Image Segmentation and Categorization

Enyu Luo, Raymond Yeh
Dept. of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
1406 W. Green St., Champaign, IL 61820, USA
enyuluo1@illinois.edu, yeh17@illinois.edu

This report reviews the task for class based pixel-wise segmentation and categorization using the single-histogram class model, and random forests. Histogram of visual words, textons, has been successfully used as the feature, for image segmentation and categorization task, to various supervised learning algorithm, such as k-nearest neighbor, or decisions trees. For this project, we investigate and implement the single-histogram class models, and random forest algorithm. These two methods are evaluated on the Microsoft research Cambridge object recognition image database [1]. We have achieved a reasonable performance on the task considering the limited running time and computing power we had.

Keywords- image segmentation; image categorization; textons; single-histogram

I. INTRODUCTION

The task of image segmentation and categorization is simultaneously segmenting an image into the semantic regions and labeling each region with the class that the region belongs to, by coloring the label image with the corresponding color. For our project, we used the Microsoft research Cambridge object recognition image database [1], illustrated in fig. 1, where the left is the actual image and the right is the pixel-wise label.

The task of image segmentation and categorization can be divided into two components, and learning phase, and the testing phase. For the *learning phase*, the input is a subset of the database, both the original image and the labeled image, as we are using supervised learning algorithms; the output of the learning phase is an object classifier to be used in the testing phase.

For the *testing phase*, the input is a subset of the database of original image, not included in the training set, and the object classifier. The output of the testing phase is the labeled image of each of the image in the testing set. The main blocks included in the system are illustrated in fig. 4.

Feature clustering and generation of texton map may or may not be necessary depending on the feature that is being used to train the classifier.

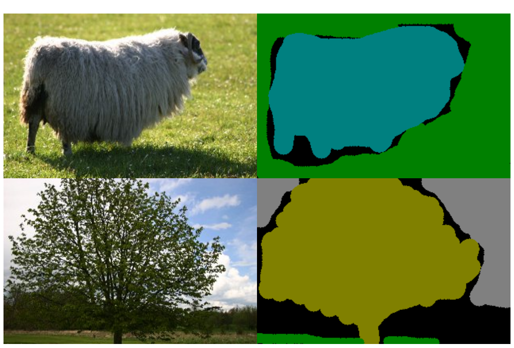

Fig. 1. Image database illustration

II. SINGLE HISTOGRAM CLASS MODEL

A. Overall Procedures

This section will describe the basic algorithm for performing the single histogram class model for the segmentation and categorization task.

First, the feature vectors used at each pixel location are raw 5×5 color patches in the CIE-LAB color space; therefore, the dimension of each feature vector is 75. The next step is to perform feature clustering to obtain V visual vocabulary called textons. The feature clustering is performed with k-means on a randomly sampled 25% subset of the feature vectors from each of the training images, to reduce the training time.

Given the set of cluster centroids return from the k-means algorithm, each pixel in the training image is assigned with the closest centroid, texton, using nearest neighbor algorithm. The result of the assignment is the texton map, which can be visualized by color-coding each centroid with an unique color, illustrated in fig. 2.

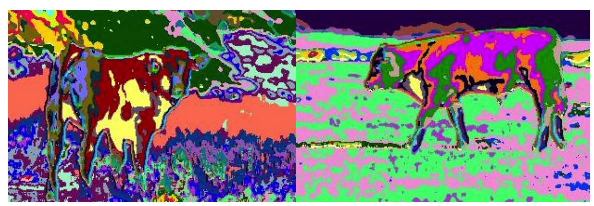

Fig. 2. Texton map illustration

Next, compute the histograms of visual words for each of the pixel in a $(2w + 1) \times (2w + 1)$ sliding window, $w = 12$ for our experiments.

Lastly, these histograms are combined into one single-histogram class model to be used as a feature to the supervised learning algorithm to generate object classifiers. The motivation behind a single-histogram class is for more efficient classification process. For example, if using a nearest neighbor algorithm, instead of comparing to all histograms in the training set, one will just need to compare to the set of class histograms, which is much smaller than the training set.

B. K-means Clustering

K-means algorithm used for the task of clustering. Meaning assigning n token in the set to k clusters, where the algorithm aims to minimize the follow objective function,

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n}(x_i^j - c_j)^2 \quad (1)$$

where x_i^j denotes a token in cluster j, *and* c_j denotes the cluster j^{th}'s centroid.
The k-means algorithm is as follows

Algorithm 1 k-means

Begin
  Initialize cluster centroid $c_1 \ldots c_k$
  While centroid has not converged
    Use centroid to classify each token into clusters
    For *i* from 1 to *k*
      Updated c_i to mean of all token in cluster *i*
    End
  End
End

C. Single-Histogram Class Model

As derived in [2], that the single-histogram class model that minimizes the overall "distance", either KL-divergence or euclidean distance from the set of histogram is as follows

$$\hat{\boldsymbol{q}} = \frac{\sum_j n^j \boldsymbol{p}^j}{\sum_j n^j} \tag{2}$$

where n^j denotes the number of pixels in the j^{th} exemplar region, $\boldsymbol{p}^j$ denotes the normalized histogram of the j^{th} exemplar region, and $\hat{\boldsymbol{q}}$ is the overall histgoram distribution of all training reigions.

KL-divergence is defined as

$$D_{KL}(P||Q) = \sum_i \ln\left(\frac{P(i)}{Q(i)}\right) P(i) \tag{3}$$

where *P* and *Q* are discrete probability dsitributions [2].

III. Random Forest of Decision Trees

A. Decision Tree

A decision tree classifier is a classifier that checks a series of node functions based on the path of the tree, and finally determines the class of the test case. A node function can be single masks, multiple masks, and even the single histogram output values. The node function can be define mathematically as follows [3]

$$t_p = \sum_{r \in \{1,2 \ldots\}} w_r \cdot f_r \tag{4}$$

where t_p denotes the node function at pixel location *p*. $\boldsymbol{f_r}$ denotes the feature vector and $\boldsymbol{w_r}$ denotes the mask, and r corresponds to the number of the masks.

The decision tree can learned by greedily choosing the node functions (attributes) and threshold that maximizes the information gain as in eq.5 until all the leaf nodes are consists of the same class.

The information gain(IG) for a patricular node is the change in entropy $\boldsymbol{H}$, define in eq. 6, from the prior state to the current state define as

$$\boldsymbol{IG} = \boldsymbol{H}(\text{Parent}) - [\text{Average } \boldsymbol{H}(\text{children})] \tag{5}$$

where entropy $\boldsymbol{H}$ is as follows

$$\boldsymbol{H} = \sum_i -p_i \cdot \log_2 p_i \tag{6}$$

where, p_i is the probability of choosing the i^{th} class tokens out of all training tokens at the node.

The intuition behind entory is the higher than entropy the more information content. Thus by computing the difference of the entropy between the parent and children of a node, it is the information gain.

B. Random Forest

A random forest is basically bags of decision trees. During the training, multiple decision trees, with random selection of features, are trained and majority voting on the class of the individual trees will determine output of the random forest. The main advantage of a random forest is that a single decision tree is likely to be over fitting, and lead to high test set error. The random forest reduces this issue and lead to a lower test set error [3].

C. Decision Tree Input Features

For this project, the feature we also used were RGB, CIELAB raw feature pixel and histogram of oriented gradients (HOG). To extract the HOG feature, first apply the mask **[-1 0 1]** and $\mathbf{[-1\ 0\ 1]^T}$ then quantize the orientation into 6 bins [4]. An illustration of the HOG feature is illustrated in Fig. 3.

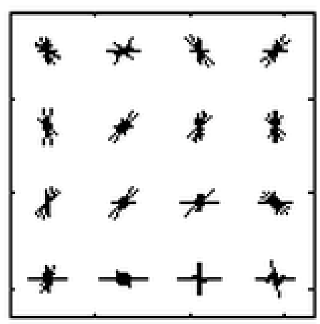

Fig. 3. HOG feature illustration

IV. System & Experiment Setup

The overall system block diagram is shown in fig. 4. The system was entirely written in MATLAB, and using the image processing toolbox. For this project, Microsoft research Cambridge object recognition image database was used. The database provided pixel-wise labeled images, which is essential for the supervised learning algorithms.

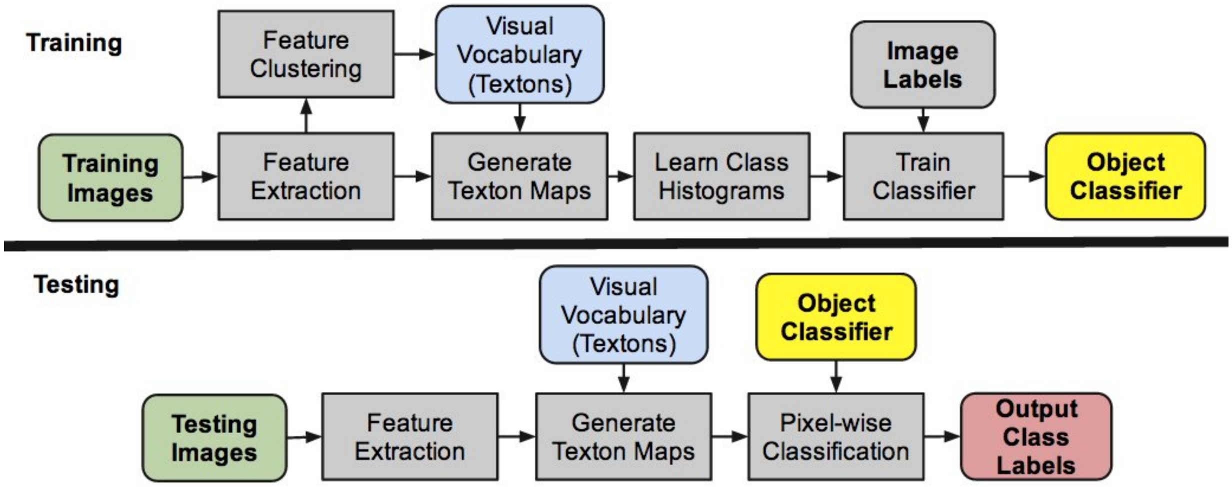

Fig. 4. System block diagram

V. Results

Two different tests were run using the Single Histogram class model. The first run contains 125 images that include cows, sheep, grass, sky, houses, and planes; a total of 11 classes. This was run with 300 clusters and the k-means algorithm did not converge at 200 iterations. The algorithm ran for 4 hours. Most of the processing time was spent on the training phase when doing k means clustering, since it was trying to cluster 300 clusters with 75 dimensions. 80% of the images (100 images) were used for training and the remaining 25 images were used for testing. The average result obtained from the testing phase was 51.9% accurate.

The second test was run with a subset of the images used in the first run. Images of houses and planes were removed from the database. This is an attempt to improve accuracy by reducing the number of classes that the algorithm has to classify. The number of clusters is reduced to 200 since we took out some categories of images, and we increase the max iterations parameter to 300 to produces better clusters although it still did not converge. 48 images were used for training and 12 images were used for testing. They have a total of 10 classes and average accuracy of 69.3%. This time the algorithm ran for about 1.5 hours.

The third test, instead of using a single Histogram class model, we used random forest classifier with the RGB & HOG feature, and the average accuracy is 61.2%.

Comparing to the paper [2] achieved 82% accuracy over 9 classes using 8000 clusters. We did not run the algorithm with this many textons, as it would probably take days of processing time on our laptops.

VI. Discussion and Future Work

Overall, our project produced reasonable results. We have achieved the average accuracy of 69.3% comparing to the 84% reported in [3]. This is main reason for this degrade in accuracy is because we were limited to using less number of textons due to the computation complexity of the k-means algorithm and learning the decision tree. The complexity for the k-means algorithm is $O(n^{dk+1} log\ n)$, and for learning the decision tree it is $O(n^2)$. Therefore, these two tasks become very expensive when the number of training example, pixels in the training images, grows.

With too few numbers of textons, it is impossible to capture enough of variance for a particular class, and results in lower accuracy in the testing case; as differently classes are captured with the same set of textons. Which explains one of the main issues with the system is that the classes were classified consistently, but with the wrong class label, as illustrated in fig.5, the left is the actual image, the middle is our system's label, and the right is provided label. In Fig.5, the cow (blue) is categorized as a sheep (cyan) consistently but incorrectly.

Fig. 5. Missed categorized result

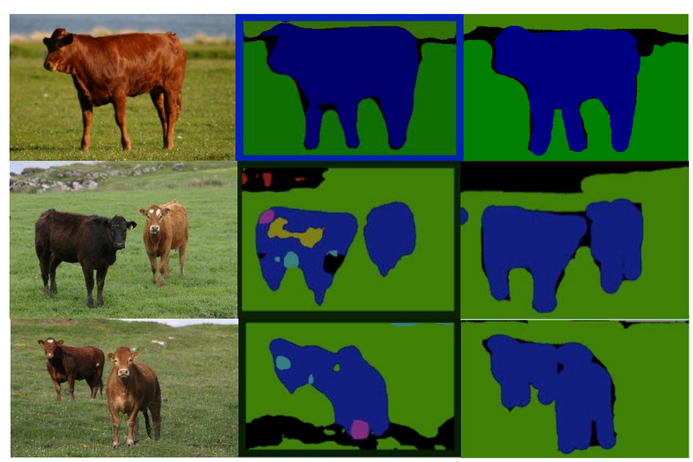

Fig. 6. Successful categorization results

Figure 6 shows some successfully categorized results from our system. From the fig.6's results, it can be observed that most of the grass are categorized successfully, however the system has difficulty capturing the entire body of the cow, as some parts are miscategorized as grass, sheep or horse.

Lastly, some future work for this project is to implement more masks for the node functions when using a decision tree, and also try to compute the single histogram class model with a large number of texton. These should lead to an accuracy that is closer to the ones reported in [2, 3]. Furthermore, we could also implement conditional random fields (CRF), which have been shown in [3] to have better result compare to the systems without CRF.

ACKNOWLEDGMENT

We would like to thank Professor Do and the TAs, Andre & Ha, for the interesting and fun semester of ECE418: Introduction to Image & Video Processing.

REFERENCES

[1] A. Criminisi. Microsoft research cambridge object recognition image database.version1.0, 2004.

[2] F. Schroff, A. Criminisi, and A. Zisserman. Single-Histogram Class Models for Image Segmentation. In Proc. ICVGIP, 2006.

[3] Schroff F, Criminisi A, Zisserman A (2008) Object class segmentation using random forests. In: Proc British machine vision conference (BMVC)

[4] N.Dala land BTriggs. Histogram of Oriented Gradients for Human Detection. In Proc.CVPR, volume 2, pages 886–893, 2005