IMAGE SCENE CATEGORIZATION USING MULTI-BAG-OF-FEATURES

WEIFENG ZHANG [1], ZENGCHANG QIN [1,2] *, and TAO WAN [2]
[1] Intelligent Computing and Machine Learning Lab (ICMLL), School of ASEE, Beihang University, China
[2] Robotics Institute, Carnegie Mellon University, USA
E-mails: zwf@asee.buaa.edu.cn, {zcqin, taowan}@andrew.cmu.edu

Abstract

Image scene classification, the classification of images into semantic categories, e.g. city, urban, sea, etc, has recently become a vigorous research focus in computer vision for its broad application prospect. In this paper, we propose a novel approach to understand image semantic scene based on multi-bag-of-features. We aim to design an efficient but simple scene classification algorithm via fusing multiple low-level image features. Experimental results demonstrate that the proposed approach offers an effective way to classify the complex image scenes by using a multi-bag-of-features model.

Keywords:

image scene classification; multi-bag-of-features; SVM classifier.

1 Introduction

In recent years, significant progress has been achieved in the field of image scene classification. However, the problem of image understanding in unconstrained environments is still a open challenge [13, 16]. Automated scene classification aims at labeling an image among a set of semantic categories, e.g. coast, sunset, and street, without human interaction. This is useful and important to provide contextual information to guide other tasks such as object recognition [15], content-based image retrieval [6], digital photofinishing [14], automatic image orientation [11], etc. The purpose of scene classification is different as the image understanding problem. The latter intends to recognize each object containing in the image, while the former depicts a scene concept that the image belongs to without even having a full knowledge of every object.

In the last few years, many different approaches concerning scene classification have been proposed. These approaches can be briefly classified into two categories: low-level feature based model and semantic model. The problem of scene modelling using low-level features has been intensively studied in image and video retrieval applications [12]. For example. Chang et al. [1] derived a image scene classification method using image color and texture features in which the images are grouped into 15 types of scene (e.g. architecture, bears, and flowers, etc). In order to narrow the gap between low-level features and high-level semantic concepts, the modelling of scene by a semantic intermediate representation has been investigated to closely match the scene model with the human perception (e.g. a garden scene mainly contains flowers and house). Fei-Fei et al. [3] proposed an image scene model based on the high-level image concepts. In their work, scale-invariant feature transform (SIFT) based features [5] and gray-level descriptors on a regular grid were used.

In this paper, we study the problem of how to understand and describe an image semantic scene by making use of low-level features to represent high-level semantic meanings. Therefore, we develop a scene classification algorithm based on the multi-bag-of-features model. One goal of this work is to examine the advantages and disadvantages of classifying image scenes by using low-level image features. We believe that the low-level strategies can yield a comparative classification outcome meanwhile maintain a low computation cost when the number of scene categories remains relatively low.

The rest of this paper is structured as follows: In Section II we give a full description about the proposed approach. The experiments on the benchmark problems and discussions are present in Section III. Section IV summarizes the work.

2 Methodology

In this section, we will describe the image scene classification approach in details. To begin with, we introduce the image representation form that is used in the method. The classification algorithm is them proposed. We also design a similar method using SIFT features for comparison between the low-level scheme with a semantic model.

*Corresponding author's email at ICMLL: zcqin@buaa.edu.cn

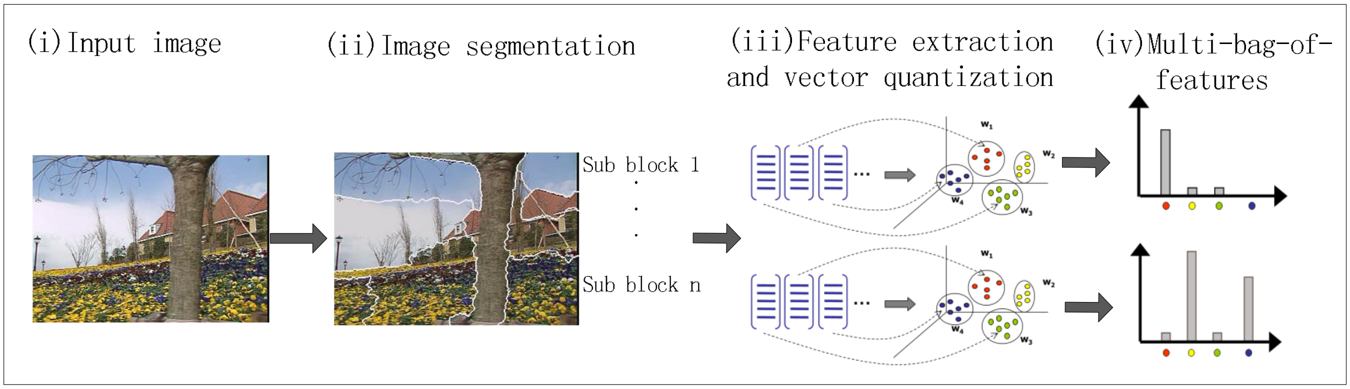

Figure 1: Image representation using multi-bag-of-features.

2.1 Image representation

Figure 1 illustrates our method of representing images. To represent images using low-level image features, every image is segmented into regions using normalized cuts algorithm [9, 10] [1]. Normalized cut algorithm treats image segmentation as a graph partitioning problem. It measures both the total dissimilarity between the different groups as well as the total similarity within the groups and it is an efficient image segmentation algorithm. Low-level features will be extracted from each of these regions based on [2]. The feature set consists of 36 low-level features: 18 color features, 12 texture features and 6 shape features. In this paper, bag-of-feature model proposed by Fei-Fei [3] is used to model images. We use bag-of-feature model to obtain vector quantization. The whole process is shown in Figure 1. Bag-of-feature model is inspired by bag-of-words model which regards document as frequencies of words from a vocabulary. Similarly, bag-of-feature model takes images for frequencies of "visual words". As shown in Figure 1, image representation approach is modified from the original bag-of-feature model. We use the following five steps in image quantization:

1. Segment images into regions.
2. Extract low-level features from each image regions.
3. Learn the "code-book" for each kind of low-level feature.
4. Quantize features using visual vocabulary.
5. Represent images by using frequencies of visual words.

[1]Normalized cuts is an efficient image segmentation algorithm. The code we are using in this paper for image segmentation is modified from *http://www.seas.upenn.edu/∼timothee/software/ncut/ncut.html*.

Therefor, suppose an image I is segmented into N regions $\{R_1, ..., R_N\}$, and the 36 features of each region can be expressed as follows:

$$feature(R_i) = \{f_1^i, f_2^i, ..., f_{36}^i\}$$

As a result, image I can be represented using the features of all of its segmented regions as follows:

$$I = \begin{pmatrix} f_1^1 & f_2^1 & ... & f_{36}^1 \\ f_1^2 & f_2^2 & ... & f_{36}^2 \\ . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \\ f_1^N & f_2^N & ... & f_{36}^N \end{pmatrix} \tag{1}$$

As shown in Equation (1), every column of the matrix shows one set of features of all the image's regions. This is not the final representation of an image. Each column of all the images' feature matrices are gathered to be learned using the bag-of-features model. K-means algorithm [7] is used to cluster these data into clusters and each center of clusters is considered as a visual word. M_i (for $i = 1, \ldots, 36$) is the size of visual vocabularies $V_i = \{v_1^i, v_2^i, ..., v_M^i\}$ and each image's first column can be represented by the frequencies of the above visual vocabulary $P^i = [p_1^i, p_2^i, ..., p_M^i]$. In our experiments, we set all M_is equal and denoted by a predefined constant M. Thus, each image can be represented by 36 bag-of-features models, which is referred to as the multi-bag-of-features model.

$$I = \begin{pmatrix} p_1^1 & p_2^1 & ... & p_M^1 \\ p_1^2 & p_2^2 & ... & p_M^2 \\ . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \\ p_1^{36} & p_2^{36} & ... & p_M^{36} \end{pmatrix} \tag{2}$$

E.g., Equation (2) is used as the representation of an image in this paper. It reveals the image's distributions over multiple feature visual words.

2.2 Scenes classification

Based on the image representation introduced above, we designed the following scene classification framework shown in Figure 2.

Training images are used to generate a bag of code-books and these data is used to train support vector machines (SVM) models [17] [2]. For 36 types of low-level features are extracted in this paper (details of the features are in [2]), we can obtain 36 SVM models - each SVM classifier is corresponding to one type of low-level feature. The system segments the input testing image and extracts features from regions which will be used in multi-bag-of-features by vector quantization with the aid of pre-trained bag of code-books. These bag-of-features will be tested using its corresponding SVM model. The final result classification is based on a weighted voting from each SVM.

The details of this scene classification approach will be illustrated by the following example. First, uppose the bag of code-books and SVM models have been trained. We can test an image I that has been represented using the multi-bag-of-features model, i.e., Equation (2). Each row of the matrix will become the input of its SVM model. For example, $P^1 = [p_1^1, ..., p_M^1]$ (where M is the size of the code-book) will be the input of SVM classifier 1 and get its classification result r_I^1. The rest rows of the matrix can be done in the same manner. Therefor, 36 SVM classifiers output a vector of classification results $[r_I^1, r_I^2, ..., r_I^{36}]$. The accuracy of each SVM classifier from training is represented by $[\omega_1, \omega_2, ..., \omega_{36}]$ which will be served as the voting weights for each SVM classifier (Equation (4)). The final classification of the testing image I can be determined by weighted voting as follows:

$$c(I) = \max_{c_i}\{V(c_1), V(c_2), ..., V(c_K)\} \tag{3}$$

Here c_i means the classes of the images and K represents the number of classes. $V(c_i)$ is the vote of class c_i and can be calculated using the following equation.

$$V(c_i) = \sum_{j=1}^{36} \omega_j \cdot \alpha_j \tag{4}$$

where ω_j is the voting weight for classifier j and,

$$\alpha_j = \begin{cases} 1, & r_I^j = c_i \\ 0, & otherwise \end{cases} \tag{5}$$

Our proposed image scene classification approach incorporates bag-of-features to represent images and use SVM as our classifier to achieve classification based on fusing low-level image features.

[2]A number of implementations of support vector machines can be downloaded from the web. The code used in this paper is modified from *http://luzhenbo.88uu.com.cn/svm.htm*. Another well-used software LIBSVM can be downloaded at: *http://www.csie.ntu.edu.tw/ cjlin/libsvm/*.

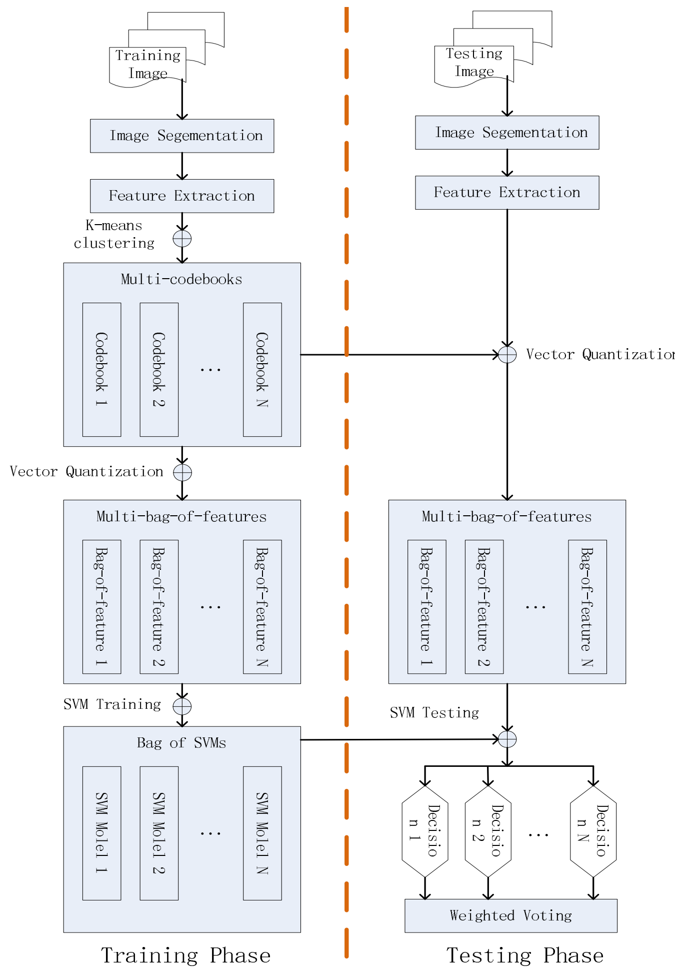

Figure 2: Work flow of the proposed scenes classification framework.

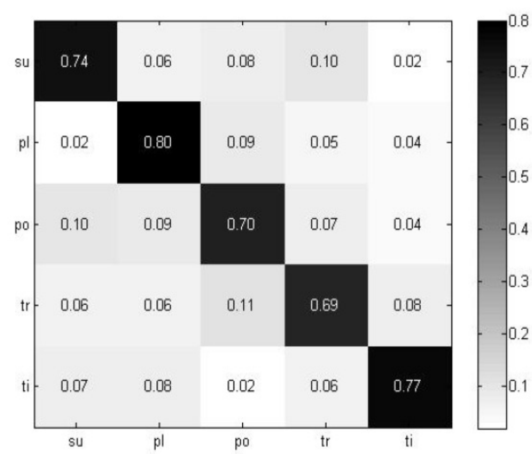

Figure 3: Classification results using 5 scene categories. The average accuracy is 74%.

2.3 Classify scenes using SIFT

In order to compare the low-level strategy with semantic modelling, we also implement an algorithm which is similar to that introduced in [4] using SIFT features. SIFT has been well-used in computer vision to detect and describe local features, which was originally proposed by David Lowe [5]. It can be used to perform reliable matching between different views of an object or scenes. The features are invariant to image scale and rotation, and it can provide robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination. Therefore, it has become the most used image feature in semantic modeling to classify scenes. Because we only want to illustrate that scene classification based on low-level image features like color, texture and shapes. SIFT is then used to be compared with those low-level features in the new multi-bag-of-features model. We simplified the algorithm in order to make it comparative with our proposed approach as follows.

1. Dense SIFT features are extracted from all the images.
2. the SIFT descriptors from training images are gathered to generate code-book using K-means clustering.
3. Train SVM classifiers on the SIFT data of images.
4. Classify test images using trained SVM.

3 Experiments and Discussions

To evaluate the performance, the proposed approach is applied to the Corel image dataset which is widely used in the research of image processing field. We randomly select N classes of images from the Corel dataset. The low-level features are extracted using the method described in section II. Figure 3 shows the classification results under 5 scenes, including: $sunset$, $plane$, $polar\ bear$, $tropical\ ocean$, and $tiger$.

In our approach, the size of code-book is the only parameter needed to be tuned manually. The effectiveness of different values of the parameter is also studied in the same experimental settings. Figure 4 illustrates the comparison of the performance with different size of the code-book. According to the experiment, 20 is a reasonable value for the size of code-book.

We also compare the performance of scene classification using the low-level strategy with the semantic model introduced in Section II. Table 1 lists the final results using different numbers of scene categories. The results show that the low-level strategy produces a comparable classification results in comparison with the more complicated semantic model when a small number of scenes is used. However, it should be noted that the performance of low-level strategy will turn to be worse when the number of scene categories increases, as shown in Figure 5. One reason is because of the descendant performance of the classifier used. Also the low-level method is incapable to distinguish the scenes with smaller dissimilarity.

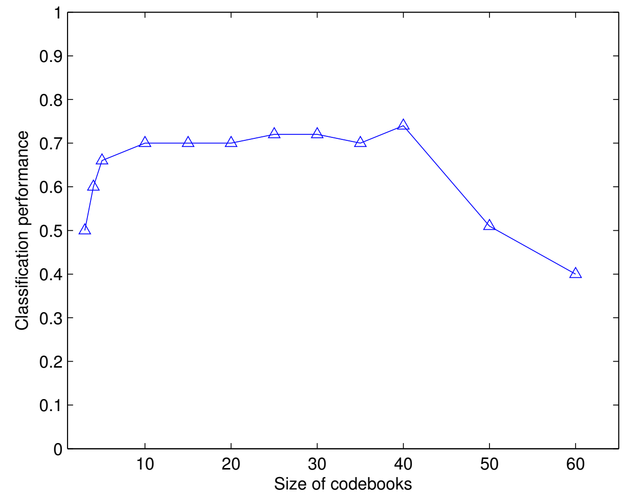

Figure 4: The illustration for effect of different size of code-book.

Table 1: Comparison of low-level strategy and semantic modelling

Accuracy/Category Number	5	6	7
Low-level strategy	0.74	0.73	0.73
Semantic modelling	0.82	0.80	0.74

4 Conclusions and future work

We have proposed a novel approach for classifying the image scenes into semantic categories. First, the image is segmented into meaningful regions and the low-level features are extracted based on each region. The image is then described using the multi-bag-of-features model. In this sense, the input image can be represented as a matrix of frequencies of multiple types of visual words. Finally, the SVM models and voting strategy are employed to classify the image scenes. The experiments indicate that the low-level strategy combined with semantic models achieves a good classification performance with reduced computation complexity when the number of scene categories is not too large. This makes the proposed approach valuable when the real applications require fast processing time and do not involve many categories. Future work focuses on further studying the relationships between these low-level features

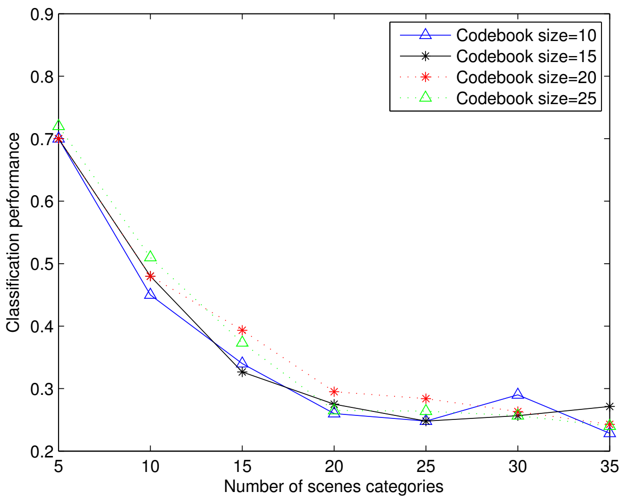

Figure 5: Classification performance descending with the increment of scene categories.

and combining the graphic models [8] to estimate the probabilistic dependency between them.

Acknowledgment

This work is partially funded by the NCET Program of MOE, China and the SRF for ROCS. The second author also thanks the China Scholar Council for visiting fellowship (No. 2010307502).

References

[1] E. Chang, K. Goh, G. Sychay, and G. Wu. Cbsa: Content-based soft annotation for multimodal image retrieval using bayes point machines. In *IEEE Transactions on Circuits and Systems for Video of Multimedia Content Description*, 13(1), pp. 26-38, 2003.

[2] P. Duyugulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceeding of 7th European Conference on Computer Vision*, pp. 97-112, 2002.

[3] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning nurural scene categories. In *Proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 524-531, 2005.

[4] J. Liu and M. Shah. Scene modeling using Co-clustering. In *Proceedings of IEEE 11th International Conference on Computer Vision*, pp. 1-7, 2007.

[5] D.G. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, 60(2), pp. 91-110, 2004.

[6] J. Luo and A. Savakis. Indoor vs. ourdoor classification of consumer photographs using low-level and semantic features. In *Proceeding of International Conference on Image Processing*, 2001.

[7] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceeding of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281-297, 1967.

[8] Z. Qin, M. Thint and Z. Huang. Ranking answers by hierarchical topic models. *IEA/AIE*, LNCS 5579 pp. 103-112, 2009.

[9] H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied fo faces and cars. In *Proceeding of IEEE International Conference of Computer Vision and Pattern Recognition*, pp. 1746-1759, 2000.

[10] J. Shi and J. Malik. Normalized cuts and image segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.

[11] A. Singhal. Bayesian Evidence Combination for Region Labeling. PhD thesis, University of Rochester, NY, 2001.

[12] A.W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. jain. Content-based image retrieval at the end of the early years. In *IEEE Transsactions on Pattern Analysis and Machine Intelligence*, 22(12), pp. 1349-1380, 2000.

[13] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*, 2rd ed. Brooks/Cole Publishing, Pacific Grove, CA, 1999.

[14] M. Szummer and R. W. Picard. Indoor-outdoor image classification. In *Proceeding of IEEE International Workshop on Content-based Access of Image and Video Databases*, 1998.

[15] A. Torralba. Contextual priming for object detection. In *International Journal of Computer Vision*, 53(2), pp. 169-191, 2003.

[16] A. Vailaya, M. Figueiredo, A. Jain, and H.J. Zhang. Content-based hierarchical classification of vacation images. In *Proceeding of IEEE Multimedia Systems*, Florence, Italy, June 1999.

[17] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, 1995.