A New CBIR System Using SIFT Combined with Neural Network and Graph-Based Segmentation

Nguyen Duc Anh, Pham The Bao, Bui Ngoc Nam, and Nguyen Huy Hoang

Faculty of Mathematics and Computer Science, University of Science,
Ho Chi Minh City, Vietnam
{ndanh,ptbao,bnnam,nhhoang}@hcmus.edu.vn

Abstract. In this paper, we introduce a new content-based image retrieval (CBIR) system using SIFT combined with neural network and Graph-based segmentation technique. Like most CBIR systems, our system performs three main tasks: extracting image features, training data and retrieving images. In the task of image features extracting, we used our new mean SIFT features after segmenting image into objects using a graph-based method. We trained our data using neural network technique. Before the training step, we clustered our data using both supervised and unsupervised methods. Finally, we used individual object-based and multi object-based methods to retrieve images. In the experiments, we have tested our system to a database of 4848 images of 5 different categories with 400 other images as test queries. In addition, we compared our system to LIRE demo application using the same test set.

1 Introduction

The need of fast and precise image retrieval (IR) systems is growing rapidly as computer power evolves quickly as well as the size of image databases. Most present IR systems fall into two categories: text-based systems and content-based ones. The simplest systems are text-based such as Google, Yahoo, etc. Though those systems are usually fast at retrieving time, they all suffer from many big drawbacks like the ambiguity of languages or the absence of precisely visual presentation in annotation as well as time consuming annotating process as stated in [1].

Unlike text-based IR systems, CBIR systems do not involve the subjective human perception in describing the semantic meaning of an image. Instead, the visual features of an image are extracted automatically. Thus, the performance of CBIR systems largely depends on what kinds of visual features being used and the extracting methods. There have been a large variety of image features proposed by many researchers as discussed in [2]. Among those, SIFT features of Lowe [3] have been proven to be very effective due to their invariance to many different transformations. Furthermore, most IR systems divide the original set of data which can be images, image objects or image annotations into subsets or clusters. Therefore, clustering methods also involve the retrieval performance.

Motivated by the effectiveness of SIFT features, in this paper, we developed our new mean SIFT features. We then applied these features to regions or objects

N.T. Nguyen, M.T. Le, and J. Świątek (Eds.): ACIIDS 2010, Part I, LNAI 5990, pp. 294–301, 2010.

segmented from images. Many approaches to image segmentation have been developed in recent years like those in[4].We used graph-based segmentation method of Pedro F. Felzenszwalb and Daniel P. Huttenlocher[5] due to its fast performance and its ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions. Like other IR systems we also clustered the database data. Specifically, we employed supervised and unsupervised clustering methods. In the first method, we clustered data manually. In the later one, the system clustered data automatically. Among existing automatic clustering methods like k-mean clustering of Stuart Lloyd [6], hierarchical clustering of S.C. Johnson [7], we decided to develop our own method by modifying the graph-based segmentation method since former techniques either failed to converge or the clustering results were not good. Given the clustered data, the goal to determine which cluster the input data belongs gives rise to training the database data. In our CBIR system, we used neural network to train our clustered data instead of some statistic techniques since many of them require either a prior knowledge or assumptions about the data, such as naive Bayes models [8]. Finally, we used two retrieval methods: individual object-based and multi object-based.

2 System Overview

As mentioned our IR system has three main functions. Each of them can be part of one or both of the following stages: the database training stage and the retrieval stage. Diagram 1 shows the database training stage and diagram 2 shows the retrieval stage.

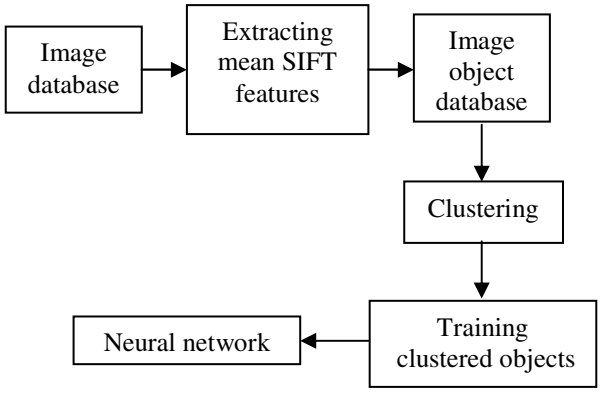

Diagram 1. Database training stage

In the first step of the database training stage, we extracted features from each image in the database as follows:

i. Extract original SIFT features from the image.
ii. Automatically segment the image into sub-regions which we considered as objects.
iii. For each object, compute its mean SIFT feature.

Given this database of objects features, we then used supervised and unsupervised clustering methods to cluster the original image database. We used our own unsupervised clustering algorithm. After that, all objects of images in the same clusters were also put into the same clusters. Finally, we trained those clusters using neural network technique to get the supervised and unsupervised neural networks accordingly.

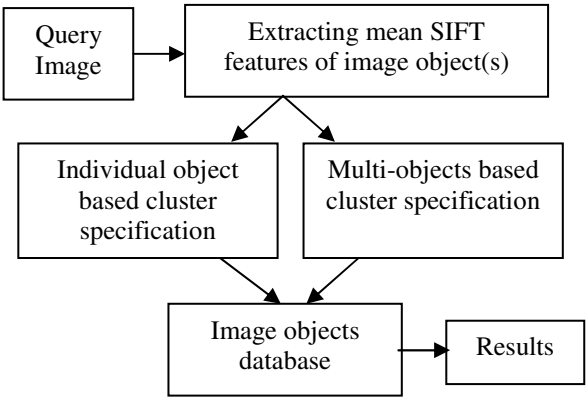

Diagram 2. Retrieval stage

The first step of the retrieval stage is similar to that of the database training stage. Given the set of objects automatically extracted from the input image, we then extracted the mean SIFT feature(s) of the object(s). As mentioned earlier, there are two neural networks. Thus we also have two distinct ways to classify objects. For each of these two neural networks, we first let user choose one object and classify that object. The results are a set of objects closest to the query object in the specified group. This is called *individual object-based retrieval*. Another way is to select the group that most objects fall into and compute the distance between all query objects and objects in that group and then return N objects of N smallest distances. We called this method *multi-object-based retrieval*. In our system, we used arcos distance for the sake of the effectiveness of our mean SIFT features.

In the next two sections, we are going to introduce our new mean SIFT feature and an automatic clustering routine based on the graph-based segmentation method mentioned earlier. Finally, we show some experimental results.

3 Mean SIFT

Before SIFT method, there had been a lot of works on image features extraction, such as Harris Corner Detector [9] or methods discussed in [2]. Those methods all have their own advantages. However, such methods fail to detect similar features at different scales. Based on scale space theory [10], SIFT method has been proven to be a successful candidate to the detection of image features at different scales. Furthermore, it has been shown that SIFT features are invariant to Affine transformations and changes in illumination. SIFT method has three main stages: locating key-points on

image, building detectors and building descriptors. The descriptors are then the features of an image. Despite outstanding advantages, SIFT method does have major drawbacks, such as the large number of descriptors, a 1024×768 image can, in practice, have up to 30000 descriptors. While the amount of descriptors is usually large, the number of matches between two similar images is small. Furthermore, sometimes, the number of matches between two similar images is the same as that of two irrelevant ones. The first issue leads to unbearable complexity in performance when dealing with large database, especially in matching time. The latter one concerns the unreliability in using the number of matching as a similarity measure between images. Many efforts have been put by researchers to tackle those issues such as using PCA to reduce descriptor's dimension [11] or removing features based on certain assumptions [12]. However, the size of a feature vector is still large in both methods. Besides, none of them addresses the unreliability issue.

In this paper, we propose our new mean SIFT feature based on the following experimental result.

Given three images A,B and C in which A and B are similar and C stands out. A has a set of descriptors $D_A = \{a_i | i = \overline{1, n_A}\}$ and its mean descriptor mA where

$$mA = \frac{\sum_{i=1}^{n_A} a_i}{n_A}. \tag{1}$$

The same notation is used for B and C. Let the arcos distance between mA and mB be denoted by

$$T_{AB} = arcos\left(\frac{mA.mB}{|mA||mB|}\right). \tag{2}$$

We shall now prove that the arcos distance between two similar images is smaller than that of two different images or $T_{AC} - T_{AB} > 0$ in this case.

Fig. 1. Each column is a triplet. Top row-image A. Middle row-image B. Last row-image C.

To obtain this result, we first conducted a test over 300 triplets of images, figure 1. Each triplet has two similar images, A and B, and an odd one, C. Let $T = T_{AC} - T_{AB}$ and f is the frequency of having $T > 0$. There are 253 triplets which have $T > 0$ in 300 samples. Based on the central limit theorem, we gain the following result:

$$\frac{f-p}{\sqrt{\frac{p(1-p)}{n}}} \sim N(0,1) \tag{3}$$

where n is 300, f is 253/300 and p is the probability of having $T > 0$.

Let the significance level be 95%, we get the formula:

$$-1.96 \leq \frac{f-p}{\sqrt{\frac{p(1-p)}{n}}} \leq 1.96 \quad (4)$$

As a result, the value of p is between 0.802 and 0.884. That means the event $T_{AC} > T_{AB}$ has the probability from 0.802 to 0.884 with a significance level of 95%.

The above proof states that if two images are more similar to each other, then chances are the distance between their mean descriptors is smaller. Thus we decided to use mean descriptor as our new mean SIFT feature. This feature can be applied for an entire image or each object segmented from an image. In either ways, the number of feature vectors is extremely small since one image has a very small number of meaningful objects.

4 An Automatic Clustering Method Based on Graph-Based Segmentation

Unlike image segmentation, data clustering in general may involve more data and the inter-relation between data may be stronger. Therefore, only one run of the original method may not result in a satisfactory segmentation. It is likely that some groups are good in the sense that their members really share similar features while some are not good meaning their members do not have anything in common. Moreover, the size of groups is another issue. By experiments we saw that using one run of the original technique in data other than image pixels may result in groups either too small or too big.

Because of those drawbacks, we decided to construct our new clustering algorithm based on the original graph-based segmentation. This method must address the two big issues mentioned above. The most appropriate solution is to put the original method into a loop and set reasonable conditions to check the outcome after each loop.

Like the original algorithm, we considered each data as a vertex and the weight of an edge connecting two vertices is the similarity measure between them.

Step 1: Initiate the set of edges E from the set of vertices V.

Step 2: Use the original graph-based segmentation to cluster V into groups:C_i $i = 1,..n$.

Step 3: For each C_i, if it satisfies condition T_1, we consider C_i as a qualified group and remove all vertices in C_i and all edges related to them. Otherwise, we loop through each edge $e_j$in C_i, if it satisfies condition T_2, we change the weight of this edge using a function: $w(e_j) = F(w(e_j))$.

Step 4: if V satisfies conditionT_3, stop. Otherwise, go back to step 1.

Conditions T_k $k = \overline{1,3}$, in general, could be any user-defined ones. In this context T_1 is the condition that the mean and standard deviation of all edges' weights in C_i are smaller than pre-defined thresholds. This is to tackle the first issue. T_2 is the

condition that an edge's weight must be smaller than a threshold, and F is a weight reduction function, particularly, we lessen the weight by 90% its original value. This is to deal with the second issue. Finally, T_3 is the condition that the number of remaining vertices in V is smaller than a threshold or after a certain amount of loops the size of V remains unchanged. This condition guarantees that the algorithm will stop.

5 Experiments

Our databaseconsists of 4848 images of five different categories: landscapes, vehicles, train-planes, signs and women. This is a very challenging database due to its variance in many properties like resolution, image domains, etc. We also use other 400 images as test queries. For each query, we returned 20 results, figure 2. As stated earlier, we use two retrieval methods: individual object-based and multi-object-based, and two clustering techniques. Thus we have four different test cases in total. The table below shows the average precision of all methods.

Table 1. Average precisionof retrieval methods

	Individual object-retrieval	Multi-object-retrieval
Supervised	80%	78.5%
Unsupervised	61.8%	59.4%

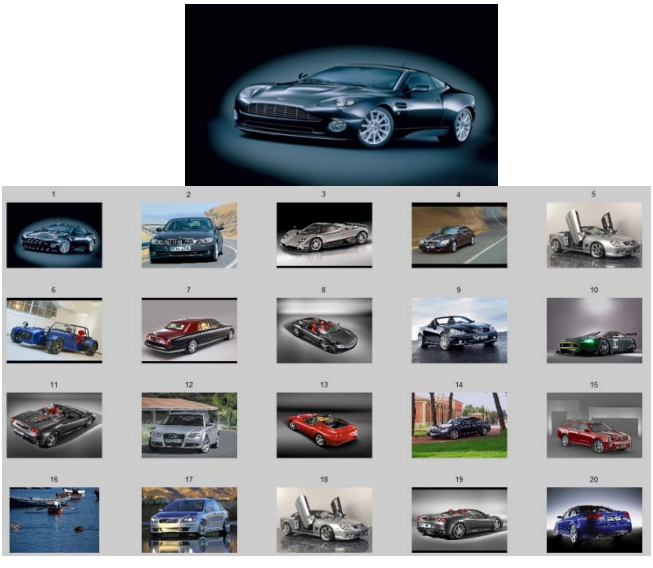

Fig. 2. A query image and 20 nearest results using individual-object retrieval and supervised clustering

We also used the same test set for the LIRE demo application, a java open source CBIR software developed by Mathias Lux and *Savvas A. Chatzichristofis* [13][14]. Particularly, we chose the method of Color and edge directivity descriptor (CEDD) [14] [15]. The precision is 50.87% which is not as good as that of our methods. In addition, in cases of images containing complex background information, our technique performs better than LIRE program, figure 3. However, the LIRE's training process is faster and less complicated than that of our system.

(a) (b) (c)

Fig. 3. The comparision between LIRE demo application and our system related to a sign image with a complex background: (a) the querry image; (b) the result of LIRE; (c) the result of our system

6 Conclusion

Since our CBIR system still lacks two important pre-processing and post-processing parts: noise removal and user-feedback reception, this result could be improved if those parts are taken into account. Nonetheless, this result is encouraging since our system was tested to a challenging database. Besides, the automatic clustering method that we developed can be used for other purposes based on the specific problems.

Acknowledgement

We would like to thank DOLSOFT Inc for providing us a large-scale image database with a wide range of categories, resolution and transformations.

References

1. Lam, T., Singh, R.: Semantically Relevant Image Retrieval by Combining Image and Linguistic Analysis. In: International Symposium on Visual Computing, pp. 770–779 (2006)
2. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-Based Image Retrieval at the End of Early Years. IEEE Transaction on Pattern Analysis and Machine Intelligence 22(12) (December 2000)
3. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
4. Gonzales, R.C., Woods, R.E.: Digital Image Processing, 2nd edn., pp. 567–634. Prentice Hall, Englewood Cliffs
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Graph-Based Image Segmentation. International Journal of Computer Vision 59(2), 167–181 (2004)
6. Lloyd, S.P.: Least squares quantization in PCM. IEEE Transactions on Information Theory 28(2), 129–137 (1982)
7. Johnson, S.C.: Hierarchical Clustering Schemes. Psychometrika, 241–254 (1967)
8. Domingos, P., Pazzani, M.J.: Beyond independence: Conditions for the optimality of the simple Bayesian classifier. In: International Conference on Machine Learning, pp. 105–112 (1996)
9. Harris, C., Stephens, M.: A combined corner and edge detector. In: Fourth Alvey Vision Conference, Manchester, UK, pp. 147–151 (1988)
10. Lindeberg, T.: Scale-space theory: A basic tool for analysing structures at different scales. Journal of Applied Statistics 21(2), 224–270 (1994)
11. Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: Proceedings of IEEE Computer Vision and Pattern Recognition, vol. 2, pp. 506–513 (2004)
12. Ledwich, L., Williams, S.: Reduced sift features for image retrieval and indoor localization. In: Australian Conference on Robotics and Automation (2004)
13. `http://www.semanticmetadata.net/lire/`
14. Mathias, L., Chatzichristofis, S.A.: Lire: Lucene Image Retrieval – An Extensible Java CBIR Library. In: Proceedings of the 16th ACM International Conference on Multimedia, Vancouver, Canada, pp. 1085–1088 (2008)
15. Chatzichristois, S.A., Boutalis, Y.S.: Cedd: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval. In: Gasteratos, A., Vincze, M., Tsotsos, J.K. (eds.) ICVS 2008. LNCS, vol. 5008, pp. 312–322. Springer, Heidelberg (2008)