CS682 Final Project Report
Location Recognition

Xing Zhou
Dept. of Computer Science
George Mason University
xzhou10@gmu.edu

Abstract

In this project, the topic of image retrieval has been investigated and implemented. Following the general pipeline of image retrieval, features are firstly extracted, then hierarchy K-means is employed to train the features from training set to build a vocabulary tree. Both the images from training set and test set are represented as a histogram of the frequency of visual words. When a query image comes, the system will return the most similar one according to the defined similarity function. In this project, the effect of several ingredients of image retrieval, such as number of visual words, different features, various similarity functions have been evaluated. Also, negative relevance feedback has been proposed to improve the performance.

1. Introduction

Suppose you are lost in a foreign city, you don't speak the language. What will you do? Existing systems such as satellite positioning may give you a direction; however in some situations they have problems. For instance, GPS positioning system cannot guide you if you were in cities where tall buildings shield you from direct line of signal with the satellites. And also the precision of GPS positioning system is 10 meters at best.

The topic of location recognition studied here will give you another option to conquer such problems. The location recognition system (LRS) is under the umbrella of content-based image retrieval systems. Given a query image as input, LRS will index some candidates in the database with high similarity scores. Then according to the existing knowledge associated with the images in the database, you can learn more about your present situation. Let's go back to the example we mentioned before, you can simply take out your cellphone, photograph the nearest building and send it to LRS system, which will work remotely and send back directions or guidelines to you. An overview of the LRS is demonstrated in Figure 1.

Figure 1. Overview of Location Recognition System.

2. Approach

2.1. Feature Extraction

Various features can be used to represent an image, such as textures, edges, colors as well as SIFT. In this project, only SIFT and color are chosen. And their respective performance as well as the performance of their combination are evaluated. In particular, the color features are computed as the average within the region of super-pixels which result from previous segmentation using watershed method [1]. In addition, the LUV color space is used which seems to be more compatible with human vision. The results of extracted SIFT features and super-pixel segmentation are illustrated in Figure 2 and Figure 3. From Figure 3, we can clearly notice that grouping pixels into super-pixels is more meaningful for training visual words than single pixels.

2.2. Building and Using the Vocabulary Tree

The vocabulary tree defines a hierarchical quantization that is built by hierarchical k-means clustering [2]. A large number of feature descriptors are used in the unsupervised training of the tree.

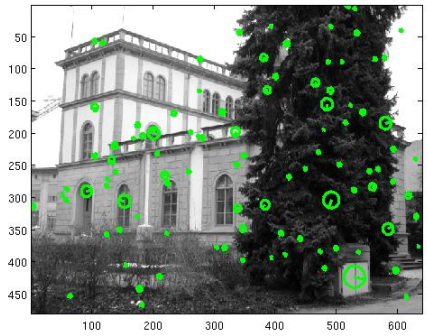

Figure 2. Extracted SIFT features.

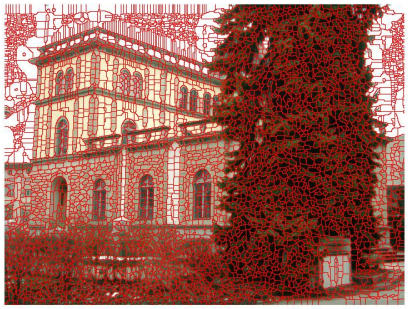

Figure 3. Segmentation of Super-pixels

Instead of k defining the number of clusters as in the standard k-means clustering, k defines the branch factor of the tree. An initial k-means clustering is running on the training data, then the training data splits into k sub-groups, where each sub-group contains the feature points closest to its cluster center. Then the same process runs on each sub-group recursively until some stop criteria is met. The process of hierarchical k-means is shown in Figure 4.

In the online phase, a descriptor traverses through the tree by at each depth comparing the descriptor to the k candidate cluster centers and choosing the closest one. The path down the tree can be encoded by integers as the number of nodes. After propagating all of the descriptors extracted from an image, a histogram of the frequency of every node being visited can be constructed. There are some issues which should be considered when constructing the histogram to represent the image.

1) How to weight the nodes in the tree? Since the information they carried is different to each other. Generally, the deeper the node lies, the more distinctive information it carries. For example, the root node carries none distinctive information since every descriptor would definitely pass through it.

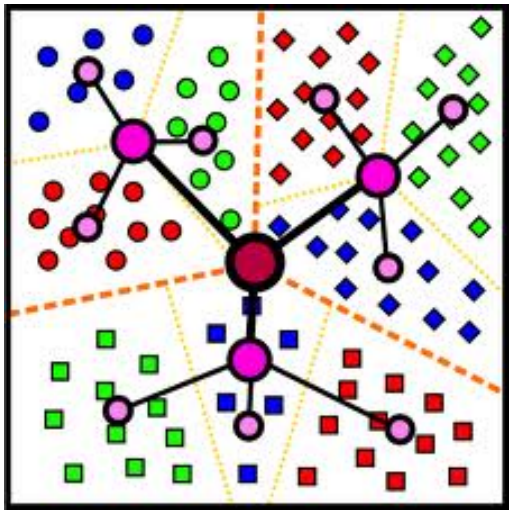

Figure 4. Illustration of the process of building the vocabulary tree.

2) Should we use all the nodes, some of the nodes or only the leaf nodes?

2.3. Relevance Feedback

In information retrieval, the idea of relevance feedback is to involve the user in the retrieval process so as to improve the final result. More specifically, the user gives feedback on the relevance of documents in an initial set of results. In this project, in order to let relevance feedback be done automatically, a set of images either positive or negative are selected randomly and are fed to the retrieval system along with the query image.Suppose we have n negative samples denoted as $S^{-}(I_1, I_2, ...I_n)$, then the similarity score of query image q_i comes to Equation (1):

$$S(q_i) = S(q_i, r_j) + \sum_{k=1}^{n} 1 - S(I_k, r_j) \qquad (1)$$

where r_j is thej^{th} reference image in the database.

If the selected negative sample is far away from the query image in feature space, this negative relevance feedback will indeed help to get the right result. But as the number of negative instances increases, their information would overwhelm the information carried by query image.

2.4. Similarity Measurement

2.4.1 Histogram Comparison Measures

Many of the features presented are in fact histograms. A lot of comparison measures have been proposed and a comparison of those measurements can be found in [4]. In this project, the following comparison measures are employed.

Minkowski Distance
Minkowski Distances are a group of distance functions defined by

$$d_p(H, H') = (\sum_{m=1}^{M} (H_m - H'_m)^p)^{\frac{1}{p}} \quad (2)$$

The well-known and widely used Euclidean distance as well as the $L1$-norm are compared in this project.

Angle between Vectors
The angle between two vectors are introduced to score the similarity of two histograms. The smaller the angle is, the more similar two feature histograms are. The angle is computed by the dot product of these two vectors.

$$d_p(H, H') = \arccos \frac{\vec{H}\vec{H'}}{\| H \| \| H' \|} \quad (3)$$

2.4.2 Weight Scheme of the Nodes

I borrow the scheme of weighting the nodes from [2]. In [2], a weight w_i is assigned to each node i in the vocabulary tree, typically based on entropy, and then the query q_i and the reference vector r_i are defined by weighted nodes as

$$q_i = n_i w_i \quad (4)$$

$$r_i = m_i w_i \quad (5)$$

where n_i and m_i are the number of descriptor vectors of the query and reference image, respectively, with a path through node i. Then the similarity score between query and reference images is given based on the normalized difference between query and reference vectors.

$$s(q, r) = \| \frac{q}{\| q \|} - \frac{r}{\| r \|} \| \quad (6)$$

In simplest case, the weights w_i could be set to a constant, but retrieval performance is typically improved by set to an entropy weighting like

$$w_i = \ln \frac{N}{N_i} \quad (7)$$

where N is the number of images in the database, N_i is the number of images with at least one descriptor passing through the node i.

3. Experiment

3.1. Data Set

ZuBuD [3] Zurich Buildings Database for Image Based Recognition is used as sample data in this project. This training data set contains 1005 color images of 201 buildings (5 images per building) or scene, which are captured at random arbitrary view point, under different seasons and weather conditions. Also in order to test some local feature based algorithm, purposely some occlusions by tree and other objects were included on some image. Figure 5 shows an example in which the building was occluded by tree in some view. In addition, another 115 images have been acquired to for recognition performance test. All these images contain the buildings included in the database; however the imaging conditions do not match exactly. Figure 6 gives an example of on query image with all 5 similar images in the database. All the images captured by cameras at 640 480 and saved as 24 bit JPEG format. In the database, all original images were transferred to png image format.

Figure 5. One example of 5 images of one building in database.

Figure 6. Query Image and five similar images in database. The first one is query image with tree occlusion. The other five images are the same building in database.

3.2. Result

The method was tested by performing queries on ZuBuD data set. The effects of the following ingredients of the method are quantitatively compared.

1) Number of Visual Words (Figure 7);

2) SIFT Features, Color Features and both of them (Figure 7);

3) Negative Relevance Feedback (Figure 8);

4) All nodes V.S. only leaves (Figure 9);

5) Histogram Distance Measurement (Figure 10).

4. Conclusion

In this project, the pipeline of context-based image retrieval was implemented. From the given experimental results, we can conclude that in context-based image retrieval SIFT feature is more effective and robust than color features, and in this data set the combination of them both helps to improve the performance. As the number of visual words increased the histogram becomes finer which results in a better retrieval result. But one thing we should notice that

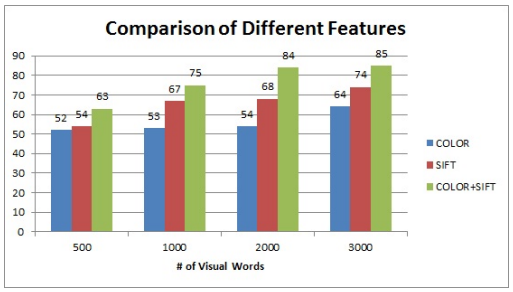

Figure 7. Comparison of the effect of the number of visual words as well as the feature used.

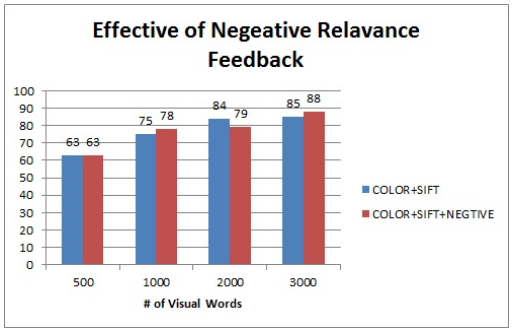

Figure 8. The effectiveness of introduction of negative relevance feedback.

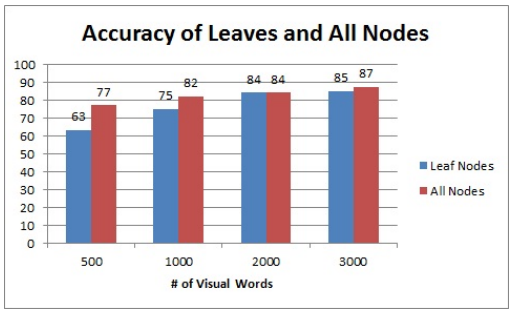

Figure 9. Comparison of the effects of using all nodes and only leaves in vocabulary tree.

thousands of visual words for color is already pretty high since in RGB color space there are only 255 bins for each channel while in LUV color space there are only 100 bins in L channel and 200 bins in U and V channel.

The technique of relevance feedback introduced in this project seems not help much to improve the results. The main reason might due to the data set containing only one category - buildings. So, it is hard to get a negative instance far from the query image in the feature space. But this might be a good idea when applying to multi-categories where great difference exists among different categories.

The importance of the nodes in vocabulary tree was also investigated. The result shown in Figure 9 is compatible

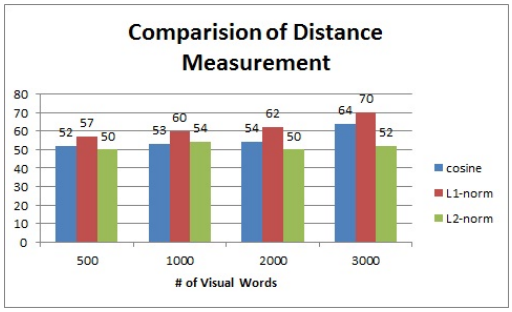

Figure 10. Comparison of different histogram distance measurements.

with our common sense, that is the nodes at deeper level carry more information than those in upper level. Although incorporating all of the nodes in the histogram seems to have the total information in hand, it makes no difference when only leaf nodes were used which consume less time as well.

References

[1] Micusik B., and Kosecka J.. Semantic Segmentation of Street Scenes by Superpixel co-occurrence and 3D Geometry. *IEEE Workshop on Video-Oriented Object and Event Classification (VOEC)*, 2009.

[2] David Nister, and Henrik Stewenius. Scalable Recognition with a Vocabulary Tree. *IEEE Conference on Computer Vsion and Patern Recognition*, 2:2161–2168, 2006.

[3] Hao Shao, Tomas Svoboda and Luc Van Gool. ZuBuD - Zurich Buildings Database for Image Based Recognition. *Technical Report*, No. 260, 2003.

[4] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Features for Image Retrieval: An Experimental Comparison. *Information Retrieval*, 11(2):77–107, 2008.