a)

True\Model								
Badminton	84	2	4	1	5	1	2	1
Bocce	1	49	13	13	5	4	6	9
Croquet	4	2	79	4	8	2	1	0
Polo	0	7	4	79	3	3	2	2
Rock-Climb	3	3	5	2	77	0	2	8
Rowing	5	9	0	4	4	57	11	10
Sailing	1	3	1	4	0	12	73	6
Snowboard	3	7	0	1	10	7	8	64

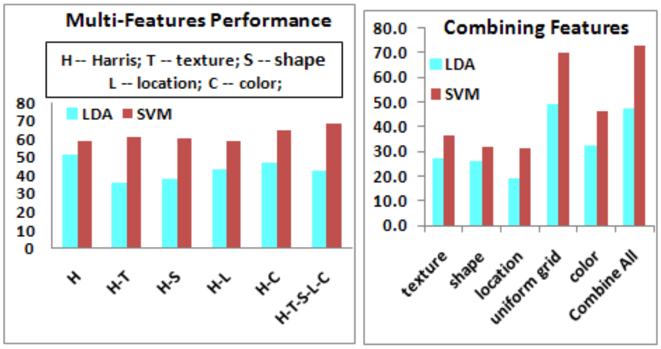
(b)

We further investigate the computation cost for extracting different types of features. In experiments, we extract approximate 100 Harris corners and SIFT interest points in each image. The number of Uniform Grid interest points is around 1500 for a typical image. Therefore, the computation cost of the Uniform Grid features is much higher, as compared to the Harris corners and the SIFT interest points, which is shown in Figure 8(b) as well. It takes average 3.9 seconds to extract the Uniform Grid features each image. However, it only takes average 0.2 seconds each image to extract the Harris corner features. Note that the feature extraction is performed over 300 images with typical resolution of 300 by 500. The average feature extraction time is reported here. All experiments are run on an Intel PC with CPU at 3.16GHZ using Matlab.

C. Results of Multiple Types of Features

Due to the complexity of the scene classification, multiple types of features with complementary information are usually necessary to achieve better performance. Our second task is to evaluate the effectiveness of the LDA model when fusing multiple types of features for scene classification.

Since the Harris corner features achieve the best performance with the LDA model by using single type of features, we evaluate the classification performance of the LDA model using various combinations of the Harris corner feature with other types of features, i.e., texture, shape, location, and color features. Figure 9(a) shows the performance of scene classification by combining different types of features for both LDA model and SVM classifier. Similar to the observations from the experiments by using single type of features, the LDA model has much lower accuracy rate than SVM over all different feature fusions. For example, SVM achieves 61% when combining the Harris corner features and the texture features, while the LDA model only achieves 36%.



(a) (b)

Figure 9. (a) Performance comparison of LDA and SVM models over fusion of different types of features. (b) Combine multiple types of features, i.e., texture, shape, location, color, and Uniform Grid features using the LDA model and the SVM model. The experimental setup of the LDA model follows closely to paper [11].

Table 3. Confusion matrices by fusing Harris corner, texture, shape, location, and color features. (a) LDA model; (b) SVM classifier. Rows are the ground truth category while columns are the classified category.

True\Model								
Badminton	68	2	3	11	4	6	4	2
Bocce	7	30	9	8	12	14	6	14
Croquet	18	10	33	15	23	0	0	1
Polo	5	13	20	39	3	6	4	10
Rock-Climb	6	5	21	1	48	4	5	10
Rowing	9	7	3	4	4	55	16	2
Sailing	7	3	0	7	3	25	50	5
Snowboard	4	21	8	12	6	9	16	24

(a)

True\Model								
Badminton	77	6	4	5	3	2	3	0
Bocce	8	53	14	5	4	7	2	7
Croquet	7	9	72	4	6	2	0	0
Polo	3	7	4	78	4	2	0	2
Rock-Climb	4	6	4	2	74	2	1	7
Rowing	5	7	2	4	8	58	10	6
Sailing	5	3	0	1	2	18	64	7
Snowboard	4	3	1	2	9	2	8	71

(b)

Combining the Harris corner features with other types of features, i.e., texture, shape, location, and color features, does not improve the performance in the LDA framework, as shown in Figure 9(a). Instead, the performance has been degraded over all different feature combinations, as compared to that of the Harris corner feature alone. When fusing the Harris corner feature with texture, shape, location, and color features, the performance of the LDA model is 42%, while it is 51% with the Harris corner features alone. The confusion matrix of the combined features is shown in Table 3(a). These results suggest that the LDA model is not effective to fusion of multiple types of features for scene classification, even though the feature types provide complementary information.

On the other hand, the performance of the SVM classifier has improved when fusing the Harris corner features with other feature types, as shown in Figure 9(a). The best performance, 68%, is achieved when combining the Harris corner feature with texture, shape, location and color feature together. It has improved more than 9%, as compared to that of using the Harris corner features alone. Table 3(b) displays the corresponding confusion matrix of the combined features. These results confirm that the Harris corner, texture, shape, location, and color features are complementary to each other for scene classification.

Li *et al.* [11] reported 54% of the overall classification performance by combining texture, shape, location, color and Uniform Grid features in the LDA framework. In addition, they also use the tag information besides the visual features. However, the authors did not address the performance gain by combining these features under the LDA framework, as compared to that of the individual features.

In order to further validate our observations that the LDA model is not effective to fuse multiple types of features for scene classification, we carefully repeat the experiments in paper [11] without the tag information and evaluate the performance of the individual features.

Under our experimental setup, the performance of the combined visual features, i.e., texture, shape, location, color, and Uniform Grid features, is 47%, which is 7% lower than the reported accuracy rate in paper [11]. We believe that the degraded performance is due to the lack of tag information of images.

After we confirm that our experimental results of the LDA model are consistent with paper [11], we investigate whether combining different types of features in the LDA framework achieves better performance as compared to that of the individual type of features.

As shown in Figure 9(b), the performance of the LDA model with the combined features is higher than that with the individual region features, i.e., texture, shape, location, and color feature. However, the multi-feature performance is almost 2% lower than that of the Uniform Grid features alone. These results confirm our previous conclusion, which states that the LDA model is not effective to fuse multiple types of features for scene classification.

Figure 9(b) also shows the corresponding performance of the SVM classifier for comparison. The multi-feature performance is improved about 3% over the Uniform Grid features with the SVM classifier. These results are consistent with our previous observations that the LDA model is inferior to the SVM model when fusing multiple feature types.

IV. Conclusion

We have experimentally evaluated the effectiveness of the LDA model over seven types of commonly used features in scene classification. LDA model obtains significantly lower accuracy rate, as compared to the SVM classifier. One possible reason is that the LDA model is originally an unsupervised model, which uses iteration algorithm to update object assignment for each feature.

The performance of the LDA model degrades as fusion of multiple types of features. In other words, our experiments demonstrate that the LDA model is not effective to fuse multiple types of features, even though these feature types provide complementary information of scene images.

Acknowledgments

This work was supported in part by NSF grant IIS-0957016 and ARO grant W911NF-09-1-0565. Shizhi Chen is funded by NOAA Cooperative Remote Sensing Science and Technology (CREST) Fellowship.

References

[1] D. Blei, A. Ng, and M. Jordan, “Latent dirichlet allocation”, Journal of Machine Learning Research, 3:993-1022, 2003.

[2] A. Bosch, A. Zisserman and X. Munoz, “Scene Classification via pLSA”, ECCV, 2006.

[3] L. Cao and L. Fei-Fei, “Spatially Coherent Latent Topic Model for Concurrent Segmentation and Classification of Objects and Scenes”, ICCV, 2007.

[4] C. Chang and C. Lin, LIBSVM : a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[5] G. Csurka, C. Dance, L. Fan, J. Willamowski, C. Bray, “Visual Categorization with Bag of Keypoints”, ECCV, 2004.

[6] L. Fei-Fei and P. Perona, “A Bayesian hierarchy model for learning natural scene categories”, CVPR, 2005.

[7] P. Felzenszwalb and D. Huttenlocher, “Efficient Graph-Based Image Segmentation”, IJCV, 2004.

[8] C. Harris and M. Stephens, “A Combined Corner and Edge Detector”, *Proc. Alvey Vision Conf.*, Univ. Manchester, pp. 147-151, 1988.

[9] T. Hofmann, “Unsupervised learning by probabilistic latent semantic analysis”, Machine Learning, 43:177-196, 2001.

[10] S. Lazebnik, C. Schmid, J. Ponce, “Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories”, CVPR, 2006.

[11] L. Li, R. Socher, and L. Fei-Fei, “Towards Total Scene Understanding: Classification, Annotation and Segmentation in an Automatic Framework”, CVPR, 2009.

[12] L. Li, “Towards Total Scene Understanding”, April 2009, http://vision.stanford.edu/projects/totalscene/index.html

[13] D. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”, IJCV, 2004.

[14] A. Oliva and A. Torralba, “Modeling the shape of the scene: A Holistic Representation of the Spatial Envelope”, IJCV, 2001.

[15] N. Rasiwasia and N. Vasconcelos, “Scene Classification with Low-dimensional Semantic Spaces and Weak Supervision”, CVPR, 2008.

[16] B. Rusell, A. Efros, J. Sivic, W. Freeman and A. Zisserman, “Using Multiple Segmentations to Discover Objects and their Extent in Image Collections”, CVPR, 2006.

[17] J. Sivic, B. Russell, A. Efros, A. Zisserman and W. Freeman, “Discovering objects and their location in images”, ICCV, 2005.

[18] M. Varma and A. Zisserman, “Classifying Images of Materials: Achieving Viewpoint and Illumination Independence”, ECCV, 2002.

[19] “Latent Dirichlet Allocation”, Wikepedia, May 20 2010, http://en.wikipedia.org/wiki/Latent_Dirichlet_allocation.