Journal of Naval Science and Engineering
2009, Vol. 5, No.2, pp. 1-19

SCENE CLASSIFICATION USING CASCADED PROBABILISTIC LATENT SEMANTIC ANALYSIS

Emrah ERGUL Lt.Jr. Grade , Asst. Prof. Nafiz ARICA, Cdr.
Turkish Naval Academy
Naval Science and Engineering Institute
Tuzla, Istanbul, Turkiye
{eergul, narica}@dho.edu.tr

Abstract

In this paper we propose a novel approach of image representation for weakly supervised scene classification that mainly combine two popular methods in the literature: Bag-of-Words (BoW) modeling and probabilistic Latent Semantic Analysis (pLSA) modeling. The new image representation scheme called Cascaded pLSA performs pLSA in a hierarchical sense after the BoW representation based on SIFT features is extracted. We associate location information with the conventional BoW/pLSA algorithm by subdividing each image into sub-regions iteratively at different resolution levels and implementing a pLSA model for each sub-region individually. Finally, an image is represented by concatenated topic distributions of each sub-region. The performance of our method is compared with the most successful methods in the literature using the same dataset. In the experiments, it has been seen that the proposed method outperforms the others in that particular dataset.

BASAMAKLI OLASILIKSAL GİZLİ SEMANTİK ANALİZ İLE SAHNE SINIFLANDIRILMASI

Özetçe

Bu makalede imgelerin analiz edilmesi ve neticesinde içerik bilgilerine göre imgelerin taşıdıkları anlamlara uygun olarak sınıflandırılmaları hedeflenmiştir. Bu kapsamda zayıf denetimle sahne

sınıflandırması sağlayan ve literatürde son zamanlarda sıkça başvurulan Görsel Kelimeler Kümesi ve Olasılıksal Gizli Anlam Analizi yöntemlerinin birleştirildiği yeni bir yaklaşım önerilmektedir. Betimlemede Olasılıksal Gizli Anlam Analizi algoritmasının hiyerarşik bir yapıda imgeye uygulanmaktadır. SIFT özniteliklerine dayalı Görsel Kelimeler Kümesinin elde edilmesini müteakip, Olasılıksal Gizli Anlam Analizi modellemesinin piramit basamaklandırma şeklinde tüm alt bölgelere ayrı ayrı uygulanır. Tüm sevilerden elde edilen gizli tema dağılımı birleştirilerek imge betimlemesi gerçekleştirilir. Önerilen yöntemin performansı, aynı veri seti kullanılarak eşit şartlarda literatürde mevcut en başarılı diğer yöntemler ile karşılaştırılmış; ve önerilen yöntemin diğerlerinden daha iyi neticeler elde ettiği görülmüştür.

Keywords: *Scene Classification, Spatial Pyramid, probabilistic Latent Semantic Analysis, Bag-of-Visual words.*

Anahtar Kelimeler: *Sahne Sınıflandırılması, Uzaysal Piramit, Olasılıksal Gizli Anlam Analizi,Görsel Kelimeler Kümesi.*

1. INTRODUCTION

In the last decade, digital imagery has grown at an incredible speed in many areas, resulting in an explosion in the number of image archives and quality of images to be managed automatically or at least with an optimum supervision. In particular, with the wide usage of high resolution digital cameras, camcorders, mobile phones with built-in cameras; and storage of personal computers, cheap flash memories reaching to huge capacities, people nowadays can easily produce thousands of personal images, share their products with social networking and photo sharing web sites in the digital world WWW without any limitation in capacity, speed and connectivity. Although people are increasingly attaching annotations to their images for semantic filtering/searching, the vast majority of the images on the internet are barely documented, making it very difficult for people to find one of interest. In order to handle overload and exploit the massive image information, we need to develop techniques to document and search images. Describing images by its semantic contents will help us organize, access and classify huge amass of data in a reasonable way. A computer system that could automatically classify objects/scenes from images would be of great importance since online resources of huge amount of digital information have covered most of our daily life. Applications are in large

scale: Surveillance, environment definition, robots with visual interactions, and smart instruments like camcorders or photographers that could sense the environment and set up automatically to capture the best snapshots.

Scene classification problem is a very impressive and multi-objective task for computer vision, and also a popular research area nowadays. It comprises of many sub-problems including segmentation of relevant components to identify objects over an image, clustering data which is extracted from dataset images into semantic exemplars to reduce storage and computing consumption, training the classification system to generate representative models in a data-driven fashion, matching between observed and unobserved images in statistical/probabilistic ways. The release of many challenging datasets with multiple classes supported by recently published papers [1, 2, 3, 4, 5, 6] has proved itself how it is hard and interesting research sub-area in computer vision. As datasets sort in large diversity and bear ambiguity with multiple classes that contain many objects; distinctive image representation, efficient training and testing algorithms are needed to cope with such complexity.

Our aim in this paper is to classify natural and man-made images among a set of challenging dataset into semantically meaningful categories. It addresses analyzing an image using computer algorithms and assigning it a category label (i.e. suburb, forest, street). While scene classification systems in the literature vary considerably, we can place them generally into 3 categories according to their image representation schemes: Low-level, Semantic and Bag-of-Words (BoW). After representation scheme is determined and conducted, a labeled set of images is used to train the system to discriminate between image classes and after training is complete, a testing image is classified by the system and compared to ground truth classes for the testing set.

Among representation schemes, BoW and Semantic modeling, also called content-based method, are the most promising methods in the previous works. Generally in the Semantic modeling, training images are partitioned into local patches and each piece is hand-labeled with one of

several classes (i.e. sky, water, and wall). Then a classifier like SVMs, Neural Networks or Bayesian Classifiers is used to model semantic classes. Eventually, low-level feature descriptors are assigned to "intermediate semantic classes" (i.e. textons or materials) and a histogram of textons is created for the image which stores a count of occurrences for each of these semantic classes in the image [7, 8, 9]. Although they use semantic concepts for image representation as human vision system does, they are mostly supervised and extra work (i.e. manually annotation for each patch in the training step) need to be established. On the other hand, low-level feature descriptors are assigned to "visual words" by establishing a clustering algorithm like k-means on the feature descriptors which are extracted from a set of training images; then each image is represented by a frequency vector of visual words in the BoW modeling. Although it needs no supervision for image representation, unlike Semantic modeling, neither is there any semantic concept in this method as it uses only the counts of visual words which account for cluster centers calculated in Euclidean metric [10, 1]. Also note that location information is ignored if only we represent an image by a histogram of occurrences in both cases.

In this paper, we propose to combine these models to achieve better results. After achieving BoW representation for an image as described shortly above, we utilize intermediate semantic representation by using probabilistic Latent Semantic Analysis (pLSA) algorithm, introduced firstly by Hofmann in text analyzing literature [11]. To add location information to the classification system; we divide the image into multiple parts, using a pyramidal division scheme as proposed by Lazebnik et. al [2], then run pLSA for each sub-region to generate a new probabilistic model which refers to mixture of topics (i.e. objects) in each relevant sub-region. After all, we achieve an intermediate semantic representation for each image which is more robust to failures in scene classification problems due to geometric and photometric changes with location information in an unsupervised manner. We call our proposed method as "Scene Classification using Pyramid of Latent Topics by Cascaded pLSA".

2. SCENE CLASSIFICATION USING PYRAMID OF LATENT TOPICS

In this section, we give a detailed description of our algorithm developed for scene classification. Shortly, our task is to classify a query image into one of given labels of scenes (e.g. suburb, kitchen, bedroom, inside city, etc.). To achieve such a challenging goal, we need to establish a chain of procedures; namely feature extraction, image representation, implementation of training and classification.

2.1 Feature Extraction

The first part of feature extraction process is the detection of interest regions over an image. The second part is to obtain a representation for those regions that allows us to find correct matches between specific objects/scenes. Although we mostly deal with semantic description in scene classification problems, the basis of semantic modeling refers to low-level descriptors of interest regions based on appearance information of an image. In scene classification tasks, most of the recent methods [2, 13, 14] use SIFT descriptor for local appearance representation which has been proposed by in [12], because of its robustness to those deformations and transformations. Besides, It has been concluded experimentally that local SIFT features achieve better results for classification issues [15]. In this paper, we extract the SIFT descriptors at regular grids all over the image for scene classification problem. However, comparative results show that utilization of dense keypoints over whole image surface work better than sparsely detected keypoints for scene classification [1, 13]. Since a dense image description is compulsory to capture uniform semantic regions such as sea, sky, forest, we have used dense SIFT representation for each image of 16 by 16 pixel patches , meaning scale 8 pixels, computed over a regular rectangular grid with spacing 8 pixels. Lazebnik et. al used the same technique to in “Spatial Pyramid” method and achieved very promising results in scene classification issue [2]. The illustration of dense SIFT descriptors over an image is depicted at Figure-1.

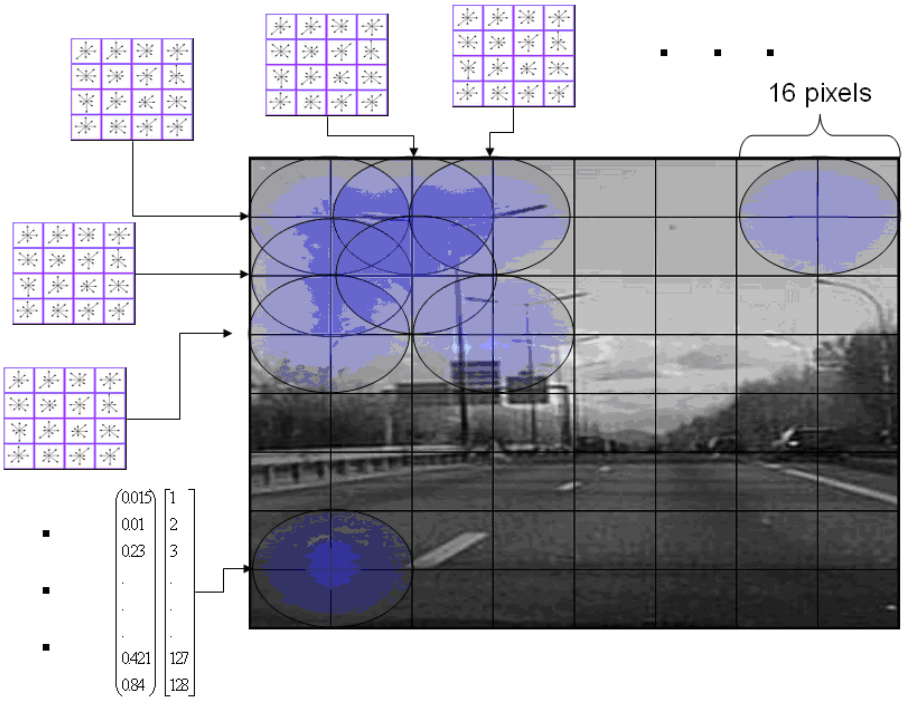

Figure 1: Illustration of dense SIFT descriptors over an image

2.2 Visual Vocabulary Generation

Our scene classification algorithm is based largely on the BoW / pLSA implementation. In this implementation, we analyze and compare images by tracking the number of occurrences of every visual word. At this point, we encounter a new concept: Visual words and word-image co-occurrence table. For example, for a set of images $I = \{i_1, i_2, i_3, i_4, ..., i_n\}$, a table of counts can be created, where each row represents a word from the visual vocabulary (i.e. cluster of visual words) $V = \{w_1, w_2, w_3, w_4, ..., w_m\}$ and each column represents an image. This two-mode table refers to BoW representation scheme for the images and it is compulsory for initializing pLSA modeling, to which co-occurrence table is the only input parameter.

After features are described in the dataset using dense SIFT algorithm as described at section 2.1, we need to group them into visual words to create aforementioned co-occurrence table which indicates a BoW representation for each image, column-wisely. In order to create the visual words of co-occurrence table, this large set of descriptors which have been extracted from the dataset densely must be reduced to a smaller set of repeated terms by clustering the SIFT descriptors. One of the simplest efficient methods of clustering in the literature is k-means, which is widely used since it is relatively quick and allows the user to choose the desired number of cluster centers as visual words. Thereafter, the entire set of images must be processed to match each descriptor with its nearest visual word. Each descriptor is compared to each of the visual words, and is assigned the label of the closest word.

At this point, each image has been converted into a list of visual words and their frequencies. To compare images, however, we need an occurrence table of counts of $N=(n(w_i,d_j))_{ij}$ where $n(w_i,d_j)$ refers to the number of times the visual word w_i occurred in image d_j. This two-mode matrix will be used in pLSA modeling as an input, which will be explained in details below.

2.3 Semantic Image Representation Based on pLSA

Probabilistic Latent Semantic Analysis (pLSA) is a statistical method of factor analysis for binary or two-mode count data to generate latent class models. It was first proposed by Hofmann in text literature based on Latent Semantic Analysis (LSA) which is derived from Frobenius norm (i.e. L_2-matrix) Singular Value Decomposition (SVD) of co-occurrence tables [11, 16]. In text/natural language analyzing field, text documents are often analyzed by counting the number of occurrences of every word. For a set of documents, a table of counts can be produced where each row represents count of a word from a specific vocabulary $V=\{w_1,w_2,w_3,w_4,...,w_m\}$, i.e. cluster of keywords, and each column represents a document $D=\{d_1,d_2,d_3,d_4,...,d_n\}$. This co-occurrence table of

counts, also called term-document matrix, is denoted as MxN matrix of $N=(n(w_i,d_j))_{ij}$ where $n(w_i,d_j)$ refers to the number of times the word w_i occurred in document d_j.

The basic idea in pLSA is to map high dimensional count vectors of documents to a lower dimensional representation, so-called "Latent Semantic Space" [11]. Representing semantic relations between words and documents by generating a new variable space, provides information beyond lexical word co-occurrences. The main difference between pLSA and LSA comes in presence when conducting SVD process to generate unobserved topic variables $Z=\{z_1,z_2,z_3,...,z_k\}$ from observed word (V) and document (D) variables.

Let us define probabilistic notations used in the algorithm: $P(d_i)$ denotes the probability that any randomly selected word belongs to the document d_i, $P(w_j \mid z_k)$ denotes the conditional probability of word w_j on unobserved topic variable z_k, and $P(z_k \mid d_i)$ indicates the probability of a random word from document d_i belongs to the unobserved topic variable z_k. Note that while word V and document D variables are observed from N co-occurrence data, topic variable Z is an unobserved "hidden" variable created in the pLSA process.

We can formulate the probability of observation pair as $P(w,d)=P(d,w)=P(d)P(w \mid d)$ in Naïve Bayesian approach, thus we get the conditional probability distribution of words over documents as:

$$P(w \mid d)=\sum_{k=1}^{K} P(w \mid z_k)P(z_k \mid d) \qquad (1)$$

This matrix decomposition seems SVD like in LSA, but topic vectors of $P(w \mid z)$ and $P(z \mid d)$ are normalized to unit to achieve a probabilistic distribution without any negative entry in pLSA model,

unlikely. In order to compare documents in latent space, however, we must work this process in reverse; starting with a term-document matrix, we want to generate a table which represents $P(z \mid d)$ in order to describe each document as a mixture of topics. The standard way to do this is to use an Expectation Maximization (EM) process [17], which alternates an expectation step where posterior probabilities are calculated for the latent variables *Z* based on current estimates of parameters and a maximization step where parameters are updated based on posterior probabilities until a model is fitted (i.e. convergence conditions are met due to sequential likelihood measurements). The Expectation (E) step uses the following equation:

$$P(z_k \mid d_i, w_j) = \frac{P(w_j \mid z_k) P(z_k \mid d_i)}{\sum_{k=1}^{K} P(w_j \mid z_k) P(z_k \mid d_i)} \qquad (2)$$

And the Maximization (M) step is formulated as:

$$P(w_j \mid z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j) P(z_k \mid d_i, w_j)}{\sum_{j=1}^{M} \sum_{i=1}^{N} n(d_i, w_j) P(z_k \mid d_i, w_j)}$$

$$P(z_k \mid d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j) P(z_k \mid d_i, w_j)}{n(d_i)} \qquad (3)$$

Where $n(d_i, w_j)$ is the number of word w_j in document d_i, and $n(d_i)$ is the number of documents which has word w_j.The E-step and M-step equations are implemented sequentially until a termination condition is satisfied. The iterative algorithm is generally stopped by giving an iteration

limit for E/M alternation steps or giving a threshold value which indicates the difference between two sequential likelihood values.

To summarize, posterior probabilities $P(w|z)$ and $P(z|d)$ are computed in E-step, and updated in M-step while maximizing the log-likelihood function which is equivalent to minimizing the Kullback-Leibler divergence of probability distributions between observed data and fitted model data [16].

2.4 Spatial Pyramid of Latent Topics by Cascaded pLSA

The key idea in our method is to fit pLSA models into sub-regions individually at different resolution levels to achieve a new representation for each image. The categorization of an unobserved image is performed in this semantic representation by using classification algorithms such as KNN and SVM.

In the training stage, the first probabilistic distributions we should find in pLSA are topic conditional visual word distributions $P(w|z)$ which will be used in other resolution levels of training images and in overall testing stage. To make this happen, a pLSA model is fit to the entire set of training images at resolution level (L) 0 (i.e. base level) where the images are intact. One point to mention further is that model fitting via pLSA is known as an unsupervised approach because we only input co-occurrence matrix (i.e. count data of visual words in each document), no other like category labels of images. The products of model fitting are $P(w|z)$ and $P(z|d_{train\_Level\,0})$; hence we will use document specific topic distributions $P(z|d)$ vector for image representation later .

At $L=1$, we split the training images into four sub-regions, generate a new co-occurrence matrix for each sub-region as an input to pLSA by using the same visual vocabulary that has been created in k-means clustering. $P(z|d)$ coefficients are computed individually by initiating the fold-in heuristic proposed by Hofmann for information retrieval [11].

Specifically, each sub-region is projected onto the triple axis space VIZ (Visual word-Image-Topic) by the $P(w|z)$ learnt when *L=0*. This is achieved by updating only the topic distributions vector $P(z|d_{train\_level=1})$ in each M-step while the learnt $P(w|z)$ kept fixed at EM iterations of pLSA until Kullback-Leibler divergence between the observed distributions and calculated $P(w|d)=\sum_{k=1}^{K}P(w|z_k)P(z_k|d)$ is minimized.

We follow the same procedure in finer resolution levels, except we divide training images into different number of sub-regions (i.e. powers of 2) at each level. For instance, we get 16 sub-regions at *L=2*, 64 at *L=3*, so on. We will use *L=3* as the maximum resolution level in our experiments.

After calculating all $P(z|d)$ coefficients of each sub-region at each resolution level, we concatenate them all with a proper weight factor which differs at each level to form a new representation for images, respectively. The weight at level ℓ is set to $\frac{1}{2^{L-\ell}}$, where *L* is the number of levels. This formulation is inversely proportional to the sub-region width at that level, means a finer resolution level is weighted more highly than a coarser level. It can be explained in that since the coarser level already includes all visual words found at the finer level, the coefficients $P(z|d)$ at the coarser level are more weighted and need to be balanced accordingly. When concatenate all $P(z|d)$ coefficients from individual sub-regions, one can obtain a new feature vector with dimensionality $D = Z\frac{1}{3}(4^{L+1}-1)$, where *Z* is the number of topics used in pLSA. For example, if we have 25 topics in pLSA modeling we achieve 25 dimensional feature vector for an image (i.e. whole image) at *L=0*, 125 dimensional at *L=1*, so on. An illustration of scene classification based on Cascaded pLSA method is illustrated at Figure-2.

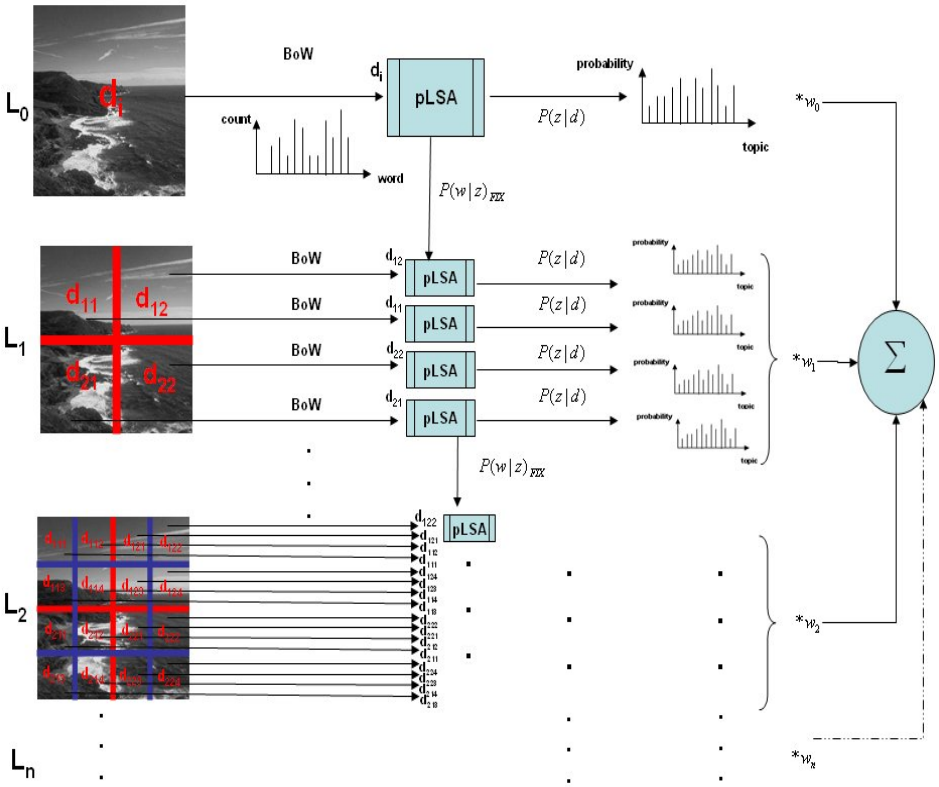

Figure 2: Illustration of Cascaded pLSA based method. BoW = Bag-of-Words histogram, P(z|d) = image specific probability distribution over latent variable space, P(w|z) = probability of a specific word conditioned on a latent variable. Note that $\sum$ indicates concatenation.

So far, we have obtained a cumulative feature vector of topic mixture coefficients for each image that are derived from sub-regions individually, using pLSA. The last part of training process is to train a multi-class classifier which takes $P(z \mid d_{train\_concatenated})$ vector and class label of each training image. The key idea in this part is to determine the observed discrimination between scene classes for further usage at testing stage. We

use KNN and SVM as a classifier for comparison. Shortly speaking, KNN gives an unobserved image the class label which is most represented within K nearest neighbors of it by calculating its Euclidean distance to training images. SVM, as a binary classifier, evaluates an image and assign it one of two classes by constructing a hyper plane as a separator region, using decision function $g(x)=\sum_i \alpha_i y_i K(x_i - x) - b$ where $K(x_i - x)$ is the response of a kernel function, x_i, x are training and test samples respectively, y_i is the class label of x_i, α_i is the learnt weight and finally b is the learnt threshold parameters. In our experiments, we use Rakotomamonjy's SVM toolbox, using Radial Basis Function (RBF) kernel with Histogram Intersection (HI) distance [18].

Testing procedure is very similar with the training process. The only difference between testing and training stages in pLSA model fitting is that $P(z \mid d_{test})$ coefficients are calculated by keeping $P(w \mid z)$ which has been already computed at *L=0* in pLSA model fitting for training images fixed at all resolution levels. The concatenated $P(z \mid d_{test\_concatenated})$ coefficients are then used to classify the test images by using a discriminative classifier that is described in the training process.

3. PERFORMANCE EVALUATION

In this section we evaluate our classification methods on a dataset which contains 13 natural scene images and compare our results with the most successful methods the scene classification literature. The dataset that we have used in the scene classification experiments was introduced by Fei-Fei et. al in the semi-supervised application of Latent Dirichlet Analysis (LDA) [1]. It contains 13 scene categories which consist of totally 3859 images in grayscale. The distribution of images per scene category is : 216 Bedroom, 241 Suburb, 210 Kitchen, 289 Livingroom, 360 Coast, 328 Forest, 260 Highway, 308 Inside city, 374 Mountain, 410 Open country, 292 Street, 356 Tall building and 215 Office. Most of the scenes display

large intra-class variability, meaning that object contents within a scene category are very different. Also note that indoor scenes (i.e. kitchen, bedroom, office and livingroom) have very similar structure, indicating a low inter-class variability. These issues make the scene classification problem hard when working with this dataset. The size of each image is varying both in a category and between categories, with an average of 250 x 300 pixels. We split whole dataset into two separate sets of images, training and testing. As speaking of training images, we select randomly 100 images per scene category which makes totally 1300 images for training set. The rest of whole dataset is used as a testing set with varying number of images from each category.

In feature extraction stage, we implement the dense SIFT with 8 pixels spacing in the Cartesian grid. Instead of generating scale space with DOG images, scale invariance is obtained by generating the SIFT descriptors using a circle of radii 8 pixels which indicates 16 pixel-width patches overlapping each other half size over the grids spacing 8 pixels. The visual vocabulary generation uses k-means algorithm to cluster SIFT descriptors which have been previously computed densely for each image. We select randomly 40 images from each scene category, totally 520 images, and input SIFT descriptors of these randomly selected images into k-means clustering algorithm, about 500000 descriptors at once. The vocabulary size is chosen as 400 in the experiments. We create a BoW representation for each image by counting the number of visual words individually and place them into a matrix $N=(n(w_i,d_j))_{ij}$ where $n(w_i,d_j)$ refers to the number of times the visual word w_i occurred in the image d_j. This two-mode matrix will be used in pLSA modeling as an input. Note that since we know the locations of SIFT descriptors over an image, we also know where the visual words in it which will be used in pLSA models of each sub-region.

In the pLSA modeling, we have experimented varying number of topics (i.e. 25,50,75 and 100) while in the classification process, we use discriminative K-nearest neighbors and Support Vector Machine classifiers

with varying parameters. The classification results of our method with SVM are displayed at Table-1; and with KNN is at Table-2. The tables show the performance rates achieved using just the highest level of the pyramid as the "single-level" columns, using multiple levels as the "pyramid" columns. Lazebnik et. al also use this format in their experiments [2].

We notice that the performance increases as we go from single level (L) 0 to finer resolution levels *L=1,2* and *3*. But it drops a little (between 1 and 2 percent) as we go from *L=2* to *L=3*. This is due to the impact of too finely subdivision at *L=3*. Although the spatial information comes in appearance and improves the performance when we divide an image into sub-regions, *L=3* is subdivided too finely and only a few number of topics exist in sub-regions at *L=3*. We notice the improvement in multi-level representations. In Table-1, the mean classification accuracy is 73.75 at *L=0* while at the highest pyramid level (i.e. *L=3*) it increases drastically to 80.29. Although it seems that single finer resolution levels support most of the improvement to the system, using all levels together make the system more robust. When we compare KNN vs. SVM, the performance of SVM is much better than that of KNN.

As speaking of the effect of number of topics used in pLSA modeling, increasing the number of topics from *T=25* to *T=100* results in generally a small performance increase (upmost 1 percent), also there are some fluctuations. We can conclude that number of topics strongly depends on the number of categories (i.e. 13 in here) in the dataset while number of visual words depends on the size of the feature vectors (i.e. 128 in here). So increase in the number of topics does not improve the performance dramatically as the visual words in the system has a linear distribution over topics, meaning that discriminative power of topic mixture vectors $P(z \mid d)$ used in the classifiers stays almost same, even displays fluctuations in some cases.

We compare the performance of our method to semi-supervised LDA of Fei-Fei et. al, [1] weakly supervised Spatial Pyramid of Lazebnik et. al [2], and Spatial Pyramid pLSA of Bosch et. al [13] using the same

dataset and the same number of training and testing images. We use SVM classifier, 400 visual words and 100 topics in comparison. pLSA indicates computation only at *L=0* (i.e. handling whole image without dividing into sub-regions) while BoW refers to Spatial Pyramid of Lazebnik et. al at *L=0.* We have implemented their algorithms (except SVM toolbox) as described in their papers except LDA of Fei-Fei et. al which we have used their maximum accuracy result noted in [1]. The comparison results are displayed at Table-3. We conclude that our method outperforms the other methods slightly.

When we focus on confusion matrix of classification, the overall performance is 80.23 percent. The best classified scenes are Suburb and Forest with a performance of 96.18 percent and 94.95 percent, respectively. The most difficult scenes are open country, Bedroom, Kitchen and Living room. There is confusion between open country and coast scenes, also between open country and mountain scenes. Indoor scenes Bedroom, Kitchen and Living room are confused each other as they have low intra-class variability among them. The confusion mainly caused by the similar structures of shape and appearance, besides potential ambiguities on the subjective manual annotations.

	T = 25		**T = 50**		**T = 75**		**T = 100**	
Level	**Single Level**	**Pyramid**	**Single Level**	**Pyramid**	**Single Level**	**Pyramid**	**Single Level**	**Pyramid**
0 (1x1)	73.12	---	72.73	---	75.09	---	74.06	---
1 (2x2)	76.33	76.70	77.07	77.27	77.11	77.44	78.22	79.34
2 (4x4)	79.66	79.95	79.33	80.07	79.74	80.20	79.62	80.77
3 (8x8)	78.59	79.87	78.22	79.95	77.93	80.44	77.93	80.90

Table 1: Classification results of Cascaded pLSA based method, using SVM. T indicates the number of topics used in pLSA.

	T = 25		**T = 50**		**T = 75**		**T = 100**	
Level	**Single Level**	**Pyramid**	**Single Level**	**Pyramid**	**Single Level**	**Pyramid**	**Single Level**	**Pyramid**
0 (1x1)	66.73	---	67.22	---	65.95	---	67.02	---
1 (2x2)	70.60	70.68	71.01	70.60	70.85	70.81	70.52	71.14
2 (4x4)	71.75	72.62	70.89	72.70	69.78	71.30	68.67	71.75
3 (8x8)	67.27	72.16	67.80	72.00	64.84	70.93	63.56	70.76

Table 2: Classification results of Cascaded pLSA based method, using KNN. T indicates the number of topics used in pLSA.

	BoW	**Bayesian Hirearchical Model (LDA) (Fei-Fei 2005A)**	**pLSA (*L= 0*)**	**Spatial Pyramid (Lazebnik 2006A)**	**SP pLSA (Bosch, 2008)**	**Cascaded pLSA**
Accuracy	72.66	65.2	74.06	77.76	79.17	**80.90**

Table 3: Compact comparison of our algorithms with other methods using the same experimental setup

4. CONCLUSION

In this paper, we focus on scene classification problem with a new method using BoW/pLSA modeling. A new image representation scheme based on Cascaded pLSA is proposed. After dense SIFT feature extraction is executed in a set of images, SIFT descriptors are clustered into visual words to achieve a BoW representation for each image as an input to pLSA modeling. We associate location information with the conventional BoW/pLSA algorithm where the spatial information is actually lost. This is achieved by subdividing each image into sub-regions iteratively at different grid levels and implementing a pLSA model for each sub-region individually. Hence each sub-region produces its own mixture of topics (i.e. $P(z \mid d)$) while staying coherent to the whole image where they belong to; since word-topic distributions (i.e. $P(w \mid z)$) of the whole image is kept

fixed. Eventually, we concatenate these sub-region specific topic distributions with a weighting scheme to obtain a new semantic image representation. One of the important contributions of pLSA modeling is reducing the dimension of representative feature vector from higher number of visual words to lower number of semantic topics, while improving classification performance considerably. We benefit this reduction when we represent an image as a concatenated mixture of topics, rather as a concatenated BoW histograms as described in [2].

We learn topics and their distributions in training images by a completely unsupervised fashion, unlike those of [1, 5]. It is mainly caused by the nature of pLSA modeling where BoW histograms are the only input to the system. We test the system in supervised classifiers (i.e.KNN and SVM) where category labels of training images are known. The performance of our method is compared with the most successful methods (i.e. Bayesian Hierarchical model (LDA), BoW with/without spatial information, pLSA without spatial information) in the literature using the same dataset and the same number of training/testing images. Our method outperforms others with a rounded percentage between 3 and 15.

REFERENCES

[1] Fei-Fei L. and Perona P. "*A Bayesian Hierarchical Model for Learning Natural Scene Categories,"* Proc. IEEE CS Conf. Computer Vision and Pattern Recognition, pp. 524-531, 2005.

[2] Lazebnik S., Schmid C. and Ponce J. "*Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories,"* Proc. IEEE CS Conf. Computer Vision and Pattern Recognition , vol. 2, pp. 2169-2178, 2006.

[3] Sivic J., Russell B.C., Efros A., Zisserman A. and Freeman W. "*Discovering Objects and Their Location in Images," IEEE* ICCV, vol. 1, pp. 370-377, 2005.

[4] Oliva A. and Torralba A. "*Modeling the Shape of the Scene: A Holistic Reprasentation of the Spatial Envelope,"* Int'l J. Computer Vision, vol. 42, no. 3, pp. 145-175, 2001.

[5] Vogel J. and Schiele B. "*Semantic Modeling of Natural Scenes for Content-Based Image Retrieval,"* Int'l J. Computer Vision, vol. 72, no. 2, pp. 133-157, 2007.

[6] Deng J., Dong W., Socher R., Li L., Li K. and Fei-Fei L. "*ImageNet: A Large-scale Hierarchical Image Database,"* CVPR, http://www.image-net.org, 2009.

[7] Vogel J. and Schiele B. "*Natural Scene Retrieval Based on a Semantic Modeling Step,*" Int'l Conf. Image and Video Retrieval, vol. 3155, 207-215, 2004.
[8] Luo J., Singhal, A. and Zhu W. "*Natural Object Detection in Outdoor Scenes Based on Probabilistic Spatial Context Models,*" Proc. IEEE Int'l. Conf. on Multimedia and Expo, 2003.
[9] Boutell M., Choudhury A., Luo J. and Brown M.C. "*Using Semantic Features for Scene Classification: How Good do They Need to Be?,*" IEEE Int'l Conf. Multimedia and Expo, pp. 785-788, 2006.
[10] Quelhas P., Monay F., Odobez J.M., Perez, D. and Tuytelaars, T. "*A Thousand Words in a Scene,*" IEEE Trans. on pattern Analysis and Machine Intelligence, vol. 29, no. 9, 2007.
[11] Hofmann T. "*Probabilistic Latent Semantic Indexing,"* Proc. SIGIR Conf. Research and Development in Information Retrieval, 1998.
[12] Lowe D. "*Distinctive Image Features from Scale Invariant Keypoints,*" Int'l J. Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.
[13] Bosch A., Zisserman A. and Munoz X. "*Scene Classification Using a Hybrid Generative/Discriminative Approach,"* IEEE Transections on Pattern Analysis and Machine Intelligence, vol. 30, no. 4, 2008.
[14] Jiang J., Ngo C.W. and Yang J. "*Towards Optimal Bag-of-Features for Object Categorization and Semantic Video Retrieval,*" ACM Int'l Conf. Image and Video Retrieval (CIVR), 2007.
[15] Mikolajczyk K. and Schmid C. "*A Performance Evaluation of Local Descriptors,*" IEEE Pattern Analysis and Machine Intelligence, vol. 27(10), pp. 1615-1630, 2005.
[16] Hofmann T. "*Unsupervised Learning by Probabilistic Latent Semantic Analysis,"* Machine Learning, vol. 41, no. 2, pp. 177-196, 2001.
[17] Dempster A.P., Laird N.M. and Rubin D.B. "*Maximum Likelihood from Incomplete data via the EM Algorithm,*" J. Royal Statist. Soc. B., vol. 39, pp. 1-38, 1977.
[18] Rakotomamonjy A. "*SVM and Kernel Methods Matlab Toolbox,*" http://asi.insa-rouen.fr/enseignants/~arakotom/toolbox/index.html, 2008.