Chapter 3

Image Retrieval with Relevance Feedback

Due to the increasing popularity of digital capturing devices such as digital camera, the dramatically large size of digital contents demands for highly efficient multimedia content management. For a particular application, a content-based image retrieval (CBIR) system often has a distinct set of configurations[14] including selected image features and a processing architecture, in order to achieve the desired matching accuracy. A known approach for constructing a satisfactory CBIR system is to incorporate semantic related features for matching. However, there are no general guidelines in designing or acquiring these features; thus, many CBIR systems have been proposed to bridge the gap between image feature space and human semantics by relevance feedback. In this chapter, we will focus on the content-based image retrieval (CBIR) methods using low-level image features.

Before developing our method, we perform a few experiments on different sizes of database in Sec. 3.1. These tests are meant to verify whether the common assumptions about feature distances are always valid. The results are interesting, and they lead to the design that we do not rely on the "distance normalization". In Sec. 3.2, we briefly discuss the concept of multiple query instances (relevance feedback) and the problems in using this technique. Based on a few assumptions, we propose a straightforward yet effective method that incorporates multiple samples and image multi-scale property for estimating user intention in Sec. 3.3. When an image is selected as a negative example, we use the method described in Sec. 3.4 to prune irrelevant results. Then, the approach of generating pseudo images using multiple (spatial or SNR) scales is described in Sec. 3.5. In Sec. 3.6, we propose a CBIR architecture that integrates the proposed techniques. It solves the feature space normalization problem, and reduces the impact of insufficient user supplied information. We also provide several screen-shots to demonstrate the subjective results of our method. In Sec. 3.7, we conduct simulations to evaluate our conjectures. Base on a fairly recognized objective performance index, we compare a few different methods. In Sec. 3.8, we experiment an alternative weighting method which relaxes the Euclidean space assumption. At the end of this chapter, we summarize this presentation with Sec. 3.10.

3.1 Distance Distribution over Image Database

To determine the similarity of two given images, feature distance is an effective method. In many designs, we have to combine several kinds of feature distances to compute the final distance. A simple way to combine them is using weighted sum of all the distances. Since different feature distance function could have different range of values, we often normalize the computed distances to prevent that one distance function may overshadow the others.

A commonly adopted normalization is based on the assumption that the distance values are near-Gaussian distributed. Hence, we can easily normalize them using the mean and the variance parameters. When we compute the Euclidean distance of two feature vectors, this assumption should be valid for most cases. However, sometimes we adopt designated distance functions (such as those proposed in MPEG-7 [4]). They may be designed to match human perceptual differentiation among features.

It is known that many perceptual measurements are not linear, we wonder whether the designated distance function produces values that satisfy this assumption or not. To verify the validity of the assumption, we test several data sets:

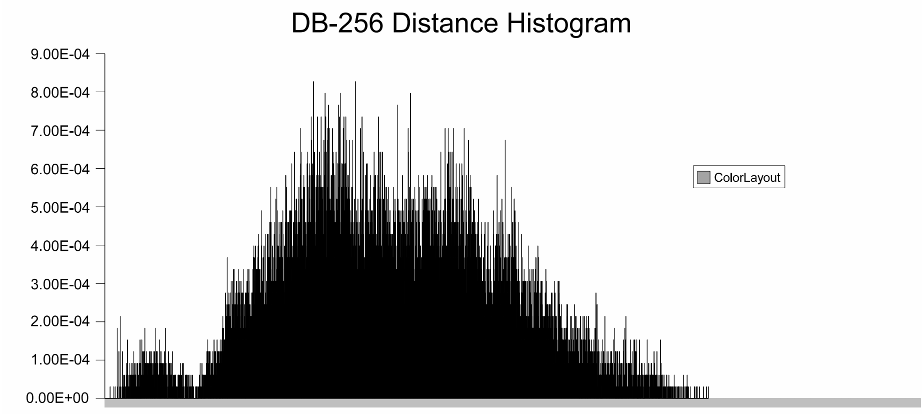
- Data set 1 (256 images): the ground-truth images used in Sec. 3.7;
- Data set 2 (794 images): 194 people (party) photos, 200 flower pictures, 200 undersea pictures, 200 outdoor scenery pictures;
- Data set 3 (17383 images): the pictures selected from the Corel gallery.

Then, we construct three sizes of image databases using the data sets:

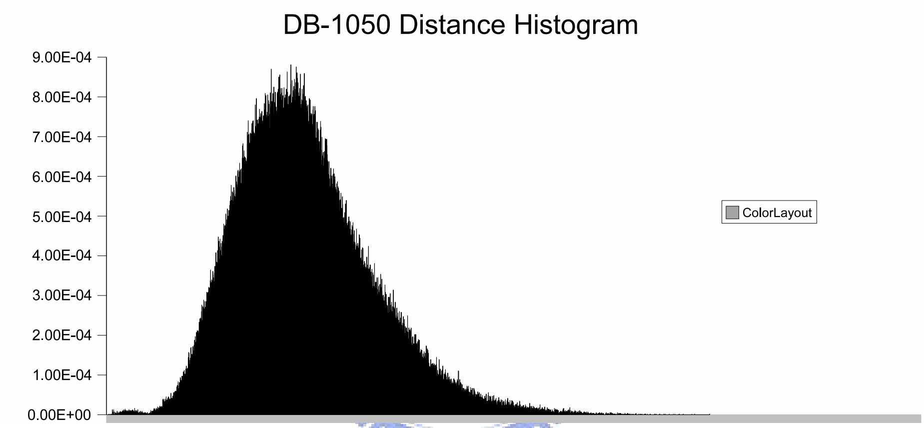
- DB-256: data set 1;
- DB-1050: data set 1 and data set 2;
- DB-18k: data set 1, data set 2, and data set 3.

The testing image features are Color Layout, Edge Histogram, and Scalable Color. All the representation feature vector and distance function are defined and proposed in the MPEG-7 visual part [24].

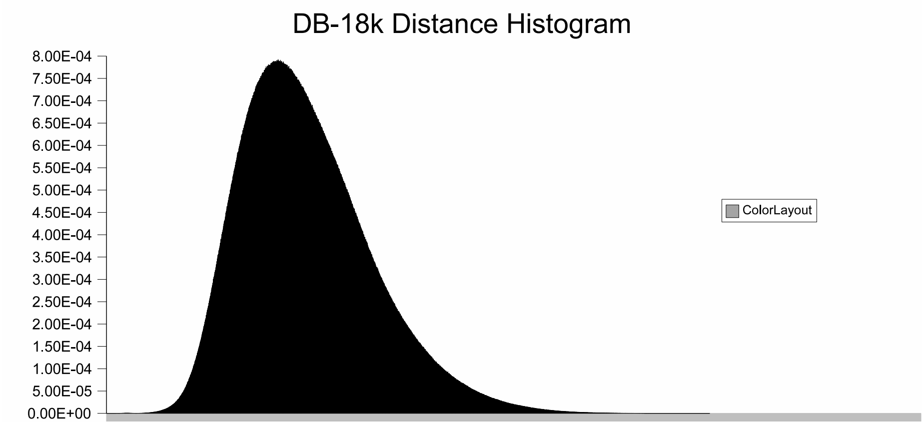
We perform the experiments as follows. From the database, compute the feature distance of each pair of the feature vectors. Then we aggregate all the values, normalize the maximum distance value to be 1.0, and plot them in a 5000-bin value histogram. Figure 3.1 shows the distance distribution of the Color Layout feature in DB-256, DB-1050, and DB-18k. Similarly, Fig. 3.2 and Fig. 3.3 show the distance distributions of the Edge Histogram feature and the Scalable Color feature, respectively. The distance distributions are often Gaussian-like. However, take the distribution of Scalable Color distances as an example, the peak value is biased to be lower end. This phenomenon indicates that *Gaussian normalization on distances may not be always valid for designated distance functions.* Another problem is the computational cost for obtaining parameters for normalization purpose. This can particularly be a difficult issue in a distributed database environment.



(a)

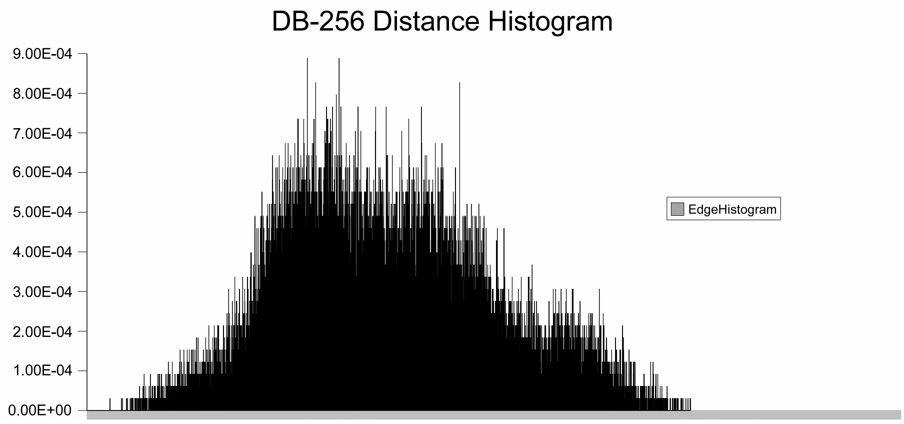


(b)

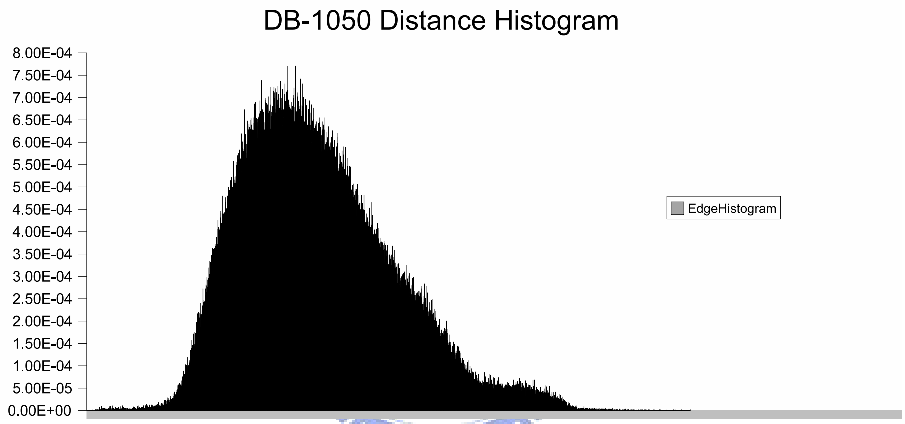


(c)

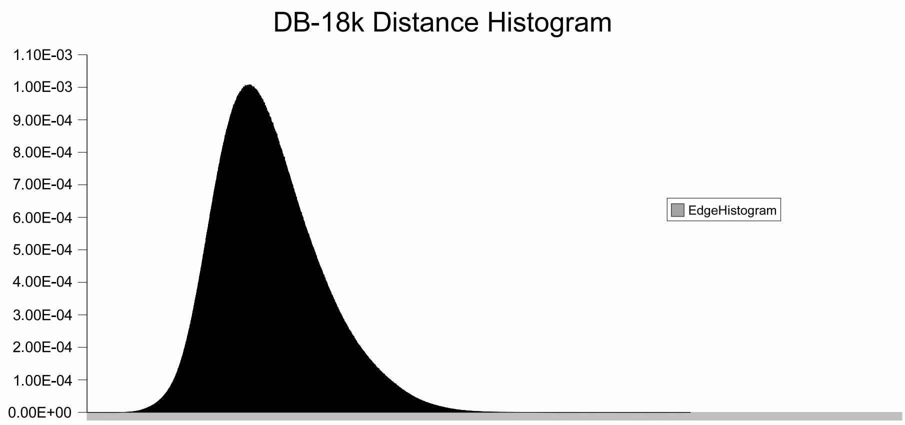
Figure 3.1: Distance distributions of Color Layout in databases. (a) DB-256 database. (b) DB-1050 database. (c) DB-18k database.



(a)



(b)



(c)

Figure 3.2: Distance distributions of Edge Histogram in databases. (a) DB-256 database. (b) DB-1050 database. (c) DB-18k database.

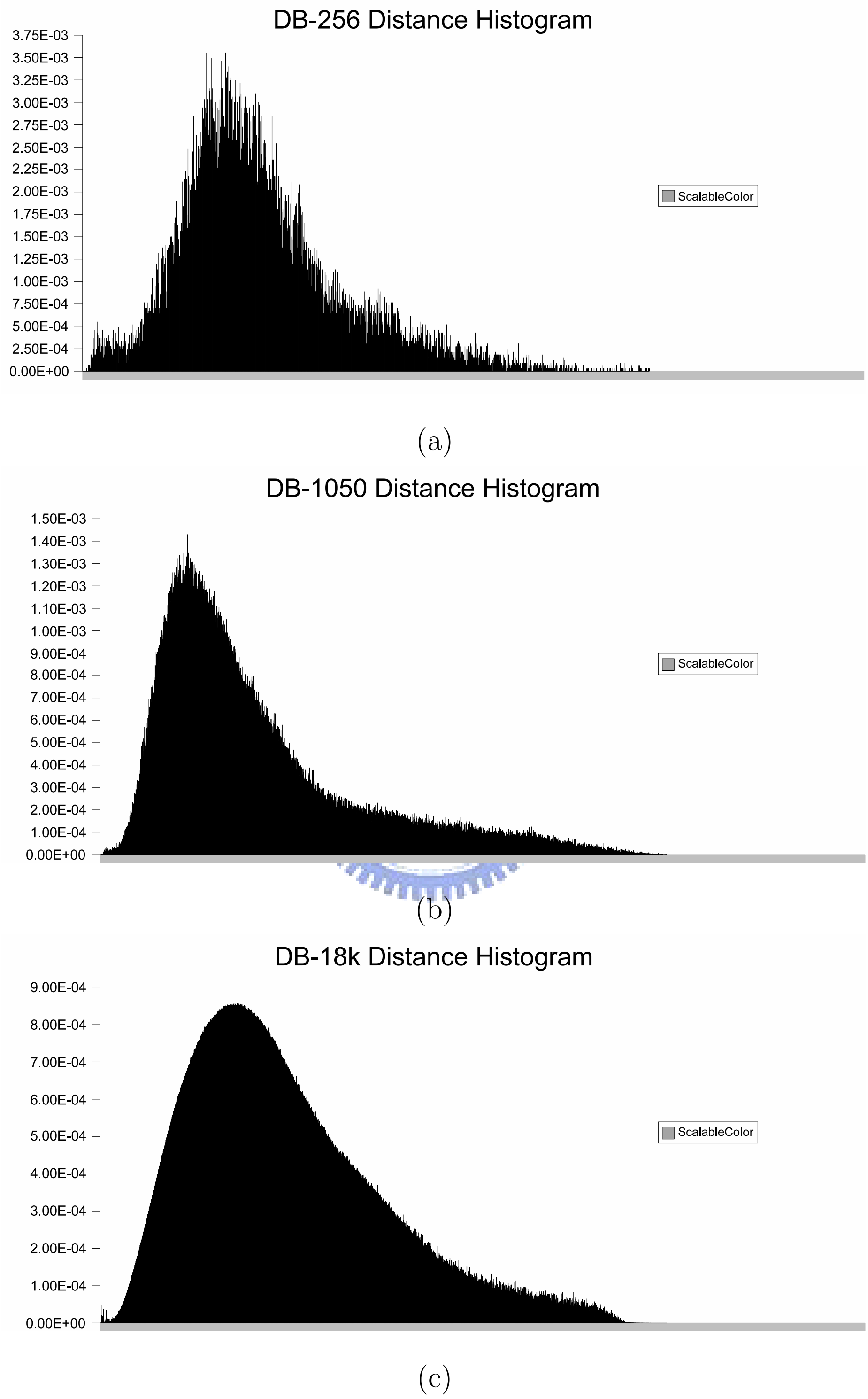


Figure 3.3: Distance distributions of Scalable Color in databases. (a) DB-256 database. (b) DB-1050 database. (c) DB-18k database.