Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 2193–2200
Expert Systems with Applications
www.elsevier.com/locate/eswa

A two-level relevance feedback mechanism for image retrieval

Pei-Cheng Cheng [a,e,*], Been-Chian Chien [b], Hao-Ren Ke [c], Wei-Pang Yang [d]

[a] Department of Computer Science, National Chiao Tung University, 1001 Ta Hsueh Rd., Hsinchu 30050, Taiwan, ROC
[b] Department of Computer Science and Information Engineering, National University of Tainan, 33, Sec. 2, Su Line St., Tainan 70005, Taiwan, ROC
[c] Institute of Information Management, National Chiao Tung University, 1001 Ta Hsueh Rd., Hsinchu 30050, Taiwan, ROC
[d] Department of Information Management, National Dong Hwa University, 1, Sec. 2, Da Hsueh Rd., Shou-Feng, Hualien 97401, Taiwan, ROC
[e] Department of Information Management, Ching Yun University, 229, Chien-Hsin Road, Jung-Li 320, Taiwan, ROC

Abstract

Content-based image retrieval (CBIR) is a group of techniques that analyzes the visual features (such as color, shape, texture) of an example image or image subregion to find similar images in an image database. Relevance feedback is often used in a CBIR system to help users express their preference and improve query results.

Traditional relevance feedback relies on positive and negative examples to reformulate the query. Furthermore, if the system employs several visual features for a query, the weight of each feature is adjusted manually by the user or system predetermined and fixed by the system. In this paper we propose a new relevance feedback model suitable for medical image retrieval. The proposed method enables the user to rank the results in relevance order. According to the ranking, the system can automatically determine the importance ranking of features, and use this ranking to automatically adjust the weight of each feature. The experimental results show that the new relevance feedback mechanism outperforms previous relevance feedback models.

Keywords: Content-based image retrieval; Relevance feedback; Image database

1. Introduction

Image capture capabilities are evolving so rapidly that extreme amount of images is produced daily. The importance of digital image retrieval techniques increases in the emerging fields of publication on the Internet, digital library, medical imaging, etc. It is a hard work to retrieve a specific image from thousands of images by browsing one by one. Attaching text annotation to images and allowing a user to query images by matching text annotation may help the retrieval of a specific image; however, attaching text annotation to images by humans is expensive and time consuming.

Content-based image retrieval (CBIR) is a promising technology to assist image finding. CBIR retrieves images by visual features inherent in images. CBIR allows the user to query an image database by image examples, partial regions of an image, or sketch contours example, etc. IBM in 1995 developed the QBIC system (Flickner et al., 1995) that allows the user to query a large image database based on visual image features such as color percentages, color layout, and textures occurring in images. The user can match colors, textures and their positions without describing them in words. CBIR offers an alternative to retrieve desired images. CBIR is more convenient and economic than annotation-based image retrieval because the visual image features of all images in database can be automatically extracted.

In the past years, CBIR has been one of the most hot research topics in computer vision. The commercial QBIC

* Corresponding author. Department of Computer Science, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 30050, Taiwan, ROC. Tel.: +886 3 4581196x7318; fax: +886 3 4683904.
E-mail addresses: pccheng@cyu.edu.tw, cpc@cis.nctu.edu.tw (P.-C. Cheng), bcchien@mail.nutn.edu.tw (B.-C. Chien), claven@lib.nctu.edu.tw (H.-R. Ke), wpyang@mail.ndhu.edu.tw (W.-P. Yang).

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.02.030

(Bartell, Cottrell, & Belew, 1995) system is definitely the most well-known system. Another commercial system for content-based image and video retrieval is Virage (Hampapur et al., 1997), which has famous commercial customers such as CNN. In the academia, systems including Candid (Cannon & Hush, 1995), Photobook (Pentland, Picard, & Sclaro, 1996), and Netra (Ma, Deng, & Manjunath, 1997) use simple color and texture features to describe image content. The Blobworld system (Bartell et al., 1995) exploits higher-level information, such as segmented objects of images, for queries. A system that is available free of charge is the GNU Image Finding Tool (GIFT) (Rocchio, 1971). A few systems are available as demonstration versions on the Web such as Viper, WIPE or Compass.

Many studies show that relevance feedback can significantly improve the effectiveness of CBIR because relevance feedback helps the system to refine the feature's weight according to user's preference. Some users may want to find images with similar colors, whereas others may want to find images with similar shapes. Relevance feedback allows the user to reflect his preference to the system, then the system can reformulate the query according to the positive and/or negative examples responded by the user. In the Spink's Spink, Greisdorf, and Bateman (1998) study show that the degree of relevance will better identify the user needs and preferences.

The similarity consideration of a user is more complex than just like or dislike. The user can point out which results are actually more relevant than others. It means that the user can offer more precise information than just positive or negative examples. The similarity degree of human is gradual and fuzzy; it is not so trivial to be categorized into just relevance or irrelevance.

In this paper we propose a two-level relevance feedback mechanism that facilitates the user to determine the preferred images and assign a relevant degree to each image. Our system offers the user a flexible environment to feedback their opinions about the results retrieved by the system. The user can rank the preferred images to create a refined query for the system. Based on the ranked images the system can predict user's preference more precisely and achieve better performance.

The application of image retrieval to general image databases has experienced limitation in success, principally due to the difficulty of quantifying image similarity for unconstrained image classes (e.g., all images on the Internet). We expect that medical imaging will be an ideal application of CBIR, because of the more limited definition of image classes, and because the meaning and interpretation of medical images is better understood and characterized. In the experiment, the medical image data was applied to evaluate the proposed relevance feedback mechanism.

This paper is organized as follows. In Section 2, we review some related relevance feedback studies. The new relevance feedback mechanism is proposed in Section 3. In Section 4, we describe the image features that we use to represent the medical images. In Section 5, we use the CasImage dataset to evaluate our proposed methods. Section 6 presents conclusion and future works of this paper.

2. Related works

Relevance feedback is a supervised learning technique for improving the effectiveness of an information retrieval system (Rocchio, 1971). For a given query, the system first retrieves a list of ranked results according to a predefined similarity metrics. Then, the user selects a set of positive and negative examples from the ranked results, and the system reformulates the query and retrieves a new list, which is expected to match the user's query goal better than the original list. The main problem is how to incorporate positive and negative examples to refine the query and how to adjust the similarity measure according to the feedback.

The original relevance feedback method, in which the vector model (Buckley & Salton, 1995; Rui & Huang, 1999) is used for document retrieval, can be illustrated by the Rocchio's formula (Rocchio, 1971) as

$$Q' = \alpha Q + \beta\left(\frac{1}{N_{R'}}\sum_{i\in D_{R'}} D_i\right) - \gamma\left(\frac{1}{N_{N'}}\sum_{i\in D_{N'}} D_i\right) \tag{1}$$

where α, β and γ are suitable constants. $N_{R'}$ and $N_{N'}$ are the number of documents in $D_{R'}$ and $D_{N'}$, respectively. That is, for a given initial query Q, and a set of relevant documents $D_{R'}$ and non-relevant documents $D_{N'}$ responded by the user, the refined query, Q', is moved toward positive examples and away from negative examples. This technique is also implemented in many content-based image retrieval systems (Ishikawa, Subramanya, & Faloutsos, 1998; Lu, Hu, Zhu, Zhang, & Yang, 2000). Experiments show that the retrieval performance can be improved considerably by using this approach.

Another method, the weighting method (Ishikawa et al., 1998; Rui & Huang, 1999), associates larger weights with more important vectors and smaller weights with less important ones. For example, (Rui & Huang, 1999) generalizes a relevance feedback framework based on low-level feature. An ideal query vector for each feature i is described by the weighted sum of all positive feedback images as

$$q_i = \frac{\pi^T Y_i}{\sum_{j=1}^{n}\pi_j} \tag{2}$$

where Y_i is the $n \times K_i$ (K_i is the length of feature i) training sample matrix for the feature i obtained by stacking the n positive feedback training vectors into a matrix. The n element vector $\pi = [\pi_1, \pi_2, \ldots, \pi_n]$ represents the degree of relevance of each of the n positive feedback images, which can be determined by the user at each feedback interaction. The system then uses q_i as the optimal query to evaluate the relevance of images in the database. This strategy is widely used by many image retrieval and relevance feedback systems (Han & Kamber, 2001; Rui & Huang, 1999).

The Bayisian estimation method has been used in many probabilistic approaches to relevance feedback. Cox,

Minka, Papathomas, and Yianilos (2000) and Vasconcelos and Lippman (1999) used Bayesian learning to incorporate user feedbacks to update the probability distribution of all images in the database. They consider the feedback examples as a sequence of independent queries and try to minimize the retrieval error by Bayesian rules. In other words, given a sequence of queries, they attempt to minimize the probability of retrieval error as

$$g(x) = \arg\max_i P(y = i|x_1, \ldots, x_t) = \arg\max_i \{P(x_t|y = i)P(y = i|x_1, \ldots, x_{t-1})\} \quad (3)$$

where $\{x_1, \ldots, x_t\}$ is a sequence of queries (feedback examples) and $P(y = i|x_1, \ldots, x_t)$ is a prior belief about the ability of the ith image class to explain the queries.

PicHunter (Cox et al., 2000) implements a probabilistic relevance feedback mechanism, which tries to predict the target image the user wants based on his actions (the images he selects as similar to the target in each iteration of a query session). A vector is used for retaining each image's probability of being the target. This vector is updated at each relevance feedback, based on the history of the session (images displayed by the system and user's actions in previous iterations). The updating formula is based on Bayes' rule. If the n database images are denoted as T_j, $j = 1, \ldots, n$, and the history of the session through iteration t is denoted as $H_t = \{D_1, A_1, D_2, A_2, \ldots, D_t, A_t\}$, with D_j and A_j being the images displayed by the system and, respectively, the action taken by the user at the iteration j, then the iterative update of the probability estimate of an image T_i being the target, given the history H_t, is

$$P(T = T_i|H_t) = P(T = T_i|D_t, A_t, H_{t-1}) = \frac{P(A_t|T = T_i, D_t, H_{t-1})P(T = T_i|H_{t-1})}{\sum_{j=1}^{n} P(A_t|T = T_j, D_t, H_{t-1})P(T = T_j|H_{t-1})} \quad (4)$$

Most current relevance feedback schemes use dichotomy relevance measurement, relevant or non-relevant. Many relevance research works indicates that users' relevance judgments exist on a continuum of relevance regions from highly relevant to lowly relevant. The criterion is based on non-binary relevance judgments that create a partial order on documents. Hence, a document is said to be superior to another one with respect to a query need when the user prefers this document to the other. The criterion they define reaches its minimum when the order created by the similarity function is the same that the order defined by the users. They show that this criterion is highly correlated to the average precision. The parameters of the retrieval system are then optimized so as to minimize this criterion. Such parameters can be weights of different similarity measures which are then linearly combined (Bartell et al., 1995), or parameters of a similarity measure (Bartell, Cottrell, & Belew, 1998). The documents with higher ranks are preferred to those with lower ranks. The target of relevance feedback algorithm is to learn the dependence between feature vectors and ranks, and predict ranks for unlabeled images.

In this paper we propose a robust relevance feedback mechanism to adjust the weighting of various features for image retrieval. In previous relevance feedback methods, a query may migrate to the mean (average) of positive examples. In this paper we propose a new relevance feedback mechanism that can detect which method is more important for user and combine the query reformulated method to refine the results.

3. Relevance feedback mechanism

CBIR uses low-level features to retrieve similar images. CBIR is more uncertain than keyword-based image retrieval about realizing human's semantic concept. Thus, a CBIR system needs to design an interface that allows the user to issue his query by giving an image example similar to the objective image. While in the process, the system keeps learning his interests until the objective image is found.

The relevance feedback mechanism attempts to extract the interests of the user from his interaction. In image retrieval, the user can determine whether an image is relevant or not at a glance; therefore, comparing with document retrieval, image retrieval is easier to interact with the user in the query process.

Previous researches allow the user to give feedback with positive examples and/or negative examples to reformulate the query. In this paper, we design a two-level relevance feedback mechanism to refine the weighting of various features according to the interests of the user. We divide the features into a logical level and a physical level. The logical level combines various methods that exploit features such as color, shape, textual, and spatial relationships to determine the relevance of images. The physical level is the vector of the feature of each method.

We propose an algorithm to judge which methods are the most suitable for the user. In the feedback process, the user ranks a sequence of relevant images in an order with respect to their similarity to the query image. Based on the ranking sequence, we can estimate how each designed method is close to user's opinion. If the feature used by one method is closer to user's opinion, then the ranking sequence generated by this method must be closer to the ranking sequence responded by the user.

If the user considers that p1 is more similar to the target image than p2, we denote p1 > p2. If the similarity degree of p1 and p2 are the same, we denote p1 = p2. (p1 > p2 > p3 > p4 > p5 > p6) is such a ranking sequence, the leftmost and rightmost of which are, respectively, the most and least similar to the target.

In the system, each feature will affect the resultant ranking sequence. We can analyze how each feature is close to the sequence responded by the user to adjust the weight of each feature. For example, suppose that the ranking sequence responded by the user is (p1 > p2 > p3 > p4 > p5 > p6). If the output sequence of the method M_1 is (p1 > p2 > p3 > p4 > p6 > p5) and that of the method M_2

is (p6 > p5 > p4 > p3 > p2 > p1), we can find that the method M_1 is closer to user's expectance than the method M_2. Therefore, reducing the weight of the feature used by M_2 and increasing the weight of the feature used by M_1 will produce a better result. Based on the above idea, the problem of evaluating the importance of each feature (and the corresponding method) becomes sequence comparison.

We employ the R_{norm} (Bollmann, Jochum, Weissmann, & Zuse, 1985) method to evaluate how close two sequences are. The R_{norm} comparison is defined as follows:

Definition 1. Let I be a finite set of images with a user-defined preference relation $\succcurlyeq$ that is complete and transitive (weak order). Let Δ^{user} be the rank ordering of I induced by the user preference relation. Also, let Δ^{system} be some rank ordering of I induced by the similarity values computed by an image retrieval system. Then R_{norm} is defined as

$$R_{\text{norm}}(\Delta^{\text{system}}, \Delta^{\text{user}}) = \frac{1}{2}\left(1 + \frac{S^+ - S^-}{S^+_{\max}}\right) \tag{5}$$

where S^+ is the number of image pairs where a better image is ranked ahead of a worse one by Δ^{system}; S^- is the number of pairs where a worse image is ranked ahead of a better one by Δ^{system}; and $S^+_{\max}$ is the maximum possible number of S^+ from Δ^{user}. It should be noted that the calculation of S^+, S^-, and $S^+_{\max}$ is based on the ranking of image pairs in Δ^{system} relative to the ranking of corresponding image pairs in Δ^{user}.

Example. Consider the following two rank orderings: $\Delta^{\text{user}} = (\text{p1}{=}\text{p4} > \text{p2}{=}\text{p3} > \text{p5})$ and $\Delta^{\text{system}} = (\text{p5} > \text{p2}{=}\text{p4} > \text{p1}{=}\text{p3})$. According to the user, p1 and p4 have the highest preference, followed by p2 and p3 at the next level of preference, followed by p5 at the lowest level of preference. The user considers that p1 is equivalent to p4 and p2 equivalent to p3. Δ^{system} is interpreted in a similar manner. Here we have, $S^+_{\max} = \{(\text{p1},\text{p2}),(\text{p1},\text{p3}),(\text{p1},\text{p5}),(\text{p4},\text{p2}),(\text{p4},\text{p3}),(\text{p4},\text{p5}),(\text{p2},\text{p5}),(\text{p3},\text{p5})\} = 8$, $S^+ = \{(\text{p4,p3})\} = 1$, $S^- = \{(\text{p5},\text{p2}),(\text{p5},\text{p4}),(\text{p5},\text{p1}),(\text{p5},\text{p3}),(\text{p2},\text{p1})\} = 5$. Therefore, $R_{\text{norm}} = 1/2(1 + (1 - 5)/8) = 0.25$.

R_{norm} values range from 0 to 1 and a value of 1 indicates that the system's rank ordering is the same as that provided by the user. A value closer to 1 is better than a value closer to 0.

R_{norm} represents the weight of each feature that the user pays attention to. Assume that there are n features $(f_1, f_2, \ldots, f_n)$ used by an image retrieval system and the weights of features are $(w_1, w_2, \ldots, w_n)$. After the user feedbacks the ranked result to the system, we can estimate the R_{norm} for each feature $(r_1, r_2, \ldots, r_n)$. Then we define the new weight of each feature as

$$w_i = \frac{r_i}{\sum_{j=1}^{n} r_j} \tag{6}$$

In the logical layer, the system then uses the new weight of diverse features to re-rank the results. This mechanism allows the exploitation of any types of features (image features or textual features) and is more flexible and robust than previous researches.

The second level attempts to decide the importance vectors of single feature. Dependent on the representation of features, different weight tuning methods can be used. We use the Rocchio's formula to reformulate the query vector in probability model representation. The color histogram representation is a probability model representation. The vector space records the probability of occurrence of each color. It is easy to realize that the user will pick up the colors more interesting to him in the query. In the moment model representation, the vector space records the value computed by predefined formulas, such as the mean value, and the standard variance. A document whose vector is closer to the query vector will be better. The scales of mean value and standard variance are different; as a result, we cannot judge which of two vectors is more important just by their values. In this case the user will prefer to adjust the weight of each vector, but it is hard to adjust the weight of vectors directly by user.

The relevance feedback method we propose does not need to focus on the real values of vectors. The R_{norm} method can easily evaluate which method or vector is more important by the ranking sequence. It is very flexible to apply to relevance feedback of different types of features. In the next section, we describe the features we use for medical image retrieval. The types of features used in our system are quite different, and they have been shown excellent performance in medical image retrieval (Cheng, Chien, Ke, & Yang, 2004).

4. Medical image features

An image consists of a large amount of pixels. In order to efficiently retrieve images relevant to a query, a CBIR system usually extracts low-level image features to represent an image in an off-line preprocessing stage. Image features can be categorized into color, shape, texture and spatial relationships. In this section, we design four features based on human's viewpoint to capture a medical image's color, shape and spatial relationships. They are *Color histogram*, *Gray Level Histogram*, *Semantic Moment*, and *Shape Correlogram*. This section describes these features in detail. The proposed features reduce semantic gap effectively and have excellent result in medical image retrieval task of ImageCLEF 2004 (Cheng et al., 2004).

4.1. Color histogram

Color histogram defines the similarity degree between color bins by a mechanism corresponding to human's perception. Color histogram (Swain & Ballard, 1991) is a basic method for representing image content and has good performance. The color histogram method gathers statistics about the proportion of each color as the signature of an image. Let I be an image that consists of pixels $p(x, y)$,

and C be a set of colors $\{c_1, c_2, \ldots, c_m\}$ that can appear in an image. The color histogram $H(I)$ of the image I is a vector $(h_1, h_2, \ldots, h_i, \ldots, h_m)$, in which each bucket h_i counts the ratio of pixels of color c_i in I. Suppose that p is the color level of a pixel. Then the histogram of I for color c_i is defined as Eq. (7)

$$h_{c_i}(I) = \Pr_{p \in I}\{p \in c_i\} \tag{7}$$

In other words, $h_{c_i}(I)$ corresponds to the probability of any pixel in I being of the color c_i. For evaluating the similarity between two images I and I', we can calculate the distance between the histograms of I and I' by using a standard method (such as the L_1 distance or L_2 distance). The image in the image database most similar to a query image I is the one having the smallest histogram distance with I.

The colors of an image are represented in the HSV (Hue, Saturation, and Value) space, which is closer to human perception than spaces such as RGB (Red, Green, and Blue) or CMY (Cyan, Magenta, and Yellow). In implementation, we quantize HSV space into 18 hues, two saturations, and four values, with additional four levels of gray values; as a result, there are a total of 148 bins.

4.2. Gray level histogram

Gray level histogram concentrating on the contrast of medical images avoids the effect of different parameters caused by the environment creating images. Gray images are different from color images in human's perception. Gray images have fewer colors than color images, only 256 gray levels in each gray image. Human's visual perception is influenced by the contrast of an image. The contrast of an image from the viewpoint of human is relative rather than absolute. To emphasize the contrast of an image and handle images with less illuminative influence, we normalize the value of pixels before quantization. In this paper we propose a relative normalization method. First, we cluster the whole image into four clusters by the K-means cluster method (Han & Kamber, 2001). We sort the four clusters in ascendant order according to their mean values. We shift the mean of the first cluster to value 50 and that of the fourth cluster to value 200; then each pixel in a cluster is multiplied by a relative weight to normalize. Let m_{c1} is the mean value of cluster 1 and m_{c4} is the mean value of cluster 4. The normalization formula of pixel $p(x,y)$ is defined as Eq. (8).

$$p(x,y)_{\text{normal}} = (p(x,y) - (m_{c1} - 50)) \times \frac{200}{(m_{c4} - m_{c1})} \tag{8}$$

After normalization, we resize each image into $128 * 128$ pixels, and use one-level wavelet with Haar wavelet function to generate the low frequency and high frequency sub-images. Processing an image using the low-pass filter will obtain an image more consistent than the original one; on the contrary, processing an image using the high-pass filter will obtain an image that has high variation. The high-frequency part keeps the contour of the image.

In a gray image, especially a medical image, the spatial relationship is very important. Medical images always contain particular anatomic regions (lung, liver, head, and so on); therefore, similar images have similar spatial structures. We add spatial information into the histogram so we call this representation as *gray level histogram* in order to distinguish from color histogram. We use the LL band for gray-spatial histogram and coherence analysis. To get the gray-spatial histogram, we divide the LL-band image into nine areas. The gray values are quantized into 16 levels for computational efficiency. The gray-spatial feature estimates the probability of each gray level that appears in a particular area. The gray-spatial histogram of an image has a total of 144 bins.

4.3. Semantic moment

Semantic moment records invariable moment of image rotation and zooming from human's viewpoint. One of the problems to design an image representation is the semantic gap. The state-of-the-art technology still cannot reliably identify objects. The Semantic moment feature attempts to describe the features from the human's viewpoint in order to reduce the semantic gap.

We cluster the pixels in an image into four classes by the K-means algorithm. For each class, we calculate the number of pixels (COH_κ), mean value of gray values (COH_μ) and standard variance of gray values (COH_ρ). Furthermore, for each class, we group connected pixels in the eight directions as an object. If an object is bigger than 5% of the whole image, we denote it as a big object; otherwise it is a small object. We count how many big objects (COH_o) and small objects (COH_ν) in each class, and use COH_o and COH_ν as parts of image features.

Because we intend to know the reciprocal effects among classes, we smooth the original image. If two images are similar, they will also be similar after smoothing. If their spatial distributions are quite different, they may have different results after smoothing. After smoothing, we cluster an image into four classes and calculate the number of big objects (COH_τ) and small objects (COH_ω). Each pixel will be influenced by its neighboring pixels. Two close objects of the same class may be merged into one object. Then, we can analyze the variation between the two images before and after smoothing. The semantic moment of each class is a seven-feature vector, (COH_κ, COH_μ, COH_ρ, COH_o, COH_ν, COH_τ, COH_ω). The semantic moment of an image is total 28-feature vector that an image contains four classes.

4.4. Shape correlogram

Shape correlogram is designed for solving the problem of partial shape match. The contour of a medical image contains rich information. A broken bone in the contour

may be different from the healthy one. Thus we choose a representation that can estimate the partial similarity between two images and can be used to easily calculate their global similarity.

We analyze the high-frequency part by our modified correlogram algorithm. The correlogram (Huang et al., 1997) is defined as Eq. (9). Let D denote a set of fixed distances $\{d_1, d_2, d_3, \ldots, d_n\}$. The correlogram of an image I is defined as the probability of a color pair (c_i, c_j) at a distance d.

$$\gamma^d_{c_i,c_j}(I) = \Pr_{p_1 \in c_i, p_2 \in I}\{p_2 \in c_j || p_1 - p_2| = d\} \tag{9}$$

For computational efficiency, the auto-correlogram is defined as

$$\lambda^d_{c_i}(I) = \Pr_{p_1 \in c_i, p_2 \in I}\{p_2 \in c_i || p_1 - p_2| = d\} \tag{10}$$

The contrast of a gray image dominates human's perception. If two images have different gray levels they still may be visually similar. Thus the coorelogram method cannot be used directly.

Our modified correlogram algorithm works as follows. First we sort the pixels of the high-frequency part in descendant order. Then we order the results of the preceding sorting by the ascendant distances of pixels to the center of the image. The distance of a pixel to the image center is measured by the $L2$ distance. After sorting by gray value and distance to the image center, we select the top 20 percent of pixels and the gray values higher than a threshold to estimate the auto-correlogram histogram. We set the threshold zero in this task. For any two pixels having a spatial relationship, we estimate the probability that the distance falls within an interval. The distance intervals we set are $\{[0,2], [2,4], [4,6], [6,8], [8,12], [12,16], [16,26], [26,36], [36,46], [46,56], [56,66], [76,100]\}$. The high-frequency part comprises $64 * 64$ pixels, thus the maximum distance will be smaller than 100. The first n pixels will have $n * (n+1)/2$ numbers of distances. We calculate the probability of each interval to form the vector of the shape correlogram.

5. The user interface

We design a graphic user interface to show how the new feedback model can be integrated into a content-based image retrieval system. Previous relevance feedback mechanisms only offer the user to choose positive or negative examples. Giving too few positive examples distorts the result; on the other hand, giving too many negative examples will confuse the system. The reason is that all positive examples are alike in a way; but each negative example is negative in its own way. Our proposed model allows the user to provide more information in the feedback phase. With the same number of judged examples we can get more information in our graphic user interface. In this manner, the iterations of feedback processes can be reduced.

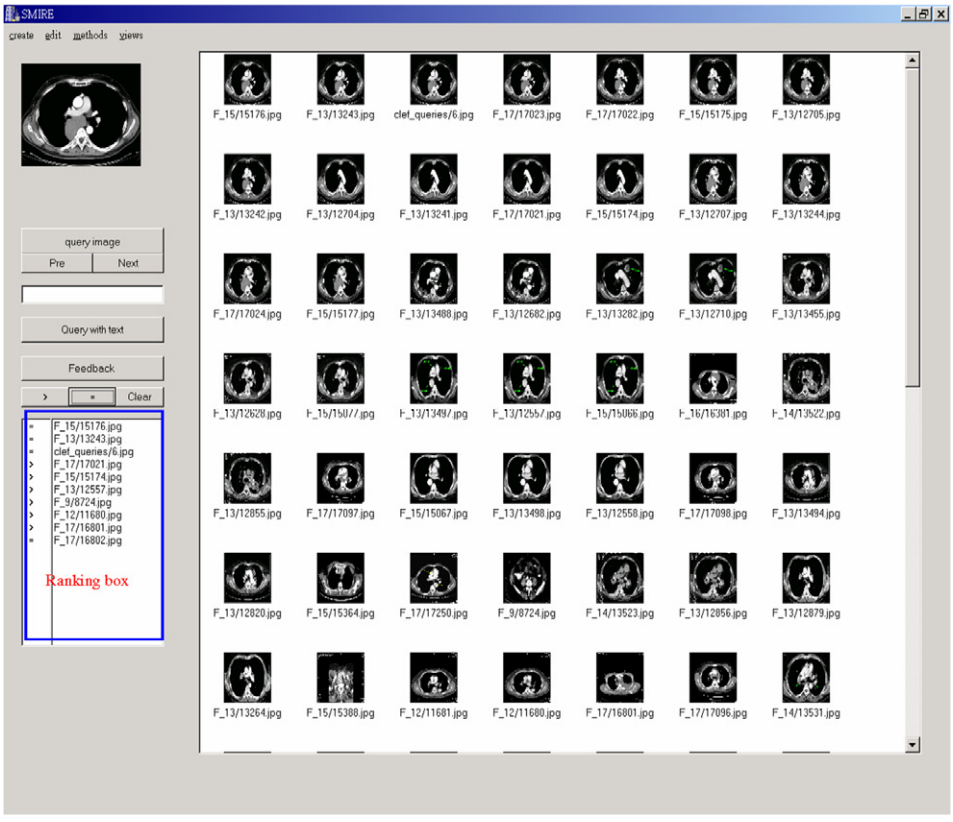

Fig. 1. Graphic user interface for the proposed CBIR system.

We define a new mechanism for the user to weight various features based on his interests. According to the result of the system, user can re-rank his preferred priority. It is inconvenient for the user to give each image a value of similarity degree. The user is usually difficult in defining a value about similarity, but the user can distinguish which images are more similar than the others. We develop a friendly user interface for the user to easily express his intention. Fig. 1 is the graphic user interface of our system. The user can click the resultant images and put it into the ranking box. The priority is reduced gradually by following ">" symbols.

As shown in Fig. 1, the top-left image is the query image. We can specify the query image from a file or the right window. The user first queries a medical image by example and obtains the list of resultant similar medical images. From the similar resultant images, the user picks up the most similar images into the ranking box. The user can use the ">" and "=" buttons to adjust the priority. The symbol ">" means that the preceding image is more important than the following image. The symbol "=" means that the importance of the preceding image is equal to the following image. This graphic user interface allows the user to easily list preferred ranking result. The system then exploits the list in the rank box to evaluate the weight of features and refine the query by the method proposed in Section 3.

6. Experiments

Although many content-based image retrieval methods have been proposed, there are few benchmarks for evaluation. In the ImageCLEF 2004 forum (Clough, Sanderson, & Müller, 2004), a forum for comparing the performance of content-based image retrieval methods is first proposed. The ImageCLEF 2004 forum contains 9916 medical images for evaluation. In this paper we follow the ImageCLEF 2004 evaluation to evaluate the performance of the feedback mechanism. The process of evaluation and the format of results employ the trec_eval tool (Clough et al., 2004). There are 26 queries for test. The corresponding answer images of each query were judged as either relevant or partially relevant by at least two assessors.

We conduct three experiments. *Color histogram*, *gray level histogram*, *semantic moment*, and *shape correlogram* are the four features for retrieving similar medical images. To show that the proposed relevance feedback mechanism is very flexible, the types of image features we use are quite different. The first experiment, called BASE, uses the visual features of the query example to query the database without relevance feedback. The comparison has been done with the method by Rui and Huang (1999), called RUI, that associates larger weights with more important dimensions and smaller weights with less important ones. This method generalizes a relevance feedback framework of the physical features based on positive feedback examples. We normalize different concept features by Gaussian normalization, and the weights of concept features are equal. Another ranked-based method for comparison has been done by B. Bartell in Ref. Bartell et al. (1998), denoted as BT.

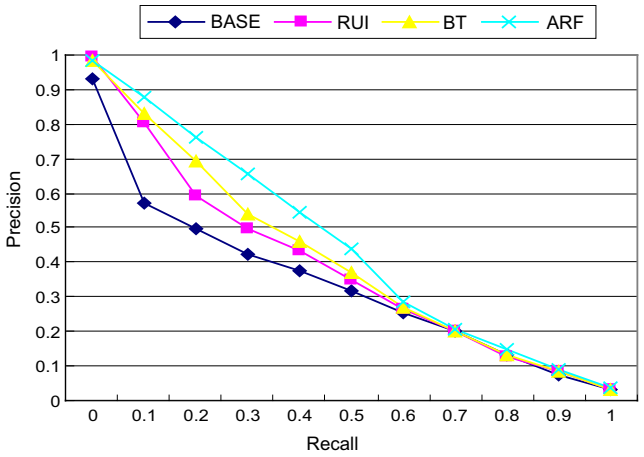

Fig. 2. The precision vs. recall graphs of average 26 queries.

The experiment, denoted as ARF (adaptive relevance feedback), is the result that uses the proposed feedback mechanism. The system integrates the four features by Gaussian normalization in the first run. While the second run, we adjust the weight of concept features by the R_{norm} method. In the physical level, the query of **color histogram** and *gray level histogram* features are reformulated by the method proposed in Rui and Huang (1999). The weight of *Coherence moment* and *Shape correlogram* features are tuned by the R_{norm} method. The test result shows that the feedback mechanisms (RUI and ARF) have better result than the mechanism without relevance feedback (BASE). While the user feedbacks its interests to the system, the proposed method (ARF) is more precise and quicker to reach user's requirement. Fig. 2 shows the precision and recall graphs. RUI, BT and ARF curves are the result after conducting relevance feedback three times.

The mean average precision of BASE is 0.3273. The mean average precision of RUI is 0.3884. The mean average precision of BT is 0.412. The mean average precision of ARF is 0.4412. Table 1 is the mean average precision and relevance feedback iterations. As shown in Table 1, the ARF method reaches the user's interests faster than the RUI method.

The experimental result shows that the proposed feedback method can be used for integrating arbitrary concept

Table 1
The mean average precision at *n*-iteration relevance feedback

	Iterations			
	0	1	2	3
RUI	0.327	0.367	0.374	0.388
BT	0.327	0.388	0.403	0.412
ARF	0.327	0.401	0.432	0.441

features. We can estimate which features are more important although the scales of features are different.

7. Conclusion

In this paper we develop a new relevance feedback mechanism to improve content-based image retrieval. The two-level feature modulation mechanism according to user's interests enhances the result significantly. Uniform and equal calibration of features is easy to adjust the feature's weight, but some features are not so trivial. The proposed method can treat various types of features in the concept level and is more robust than previous works.

It is easy to integrate our feedback mechanism into existent content-based image retrieval methods. Furthermore, the feedback mechanism can be applied to CBIR applications other than medical images. In the future, we will use the feedback mechanism to combine visual feature and textual features.

Acknowledgements

This work was supported by the National Science Council (Grant Number: NSC-95-2221-E-259-044). Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors only and do not necessarily reflect the views of the National Science Council.

References

Bartell, B., Cottrell, G. W., & Belew, R. (1995). Learning to retrieve information. In *Current trends inconnectionism: proceedings of the swedish conference on connectionism*. Hillsdale: Lea.

Bartell, B., Cottrell, G. W., & Belew, R. (1998). Optimizing similarity using multi-query relevance feedback. *Journal of the American Society for Information Science, 49*(8), 742–761.

Bollmann, P., Jochum, F., Weissmann, V., & Zuse, H. (1985). The live-project-retrieval experiments based on evaluation viewpoints. In *Proceedings of the 8th annual international ACM/SIGIR conference on research and development in information retrieval*. New York (pp. 213–214).

Buckley, C., & Salton, G. (1995). Optimization of relevance feedback weights. In *Proc. SIGIR'95*.

Cannon, P. M., & Hush, D. R. (1995). Query by image example: the CANDID approach. In: Niblack, W., Jain, R. C. (Eds.), *Storage and retrieval for image and video databases III*, *SPIE proceedings* (vol. 2420, pp. 238–248).

Cheng, P. C., Chien, B. C., Ke, H. R., & Yang, W. P. (2004). KIDS's evaluation in medical image retrieval task at ImageCLEF 2004. Working Notes for the CLEF 2004 Workshop September, Bath, UK.

Clough, P., Sanderson, M., & Müller, H. (2004). The CLEF cross language image retrieval track. In: *Working Notes of the CLEF 2004 Workshop*.

Cox, I. J., Minka, T. P., Papathomas, T. V., & Yianilos, P. N. (2000). The Bayesian image retrieval system, pichunter: theory, implementation, and psychophysical experiments. *IEEE Transactions on the Image Processing*, Special issue on digital libraries.

Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., et al. (1995). Query by image and video content: the QBIC system. *IEEE Computer, 28*(9), 23–32.

Hampapur, A., Gupta, A., Horowitz, B., Shu, C.-F., Fuller, C., Bach, J. et al. (1997). Virage video engine. In: I.K. Sethi, & R.C. Jain, (Eds.), *Storage and retrieval for image and video databases V, SPIE proceedings* (vol. 3022, pp. 352–360).

Han, J., & Kamber, M. (2001). *Data mining: concepts and techniques*. Academic press, San Diego, CA, USA.

Huang, J., Kumar, S. R., Mitra, M., Zhu, W. J., & Zabih, R. (1997). Image indexing using color correlograms. In *Proceedings of IEEE conference on computer vision and pattern recognition*, San Juan, Puerto Rico.

Ishikawa, Y., Subramanya, R., & Faloutsos, C. (1998). Mindreader: query databases through multiple examples. In *Proceedings of the 24th VLDB conference*, New York.

Lu, Y., Hu, C., Zhu, X., Zhang, H., & Yang, Q. (2000). A unified framework for semantics and feature based relevance feedback in image retrieval systems. In *Proceedings of the 8th ACM Multimedia International Conference*, Los Angeles, CA.

Ma, W. Y., Deng, Y., & Manjunath, B. S. (1997). Tools for texture- and color-based search of images. In B.E. Rogowitz, T.N. Pappas (Eds.), *Human vision and electronic imaging II*, *SPIE Proceedings*, San Jose, CA (vol. 3016, pp. 496–507).

Pentland, A., Picard, R. W., & Sclaro, S. (1996). Photobook: tools for content-based manipulation of image databases. *International Journal of Computer Vision, 18*(3), 233–254.

Rocchio, J. J. (1971). Relevance feedback in information retrieval. In *The SMART retrieval system: experiments in automatic document processing* (pp. 313–323).

Rui, Y., & Huang, T. S. (1999). Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Circuits Systems and Video Technology, 8*(5).

Spink, A., Greisdorf, H., & Bateman, J. (1998). From highly relevant to nonrelevant: Examining different regions of relevance. *Information Processing and Management, 34*(5), 599–622.

Swain, M. J., & Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision, 7*, 11–32.

Vasconcelos, N., & Lippman, A. (1999). Learning from user feedback in image retrieval systems. In *Proceedings of NIPS'99*, Denver, CO.