IEICE TRANS. INF. & SYST., VOL.E89–D, NO.5 MAY 2006

PAPER

A Relevance Feedback Image Retrieval Scheme Using Multi-Instance and Pseudo Image Concepts

Feng-Cheng CHANG[†a)], *Nonmember and* **Hsueh-Ming HANG**[†b)], *Member*

SUMMARY Content-based image search has long been considered a difficult task. Making correct conjectures on the user intention (perception) based on the query images is a critical step in the content-based search. One key concept in this paper is how we find the user preferred low-level image characteristics from the multiple positive samples provided by the user. The second key concept is how we generate a set of consistent "pseudo images" when the user does not provide a sufficient number of samples. The notion of image feature stability is thus introduced. The third key concept is how we use negative images as pruning criterion. In realizing the preceding concepts, an image search scheme is developed using the weighted low-level image features. At the end, quantitative simulation results are used to show the effectiveness of these concepts.

key words: *image retrieval, perception weighting, relevance feedback*

1. Introduction

Due to the increasing popularity of digital capturing devices such as digital camera, the dramatically large size of digital contents demands highly efficient multimedia content management. For a particular application, in order to achieve the desired matching accuracy, a content-based image retrieval (CBIR) system often has a distinct set of configurations [1] including selected image features and a processing architecture. A known approach for constructing a satisfactory CBIR system is to incorporate semantic related features for matching. Researches on CBIR topics show that semantic features are critical in boosting the query accuracy. These high-level features can be either extracted at the stage of content analysis, or acquired at run-time. The former is used to provide semantic content descriptions and often requires sophisticated feature extraction processes, such as object segmentation or textual descriptions. The latter is often used to capture the user preferences, and often tries to derive the information from run-time input sources, for example, the user-provided query images.

Features can be roughly categorized into two groups according to their lifetime. The first group is used to describe the static information of an image, such as the color and the textual description; the second group is used to describe the run-time properties of a content-based processing system, such as the user preferences and the content distribution in a database. Although these two kinds of features represent different aspects of a CBIR problem, they are often coupled. A CBIR application is often designed for a specific problem domain, and only processes a limit number of static features. Based on these selected features, it incorporates means to conjecture about user preferences for each query. There are no general rules in acquiring user preferences; thus, many CBIR systems have been proposed to interact with users by using relevance feedback. A typical Query-by-Example (QBE) CBIR system with relevance feedback generally analyzes the user query images and/or relevant feedback images to derive the necessary search parameters. The search parameters are often defined in terms of the image features pre-chosen in the system. Then, the system searches the database and returns a list of the top-N similar images for further feedback actions. This process can be repeated and hopefully it will eventually produce satisfactory results to that particular user and query. If we treat the relevance feedback function as a means to obtain multiple user inputs, the system can be generalized to be an architecture which:

- acquires multiple query instances (positive ones and negative ones).
- analyzes the collected instances based on the pre-chosen features.
- makes a "guess" on the user intention, represented by using the pre-chosen features.
- performs the search according to the derived user intention.

In addition to the user preference issue, we often expect that a CBIR application can be efficiently implemented. Many of the conventional CBIR methods adopt classification techniques, and assume that some global distribution parameters are available without penalty. For example, the MARS [2] scheme needs the mean and the variance values in normalizing distances. However, when an application is applied to a very large database or a collection of distributed databases, it not only faces the computational complexity problem, but also it needs to solve the frequent database update problem. Thus, updating the database distribution parameters can be a costly task. Although the terms, positive and negative queries, appear in [3], they are referred to (image) regions that compose an image. Image similarity is decided by a number of set operations on the pre-classified (region) feature clusters. The approach in [3] is very different from the approach in this paper.

In this paper, we are interested in estimating low-

Manuscript received February 24, 2005.
Manuscript revised October 28, 2005.

[†]The authors are with Department of Electronics Engineering, National Chiao Tung University (NCTU), 1001 Ta-Hsueh Road, Hsinchu, Taiwan 30010, R.O.C.
a) E-mail: fcchang.ee88g@nctu.edu.tw
b) E-mail: hmhang@mail.nctu.edu.tw

DOI: 10.1093/ietisy/e89–d.5.1720

level user perception from multiple query images on-the-fly. Thus, we will focus on the content-based image retrieval (CBIR) methods using only low-level image features. In designing the user-intention estimation algorithm, we try to keep the algorithm independent of the distribution parameters to reduce the computational cost. As shown in Sect. 4, it may degrade the accuracy in some cases. However, it is a trade-off between accuracy and scalability.

Our paper is organized as follows. In Sect. 2, we briefly discuss the concept of multiple query instances (relevance feedback) and the problems in using this technique. In Sect. 3, 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. Also in this section, we propose a method to deal with the negative samples. In Sect. 4, we conduct simulations to evaluate our conjectures. Base on a fairly recognized objective performance index, we compare a few different methods. At the end, we conclude this presentation with Sect. 5.

2. Problem and Design Target

As described in the previous section, relevance feedback is a method to acquire multiple user-provided query images. The problem is how one utilizes multiple image features and multiple query instances (images) to derive the suitable parameter values for searching purpose. Multiple features and multiple instances represent two different aspects. The former is how we describe an image in an application; the latter is how we guess the user intention using the given instances. There exist many proposals on combining multiple features for image search such as using Borda counts [4]. Methods of combining multiple instances are usually considered as a part of a relevance feedback operation. There are several existing CBIR proposals containing relevance feedback such as MARS [2], [5] and iPURE [6]. Typically, the multi-instance analysis process uses the pre-selected features. Since feature selection is a design-time issue, the analysis method varies from application to application. For example, if the features are expressed as a vector of moments, the weighting factors for each moment can be computed by the boosting method [7], as described in [8] and [9].

In our previous project, we developed an MPEG-7 testbed [10] and thus have used it to examine several low-level MPEG-7 features. We observed that subjectively similar pictures tend to be close (near) in one or more feature spaces. Another observation is that a low-level feature often has (somewhat) different values when it is extracted from the same picture with different spatial resolutions and/or picture quality (SNR scalability). Our investigation finds that people often design a QBE system with feedback under the assumption that a sufficient number of query instances or feedback iterations can be provided by the user. However, this assumption is not always true in a real-world application [11], [12]. Often, the sample size is very small (one to three) and the information contained in the samples may not be all consistent in the sense of data clustering.

Based on our observations, we are motivated to develop a user perception estimation algorithm, which tries to make a correct conjecture on the user intention based on a small number of samples (instances) provided by the user. For simplicity and fast calculation, our system uses only low-level features for high-volume feature extraction and matching. Another consideration is that it incorporates only simple distance-based weighting and matching scheme to make it easily integrated into various application scenarios. For a large database (especially a collection of distributed databases), calculating global distribution parameters often accompanied with some penalty. In our design, several elements are different from the previous works: the first is that our system is independent of global distribution parameters and thus it is suitable for large database; the second is that we treat the negative query images in a special way as described later; the third is that we adopt a simple user interface for relevance feedback. Our system only asks users to label positive or negative images without restriction on their numbers.

3. Proposed Weighting Method

In the following discussions, we focus on a geometrical approach that combines multiple low-level features together to form a "good" similarity function for retrieving similar images. We first describe the feature weights produced by multiple instances (query set) in Sect. 3.1. When an image is selected as a negative example, we use the method described in Sect. 3.2 to prune irrelevant results. Then, the approach of generating pseudo images using multiple (spatial or SNR) scales is described in Sect. 3.3. In Sect. 3.4, we propose a CBIR architecture that uses the multi-instance and pseudo image concepts. It solves the feature space normalization problem, and reduces the impact of insufficient user supplied information. In this section, we also provide several screen-shots to demonstrate the subjective results of our method. More detailed simulations with objective metrics are provided in Sect. 4.

3.1 User Perception Estimation

There are several ways to combine different low-level features. Here we adopt a straightforward one: weighted sum of feature distances. We use the originally designated distance definition of individual features. Our focus is to find the most appropriate weight of each feature to produce an effective combined distance measure. Thus, our method preserves the individual feature space properties. In this scheme, the user perception is expressed by a weighting vector. Note that the weighting vector is derived from the multiple instances provided by the user.

Similar to many other image retrieval schemes, we assume the following conditions are satisfied:

- All the basic feature distance metrics have finite values

$[0, \infty)$, and a zero-distance means the two features are equivalent.

- Two perceptually similar images have a small distance in at least one feature space.
- Low-level features are locally inferable [13]. That is, if all the feature values of two images are fairly close, then the two images are perceptually similar.

In addition to the above assumptions, we add another conjecture: if two images have a large distance value in a specific feature space, we cannot determine the perceptual similarity of them based merely on this feature. Note that this feature space is simply irrelevant to our perception. It does not necessarily decide dissimilarity in perception.

Different from several well-known CBIR systems, our system does not rely on *a priori* feature distributions. These distributions may help to optimize inter-feature normalization, as in MARS [2], to produce better matching performance. However, they often introduce overheads and require high computation. Even if feature distributions are available, they may not lead to appropriate normalization. More importantly, user perceptions do not necessarily match the feature distributions in the database. Thus, we try to design our method to be independent of feature distributions as shown below. The need of normalization is eliminated because of the way we define distance function.

In summary, our feature weighting and combination principle is: *given two user-input query images, if they are farther apart in a certain feature space, this feature is less important in deciding the perceptual similarity for this particular query.* Suppose we have a query image set with n samples, $Q = \{q_i \mid i = 1..n\}$, and an available basic feature set $F = \{F_j \mid j = 1..m\}$. Let f_{ij} denote the value of the j-th feature (F_j) for the i-th image (q_i). The normalized distance function for feature F_j is $d_j(f_{1j}, f_{2j}) = n_j * D_j(f_{1j}, f_{2j})$, where $D_j(f_{1j}, f_{2j})$ is the designated distance function for F_j, and n_j is the normalization factor for F_j, which sets the normalized value $d_j(f_{1j}, f_{2j})$ in the range of $[0, 1]$. Though n_j is an *a priori* information, we will see that it can be safely discarded at the end of this section.

To measure the sparseness of a feature point set, we assume all the feature distances satisfy the properties of an Euclidean space, for example, the triangular inequality. We will also suggest an alternative definition of sparseness in Sect. 4.4, which relaxes the assumption of Euclidean space and produces comparable results. Based on the finite distance assumption, there exists at least one hyper-sphere, inside which all the feature points are located. A hyper-sphere can be defined by a "pivot" (centroid) and a radius. In the following description, $r_j^{(k)}$ is the radius of the k-th hyper-sphere in the F_j space. Among all possible spheres in the F_j space, we call the smallest one as the bounding sphere, and its radius is defined as the scatter number in this space.

Based on the above discussion, we define the scatter number (s_j) of Q for feature F_j as follows:

$$s_j = \begin{cases} 1, & \text{if } \mid Q \mid = 1 \quad (1) \\ \frac{1}{2} d_j(f_{1j}, f_{2j}), & \text{if } \mid Q \mid = 2 \quad (2) \\ \max_{\forall k} r_j^{(k)}, & \text{if } \mid Q \mid \geq 3 \quad (3) \end{cases},$$

where $\mid Q \mid$ is the size of set Q.

In condition (1), because we do not have enough information to determine the scatter of each feature, we simply assign a default value (= 1) to s_j. In case (2), we only have two query samples $Q = \{q_1, q_2\}$. Thus, the minimal bounding radius is half of the distance between them. In case (3), we have more than two query instances, and we can derive the bounding radius using geometry theorems. Let $Q^{(k)} = \{q_1^{(k)}, q_2^{(k)}, q_3^{(k)}\}$ be the k-th combination out of the total C_3^n combinations of the query set Q, and they satisfy the following criterion:

$$\begin{aligned} t_1^{(k)} &= d_j(f_{1j}^{(k)}, f_{2j}^{(k)}) \\ t_2^{(k)} &= d_j(f_{2j}^{(k)}, f_{3j}^{(k)}) \\ t_3^{(k)} &= d_j(f_{3j}^{(k)}, f_{1j}^{(k)}) \\ t_1^{(k)} &\geq t_2^{(k)} \geq t_3^{(k)} \end{aligned},$$

Under this condition, there are two sub-cases: one is $(t_1^{(k)})^2 \geq (t_2^{(k)})^2 + (t_3^{(k)})^2$, and the other is $(t_1^{(k)})^2 < (t_2^{(k)})^2 + (t_3^{(k)})^2$. When the former one occurs, the bounding radius is $r_j^{(k)} = \frac{1}{2} t_1^{(k)}$; when the latter one occurs, $r_j^{(k)}$ is the solution of the two equations:

$$\begin{aligned} (t_1^{(k)})^2 &= (t_2^{(k)})^2 + (t_3^{(k)})^2 - 2(t_2^{(k)})(t_3^{(k)}) \cos\theta \\ (t_1^{(k)})^2 &= (r_j^{(k)})^2 + (r_j^{(k)})^2 - 2(r_j^{(k)})(r_j^{(k)}) \cos 2\theta \end{aligned}$$

The scatter numbers may be interpreted as an "importance" indicator of that feature, because a larger bounding sphere means that feature points are spreaded over a large region. Based on the previously described principles, we give less *perception weight* to a more scattered feature (F_j):

$$w_j = \frac{1}{s_j} * \left(\sum_{k=1}^{m} \frac{1}{s_k} \right)^{-1}.$$

The distance function (of two images, q_1 and q_2) combining m features is then defined as

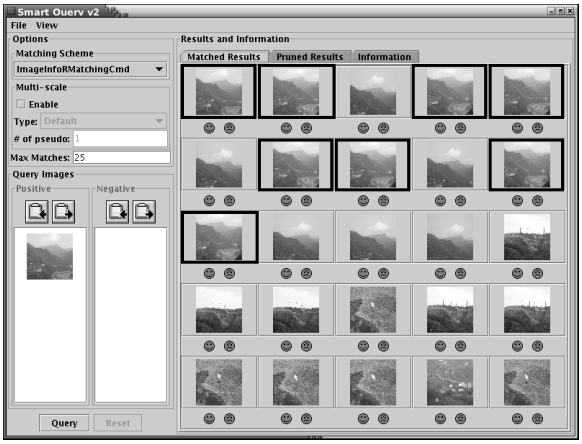
$$D(q_1, q_2) = \sum_{j=1}^{m} w_j * d_j(f_{1j}, f_{2j}).$$

Finally, the distance function between image I and n query instances (Q) is defined by

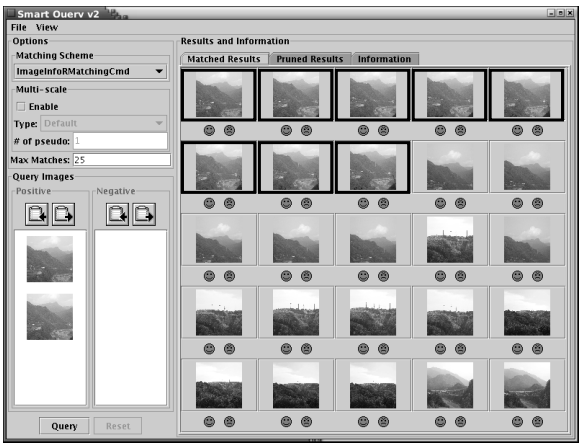
$$D(I, Q) = \min_{i=1..n} D(I, q_i).$$

Note that the normalization factor n_j is canceled in every $w_j * d_j(f_{1j}, f_{2j})$ term. This implies that we can safely ignore the distance normalization problem as long as all the feature metrics are bounded.

To test the efficiency of the method, we perform subjective queries against the image database that will be described



(a)

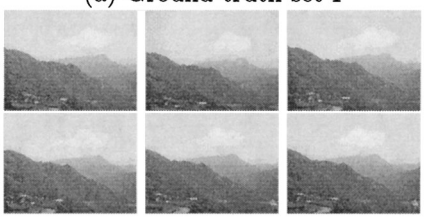


(b)

Fig. 1 Subjective result of multi-instance effects.

in detail in Sect. 4.2. Three image global features defined by MPEG-7 [14] are adopted. They are scalable color, color layout, and edge histogram. Figure 1 illustrates how our proposed method improves the query accuracy. Figure 1 (a) shows the results when a user specifies one image as the query example. Since we do not have enough information to weight the features, we simply assign them equal weights. The top-25 results are listed from left to right and top to bottom, with the most similar at the top-left corner. The boxed images are the ground-truth images. We may see that three non-ground-truth images are considered more similar than the ground-truth images. Figure 1 (b) shows the results when a user gives two of the ground-truth images as the query examples. The system derives weighting factors for each feature, and the results are improved. All the ground-truth images occupy the top ranks, a desired result. Figure 2 shows the two ground-truth sets which occupies top ranks of Fig. 1 (a). The top-left image of Fig. 2 (a) is the query instance, and the second set (Fig. 2 (b)) interferes the query results in the condition of Fig. 1 (a).

(a) Ground-truth set I



(b) Ground-truth set II

Fig. 2 The two interfering ground-truth sets in Fig. 1 (a).

3.2 Negative Images

In this section, we will describe how to use negative feedback images to improve the query accuracy. For a typical QBE search, a user provides a non-empty set of relevant (positive feedback) images. Suppose we can also ask the user to select negative images. In the following proposed scheme, negative images are not included in computing the perceptual weights. This is due to the following observations.

1. *All positive examples are alike; each negative example is negative in its own way* [12].
2. The human perception of similarity and dissimilarity may not be (linearly) additive.
3. When an image is considered dissimilar to the query one, we do not know which features (one or many) dominate in producing the perceptual dissimilarity.

So we use negative images in the following way: they create "holes" in the feature space. That is, the database images located inside the pruning radius and close to the negative images are removed from the top-N (similar) list. As shown in Fig. 3, a negative sample is denoted as g_i and the positive samples are denoted as p_1, p_2, and p_3. Essentially, we conduct a pruning process for removing positively correlated images based on the given negative image(s). Let Q_p and Q_n are the positive and the negative image sets respectively. A *pruning radius* associated with a negative image $g_i \in Q_n$ is specified by $r_p(g_i) = D(g_i, Q_p)$. An image I_r is thus removed from the top-N list if I_r is located in a pruning region:

$$\exists g_i \in Q_n \text{ satisfies } \begin{cases} D(I_r, g_i) < r_p(g_i) \\ D(I_r, Q_p) > D(I_r, g_i) \end{cases}.$$

There are two conditions given in the preceding equation. An intuitive explanation to the second condition is that if an

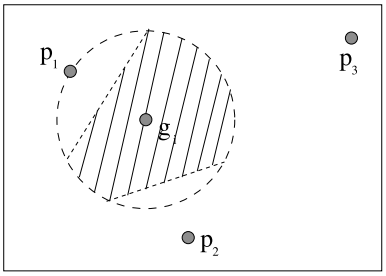
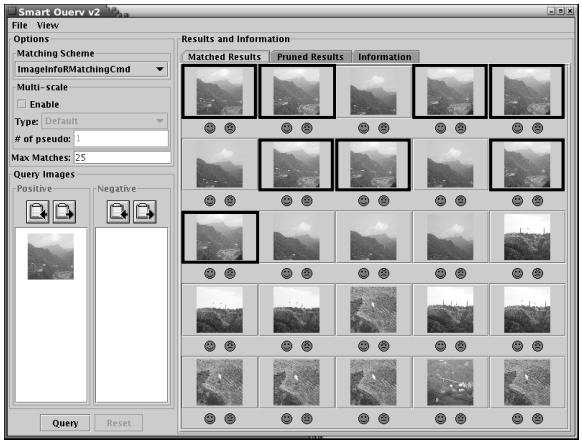


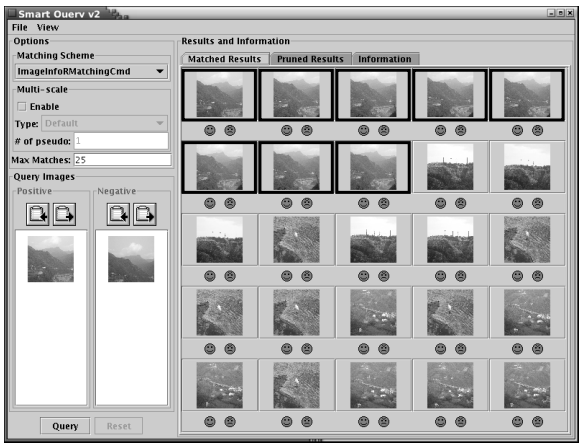
Fig. 3 Pruning area in the combined feature space.

image is closer to Q_n than Q_p, it is excluded from the top-N list. Since the negative feedback sample set is small and incomplete, we do not want to exclude the images that are a bit far away from both sets but are slightly closer to a negative sample. Therefore, the first condition gives a maximum pruning radius. Thus, our pruning operation starts from the highest priority item on the top-N list. If an image is closer to Q_n than Q_p and is located inside the pruning radius, it is excluded from the top-N list.

Figure 4 illustrates the matching results with and without the negative query instance. Figure 4 (a) is the query results of using a single positive instance. After assigning the highest-ranked non-ground-truth image as the negative feedback, the query results is shown as Fig. 4 (b). We may see that the query accuracy is improved even when using equal-weighted combined distance function (remember that only positive instances participate in the weighting estimation).



(a)



(b)

Fig. 4 Subjective result of negative instance effects.

3.3 Pseudo Query Images

In the case that the number of query images is too small, we use the multi-scale technique to create pseudo query images. The term "scale" here refers to either the spatial resolution or the SNR quality. The idea is based on the conjecture that the down-sampled or noise-added images are subjectively similar to the original version. We also observe that a low-level feature often have somewhat different values at different scales.

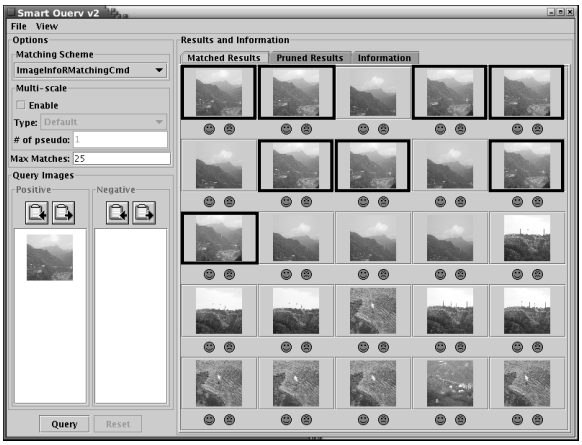
The pseudo images can be generated in various ways, such as using wavelet transforms [15]. In this paper, we examine the effects of spatial scaled pseudo images and SNR-scaled pseudo images. The spatial scaled images are generated by down-sampling the original image in both width and height by a factor of α, $0 \le \alpha < 1$; the SNR-scaled images are generated by lowering the quality factor q in JPEG compression by a ratio of β, $0 \le \beta < 1$.

An unstable (sensitive) feature in our definition yields a large distance value among the scaled images derived from the original with different scales. The measure of instability is again specified by the scatter number s_j defined in Sect. 3.1. Stable features often represent the most noticeable features of an image and they in term are often the features that the inquiring users desire. Therefore, we come up with another principle: *We give the stable features of a query image more confidence (more weight) in searching for its similar images.* Thus, we include these pseudo images into the query set. The combined procedure thus puts less weight to more scattered features, which may be due to either perceptual irrelevance or feature instability. We will see that the pseudo image improves accuracy when the number of input images is one or two. Hence, the feature stability principle is justified mostly by observations and experiments.

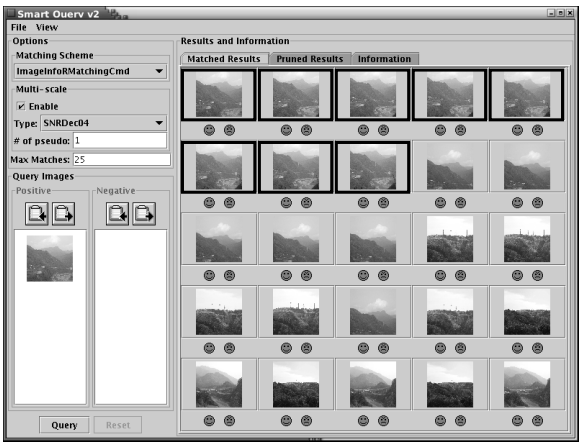
The effect of pseudo-image generation is illustrated by Fig. 5. As before, Fig. 5 (a) is the single positive image query. When we enable the multi-scale pseudo-image generation (one SNR pseudo image in this example), the query returns the desired result, as shown in Fig. 5 (b). Often, by incorporating pseudo image concepts, the system gives users the best results at the first query iteration.

3.4 Architecture

The proposed CBIR query system architecture is summarized by Fig. 6. The original positive query (input) images are used to generate pseudo-images. Together they form the query set. The query set is fed into the user perception analysis process to estimate the weighting factors. Then, the



(a)



(b)

Fig. 5 Subjective result of pseudo instance effects.

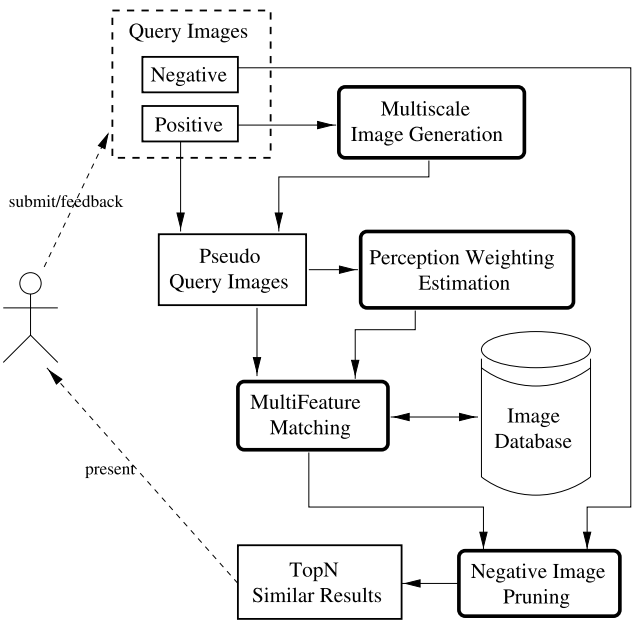


Fig. 6 Proposed perception estimation and query system.

query set and the weighting factors are passed to the image matching process to compute image similarity. A tentative matching list is thus produced. Then, the pruning process based on the supplied negative images is applied to the tentative matching list and some "negative" images may be removed. At the end, we receive the final top-N list.

4. Experiments and Discussions

In this section, we examine our design using objective measures. We first explain the adopted accuracy metric in Sect. 4.1. Then, the simulation environment and conditions are described in Sect. 4.2. The simulation results are summarized in Sect. 4.3. Finally, we modify the proposed method to reduce its complexity. Also, our scheme is compared against with two other schemes in Sect. 4.4.

4.1 ANMRR

Many researches use precision and recall analysis to evaluate a CBIR system. In fact, these two rating metrics represent two different viewpoints. The former one is the ratio between the number of the retrieved relevant images and the number of the total retrieved images. The latter is the ratio between the number of the retrieved relevant images and the number of the pre-defined relevant (so-called ground truth) images. These two rates are influenced by the chosen size of the top-N list. In searching for a suitable objective measure, we finally adopt the *Average Normalized Modified Retrieval Rank* (ANMRR) [16] metric. The ANMRR is used in the MPEG-7 standardization process to quantitatively compare the retrieval accuracy of competing visual descriptors. This metric is a modified combination of precision and recall metrics, and is a normalized index to rate the overall query accuracy of a system. For a query image, this measurement favors a matched ground-truth result and penalizes a missing ground-truth. We briefly describe the formula of ANMRR in the following paragraphs. Details can be found in the references [16], [17].

For a query image q with a ground-truth size of $NG(q)$, $rank(k)$ is the rank of the kth ground-truth image on the top-N result list. Then,

$$Rank(k) = \begin{cases} rank(k) & \text{if } rank(k) \le K(q) \\ 1.25 \cdot K(q) & \text{if } rank(k) > K(q) \end{cases}$$
$$\text{where } K(q) = \min\{4 \cdot NG(q), 2 \cdot \max_{q}[NG(q)]\}.$$

The average retrieval rank is computed and normalized with respect to the ground-truth set to yield the *Normalized Modified Retrieval Rank* (NMRR):

$$NMRR(q) = \frac{\frac{1}{NG(q)} \sum_{k=1}^{NG(q)} Rank(k) - 0.5 \cdot [1 + NG(q)]}{1.25 \cdot K(q) - 0.5 \cdot [1 + NG(q)]}.$$

The range of $NMRR(q)$ is $[0, 1]$. The value 0 indicates a perfect match that all the ground-truth pictures are included in the top-rank list. On the other hand, the value 1 means no match. Finally, we have the *Average Normalized Modified Retrieval Rank* (ANMRR) over the test cases:

$$ANMRR = \frac{1}{NQ}\sum_{q=1}^{NQ} NMRR(q),$$

where NQ is the number of queries.

4.2 Experiments

In our previous work [18], we have conducted a preliminary experiment to evaluate the proposed method against a 1050-image database. The results show that the multi-instance user perception weighting method is promising, and the pruning concept always improves the query accuracy in our method; also, the pseudo-image concept improves the accuracy in many cases.

In this paper, we extend the evaluation process to a much larger scale. The database consists of 18433 images including 256 test (ground-truth) images, 194 people (party) photos, 200 flower pictures, 200 undersea pictures, 200 outdoor scenery pictures, and 17383 images from the Corel gallery.

We collect 38 sets of outdoor scenic images as the ground truth. They are similar in terms of low-level descriptions. We prepare the ground-truth images as follows: each set of ground-truth images is taken on the same spot with slightly different camera pan and tilt angles by hand. The size of a ground-truth set varies from 4 to 10. Images in each set are perceptually similar. However, by examining the low-level features, we observe that the feature values can be quite different. There are several possible causes. The first is that these pictures are taken by hands. They are inevitably somewhat shifted and blurred. The second is that different shots have slightly different focus and shutter speed. The third is that photos with shooting angle variation may have different background lighting, which may change the shade of each picture.

Our experiments simulate a typical image query scenario. A user first chooses one or a few "similar" input images to start a query. The matching process returns an ordered list of results; we call it the positive-only query result. If the result is not perfect; that is, not all ground-truth images occupy the highest ranks, or simply $NMRR \neq 0$, then the highest ranked non-ground-truth image is assigned as the negative feedback item. Then, we repeat the query process with both positive and negative images and produce the positive-and-negative query result. If the positive-only result is perfect, both $NMRR_{positive\text{-}only}$ and $NMRR_{positive\text{-}and\text{-}negative}$ are set to zero. Since the smallest ground truth set has only four images, we simulate the conditions of one to three positive images per query. All possible combinations of images in all ground truth sets are tested to derive the *ANMRR* values.

Two multi-scale schemes are tested: spatial and SNR. The spatial scaling factor (for both width and height) for each down-sampled image is defined as follows: the n-th scale factor (for the n-th pseudo image) is $\alpha - 0.1(n-1)$, where $n = 1, 2$. We perform experiments at $\alpha = 0.9, 0.8, 0.7, 0.6, 0.5$ to look for the best parameter values that lead to the best ANMRR. The SNR-scaled images are generated by applying JPEG compression with a quality factor of $\beta - 0.1(n-1)$ for the n-th scaled version. The test values are $\beta = 0.7, 0.6, 0.5, 0.4, 0.3$.

To examine the effectiveness of our method, we simulate another two weighting schemes under the same assumptions. The first scheme is a variation derived from the MARS system. In this scheme, the distance metric $d_j(f_1, f_2)$ for each feature F_j is normalized as follows:

$$d'_j(f_1, f_2) = \frac{D_j(f_1, f_2) - \mu_j}{3\sigma_j},$$

where μ_j and σ_j^2 are the mean and variance of the distances of F_j in the database. This step ensures that under normal distribution assumption about 99% of the distance values are within the range of $[-1, +1]$. The second parameter-shifting step guarantees that these 99% values are within $[0, 1]$:

$$d''_j(f_1, f_2) = \frac{d'_j(f_1, f_2) + 1}{2}.$$

The final step clamps all calculated distance values between zero and one.

The original MARS system adopts a 5-level relevance feedback. To make it comparable with our simulation environment, we reduce the relevance feedback levels to three: positively relevant ($Score_l = +1$), no opinion ($Score_l = 0$), and negatively relevant ($Score_l = -1$). The weighting process is similar to that in the original MARS. Assume the overall query result list is RT, and the result list of feature F_j is RT_j. To calculate the weight w_j, we first initialize $W_j = 0$, and then update W_j as follows:

$$W_j = W_j + Score_l,$$
for each item l
which appears in both RT and RT_j.

After all W_j have been updated, the negative weights are adjusted to 0 (means "irrelevant" in MARS). Then, we compute the weighting factor for each feature F_j as

$$w_j = \frac{W_j}{\sum_{\forall j} W_j}.$$

A final remark about this MARS-like scheme is the RT list. According to the original proposal [2], it is an iterative procudure that leads to the "optimal" RT. The original proposal selects $P_{fd} = 3$ as the maximum number of iterations and shows good convergence in general. In our simulation, we set $P_{fd} = 5$. This is called Scheme A in the rest of this section.

The second scheme we simulate has the same basic

structure as our proposed scheme in Sect. 3. However, it adopts the same distance normalization in MARS. This is Scheme B. We simulate this scheme for two reasons. One is to compare with a MARS-like scheme to see the effects of different weighting estimation procedures. The other is to compare it with our scheme to see the effects of different distance normalization methods. Our scheme is labeled as Scheme C.

4.3 Simulation Results

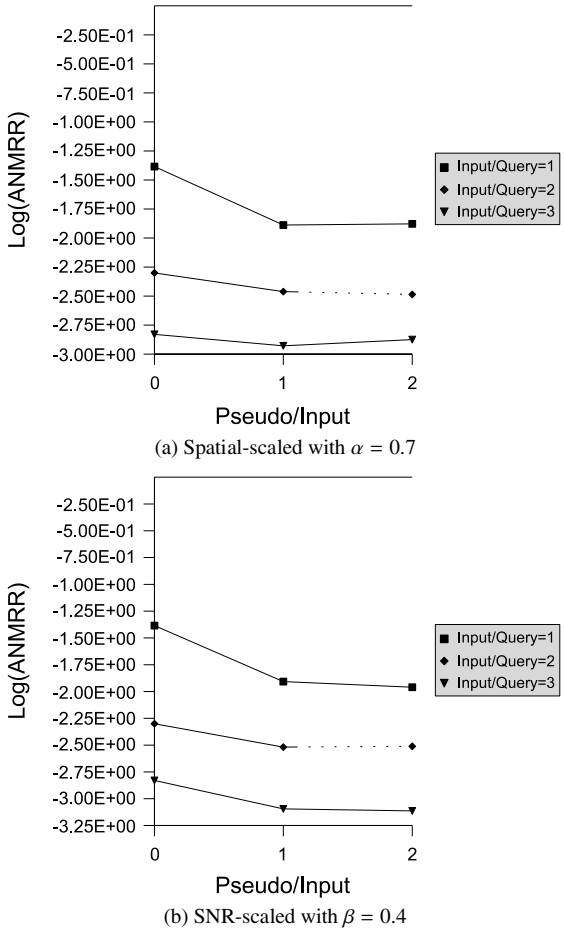
In this section, we show the simulation of query accuracies, under the environments defined in Sect. 4.2. In Sect. 4.3.1, we examine the effects of multi-instance and pseudo-images. Then, we put all simulation statistics together, to compare the efficiency of different matching schemes. The simulation results are summarized in Tables 1, 2, 3, and 4. The bold-faced numbers are the winners among all tests with the same query parameters, and the underlined numbers are the poorest performers. The row of *Input/Query* means the number of positive input images selected by the user in a query. The row of "Pseudo/Input" means the number of *pseudo images* created from each (user selected) input image. The first column (Scheme A) is the MARS-like scheme, and the second and third columns (Scheme B and C) are our schemes with different normalization formulas. To see more clearly the differences among various methods and parameters, the ANMRR values are shown in log scale. In the following paragraphs, we will examine these results and discuss the performance of the aforementioned methods.

4.3.1 Multi-instance and Pseudo-images

Here we examine the effect of multi-instances and pseudo-images. Two multi-scale schemes are shown in Fig. 7: spatial and SNR scaled pseudo image generation schemes. Figure 7 (a) is spatial down-sampling with a spatial scaling factor of $\alpha = 0.7$. We examine the effect of different pseudo/input image ratios. Under the same pseudo/input ratio, the more the input images (user provided), the better the query accuracy. For the same number of input images, pseudo images can improve the accuracy, especially when the input images is one or two. However, when input (query) images are higher in number, the addition of pseudo images may lower the matching accuracy. Figure 7 (b) shows the results of using SNR-scaled pseudo images. The noisy versions (pseudo images) are generated by a scaling factor of $\beta = 0.4$. The general trend of Fig. 7 (b) is similar to that of Fig. 7 (a). However, the average ANMRR is better in SNR multi-resolution approach. The other scaling factor values have been tested but the results are less favored.

4.3.2 Observations on Positive-only Query Results

We first look at the results of positive-only spatial-scaled



(a) Spatial-scaled with $\alpha = 0.7$

(b) SNR-scaled with $\beta = 0.4$

Fig. 7 The examples showing the accuracy improvements by using multi-instances and pseudo-images.

Table 1 Best *log(ANMRR)* of spatial-scaled pseudo images (positive-only).

	Scheme A	Scheme B	Scheme C
Input/Query = 1			
Pseudo/Input = 0	**−2.23**	**−2.23**	−1.38
Pseudo/Input = 1	**−2.22**	−2.18	−1.94
Pseudo/Input = 2	**−2.19**	−2.15	−1.93
Input/Query = 2			
Pseudo/Input = 0	−2.40	**−2.45**	−2.30
Pseudo/Input = 1	−2.40	−2.45	**−2.49**
Pseudo/Input = 2	−2.33	−2.37	**−2.49**
Input/Query = 3			
Pseudo/Input = 0	−2.40	−2.48	**−2.83**
Pseudo/Input = 1	−2.41	−2.48	**−2.93**
Pseudo/Input = 2	−2.33	−2.38	**−2.93**

experiments (Table 1). The cases shown here are the spatial scaled pseudo images with the best scaling factors. For each method, multiple input images (all the cases where Pseudo/Input = 0) improve the query accuracy. This shows that more "positive" query information would result in better query precision, regardless which scheme is in use. Next, we examine the effect of pseudo images. Our scheme with one pseudo image has the best accuracy in all Input/Query

cases. However, the pseudo images do not improve the other two methods as much. Even worse, more pseudo images would degrade the query accuracy. The simulation results also show that increasing pseudo images does not always improve accuracy. Under our current scheme, one pseudo image per input image is the best. Our conjecture to this phenomenon is as follows. The pseudo image concept is based on an assumption that the stable image features are the features of similar values in human perception. This assumption provides additional information to "guess" the user intention in a query. However, too many pseudo images may overly weight the chosen features, not reflecting their true weights.

For each query parameter set (Input/Query and Pseudo/Input), we compare the results of different estimation methods. Comparing the MARS-like method (Scheme A) with the Gaussian-normalized method (Scheme B), we observe that when input images are few, the MARS-like scheme wins. In contrast, Scheme B wins when more input images are provided. Since these two methods use the same distance normalization procedure, the difference comes from the weight computing procedures. When few images are available for estimation, iterative training would provide a better guess on the user perception. When more images are provided by a user, the ranking-list-based Scheme A does not provide as precise guess as the distance-based Scheme B. Comparing the Gaussian-normalized scheme (Scheme B) with our scheme (Scheme C), the former wins when input images are few and loses when more images are provided. The two methods use the same distance definition and the estimation procedure, so the difference comes from the distance normalization procedures. It is reasonable that the Gaussian-normalized scheme wins for few inputs cases, because the distance metrics are optimized according to the data distribution. This implicitly provides clustering information of the database, and thus produces better results than our method. However, feature distributions in a database may not be the same as the distance distribution viewed from the user perception for a particular query. This may explain why our method wins when more input images are provided. Our user perception (intention) estimation is based only on the user provided information (not the entire database).

We conduct the same analysis on the SNR-scaled case (Table 2), and similar conclusions can be drawn. However, there are two noticeable differences. The first one is that in several test cases, Pseudo/Input = 2 outperforms Pseudo/Input = 1 in the SNR-scaled case. The second is that in most cases, the SNR-scaled pseudo images outperforms the spatial-scaled ones.

4.3.3 Observations on Positive-and-Negative Query Results

Next, we look into the Positive-and-Negative Query cases. As mentioned in Sect. 4.2, the simulation is done using the typical query scenario. For each positive-and-negative query, there is zero or one negative image depending on whether the positive-only query is a perfect match or not. Similar to what we did in Sect. 4.3.2, we first examine the simulation results of positive-and-negative feedback with spatial-scaled pseudo images (Table 3). For all schemes, multiple input images improve the accuracy. Effects of pseudo images are similar to that of the positive-only results. Our method seems to be able to utilize pseudo images better for improving the accuracy. For the other two schemes, pseudo images do not provide significant improvements. The ANMRR values show that the Pseudo/Input = 1 cases give the most significant improvement. Additional pseudo images offer much less improvement if any.

Comparing Scheme A (MARS-like scheme) with Scheme B (Gaussian-normalized scheme), we found that the Gaussian-normalized scheme wins in most cases. Our explanation is that in our proposed procedure, the negative feedback does not participate in weights estimation. Since the negative instances may be too diverse to be useful in weighting estimation, their roles are more appropriate when used in pruning. The simulation results seems to prove this concept. Comparing Scheme C (our scheme) with Scheme B (Gaussian-normalized scheme), the results show that ours wins when sufficient input images are available. The ANMRR values show that the best accuracy is the Input/Query = 3 case in our scheme. The reason is that the pruning distance relies on the estimated distance func-

Table 2 Best *log(ANMRR)* of SNR-scaled pseudo images (positive-only).

	Scheme A	Scheme B	Scheme C
Input/Query = 1			
Pseudo/Input = 0	**−2.23**	**−2.23**	−1.38
Pseudo/Input = 1	**−2.19**	−2.18	−1.95
Pseudo/Input = 2	**−2.18**	**−2.18**	−2.00
Input/Query = 2			
Pseudo/Input = 0	−2.40	**−2.45**	−2.30
Pseudo/Input = 1	−2.45	−2.49	**−2.52**
Pseudo/Input = 2	−2.47	−2.49	**−2.52**
Input/Query = 3			
Pseudo/Input = 0	−2.40	−2.48	**−2.83**
Pseudo/Input = 1	−2.54	−2.63	**−3.09**
Pseudo/Input = 2	−2.63	−2.71	**−3.11**

Table 3 Best *log(ANMRR)* of spatial-scaled pseudo images (positive-and-negative).

	Scheme A	Scheme B	Scheme C
Input/Query = 1			
Pseudo/Input = 0	−1.97	**−2.12**	−1.39
Pseudo/Input = 1	−2.07	**−2.21**	−1.97
Pseudo/Input = 2	−2.06	**−2.22**	−2.00
Input/Query = 2			
Pseudo/Input = 0	**−2.67**	−2.61	−2.40
Pseudo/Input = 1	−2.65	**−2.71**	−2.63
Pseudo/Input = 2	**−2.64**	−2.62	−2.60
Input/Query = 3			
Pseudo/Input = 0	−2.72	−2.76	**−3.09**
Pseudo/Input = 1	−2.73	−2.76	**−3.22**
Pseudo/Input = 2	−2.62	−2.63	**−3.21**

Table 4 Best *log*(*ANMRR*) of SNR-scaled pseudo images (positive-and-negative).

	Scheme A	Scheme B	Scheme C
Input/Query = 1			
Pseudo/Input = 0	−1.97	**−2.12**	−1.39
Pseudo/Input = 1	−1.95	**−2.17**	−2.03
Pseudo/Input = 2	−1.94	**−2.17**	−2.04
Input/Query = 2			
Pseudo/Input = 0	**−2.67**	−2.61	−2.40
Pseudo/Input = 1	−2.60	**−2.71**	−2.67
Pseudo/Input = 2	−2.63	**−2.73**	−2.66
Input/Query = 3			
Pseudo/Input = 0	−2.72	−2.76	**−3.09**
Pseudo/Input = 1	−2.89	−2.96	**−3.47**
Pseudo/Input = 2	−2.99	−3.05	**−3.49**

tion. Thus, the more precise distance function would lead to a lower "mis-pruning" probability.

The ANMRR values shown in Table 4 for the SNR-scaled pseudo images lead to similar conclusions as before. First, multiple input instances improve query accuracy. Second, our method benefits more from the pseudo images. Third, the Gaussian-normalized scheme (Scheme B) wins in almost all cases when comparing to the MARS-like scheme (Scheme A). Fourth, our method (Scheme C) performs better than the Gaussian-normalized when more input images are available. Finally, our method has a significant performance improvement at Input/Query = 3, which indicates a good potential of our approach for even more input images.

4.3.4 Observations on Different Feedback Schemes

In Sects. 4.3.2 and 4.3.3, we discuss the effects of different weights estimation methods in each specific scheme. In this section, we will discuss the general effect of negative instances and the generation of pseudo images.

Negative instances are important, because they tell us about the "undesired" image properties (or image feature values). That is, the user does not want pictures similar to a negative image. However, the negative images do not provide information about a particular feature whether it is good for matching purpose or not. Two negative images can be close or far away, but positive images should always be close together on the user preferred features. The simulation results show that negative feedback improves query accuracy in many cases, especially when enough positive instances are given. If the number of input instances is small, only our method can consistently improve the accuracy using the negative instances.

Although both multi-scale schemes that generate pseudo images can enhance the query accuracy (especially for our method), we notice that the SNR multi-scaled images not only produces better performance than the spatially scaled ones, they also have consistently improved results. This may be due to the fact that the spatial-scaled images suffer from the aliasing effect when pictures are down-sampled and thus image features are distorted more than those of the SNR-scaled ones. Overall, Scheme C significantly improves the query accuracy by combining multi-instance and pseudo-image concepts.

Note that our scheme does not produce as good accuracy as the other two schemes when very few samples are available. This may due to the fact that our feature weights are derived solely from the samples; no distance distribution information is used to normalize the weights. Our estimate is getting much better when the number of samples increases. That is, our scheme is more suitable for multiple-instance cases, which is the goal of this contribution.

4.3.5 Summary

Based on the above simulation results, we briefly summarize our observations below.

- Distance-based weight estimation outperforms when multiple input instances are available.
- Pseudo images improve query accuracy in many cases, especially when our method is used with SNR scalability.
- Experiments show that one pseudo image per input image gives significant performance boost in most cases.
- Negative instances used as a pruning criterion produce better results than those used as negative samples in weight calculation.
- When input instances are few, negative feedback may even degrade the performance of the MARS-like and the Gaussian-normalized schemes.
- When sufficient input instances are available, the Gaussian normalized feature distance does not provide as precise estimation as our method.
- The SNR multi-scaled pseudo images provide better ANMRR values; they also lead to more consistent improvements in accuracy.

An overall comment about the performance of our scheme is as follows:

- When only one input image is available, our scheme looses about 0.8 in *log*(*ANMRR*). However, with the assistance of pseudo images, the gap shrinks to about 0.25.
- In the case of two input images, our scheme improves. Without pseudo images, ours looses about 0.2; with pseudo images, ours may win or loose in the average of 0.05.
- When we have three input images, our scheme wins. The figure in *log*(*ANMRR*) is about 0.3 to 0.5.
- From the above results, we can summarize that Scheme C is good for sufficient query images. For small-sample cases, though not as good as other schemes, it produces comparable accuracy by including pseudo images.

4.4 Another Distance Measure

In Sect. 3, we assume the matching function produces distances that satisfy triangular inequality. This may not be

necessary because a CBIR system may adopt non-linear operations either in the extraction or in the matching process. In this section, we define the scatter number by a statistical approach, which does not rely on the geometry theorems and eliminate the sinusoidal operations. Its calculation is thus simpler.

The assumptions and the conjectures are the same as described in Sect. 3, except that we do not assume the distances satisfy the triangular inequality. To measure the sparseness of a set of feature points, firstly we define a value $scatt_{ij}$ which represents how far the instance q_i is away from the rest of the query instances:

$$scatt_{ij} = \mu_{ij} + \sigma_{ij},$$

where

$$\mu_{ij} = \frac{1}{n-1} \sum_{k=1,k\neq i}^{n} d_j(f_{ij}, f_{kj})$$

$$\sigma_{ij}^2 = \frac{1}{n-1} \sum_{k=1,k\neq i}^{n} (d_j(f_{ij}, f_{kj}))^2 - \mu_{ij}^2.$$

The $scatt_{ij}$ is similar to the average distance, except that it includes the variation information. The second term (standard deviation) is added into this measure because experiments indicate that an "inconsistent" feature (large standard deviation) is less important.

Then we express the scatter number in a conservative way: calculate the closeness between the given instance and any other point in the set and pick up the maximum; that is,

$$s_j = \max_{\forall i} scatt_{ij}.$$

Note that in this method, the normalization factor is also canceled in each term of the weighted distance function.

We call this weighting scheme as scheme D. The simulation results of scheme A, B, and D are reported in our previous paper [19]. Since the results of scheme A and B are the same as listed in Table 1 to 4, we only show the results of scheme D in Table 5. By comparing to scheme C in Table 1 to 4, we may see that these two schemes have similar accuracy but scheme D has the computational advantage. The bold-faced values represent the better ANMRR in scheme D; while the underlined values represent the better ANMRR in scheme C.

Table 5 Best *log*(*ANMRR*) of scheme D for all query schemes.

	Spatial scaling without negative	SNR scaling without negative	Spatial scaling with negative	SNR scaling with negative
Input/Query = 1				
Pseudo/Input = 0	−1.38	−1.38	−1.39	−1.39
Pseudo/Input = 1	−1.94	−1.95	−1.97	−2.03
Pseudo/Input = 2	−1.93	−1.99	−1.99	**−2.06**
Input/Query = 2				
Pseudo/Input = 0	−2.30	−2.30	−2.40	−2.40
Pseudo/Input = 1	**−2.51**	−2.52	**−2.65**	−2.67
Pseudo/Input = 2	**−2.50**	−2.52	**−2.65**	−2.66
Input/Query = 3				
Pseudo/Input = 0	−2.83	−2.83	−3.09	−3.09
Pseudo/Input = 1	−2.92	−3.07	−3.21	**−3.49**
Pseudo/Input = 2	−2.92	−3.10	**−3.23**	**−3.51**

5. Conclusions

In this paper, the multi-instance image retrieval problem was investigated. The main contributions of this paper are listed below.

1. Two distance-based methods to estimate the user perception based on the given positive instances were proposed. One is a geometric approach and the other is a statistical approach.
2. Two schemes for generating consistent pseudo images were investigated. We showed that the pseudo image concept improves the query accuracy in many cases, particularly when the query set is too small.
3. A method of pruning irrelevant outcomes based on the given negative images was proposed.

The first concept was realized by analyzing the scattering magnitude of the query instances in the feature space. Our conjecture is that a scattered feature implies less importance in deciding the perceptual similarity. The second concept was realized through the notion of feature stability. Our conjecture is that a stable image feature (for a particular image) would have similar numerical values (small scatter numbers) at different spatial or SNR scales of the same image. Therefore, pseudo images were created by scaling the original image at various spatial and SNR resolutions. The third one was realized by creating pruning regions in the combined feature space. Our conjecture is that negative instances carry only the information of the undesired image feature values. Namely, undesired images should not look like the opposites of positive images. Because negative images may be close or far away, they are only suitable for pruning not for distance estimation.

All the preceding concepts can be integrated into one algorithm using the same basic structure. We examined the performance of our scheme using the ANMRR criterion. Simulations showed that multiple instances are helpful in achieving better query accuracy. In the case that the user input set is small, the synthesized pseudo images improve the results in most cases. As we conjectured, negative feedbacks used for pruning performs better than those used for weight estimation.

Additional conclusions are:

1. the SNR multi-scaled images generally perform better than the spatial multi-scaled ones for pseudo images purpose;
2. although distance normalization is conceptually useful for weight estimation, simulations show that normalization is not always needed for producing good results;
3. our method does not require *a priori* information about

the data distribution in the database, which not only reduces the computational complexity but also makes it more suitable for searching in a distributed networking environment.

Acknowledgments

This work is partially supported by the Lee & MTI Center for Networking Research at National Chiao Tung University and by National Science Council (Taiwan, R.O.C.) under Grant NSC 91-2219-E-009-041.

References

[1] A.W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," IEEE Trans. Pattern Anal. Mach. Intell., vol.22, no.12, pp.1349–1380, Dec. 2000.

[2] Y. Rui, T.S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A power tool for interactive content-based image retrieval," IEEE Trans. Circuits Syst. Video Technol., vol.8, no.5, pp.644–655, Sept. 1998.

[3] J. Fauqueur and N. Boujemaa, "Logical query composition from local visual feature thesaurus," Third International Workshop on Content-Based Multimedia Indexing (CBMI'03), Rennes, France, Sept. 2003.

[4] S. Jeong, K. Kim, B. Chun, J. Lee, and Y.J. Bae, "An effective method for combining multiple features of image retrieval," TENCON 99. Proc. IEEE Region 10 Conference, Korea, pp.982–985, Sept. 1999.

[5] Y. Rui, T.S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," Proc. IEEE Int. Conf. Image Processing, pp.815–818, 1997.

[6] G. Aggarwal, T.V. Ashwin, and S. Ghosal, "An image retrieval system with automatic query modification," IEEE Trans. Multimed., vol.4, no.2, pp.201–214, June 2002.

[7] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting." Dept. of Statistics, Stanford University Technical Report, 1998.

[8] C.Y. Liu, J.J. Chen, and F.C. Chang, "A dynamically adapted retrieval algorithm for multi-instance image query with heterogeneous features," IEEE Consumer Communications and Networking Conference (CCNC), pp.627–629, Las Vegas, Nevada USA, Jan. 2004.

[9] J.J. Chen, C.Y. Liu, and F.C. Chang, "The content-driven preprocessor of images for mpeg-7 descriptions," Journal on Systemics, Cybernetics and Informatics, vol.1, no.3, 2003.

[10] F.C. Chang, H.M. Hang, and H.C. Huang, "Research friendly MPEG-7 software testbed," Image and Video Communication and Processing Conf., pp.890–901, Santa Clara, USA, Jan. 2003.

[11] T. Ashwin, N. Jain, and S. Ghosal, "Improving image retrieval performance with negative relevance feedback," ICASSP, pp.1637–1640, May 2001.

[12] X.S. Zhou and T.S. Huang, "Small sample learning during multimedia retrieval using biasmap," Computer Vision and Pattern Recognition (CVPR), pp.I-11–I-17, Dec. 2001.

[13] C. Zhang and T. Chen, "An active learning framework for content-based information retrieval," IEEE Trans. Multimed., vol.4, no.2, pp.260–268, June 2002.

[14] Multimedia Content Description Interface—Part 3: Visual, ISO/IEC JTC1/SC29/WG11, FDIS N4203, MPEG Committee, July 2001.

[15] M. Kobayakawa, M. Hoshi, and T. Ohmori, "Interactive image retrieval based on wavelet transform," Proc. SCI'99 and ISAS'99, pp.76–85, Orland, Florida, July 1999.

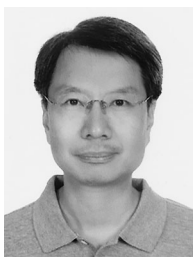
[16] Subjective Evaluation of the MPEG-7 Retrieval Accuracy Measure (ANMRR), ISO/IEC JTC1/SC29/WG11, M6029, MPEG Committee, May 2000.

[17] B.S. Manjunath, P. Salembier, and T. Sikora, eds., Introduction to MPEG-7, John Wiley & Sons Ltd., Baffins Lane, Chichester, West Sussex PO19 1UD, England, 2002.

[18] F.C. Chang and H.M. Hang, "Content-based image retrieval using both positive and negative feedback," International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, June 2004.

[19] F.C. Chang and H.M. Hang, "A relevance feedback image retrieval scheme using multi-instance and pseudo image concepts," IS&T/SPIE Electronic Imaging 2005, San Jose, California, USA, Jan. 2005.

Feng-Cheng Chang received the B.S. and M.S. degrees in electronics engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1994 and 1996 respectively. He joined the Java center of the Institute for Information Industry (III), Taiwan, in July 1998. After working at III for a year, he came back to NCTU and is working toward the Ph.D. degree. He participated the development of web middleware when he was in the Java center, and built his knowledge about flexible framework design. His research interests include image processing, multimedia database, digital rights management, and internet applications.



Hsueh-Ming Hang received the B.S. and M.S. degrees from National Chiao Tung University, Hsinchu, Taiwan, in 1978 and 1980, respectively, and Ph.D. in Electrical Engineering from Rensselaer Polytechnic Institute, Troy, NY, in 1984. From 1984 to 1991, he was with AT&T Bell Laboratories, Holmdel, NJ, and then he joined the Electronics Engineering Department of National Chiao Tung University, Hsinchu, Taiwan, in December 1991. He has been actively involved in the international video standards since 1984 and his current research interests include multimedia compression, image/signal processing algorithms and architectures, and multimedia communication systems. Dr. Hang holds 10 patents (ROC, US and Japan) and has published over 130 technical papers related to image compression, signal processing, and video codec architecture. He was a conference co-chair of the Symposium on Visual Communications and Image Processing (VCIP) in 1993, and the program chair in 1995. He was a co-program chair for the IEEE International Symposium on Consumer Electronics in 1998 and the IEEE Signal Processing Systems Workshop in 1999. He was an associate editor of the IEEE Transactions on Image Processing (1992–1994) and the IEEE Transactions on Circuits and Systems for Video Technology (1997–1999). He is co-editor and contributor of the Handbook of Visual Communications published by Academic Press. He is a Fellow of IEEE and is recipient of the IEEE Third Millennium Medal and the IEEE ISCE Outstanding Service Award.