Multimed Tools Appl
DOI 10.1007/s11042-008-0212-5

Confidence interval approach to feature re-weighting

Robert S. Lee · Chin-Wan Chung · Seok-Lyong Lee · Sang-Hee Kim

Abstract Relevance feedback is commonly incorporated into content-based image retrieval systems with the objective of improving retrieval accuracy via user feedback. One effective method for improving retrieval performance is to perform feature re-weighting based on the obtained feedback. Previous approaches to feature re-weighting via relevance feedback assume the feature data for images can be represented in fixed-length vectors. However, many approaches are invalidated with the recent development of features that cannot be represented in fixed-length vectors. In addition, previous approaches use only the information from the set of images returned in the latest query result for feature re-weighting. In this paper, we propose a feature re-weighting approach that places no restriction on the representation of feature data and utilizes the aggregate set of images returned over the iterations of retrieval to obtain feature re-weighting information. The approach analyzes the feature distances calculated between the query image and the resulting set of images to approximate the feature distances for the entire set of images in the database. Two-sided confidence intervals are used with the distances to obtain the information for feature re-weighting. There is no restriction on how the distances are calculated for each feature. This provides freedom for how the feature representations are structured. The experimental results show the effectiveness of the proposed approach and in comparisons with other work, it is shown that our approach outperforms previous work.

Keywords Content-based image retrieval · Relevance feedback · Feature re-weighting · Image databases

R. S. Lee · C.-W. Chung (✉)
Division of Computer Science, Korea Advanced Institute of Science and Technology, Daejeon, South Korea
e-mail: chungcw@cs.kaist.ac.kr

S.-L. Lee
School of Industrial Information & Systems Engineering, Hankuk University of Foreign Studies, Seoul, South Korea

S.-H. Kim
Agency for Defense Development, Daejeon, South Korea

1 Introduction

Content-based image retrieval (CBIR) is regarded as an effective approach to managing large collections of images. CBIR systems allow users to search for images via query-by-example, where the user provides a sample image or sketch of the type of images they are looking for within the collection. In this approach, the similarity between the query image and the images in a database are determined by a comparison of the features that are used to describe each image; such features may describe color, texture and/or shape. One inherent problem with this approach is the semantic gap between the feature descriptions and the high-level visual perception of an image. In other words, although the feature values of some images may be similar, their actual visual appearances may not be similar. An effective methodology for overcoming this semantic gap involves relevance feedback to perform feature re-weighting. Incorporating relevance feedback into a CBIR system leads to an image search becoming an interactive session, where the user provides feedback on the quality of the results. The system uses this feedback to weight higher the features that can better determine visual similarity, then using the updated feature weights, determine the set of similar images based on the features. This set of images should contain more relevant images than the previous result set. Existing approaches for feature re-weighting via relevance feedback assume that the feature data can be represented in fixed-length vectors, where the corresponding elements of the vectors of each image are located in the same vector positions [1, 9, 11, 12, 15, 18, 21]. Following this assumption, feature re-weighting can be seen to have the effect illustrated in Fig. 1. In (a) we see the initial plot for two features of image A and image B. From user feedback, the system determines that feature *x* is more important than feature *y*. Since feature *x* is more important, its axis is compressed. The opposite occurs for the axis corresponding to feature *y*. These changes are shown in (b). With the scaling of the axes, the point corresponding to image B is brought closer to the query point and, thus, is considered to be more similar to the query image. Also, the distances are calculated using some Euclidean-type function. This assumption is acceptable when using features such as colour histograms since the number of histogram bins for each image is the same. However, this viewpoint is invalidated by features such as the dominant color descriptor [17], curvature scale space [18], and incomplete contour representations [6], which have representations that may differ in size between images. High-level features that are currently being researched, such as 9D-SPA [10], which describe the spatial relations between objects in an image also invalidate the assumption that features can be represented in fixed-length vectors since the number of objects may differ between images.

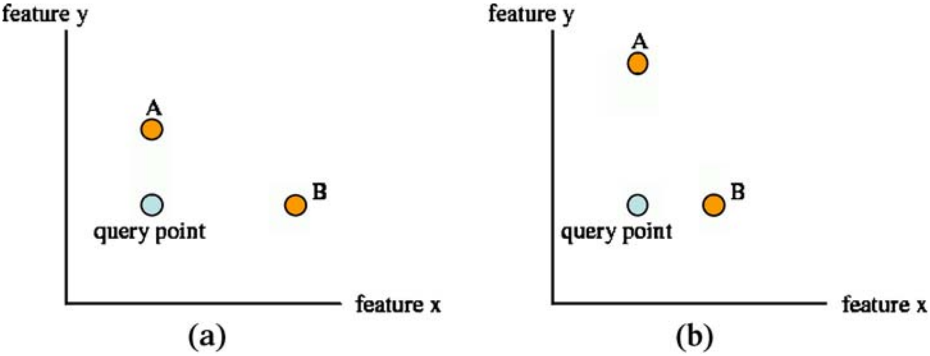


Fig. 1 Effect of feature re-weighting

Our proposed approach does not directly analyze the feature values to perform feature re-weighting. Instead, using the result set of images, the feature distances obtained using the respective distance function of each feature are analyzed to approximate the distances for all the relevant and non-relevant images in the database via two-sided confidence intervals. By doing so, information approximated from the whole image database is used for feature re-weighting. In addition, since the feature values are not directly analyzed, there is no restriction placed on the structure of the feature representations. Taking this approach leads to the complete freedom in the feature representations and distance functions used for a given feature.

Figure 2 shows how our feature re-weighting mechanism is incorporated into an image search session that involves relevance feedback. First the user provides the query image the system will use to find similar images from the database. The features are extracted from the image which the query processor then compares to the features of the images in the database. The most similar images, as determined by the query processor with the feature weights initially all equal, are then returned to the user. The user identifies the images that are relevant for the search and then requests a new set of images. The feature re-weighting mechanism uses this feedback to update the weights, which are then passed onto the query processor and used for the determination of the next set of similar images. The user provides feedback on this returned set of images and the cycle continues until the resulting image set does not change or the user is satisfied with the results.

The contributions of this paper are as follows:

- Effective re-weighting of features. Since our approach is able to use information approximated from the whole image database, the recall of the proposed approach was on average better than an existing approach by 144% at the 2nd iteration, 78% by the fifth iteration, 63% for the 10th iteration, and 53% for the 20th iteration of retrieval.
- A feature re-weighting approach that supports features regardless of whether or not they can be represented in a fixed-length vector. The combination of features used for the experiments show that it can effectively handle features regardless of the structure of the feature representation. When using feature representations of both fixed and varying lengths, our proposed approach was on average better than an existing approach by 185% at the second iteration, 51% by the fifth iteration, 36% for the tenth iteration, and 26% for the 20th iteration of retrieval.
- Simplified user interaction for relevance feedback. The user is not required to judge the degree of relevance or non-relevance of an image. The user only needs to identify the relevant images.

The remainder of this paper is organized as follows. In Section 2, we discuss related work. Section 3 describes our proposed feature re-weighting approach after an explanation

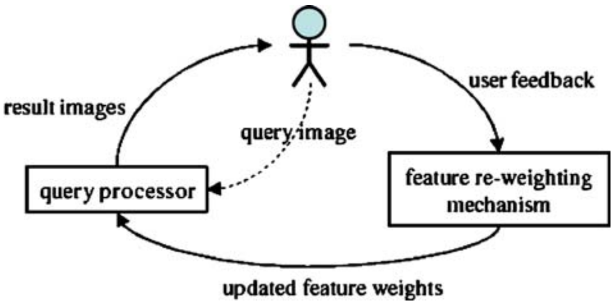


Fig. 2 Feature re-weighting incorporated into relevance feedback cycle

of the basis of the approach. Experimental results are presented in Section 4. Finally, Section 5 provides a summary of our work.

2 Related work

A great deal of research has been focused on content-based image retrieval since the early 1990s. The first commercial CBIR system to be developed is IBM's Query By Image Content (QBIC) [5] which allows a user to search for images using colour, texture, shape and text. VisualSEEk [22] introduced the identification of regions and objects within an image, which provides significant gains in similarity retrieval compared to the use of global image features. Since then, systems which demonstrate improved segmentation, feature extraction, and query processing have been developed. However, many of those systems cannot overcome the semantic gap without the use of relevance feedback.

The Multimedia Analysis and Retrieval System (MARS) [19–21] introduced the concept of relevance feedback in content-based image retrieval. The feature model described in the system is designed to support two levels of feature re-weighting, intra-feature and inter-feature re-weighting. Intra-feature re-weighting updates the influence each component of a feature representation has in the feature's distance calculation, e.g. the influence of each bin in a colour histogram. Inter-feature re-weighting adjusts the weight a feature's distance has in the overall distance calculation when multiple features are used. For intra-feature re-weighting in MARS, the variance in the values of the corresponding feature components of the relevant images is analyzed. The weight for the jth feature component of feature$_i$ is updated to the inverse of the standard deviation of the component values from the relevant images. This step of re-weighting determines the amount the jth component contributes to the feature$_i$ distance. The determination of the significance of each feature (i.e., how much the feature$_i$ distance contributes to the overall distance) is called inter-feature re-weighting. For each iteration of retrieval, the system calculates the set of most similar images with respect to the overall distance along with the sets of similar images calculated with respect to each individual feature. Let S and S_i, where $i=1\ldots n$, where n is the maximum number of features, be the sets of top-k images determined using the overall distance and distance based on only feature$_\mathrm{i}$, respectively. Scores are assigned to the retrieved images, those in set S, by the user. The scores correspond to the degree of relevance as judged by the user. To update the weight for feature$_i$, the score for each feature is initialized to 0. The following steps are then performed.

```
$S=\{\mathrm{obj}_1,\ldots, \mathrm{obj}_k\}$
$S_i=\{\mathrm{obj}_l^i,\ldots, \mathrm{obj}_k^i\}$
for $i=1$ to $n$
  for $x=1$ to $k$
  if ($\mathrm{obj}_x^i$ exists in $S$)
    $F_i=F_i+\mathrm{Score}_x$
```

The feature score (F_i) is used to calculate the weight of feature$_i$ to determine how much the feature$_i$ distance contributes to the overall distance calculation when retrieving the next set of top-k similar images. One inherent problem with MARS is that the intra-feature re-weighting approach cannot support features that cannot be represented in fixed-length vectors. In addition, the set of possible values for Score in the previous steps is determined arbitrarily. Most importantly, the information obtained to re-weight a feature is limited to the size of the intersection between S and S_i in the inter-feature re-weighting procedure. The inability to use the information of the entire image database leads to the degradation of the accuracy of the system.

The approach proposed in [25] uses an intra-feature re-weighting technique that includes the use of non-relevant images. A discriminant ratio dr_{ij} is used to determine the ability of component j of feature$_i$ to separate non-relevant from relevant images and is calculated by Eq. 1, where m is the number of non-relevant images and $or\left(f_{ij}^{non-rel,l}\right)$ is the value of the jth component of the lth non-relevant image that is outside the range of values for the jth component of relevant images, and 0 otherwise.

$$dr_{ij} = 1 - \frac{\sum_{l=1}^{m} or\left(f_{ij}^{non-rel,l}\right)}{\sum_{l=1}^{m} f_{ij}^{non-rel,l}} \tag{1}$$

The weight w_{ij} for the jth component of the ith feature representation is then determined using Eq. 2, where θ_{ij}^{rel} is the standard deviation of the jth component of the ith feature among the relevant images.

$$w_{ij} = \frac{dr_{ij}}{\theta_{ij}^{rel}} \tag{2}$$

The inter-feature re-weighting, or determination of the weight for each feature$_i$, is performed using Eq. 3, where K is the number of features and δk is the total distance between the kth feature of the query image and those of the relevant images.

$$W_i = \sum_{k=1}^{K} \sqrt{\frac{\delta_k}{\delta_i}} \tag{3}$$

Later works on relevance feedback and feature re-weighting [1, 2, 11, 24] also assume that features can be represented in fixed-length vectors. Following this assumption the components of the features are all placed into a single vector. The corresponding components of each image's feature vector are analyzed to determine how much each element contributes to the distance between the query image and the images in the database. The distance calculations are performed using a weighted Euclidean-type function as illustrated with Eq. 4, where $\overrightarrow{q}$ is the feature vector for the query image, $\overrightarrow{f}$ is the feature vector for an image in the database, and **W** is a matrix that contains the weights for the vector components. Similar to the intra-feature re-weighting approach of MARS, these feature re-weighting approaches cannot support features that cannot be represented in fixed-length vectors. The similarity between images using features such as [6, 10, 17, 18] which have representations of varying length are calculated by performing a best-fit pair-wise matching approach. Best-fit pair-wise matching algorithms attempt to find the closest matching of pairs between the components of two feature representations, then compute a distance metric using a function that measures the quality of matches. The feature components that were not paired, due to the difference in length of the feature representation, would be factored in to increase the distance that is calculated.

$$\text{distance (query image, database image)} = \left(\overrightarrow{q} - \overrightarrow{f}\right)^T \cdot w \cdot \left(\overrightarrow{q} - \overrightarrow{f}\right) \tag{4}$$

More recent research, such as that of [7] proposes approaches that support feature representations of varying length while incorporating the training mechanism of a support vector machine (SVM) to improve retrieval quality. While SVMs cannot support feature representations of varying lengths, [7] utilizes vectors for each image representation where the vector elements are the distances of a given image to each of the training images. With such fixed length vectors SVMs can be incorporated in a standard manner. The downside of using the approach, however, is the possible degradation of performance that coincides with the increase in the number of training images and images in the dataset.

Feature re-weighting mechanisms obtain feedback information via a graphical user interface. In some systems [4, 23], one-class feedback is utilized, where a user simply identifies the relevant images in the result set. However, these approaches lack the ability to extract as much information from the user feedback as systems with more elaborate user interfaces. Some systems [1, 3, 21] incorporate multi-class feedback to obtain more information via user feedback. The drawback in using multi-class feedback is the burden on the user having to judge the degree of relevance or non-relevance of each returned image. Another work [8] has extended the interface further so that the user places the most similar results at the center of the window, and the further an image is from the center, the less relevant it is considered. However, such an interface is too complex.

3 Obtaining distances and re-weighting features

In this section, we go into detail regarding the proposed relevance feedback mechanism for feature re-weighting which we introduced in [16]. First, the distance calculation model will be described to provide insight into how the feature data corresponding to images are evaluated to determine their similarity to the query image.

3.1 Distance model

Since the distances for different feature representations are not of comparable magnitudes, the distances are normalized using Eq. 5. f_i(query,dbImage) corresponds to the distance between the query image and the database images with respect to feature$_i$, for i=1…n, where n is the number of features used. As a result, the distances for each feature will be in the range [0, 1] and the overall distance between the query image and the database images will be in the range [0, n].

$$d_{feature\_i} = \frac{f_i(\text{query}, \text{dbImage})}{\max(f_i(\text{query}, \text{dbImage}))} \tag{5}$$

Figure 3 illustrates the model used to determine the distance between a query image and the images stored in the database. The f_i(query, dbImage) represents the distance function for feature$_i$. Each of these distances is associated with a corresponding weight. As mentioned before, the better the feature is at identifying the visual similarity between images, the higher the weight. These feature weights are normalized to sum to one.

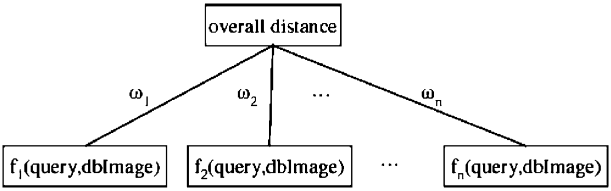


Fig. 3 Distance model

The distances calculated for each feature are multiplied by their associated weight, and then summed to obtain the overall distance value. In effect, the distance is defined by

$$\text{overall distance} = \sum_{i=1}^{n} w_i f_i(\text{query}, \text{dbImage}) \qquad (6)$$

where n is the number of features and w_i is the feature weight for feature$_i$.

The benefit of this distance model is the freedom it provides in how distances are calculated and how the re-weighting information is obtained. Unlike the approaches mentioned in the related work, this model does not restrict the feature data to being represented in vectors of fixed length. In addition, distances do not need to be calculated using a matrix of feature weights, which is the case when using a Euclidean-type distance function. There are no restrictions placed on the data structures used to represent the features and their associated distance measures.

3.2 User interface

The user interface for this work is designed so that the user simply clicks on the relevant images that are returned. By appearance it is one-class feedback, but in essence it provides multi-class feedback since objects that are not identified as relevant are assumed to be non-relevant. From the user feedback, the system obtains two sets of images, the relevant and the other non-relevant, which are analyzed to update the feature weights.

3.3 Basis of proposed approach

The data obtained from the two sets determined via user feedback are used to update the weights for the features by means of two-sided confidence intervals [8]. Two-sided confidence intervals are used to make inferences on the differences between two population proportions. The use of two-sided confidence intervals assumes the sample data are obtained from a population with normal distribution since confidence intervals are based on the central limit theorem. Certain features may be optimal in a given image search, however, while others are ineffective. Since each feature is equally weighted for the first iteration, there is no bias placed on any feature. As a result, this leaves a degree of randomness in the images retrieved for the initial retrieval.

An example of how two-sided confidence intervals are used is first provided for clearer understanding. Following that, an explanation of how they are used in the proposed approach for feature re-weighting will be given.

Suppose there are two drugs, drug A and drug B, that have been developed to treat some particular symptoms. Two clinical trials will be performed to test their effectiveness. One group of n patients is given drug A and a separate group of m patients is given drug B. From the results of the clinical trials, let $p_A^{'}$ be the success rate for drug A and $p_B^{'}$ be the success rate for drug B.

Let p_{A} be the actual success rate for drug A and p_{B} be the actual success rate for drug B. However, the values for p_{A} and p_{B} cannot be obtained since it is not feasible to test every patient that exists around the world. What are available are the results from the clinical trials, thus, two-sided confidence intervals are used to approximate the range for the difference between p_{A} and p_{B}.

The bounds of a two-sided confidence are calculated as follows:

$$p_A - p_B \in (lb, ub) \tag{7}$$

$$lb = p'_A - p'_B - z_{\alpha/2}\sqrt{\frac{p'_A\left(1-p'_A\right)}{n} + \frac{p'_B(1-p'_B)}{m}} \tag{8}$$

$$ub = p'_A - p'_B + z_{\alpha/2}\sqrt{\frac{p'_A\left(1-p'_A\right)}{n} + \frac{p'_B(1-p'_B)}{m}} \tag{9}$$

In Eqs. 8 and 9, $z_{\alpha/2}$ is the confidence coefficient that is dependent on the confidence desired. The choice of the confidence coefficient is arbitrary. A 95% confidence interval does not mean that there is a 95% probability that the interval contains the true difference between the effectiveness of drug A and drug B. Rather, for a 95% confidence interval, if many patients are tested and confidence intervals are calculated continuously as more patients are tested, then in the long run, 95% of these intervals would contain the true difference in effectiveness between drug A and drug B.

For the difference in the success rates between the two drugs, the confidence interval must lie somewhere in the range [−1, 1] since the success rate for a drug is 0 if it was effective for none of the patients, and 1 if it was effective for every patient.

The location of the confidence interval determines the conclusion that can be derived between the effectiveness of the two drugs.

- If the confidence interval lies completely in the positive range, then with certain confidence we know that the effectiveness of drug A is approximately (lb × 100) to (ub × 100)% greater than drug B.
- If the confidence interval covers both positive and negative values, then we cannot be sure that drug A is any more effective than drug B.
- If the confidence interval lies completely in the negative range, then with certain confidence we know that the effectiveness of drug B is approximately (|ub| × 100) to (|lb| × 100)% greater than drug A.

3.4 Feature re-weighting via two-sided confidence intervals

The proposed re-weighting technique is based on the use of two-sided confidence intervals to approximate the difference between the feature distances for the sets of relevant and non-relevant images.

The user provides a query image for which the system must retrieve the top-k most similar images in the database. However, the images considered most similar by the system with regards to the calculated distances may not reflect the user's perspective of visual similarity. With respect to each feature$_i$ individually, it is not possible to determine the exact average feature$_i$ distance for all relevant images in the database $d_i(r)$ and the same for non-relevant images $d_i(nr)$ since the user cannot be expected to check every image in the database. By means of the user feedback on the returned images, the system can calculate values for an approximation of the average feature distance for relevant and non-relevant images. Let d_i'be the average feature$_i$ distance for the images marked relevant in the result set and $d'_i(nr)$ be the average feature$_i$ distance for the remaining images that are assumed to be non-relevant. Using the values $d'_i(r)$ and $d'_i(nr)$, the range for the difference between $d_i(r)$ and $d_i(nr)$ can be approximated via two-sided confidence intervals. In other words, we approximate feature distances for the relevant and non-relevant images in the database using the returned set as a small sample.

The bounds of a two-sided confidence for feature re-weighting are calculated as follows:

$$d_i(r) - d_i(nr) \in (lb, ub) \tag{10}$$

$$lb = d'_i(r) - d'_i(nr) - z_{\alpha/2}\sqrt{\frac{d'_i(r)(1 - d'_i(r))}{n} + \frac{d'_i(nr)(1 - d'_i(nr)}{m}} \tag{11}$$

$$ub = d'_i(r) - d'_i(nr) + z_{\alpha/2}\sqrt{\frac{d'_i(r)(1 - d'_i(r))}{n} + \frac{d'_i(nr)(1 - d'_i(nr)}{m}} \tag{12}$$

Similar to the previous scenario, for Eqs. 11 and 12, $z_{\alpha/2}$ is the confidence coefficient corresponding to the confidence interval desired. n is the number of images marked relevant by the user and m is the number of images considered non-relevant (i.e. $m=k-n$). Again, for the difference between the average feature distance for relevant images and that for non-relevant images, the confidence interval must lie somewhere in the range [−1, 1] since the feature distances have been normalized to a maximum value of 1.

The location of the confidence interval determines how feature$_i$ will be re-weighted. Observing the location of the upper and lower bounds, one of the following cases will arise:

- If both the upper bound and lower bound are greater than 0, then the approximate average feature$_i$ distance for all relevant objects in the database, for the feature under consideration, is greater than that for the non-relevant images. As a result, one can infer that this feature does not appropriately capture visual similarities for this query and its weight is set to 0.
- If the bounds straddle 0, that is, if the upper bound is positive and the lower bound is negative, one can infer that the feature is somewhat good, but cannot fully distinguish relevant images from those that are non-relevant. In this case, the feature weight is set using Eq. 13. As can be seen, the further the confidence interval slides into the negative range, the better the feature must be at distinguishing relevant images from non-relevant images. Likewise, Eq. 13 reflects this, as the weight increases the further the interval slides into the negative region.
- If both the upper bound and lower bound are less than 0, then the approximate average feature$_i$ distance for all relevant images in the database is smaller that that for the non-relevant images. Thus, one can infer this feature can successfully distinguish visual similarities and the feature weight is determined using Eq. 14. As in the previous case, the further negative the confidence interval is, then one can infer, the better the feature can distinguish the user's perspective of visual similarity. Thus, the upper bound is placed in the numerator to reflect this characteristic. Also, the closer the lower bound approaches −1, again it reflects the characteristic of the more negative the interval, the better the feature. Thus, 1-|lb| is placed in the denominator. Finally, the boundary condition where the upper bound is 0 must be considered. The feature weight must be equal to that when

using Eq. 13 for when the upper bound is 0. When the upper bound is 0, using Eq. 13, the feature weight is set to 1. Thus, we add the constant 1.

$$\text{Feature weight} = \text{length of interval in negative range}/\text{total interval length} \tag{13}$$

$$\text{Feature weight} = 1 + \frac{|ub|}{1 - |lb|} \tag{14}$$

An illustration of each of the cases described above is provided in Fig. 4.

From the above three cases, we can observe that the lower in value the midpoint of the confidence interval, the higher the corresponding feature weight. The following theorem states that Eqs. 12 and 13 satisfy this requirement.

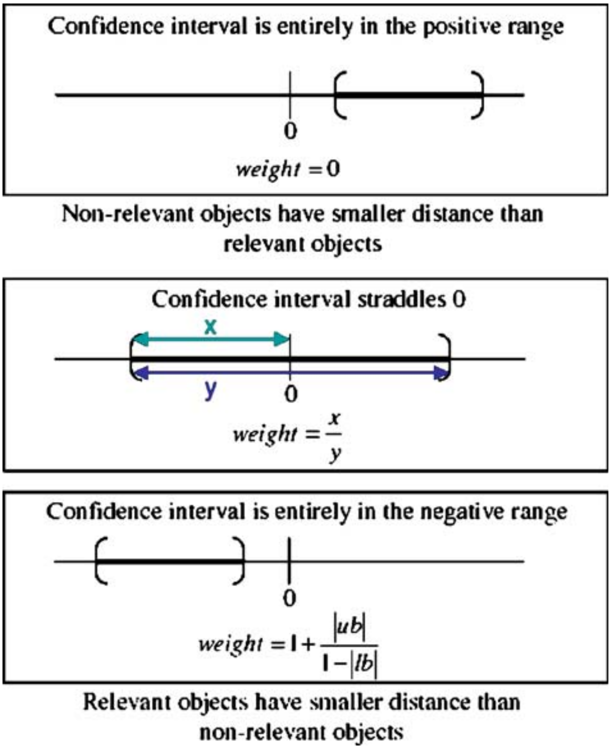
Theorem 1: *Let* lb_1 *and* lb_2 *be the lower bounds for confidence intervals 1 and 2, respectively, and let* ub_1 *and* ub_2 *be the upper bounds. Then*

$$\frac{lb_1 + ub_1}{2} < \frac{lb_2 + ub_2}{2} \Leftrightarrow f(lb_1, ub_1) > f(lb_2, ub_2)$$

where $f(\mathrm{lb}_i, \mathrm{ub}_i)$ is the function used to determine the feature weight. The proof for Theorem 1 is provided in Appendix 1.

Having obtained the updated weight for each feature, the feature weights are normalized by total weight, i.e., the features weights are normalized to sum to 1.

Fig. 4 Possible locations of confidence intervals approximating $d_i(r) - d_i(nr)$


Confidence interval is entirely in the positive range
0
weight =0
Non-relevant objects have smaller distance than relevant objects
Confidence interval straddles 0
x
y
0
$weight = \frac{x}{y}$
Confidence interval is entirely in the negative range
0
$weight = 1 + \frac{|ub|}{1 - |lb|}$
Relevant objects have smaller distance than non-relevant objects

From the first iteration of retrieval, the system has only the single set of feedback to update the weights. When the feature weights are updated, the next iteration of retrieval will return a different set of images. Then, for the following iterations, instead of using the distance information obtained from the images in the most recent iteration of retrieval, the information from the images returned in the previous iterations can be used as well.

The following is a description of the notation that will be used to describe the images that are used to obtain the distance information for feature re-weighting.

K	the number of iterations of retrieval
R_k	the set of images returned in the kth iteration
R_k^{rel}	the relevant images from the kth iteration
$R_k^{\mathrm{non-rel}}$	the non-relevant images from the kth iteration
R_k^{unique}	the set of unique images that have been returned up to the kth iteration
$R_k^{\mathrm{unique,rel}}$	the set of unique images identified as relevant in the k iterations
$R_k^{\mathrm{unique,non-rel}}$	the set of unique images identified as non-relevant in the k iterations

Take note

$$R_k = R_k^{\mathrm{rel}} \cup R_k^{\mathrm{non-rel}},$$
$$R_k^{\mathrm{unique}} = R_k^{\mathrm{unique,\,rel}} \cup R_k^{\mathrm{unique,\,non-rel}},$$
$$R_l^{\mathrm{unique}} = R_1,$$
$$R_l^{\mathrm{unique,\,rel}} = R_l^{\mathrm{rel}}, \text{ and}$$
$$R_l^{\mathrm{unique,\,non-rel}} = R_l^{\mathrm{non-rel}}$$

Then for iterations $k+1$, where $k>0$, $R_{k+1}^{\mathrm{unique}}$ and $R_{k+1}^{\mathrm{unique,\,non-rel}}$ are defined as follows.

$$R_{k+1}^{\mathrm{unique}} = \mathrm{R_{k+1}} \cup \mathrm{R_k^{unique}}$$
$$R_{k+1}^{\mathrm{unique,\,rel}} = R_{k+1}^{\mathrm{rel}} \cup R_k^{\mathrm{unique,\,rel}}$$
$$R_{k+1}^{\mathrm{unique,\,non-rel}} = R_{k+1}^{\mathrm{non-rel}} \cup R_k^{\mathrm{unique,\,non-rel}}$$

Thus, to incorporate the information obtained from the previous retrieval iterations for feature re-weighting, the images in $R_{k+1}^{\mathrm{unique,\,rel}}$ and $R_{k+1}^{\mathrm{unique,\,non-rel}}$ are used to calculate the confidence intervals at the (k+ *1*)th iteration. More specifically, when using Eqs. 11 and 12 to update the weights at the (k+ *1*)th iteration, $d_i'(rel)$ and $d_i'(non-rel)$ are now obtained using the feature distances for the images in $R_{k+1}^{\mathrm{unique,\,rel}}$ and $R_{k+1}^{\mathrm{unique,\,non-rel}}$, respectively. Also, n now corresponds to the number of images in $R_{k+1}^{\mathrm{unique,\,rel}}$ and m corresponds to the number of images in $R_{k+1}^{\mathrm{unique,\,non-rel}}$.

Note that as the sets of relevant and non-relevant images are generated through the retrieval iterations, the number of images used to obtain the sample data for the two-sided confidence calculation continues to increase. This justifies the assumption of a normal distribution of the distances of the image features. Also, as the sets of sample data increase with each iteration, the approximation of the two-sided confidence intervals becomes more accurate. This is a favourable characteristic as the retrieval performance should improve at each iteration.

4 Experimentation

In this section, the experimental results are presented to validate the effectiveness of the proposed feature re-weighting technique. In addition, the retrieval accuracy using the feature re-weighting techniques of MARS, VisiMine [24], and [25] which will be referred to as DD, are compared with that of the proposed approach.

The experimentation was performed on the Windows platform powered by a Pentium 4 2.6 GHz CPU using 512 MB of RAM. The prototype system is implemented using C++ and the.NET framework. Images and their associated feature data are stored to an Oracle 10 g database located remotely. The image dataset used is based on the shape silhouettes used for the MPEG-7 Core Experiment CE-Shape-1 part B [14]. The dataset consists of 8,400 images that are categorized into 70 groups. The filenames identify the group to which an image belongs. The images are monochrome and consist of a single object and a plain background. As a result, any challenges with image segmentation are completely avoided. The dataset contains the original 1,400 images and modifications of this set by scaling to 50% and 200% and rotating the images by 90°, 180°, and 270°. A subset of the images is illustrated in Appendix 2.

In a CBIR system, multiple features can be extracted from an image to provide a low level description of the object contained within. The set of features used for each of the comparison systems in the experimentation is described below. Only a brief description will be provided for each. Further details can be found in the references.

- Polar Projections [13] capture the details of a shape based on outward projections from the center. The resolution of the projections is user-definable and has been set to 120 for this experimentation.
- Curvature Scale Space [18] represents the curvature zero-crossing points of a contour as it evolves. For CSS extraction, this requires contour evolution (i.e. applying a filter to a convex shape to gradually morph it into an ellipse). During this process the curvature is measured and the zero-crossing points are captured.
- Eccentricity is the ratio of the major axis to the minor axis.
- Compactness is defined as the ratio of the squared perimeter to the area, i.e., compactness = square perimeter/area
- Perimeter is the length of the outer boundary of the object in the image.
- Circularity is defined as the ratio between the perimeter of the object and the perimeter of a circle of equivalent area, i.e., circularity=square perimeter/(4π×area)

For the above features, with the exception of curvature scale space which uses a best-fit pair-wise matching algorithm to determine the distance between two image representations, the Euclidean distance is used. Note, the VisiMine system cannot support variable length feature representations and thus is excluded from the experimental comparisons involving the Curvature Scale Space feature.

Each of the comparison systems is setup for an image retrieval session to be initiated by providing a single image to be used as the query. Of the 120 images that are returned by each system per iteration, relevant images are determined based on their groupings.

In each of the experimental comparisons, any images not identified by the user as being relevant are automatically excluded as candidates for retrieval in the following iterations. The benefit of this approach is intuitive as there is no reason for the user to see a non-relevant image more than once. The measures used throughout this section to describe retrieval accuracy are recall and precision which are defined as follows:

$$\text{Recall} = \text{retrieved}_{\text{relevant}}/\text{total}_{\text{relevant}}$$
$$\text{Precision} = \text{retrieved}_{\text{relevant}}/\text{retrieved}$$

where $\text{retrieved}_{\text{relevant}}$ represents the number of relevant images retrieved for the given iteration, $\text{total}_{\text{relevant}}$ represents the number of images in the database that are relevant to the query, and retrieved represents the number of images retrieved for the given iteration.

As 95% is the most commonly used percentage for two-sided confidence intervals, such confidence intervals will be used for the remaining experimental results. The results presented are the average over 50 queries with the top 120 similar images retrieved per iteration.

Figure 5 compares the recall of the proposed approach, which is denoted as CIA for confidence interval approach, to that of MARS, VisiMine and DD. Existing re-weighting

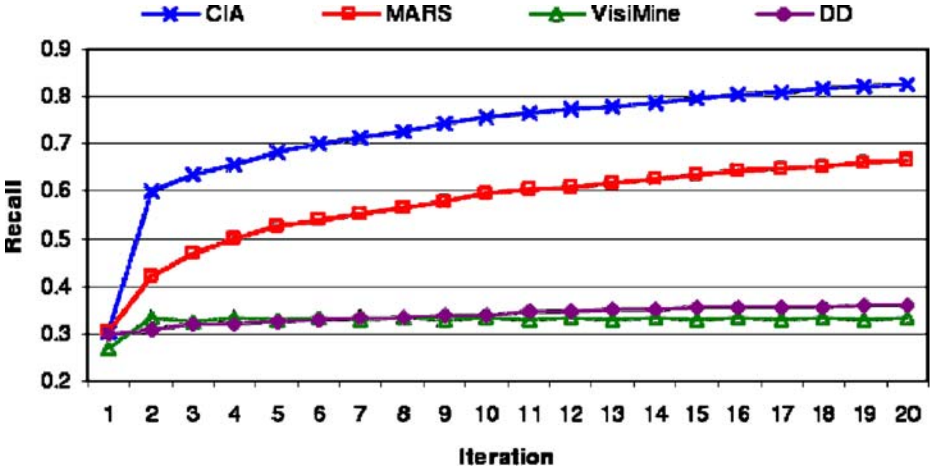


Fig. 5 Recall-Iteration comparison with MARS, VisiMine, and DD

techniques assume that feature representations are described in vectors of the same length and type (e.g. MARS, VisiMine and DD). To allow the use of the intra-feature re-weighting techniques of MARS, VisiMine and DD there are several changes that were required. Firstly, the CSS feature is not considered as it cannot be described in a fixed-length vector since the number of peaks extracted from various images is unlikely to be the same. In addition, the best-fit pair-wise matching scheme used to calculate distances between CSS feature representations is not supported by the comparison systems. Secondly, for the polar projections of an image, the angle offset must be stored so that the rotational orientation of the polar projection representations for the retrieved images can be matched to that of the query image for which their minimal distances were calculated (e.g., to ensure the corresponding elements of each image's feature vector are in corresponding positions). It can be seen that CIA outperforms the comparison systems. The performance of VisiMine is substantially lower than CIA since it uses a weighted-Euclidean type distance function without performing any normalization on the values of the vector components. Thus, the re-weighting of the feature vector elements is less effective since the values of the components may not necessarily be of comparable magnitudes.

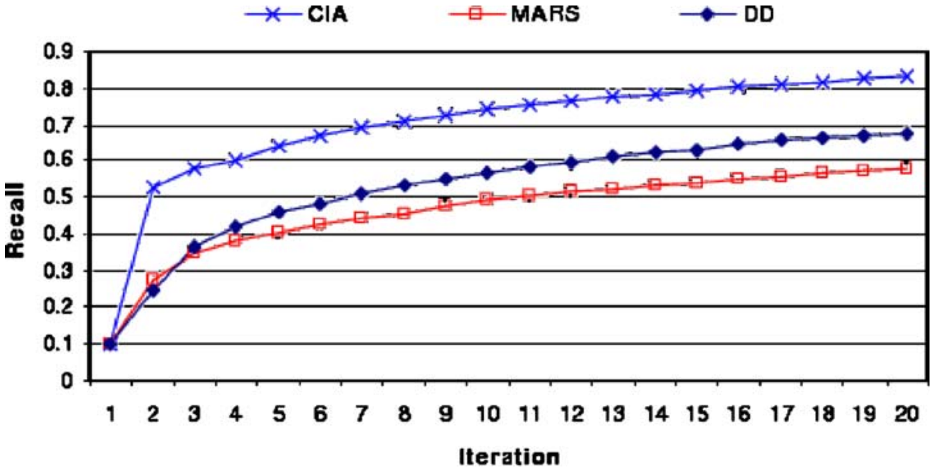


Fig. 6 Recall-Iteration comparison with MARS and DD

Table 1 Improvement in recall

	Iteration 2	Iteration 5	Iteration 10	Iteration 20
Figure 5				
CIA	0.2987	0.3782	0.4538	0.5237
MARS	0.1192	0.2223	0.2910	0.3623
Figure 6				
CIA	0.4288	0.5423	0.6407	0.7307
DD	0.1502	0.3592	0.4698	0.5777

The poor retrieval performance of DD can be attributed to the small range of values in the crossings components of the polar project feature representation which affects its intra-feature re-weighting mechanism. For the scoring scheme used by MARS, the images marked relevant by the user are assigned 1, while the remainder are assigned 0.

To demonstrate the advantage of our re-weighting approach over existing approaches, we incorporate the use of the CSS feature which cannot be represented using a fixed length vector and does not use a Euclidean-type distance. As a result, VisiMine is eliminated from the comparison, and the use of MARS and DD is restricted to their inter-feature re-weighting techniques. Figure 6 shows a comparison of the recall performance between CIA, MARS and DD. Notice how the performance of DD has drastically improved when its intra-feature re-weighting is not used; showing how its intra-feature re-weighting does not effectively support the polar projection feature representation. Our proposed approach continues to succeed in improving retrieval accuracy and continues to outperform the comparison systems.

Table 1 displays the amount of improvement in recall after 2, 5, 10 and 20 iterations of retrieval in the last comparison. The lower performance of MARS can be attributed to the fact that it is unable to use information based on the entire image database, but use information based only on the returned images. Table 2 provides a quantitative value for the amount of improvement in recall that CIA has over the comparison systems at the second, fifth, tenth and 20th iterations. The values are calculated using Eq. 15. It is apparent that CIA provides a significant improvement in retrieval accuracy in the early iterations of an image search session. This is a desirable characteristic as it is favourable for users to receive improved results without having to provide much feedback.

$$\frac{\text{Improvement} = \text{recall improvement}_{\text{CIA}} - \text{recall improvement}_{\text{other}}}{\text{Recall improvement}_{\text{other}}} \tag{15}$$

Figure 7 provides precision-recall plots for CIA at the first, second, fifth, tenth and 20th iterations. As each of the top-k images are retrieved in order, the precision and recall values are plotted. It can be seen that as the iterations progress, the relevant images continue to move to the front of the retrieved images. Comparisons of the precision-recall plots between CIA, MARS, and DD for the second, fifth, tenth, and 20th iterations are provided in Fig. 8. In the precision-recall plots, it shows that CIA outperforms MARS and DD since the precision-recall plots for CIA lie above the majority of the plotted points for MARS and DD.

Table 2 Improvement over MARS and DD

	Iteration 2	Iteration 5	Iteration 10	Iteration 20
Figure 5				
Vs MARS	1.5059 (151%)	0.7013 (70%)	0.5595 (56%)	0.4455 (45%)
Figure 6				
Vs DD	1.8549 (185%)	0.5097 (51%)	0.3638 (36%)	0.2648 (26%)

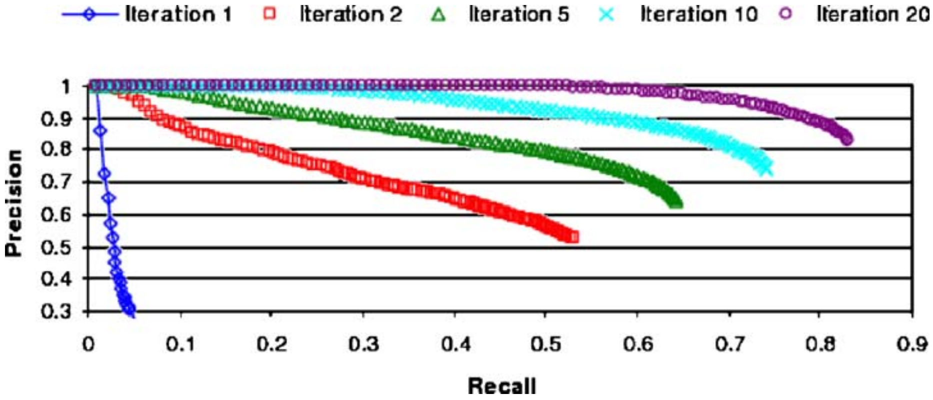


Fig. 7 Precision-Recall plots for CIA

Figure 9 provides some insight into the number of unique images R_k^{unique} that are retrieved over the iterations. Notice that the number of images retrieved for CIA is much smaller. This illustrates the ability for CIA to provide improved recall performance while obtaining feedback information from fewer images than the comparison systems.

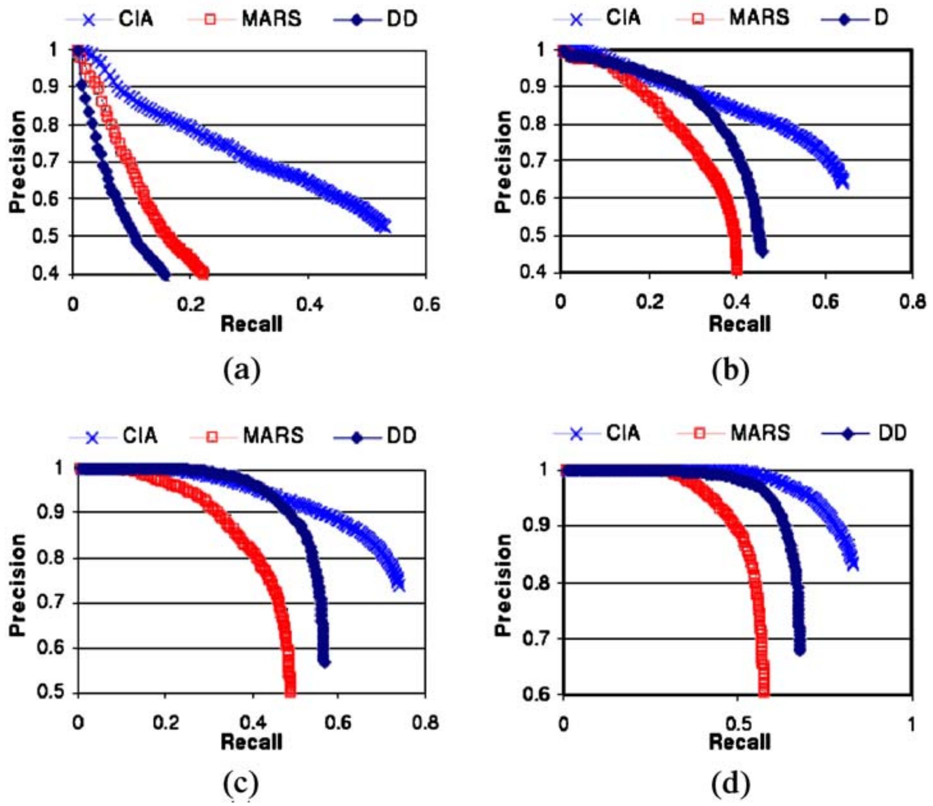


Fig. 8 Precision-recall comparison between CIA, MARS and DD for **a** second iteration, **b** fifth iteration, **c** tenth iteration, **d** 20th iteration

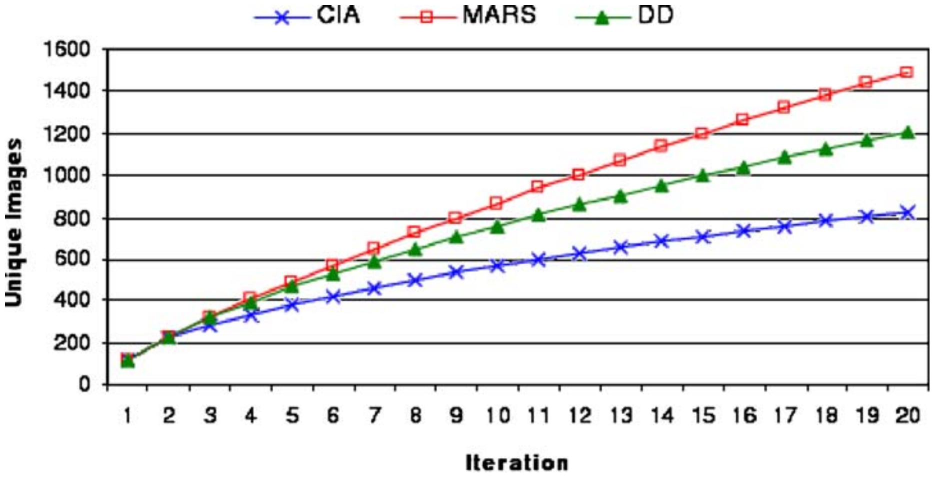


Fig. 9 Number of unique images retrieved

To evaluate the effect of marking only a subset of the relevant images as feedback per iteration, we performed additional experimentations. When limiting the number of images marked relevant to a maximum of 15, 30, and 60 images, the recall after 10 iterations is 12.5%, 25%, and 50% respectively. Therefore, the user must mark all relevant images for each iteration for optimal results.

For another experimental environment, 20 images are retrieved per iteration. This environment may be applied when the total number of relevant images in a database is smaller or the user opts to view fewer results at a time. The recall and precision results are listed in Tables 3 and 4. As expected, the recall values are lower since the maximum achievable recall when retrieving 20 images is 0.167. Conversely, the precision values are higher since the number of retrieved images is significantly smaller than the previous experimental environment. Similar to the results of the previous environment, by using all 20 returned images as feedback, CIA outperforms MARS and DD, both of which obtain data for feature re-weighting from only the images marked relevant.

The time taken by each approach for the process of feature re-weighting was measured. For all approaches, each iteration of feature re-weighting registered zero time, where the smallest time unit measurable is 100 ns. Since the scope of this paper is feature re-weighting, indexing was not incorporated for any of the approaches.

5 Conclusion

This paper proposes a feature re-weighting technique for content-based image retrieval systems that incorporate relevance feedback. Unlike many earlier works, feature representations are not required to have a fixed-length representation for each image. Instead, the feature distances are used with the statistical technique of two-sided confidence

Table 3 Recall for top-20 retrieval

	Iteration 1	Iteration 2	Iteration 5	Iteration 10
CIA	0.0353	0.1051	0.1365	0.1404
MARS		0.0970	0.1244	0.1304
DD		0.0590	0.0897	0.0987

Table 4 Precision for top-20 retrieval

	Iteration 1	Iteration 2	Iteration 5	Iteration 10
CIA	0.2115	0.6308	0.8192	0.8423
MARS		0.5821	0.7464	0.7821
DD		0.3538	0.5385	0.5923

intervals to update the feature weights. In addition, it places no restrictions on the distance functions used for a feature. Each feature can use the distance function that is specifically designed for the feature instead of being restricted to a Euclidean-type distance function.

Another advantage of the proposed approach is the simplicity of the user feedback. The user simply identifies the relevant images to provide feedback. Since two-sided confidence intervals perform inferences on two populations, we only require the user to identify the relevant images in the result set, and consider the remainder to be non-relevant.

The experimental results show that the proposed feature re-weighting approach provides effective feature re-weighting regardless of the structure of the feature representations. One desirable characteristic that is evident from the results is that the proposed approach provides a significant improvement in the early iterations of retrieval, which is evident with the on average 151% improvement over MARS for the 2nd iteration of retrieval. The proposed approach continued to outperform MARS on average 70% at the 5th iteration, 56% at the 10th iteration, and 45% at the 20th iteration of retrieval. In the case where a feature representation of varying length was used, our approach outperformed DD on average by 185% at the second iteration, 51% by the fifth iteration 36% for the tenth iteration and 26% at the 20th iteration.

Acknowledgments This research was supported by the Defense Acquisition Program Administration and the Agency for Defense Development, Korea, under the contract UD030000AD, through the Image Information Research Center at Korea Advanced Institute of Science and Technology.

Appendix

Appendix 1 Proof of Theorem 1

Since the proof for Eq. 12 is straightforward, we will omit the proof for Eq. 12.

Every possible case for the placement of two confidence intervals in the negative range must be considered. Consider Fig. 10a where both bounds for interval 2 are greater than interval 1. Then shift the lower bound of interval 2 to be in the range of interval 1 to obtain case (b). Next, further shift the lower bound of interval 2 so that it is more negative than interval 1 to obtain case (c). All cases where the upper bound of interval 2 is greater than interval are covered. Thus, shift the upper bound of interval 2 into the range of interval 1. If the lower bound of interval 2 is also placed inside the range of interval 1, case (d) is obtained. Then, shift the lower bound of interval 2 so that it is more negative than interval 1 to obtain case (e). Finally, shift the upper bound of interval 2 to be more negative than interval 1, and this leaves only the possibility of the lower bound of interval 2 being more negative than interval 1 as displayed in case (f). As can be seen, case (c) and (d) are the same, as well as case (b) and (e), and likewise for cases (a) and (f). This leaves three cases that must be checked.

- Case 1: Independent (Fig. 10a and f). Figure 11 illustrates the case where there are two confidence intervals that do not overlap.

  Let

1. $\mathrm{lb}_1+x=\mathrm{lb}_2$,
2. $\mathrm{ub}_1+y=\mathrm{ub}_2$

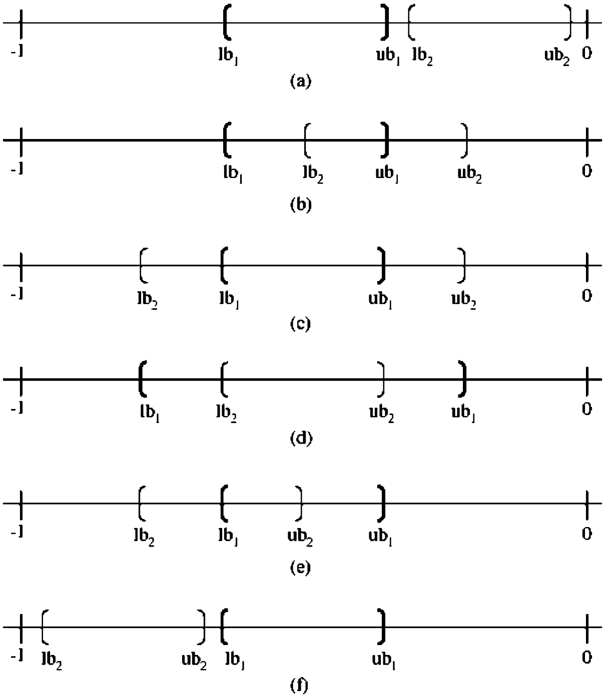


Fig. 10 Possible cases for confidence intervals in the negative range

where

3. $-1<\mathrm{lb}_1<\mathrm{ub}_1\le 0$,
4. $-1<\mathrm{lb}_2<\mathrm{ub}_2\le 0$,
5. $x>0$,
6. $y>0$.

Then

$$\begin{aligned}
\frac{lb_1+ub_1}{2}<\frac{lb_2+ub_2}{2} &\Leftrightarrow 1+\frac{|ub_1|}{1-|lb_1|}>1+\frac{|ub_2|}{1-|lb_2|} \\
ub_1+lb_1<ub_2+lb_2 &\Leftrightarrow 1+\frac{|ub_1|}{1-|lb_1|}>1+\frac{|ub_2|}{1-|lb_2|} \\
ub_1+lb_1<ub_2+lb_2 &\Leftrightarrow \frac{|ub_1|}{1-|lb_1|}>\frac{|ub_2|}{1-|lb_2|} \\
ub_1+lb_1<ub_2+lb_2 &\Leftrightarrow \frac{|ub_1|}{1+lb_1}>\frac{|ub_2|}{1+lb_2} \\
ub_1+lb_1<ub_2+lb_2 &\Leftrightarrow \frac{ub_1}{1+lb_1}<\frac{ub_2}{1+lb_2} \\
ub_1+ub_1lb_2 &< ub_2+ub_2lb_1
\end{aligned} \tag{16}$$

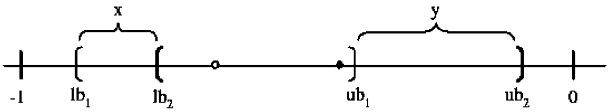
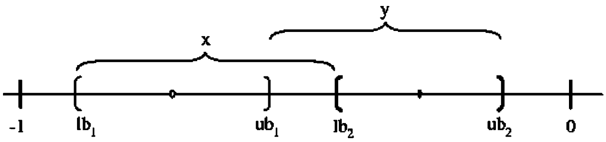


Fig. 11 Confidence intervals with no overlap

Fig. 12 Overlapping confidence intervals



$ub_1 + ub_1 lb_1 + ub_1 x < ub_2 + ub_2 lb_1$ by substituting (1) $ub_1 + ub_1 lb_1 + ub_1 x < ub_1 + y + ub_1\, lb_1 + lb_1 y$ by substituting (2) $ub_1\, x < y + lb_1\, y$

$$0 < y + lb_1\, y - ub_1\, x$$

$0 < y(1 + lb_1) - ub_1\, x$ is true considering (3), (4), (5), (6).

- Case 2: Overlapping (Fig. 10b and e)

Figure 12 illustrates the case where there are two confidence intervals that have partial overlap. Let

1. $\mathrm{lb}_1 + x = \mathrm{lb}_2$,
2. $\mathrm{ub}_1 + y = \mathrm{ub}_2$

where

3. $-1 < \mathrm{lb}_1 < \mathrm{ub}_1 \leq 0$,
4. $-1 < \mathrm{lb}_2 < \mathrm{ub}_2 \leq 0$,
5. $x > 0$,
6. $y > 0$.

Starting from Eq. 16,

$$ub_1 + ub_1 lb_2 < ub_2 + ub_2 lb_1$$

$ub_1 + ub_1 lb_1 + ub_1 x < ub_2 + ub_2 lb_1$ by substituting (1)$ub_1 + ub_1 lb_1 +\ ub_1 x < ub_1 + y + ub_1 lb_1 y$ by substituting (2)

$$ub_1\, x < y + lb_1\, y$$

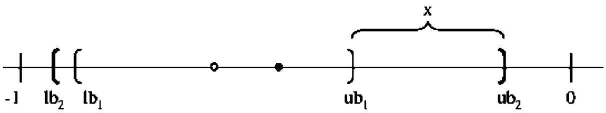
$$0 < y + lb_1\, y - ub_1\, x$$

$0 < y(1 + lb_1) - ub_1\, x$ is true considering (3), (4), (5), (6).

- Case 3: Contained (Fig. 10c and d)

Figure 13 illustrates the case where one confidence interval is fully contained in another confidence interval.

Fig. 13 Confidence interval contained in another



Let

1. $\mathrm{ub}_1 + x = \mathrm{ub}_2$

where

2. $-1 < \mathrm{lb}_1 < \mathrm{ub}_1 \leq 0$,
3. $-1 < \mathrm{lb}_2 < \mathrm{ub}_2 \leq 0$,
4. $x > 0$.

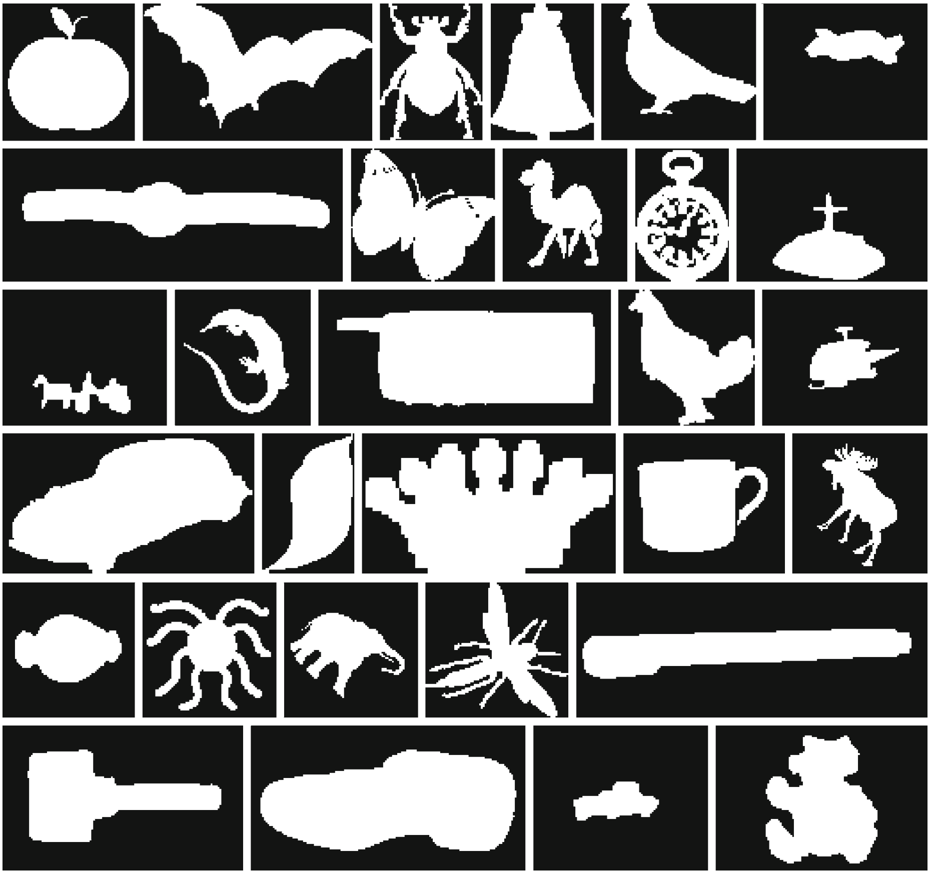
Starting from Eq. 16,

$$ub_1 + ub_1\, lb_2 < ub_2 + ub_2 lb_1$$

$ub_1 + ub_1 lb_2 < ub_1 + x + ub_1\, lb_1 + lb_1 x$ by substituting (1)$0 < x(1 + lb_1)$ is true considering (2), (3), (4).

Appendix 2 MPEG-7 Core Experiment CE-Shape-1 Part B

The following is a subset of the images from the MPEG-7 Core Experiment CE-Shape-1.



References

1. Aggarwal G, Ashwin TV, Ghosal S (2002) An image retrieval system with automatic query modification. IEEE Trans Multimedia 4–2:201–214
2. Cheikh FA, Cramariuc B, Gabbouj M (2003) Relevance feedback for shape query refinement. In Proceedings of the IEEE International Conference on Image Processing, September, Barcelona, Spain, pp 745–748
3. Ciocca G, Schettini R (2001) Content-based similarity retrieval of trademarks using relevance feedback. Pattern Recogn 34–8:103–119
4. Cox IJ, Miller ML, Minka TP, Papathomas TV, Yianilos PN (2000) The Bayesian image retrieval system, pichunter: theory, implementation, and psychophysical experiments. IEEE Trans Image Process 9–1:20–37
5. Faloutsos C, Flickner M, Hafner J, Niblack W, Petkovic D, Equitz W (1994) Efficient and effective query by image content. Journal of Intelligent Information Systems 3–3/4:133–150
6. Ghosh A, Petkov N (2005) Robustness of shape descriptors to incomplete contour representations. IEEE Trans Pattern Anal Mach Intell 27–11:1793–1804
7. Gondra I, Heisterkamp DR (2004) Learning in region-based image retrieval with generalized support vector machines. IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp 149–156
8. Hayter AJ (2002) Probability and statistics for engineers and scientists. Duxbury, Pacific Grove, CA
9. Heesch R, Ruger S (2003) Performance boosting with three mouse clicks—relevance feedback for CBIR. In 25th European Colloquium on IR Research, Springer LNCS 2633, pp 363–376
10. Huang P-W, Lee C-H (2004) Image database design based on 9D-SPA representation for spatial relations. IEEE Trans Knowl Data Eng 16–12:1486–1496
11. Ishikawa Y, Subramanya R, Faloutsos C (1998) Mindreader: querying databases through multiple examples. In Proceedings of the 24th International Conference on Very Large Data Bases, pp 218–227
12. Kim D-H, Chung C-W (2003) Qcluster: relevance feedback using adaptive clustering for content-based image retrieval. In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp 599–610
13. Kwon Y-I, Park H-H, Lee S-L, Chung C-W (2005) A shape feature extraction for complex topographical images. In Proceedings of the International Symposium on Remote Sensing, pp 575–578
14. Latecki LJ (2002) Shape data for the MPEG-7 Core Experiment CE-Shape-1. http://www.cis.temple.edu/~latecki/TestData/mpeg7shapeB.tar.gz
15. Lee K-M, Street WN (2004) Cluster-driven refinement for content-based digital image retrieval. IEEE Trans Multimedia 6–6:817–827
16. Lee RS, Kim S-H, Park H-H, Lee S-L, Chung C-W (2006) A feature re-weighting approach for the non-metric feature space. Journal of Korea Information Science Society: Databases 33–4:372–383
17. Manjunath BS, Ohm JR, Vasudevan VV, Yamada A (2001) Color and texture descriptors. IEEE Trans Circuits Syst Video Technol 11–6:703–715
18. Mokhtarian F, Bober M (2003) Curvature scale space representation: theory, applications, and MPEG-7 Standardization. Springer, New York
19. Porkaew K, Chakrabarti K, Mehrotra S (1999) Query refinement for multimedia similarity retrieval in MARS. Technical Report TR-DB-99-05, University of California at Irvine
20. Porkaew K, Mehrotra S, Ortega M (1999) Query reformulation for content based multimedia retrieval in MARS. Technical Report TR-DB-99-03, University of California at Irvine
21. Rui Y, Huang TS, Ortega M, Mehrotra S (1998) Relevance feedback: a power tool for interactive content-based image retrieval. IEEE Trans Circuits Syst Video Technol 8–5:644–655
22. Smith JR, Chang S-F (1997) VisualSEEk: A fully automated content-based image query system. In Proceedings of the ACM International Conference on Multimedia, pp 87–98
23. Taycher L, La Cascia M, Sclaroff S (1997) Image digestion and relevance feedback in the imagerover WWW Search Engine. In Proceedings of the 2nd International Conference on Visual Information Systems, pp 85–94
24. Tusk C, Koperski K, Aksoy S, Marchisio G (2003) Automated feature selection through relevance feedback. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, pp 3691–3693
25. Wu Y, Zhang A (2002) A feature re-weighting approach for relevance feedback in image retrieval. In Proceedings of the IEEE International Conference on Image Processing, pp 581–584

Robert S. Lee received his B.S. in Computer Science from the University of British Columbia in Vancouver, Canada. In 2006 he received his M.S. in Computer Science from the Korea Advanced Institute of Science and Technology (KAIST) in Daejeon, South Korea. He is currently working as a software developer for Business Objects, an SAP company. His main areas of interest include image retrieval, relevance feedback, and software development life cycles.

Chin-Wan Chung received a Ph.D. degree from the University of Michigan, Ann Arbor in 1983. He was a Senior Research Scientist and a Staff Research Scientist in the Computer Science Department at the General Motors Research Laboratories (GMR). While at GMR, he developed Dataplex, a heterogeneous distributed database management system integrating different types of databases. Since 1993, he has been a professor in the Division of Computer Science at the Korea Advanced Institute of Science and Technology (KAIST), South Korea. At KAIST, he developed a full-scale object-oriented spatial database management system called OMEGA, which supports ODMG standards. His current research interests include multimedia databases, sensor network and stream data management, and the semantic Web.

Seok-Lyong Lee is a professor at School of Industrial and Information Systems Engineering, Hankook University of Foreign Studies. He received his Ph.D. in Department of Information and Communication Engineering from the Korea Advanced Institute of Science and Technology (KAIST) in 2001. He received BS degree in Mechanical Engineering in 1984 and MS degree in Industrial Engineering in 1993 from Yonsei University. He was an Advisory S/W Engineer at IBM Korea from 1984 to 1995. His research interests include Multimedia Databases, Data Mining and Warehousing, and Multimedia Information Retrieval.

Sang-Hee Kim received his B.Sc. and M.Sc. degrees in Computer Science from Sogang University, Korea in 1985 and 1987, respectively, and his Ph.D. degree in Electrical Engineering and Computer Science from Korea Advanced Institute of Science and Technology (KAIST). He is working for Agency for Defense Development (ADD) in the field of satellite image processing and three-dimensional terrain rendering. His main research interests are image database, feature extraction and image registration, real-time three-dimensional terrain rendering, geographical information system, virtual reality.