A collaborative, distributed, long-term learning approach to using relevance feedback in content-based image retrieval systems

Vladimir Nedovic and Oge Marques *

Department of Computer Science and Engineering
Florida Atlantic University
777 Glades Road, Boca Raton, FL 33431-0991
omarques@fau.edu

Abstract — **In recent years, an extensive research in the area of Content-Based Image Retrieval (CBIR) has been focused on Relevance Feedback (RF) techniques to improve the retrieval of images. In relevance feedback systems, a search engine dynamically updates the weights of various visual features in the query based on the user's measure of retrieved images' (ir)relevance. In this paper, we first present current state of the art in this area, and then propose a novel; collaborative relevance feedback approach that relies on the postulation that human-perception subjectivity is narrower than the semantic gap. As relevant results of each query are being remembered by the system, various users over different sessions can contribute to improve the successive query results.**

Index Terms — **collaborative distributed learning, Content-Based Image Retrieval, Relevance Feedback, Query-By-Example, semantic gap, human-perception subjectivity.**

1. Introduction

With the increasing use of digital images and especially the proliferation of the World Wide Web, a vast amount of images has been generated, stored, transmitted, and accessed. In order to make use of such data, efficient methods for searching and retrieving images had to be developed.

As various text search engines have proved to work successfully, the initial efforts in image retrieval were directed towards a keyword annotation of images. In this approach, all images are annotated manually before being stored; they can be retrieved later by searching for the corresponding keywords. However, there are many difficulties with this approach, most notable of which are the huge amount of effort required to annotate every single image in the database and the differences in interpretation (i.e. in keyword choice and assignment) of image's visual contents by different indexers [5]. Furthermore, as the size of image repository increases, the task of manually assigning textual descriptors becomes infeasible.

To avoid manual annotation and automate the process of image retrieval, Content-Based Image Retrieval (CBIR) techniques based on image's visual contents have been developed in the early 1990's. CBIR systems use low-level features (mostly color, texture, and shape) to represent image's content and find similar images during the retrieval process. This approach avoids manual annotation and can be completely automatic because the image's visual contents do not change. CBIR has attracted huge research attention; however, despite all the research efforts, the systems lagged behind the text search engines. There are two main reasons for inefficiency of such systems [5],[8]:

- *The gap between the low level features and high level semantic concepts, i.e. the semantic gap*

  The problem stems from the fact that image primitives, such as color, texture, and shape do not always convey image's meaning; therefore, various visual similarity measures used in the retrieval process, such as color histograms or Fourier descriptors, are not sufficient because they do not necessarily match the semantics ascribed to some image by the user
- *The subjectivity of human perception*

  Different persons, or even the same person in different circumstances, often have different interpretations of the same visual content. This human perception subjectivity exists at various levels: one person may be more interested in image's color feature, while another can give more relevance to the texture feature; even if they are both interested in texture, the way they perceive the similarity of texture can be very different.

To address some of the CBIR shortcomings, a technique called Relevance Feedback has been introduced, in which human and computer iteratively interact to refine high level queries to representations based on low-level image features. The technique has long been successfully applied in document retrieval, but has attracted even more attention in the CBIR community. The reasons for this are certainly the abovementioned semantic gap and human subjectivity problems (i.e. the need for interaction with a user), but also the fact that unlike with text documents, judging an image and deciding whether it is relevant or not does not present a big burden for a user [14],[2]– this makes the feedback

* Contact author

process more sensible and faster. The chief fact, however, that contributed to relevance feedback being the most active research topic in CBIR is that retrieval accuracy of a general CBIR engine was so low that even applying the algorithms developed for document retrieval significantly improved the accuracy [8].

In this paper, we provide a brief overview of various approaches and techniques used in relevance feedback area of CBIR, their nature and limitations. Then we propose a novel RF strategy in which users' relevant results are remembered and their habits learned over time; as retrieved results improve the successive query results, this strategy has the potential of narrowing or even completely bridging the semantic gap. Relying on humans' judgment of semantic relevance, we also try to find meaningful strings within images' filenames and thus incorporate keyword description into the system as well.

This paper is organized as follows: Section 2 gives relevant background information and presents an overview of related work. Section 3 describes our approach in more detail, both from conceptual as well as implementation point of view. Finally, our conclusions and suggestions for future work are in Section 4.

2. Background and related work

Relevance feedback in CBIR is a process of dynamically adjusting an existing query using the information fed-back by the user about the relevance of previously retrieved images such that the adjusted query is a better approximation to the user's need [5]. The goal of such a process is to capture the user's high level query and perception subjectivity by interacting with him/her and automatically adjusting the weights based on the provided feedback. A typical scenario in an RF system is as follows:

1) The user gives a query example image and/or keyword description to the system

2) The system provides initial retrieval results based on some predefined similarity metrics (e.g. based on their overall similarity, using all the features)[8],[5]

3) The user marks the retrieved images by judging whether (and to what degree, in some cases) they are relevant to his/her query

4) Based on the information fed-back by the user, the system refines the query and retrieves a new list of images for him/her – the algorithm is back to Step 3

Further in this section, we present some major variants of relevance feedback techniques. As different methods (all under the "cloud" of Relevance Feedback) have made different assumptions and taken different approaches to the problem, they can be divided into groups by various criteria [2]. We give an overview of such criteria and classify the systems accordingly.

2.1 Target vs. Similarity Search

Most of the early CBIR systems assumed that the user is looking for a class of similar items to the given example image (similarity or category search). However, Cox et al. [15] assume that the user is looking for a particular target item (target search); in the latter approach, the retrieved images marked as relevant by the user are not considered target images, but more similar ("closer") to the target than others (i.e. relative judgment). However, the target search imposes a challenging question to the user, as it is not always easy for a human to tell between two images which one is "closer" to the third one. Furthermore, user consistency is hard to achieve, i.e. it is even more difficult to determine relative distances consistently in accordance with the underlying feature representations adopted by the search engine [2].

2.2 Visual Features vs. Keyword Descriptors

Relevance feedback techniques were introduced into the CBIR community in an attempt to bridge the semantic gap; however, even with the online query learning and refinement it is difficult to capture user's high-level concepts. Therefore, many algorithms were developed that take into account both textual and visual cues in order to arrive at the satisfying results faster.

Some of the approaches only consider images found on the World Wide Web [13], concentrating on the strings found in the IMG and ALT tags, as well as the text around images. In an attempt to facilitate user's labeling, but still enable him/her to give subjective annotations, Nakazato et al. [3] propose annotation by groups instead of performing the task for each particular image. They try to avoid repeating the same keyword assignment for a set of very related images and enable hierarchical annotation. Many others try to include automatic image annotation into the process whenever possible; Santini et al. [10] use some elementary techniques from information retrieval to associate labels with a small subset of images and create a visual dictionary – the textual search is then a starting point for the visual search. Zhou et al. [4] also assume a labeled subset (not necessarily manually annotated) from which they create a real-time thesaurus based on the association information obtained from the feedback (word association based on relevance feedback, or WARF). By using soft vector representation for keywords, they combine textual information with the feature vectors based on visual cues. A similar method is presented in [9].

2.3. Feature Selection

The task of selecting a limited number of low-level visual features and their variants to represent image's semantics is a critical one considering the issue of a semantic gap. Furthermore, users pay more attention to the semantic content or to a certain object of an image than to the background, which may carry significant visual content [8].

Although Tieu and Viola [16] used more than 45,000 "highly selective features" which were argued to be expressive for high-level semantic concepts, majority of CBIR systems uses standard visual features such as color, shape, texture, and structure. For color, the HSV space has been the one of choice because it was designed to model human perception; some systems use color histograms of each component [3], [10], but most rely on two or three first color moments as defined by Stricker and Orengo in [18]. Shape features were divided into the region-based, which are suitable for complex objects with disjoint region, contour-based, which are argued to preserve semantics of the object, and edge-based [1]. Common shape representations include a Fourier descriptor and chamfer descriptor [5], as well as a Curvature Scale-Space (CSS) [19], which is suitable for contour-based shapes [1]. Wavelet moments, as well as co-occurrence matrix and Tamura features have been used to describe texture [5].

Many systems extract only the global features, but there are other approaches that are region-based. The latter can be further classified as segmentation-based (images segmented by color/texture similarities) and grid-based (images divided into blocks and features calculated from each block) which aim at capturing the local content of an image [10]. Dagli and Huang go further and combine grid and segmentation approaches [1].

In all the systems, typically one feature per image/region is calculated and features are represented by their corresponding feature vectors.

2.4. Machine Learning vs. Pattern Recognition

Relevance feedback can be considered a learning problem: a user provides feedback examples from the retrieval and the system learns how to refine the query to retrieve better results. Using Mitchell's definition, machine learning is concerned with the question of how to construct computer programs that automatically improve with experience. In CBIR, relevance feedback is a task of improving the retrieval performance and the experience are the results of user's feedback. Almost all of the current state of the art relevance feedback methods (such as the one presented in [7]) are classic machine learning methods [8]. However, the problem that makes many of these methods ineffective is the fact that the number of feedback examples provided by the user is very small (typically less than 20 per iteration) compared to the dimension of the feature space. This makes the learning difficult as the algorithms are supposed to learn from a small number of training samples in a very high-dimensional feature space [2]. Furthermore, class densities, especially that of the negative examples cannot be reliably modeled with such small sample size [7].

Some researchers consider relevance feedback process in CBIR a pattern recognition (classification) problem [8]. A classifier is trained with both positive and negative examples and then it classifies all images in the database into the relevant and irrelevant groups with respect to the given query. But in CBIR environment, the classes are not clearly defined and even if a predefined class structure exists, it is difficult to determine which category an image should belong to (because of the subjectivity of human perception). Therefore, even though many classifiers have been utilized, such as linear, nearest-neighbor, Bayesian, support vector machine (SVM), etc. the classification process does not have a clearly defined goal and its efficiency will be limited.

2.5. Re-weighting vs. Query Point Movement

The underlying assumption that most relevance feedback algorithms make is that the user has an ideal query point q that he/she is looking for, and then they try to estimate that ideal point, as well as the relative importance of individual features in the query vector [11]. Most proposed systems use one or the combination of the following methods:

(a) query-point movement

(b) re-weighting

As stated in [11], these two methods are actually orthogonal and can be combined, although not all previous methods have done so.

Query-point movement tries to improve the estimate of the ideal query point by moving it in the feature space towards positive and away from negative example points. This concept has been adopted from the information retrieval field, where query-point movement is usually described by Rocchio's formula that is based on the vector space model [11]; although the formula uses similarity-based query processing, the similarity can easily be translated into simple, even Euclidean, distances (see below).

Re-weighting is a very intuitive approach: if the variance of positive examples is high for some feature, than that feature is not important to the user and should receive smaller weight; therefore, the inverse of variance (or standard deviation) of some feature is used as a weight for that feature.

The MARS system presented in [6] combines both methods in its learning and refinement process.

2.6. Online Feature Weighting

Instead of a user manually specifying the relative importance of particular features in the "ideal query" he/she has in mind, relevance feedback algorithms were developed to learn those feature weights automatically from the feedback examples. In online feature weighting, the system determines a feature-weighting matrix W:

$$D(x, q) = (x\text{-}q)T\ W\ (x\text{-}q)$$

where D is the distance function, $q=[q_1, ..., q_n]^T$ is an n-D vector representing the "ideal" query the user has in mind, and $x=[x_1, ..., x_n]^T$ is a feature vector corresponding to an image in the database.

Different algorithms take different approaches as to how to calculate the weighting matrix W. Some take into the account only the positive feedback examples (*one-class*); one such system is developed in *MARS* interface [6]. As mentioned above, *MARS* uses both query-point movement and re-weighting methods and gives more weights to common features among example images (low variance features). The distance function used is the Mahalanobis distance, or the generalized Euclidean distance that uses the inverse of the covariance matrix of positive examples (i.e. $D(x,y) = [(x\text{-}y)^T\ C^{-1}\ (x\text{-}y)]$), [7]. Some other algorithms take both positive and negative examples to refine the query and divide them into two distinct classes (*two-class*). A linear transformation commonly used for feature-weighting in such cases is *Fisher's Discriminant Analysis (FDA)*, which attempts to find a weighting matrix W that minimizes the scatter within each cluster and maximizes the scatter between the clusters [7], [1]. When each negative example is treated as a separate class, the problem becomes that of a (1+x)-class, in which case *Multiple Discriminant Analysis (MDA)* or *Biased Discriminant Analysis (BDA)* are used. In the former method, while within-scatter is minimized, part of the objective is that all negative examples are apart from one another; since this can be potentially damaging, as several negative examples can come from the same class, BDA only tries to "pull" positive examples closer to one another, and to "push" negative examples from the positive ones (where negative examples can come from a large, but uncertain number of classes). Finally, some methods create many positive and many negative classes (*(x+y)-class*) and use *Group BDA* transformation for feature weighting (classical BDA applied to multiple positive classes) .

2.7. With vs. Without Memory

A big disadvantage of classic relevance feedback approaches such as [6] is that the acquired knowledge from one session is not stored and cannot be reused to improve the system's performance in successive sessions. In other words, even if the same user repeats the same query (with the same "ideal query" in mind and the same feedback), he/she would have to go through all the iterations of the relevance feedback procedure again. Therefore, the abovementioned machine learning process is limited to only one session and does not accumulate its "experience" over multiple sessions. To address this issue, some have proposed other approaches in which they attempt to memorize results for some user, assuming that they represent his/her subjectivity characteristics [8]. Of course, the problem of human perception subjectivity persists and it remains a challenge to handle the differences across different users and across different sessions of the same user. However, it is claimed that previous query results can be utilized to reduce the number of iterations before the convergence of the RF algorithm.

3. Our system

Even though relevance feedback algorithms and interfaces have been the subject of a very active research in recent years, we still think there is room for improvement. The reason for that is not only the fact that some methods simply do not work with the current approach to the problem, but also that some combinations of methods have not been tried yet. Claiming that human perception subjectivity is narrower than the semantic gap and that we should thus rely more on user's judgment than on the low-level feature-extraction engine, we propose a system in which collaborative contribution of many users over different sessions helps to improve the results of successive queries. This can be very helpful in a scenario similar to the one in which employees of a magazine company, which could have offices at different parts of the world, share not only the same image database, but also the same interests, educational backgrounds, and presentation styles (i.e. belong to the same "user profile"), which are all very important when the judgment of image similarity is shared among users. In such a scenario, the image similarity as determined by one user could certainly facilitate the queries with the same example image performed by the user's colleagues , since they would most probably perceive the same images as similar or relevant to their needs (which might not be the case, for example, when a general user and a botanist judge the semantic similarity between a red rose and a red carnation).

A very important feature of our interface is that the collaborative action does not require any additional effort by the user than typical relevance feedback interfaces, since it is incorporated into the usual interface in which images are only marked as relevant to the query or not. The system simply remembers users' "habits" during query sessions by storing relevant results and it uses these results in successive queries to classify and cluster semantically similar images. A simplified block-diagram of our method is given in Figure 1.

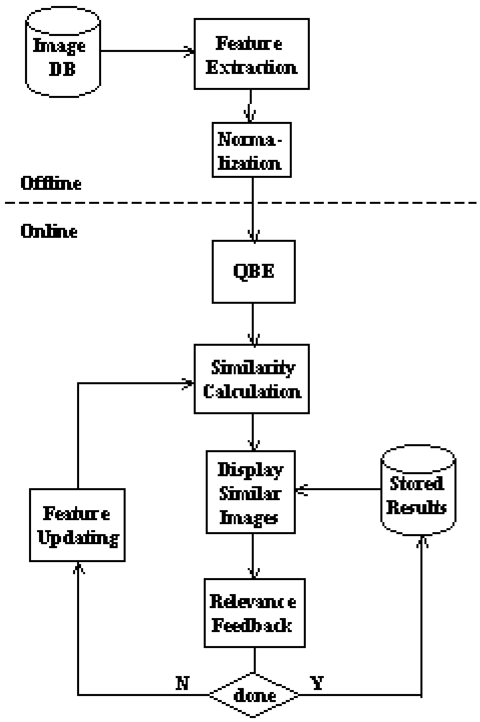

Figure 1: Simplified block-diagram of our approach

3.1 The Interface

In the proposed method, we try to take advantage of two limitations of current relevance feedback algorithms:

(1) most systems do not store previous query results to improve the performance of the CBIR system in successive query sessions and

(2) in most cases, keyword search and visual search in CBIR are mutually exclusive modes of interaction

While most of the systems do not store previous query results and start the retrieval process all over with a new query example, none have tried to solve the issue of the semantic gap using the users' remembered habits. Acknowledging the difficulties arising from human's perception subjectivity, we propose a novel user interface in which results of the same queries by previous users (or, even better, of the same user at different times) are displayed to the current one in an attempt to expedite the retrieval process (Figure 2).

As in typical Relevance Feedback interfaces, in our case the process also starts with a user presenting a query example to the system. This can be an image outside of the database (as is typical in a QBE approach), however for testing purposes we decided to use the images already present in the database itself (to verify that the best match will always be the equivalent image). Once the image has been presented to the system, it responds by retrieving images visually similar to the selected one in terms of color and shape (middle column in the figure). If the same example image has already been used in the previous queries, the results of those queries are displayed below it; furthermore, the results of queries in which the retrieved (visually similar) images have been included are also shown in the rows of the rightmost column of the figure. The latter introduces a novelty feature of our user interface: since it is difficult to bridge the semantic gap and describe image semantics by its visual features, we rely on other humans' perception of semantic similarity (claiming that semantic gap is wider than the differences that arise from perception subjectivity). If interested in the particular retrieved image, the user can keep its "satellites" or he/she can delete any number of them. The user is provided with four options in judging the degree of relevance for the retrieved results: he/she can decide that the query example and a particular retrieved image are similar:

(1) semantically (since that is the overall goal of the process, the system "glues" those two images together – the glued images are displayed as "satellites" of each others in the successive rounds of this session or in other sessions)

(2) visually (similar common visual features of both images are given more weight in the query)

(3) both (do both: glue + adjust feature weights)

(4) neither (disregard the retrieved image and its satellites, except those deleted, for the rest of this retrieval session)

The degrees of relevance for the options above are stored as *Scores* with values 2, 1, 3, and –2, respectively; the scores are used in the process of feature weight updating, which closely followed the algorithm proposed in [1].
Based on the feedback provided by the user, the system refines its query and retrieves new results; the retrieval process should terminate (and hopefully converge) after three or four such rounds so as not to overwhelm the user. The final results ("glued-pairs" or those images marked by the user as *Semantically* similar or *Both*) are stored to be used in the next session, involving either the same user or someone different.

3.2 Keyword Matching

It is very straightforward to incorporate a keyword search into a standard CBIR system, which relies solely on visual features. We try to avoid manual annotation altogether and only rely on those images whose filenames have already been assigned by the user. We assume that those filenames convey some semantic information relevant to the filename's creator and that they are generally a much better descriptor of image's semantic contents than those that could be extracted from the visual features alone (that is, we assume that the human perception subjectivity is narrower than the semantic gap). Acknowledging that a large number of images, especially those originating from new digital cameras, do not have meaningful filenames (they are typically named something like "DCM0179.JPG"), the processing overhead of matching the keywords with the images' filenames should not be an issue and therefore does not impose too big a burden to implement. We take into account common substrings with length of more than 3 characters between keywords and image's filename (excluding nebulous combinations of alphanumerical characters mentioned above, which should not pose a problem if the keywords are not just random combinations of characters, but the existing words instead) and transform the result into another weight of our online-weighting algorithm.

3.3 Relevant results

For this project, a subset of images from Corel's collection has been used for offline feature extraction. The images usually represent objects in high contrast with the background so that the segmentation process could give desirable results. There are always a couple of semantically (and usually visually) similar objects in the database which should be found by the first or the second iteration of the algorithm.

We decided to use only two low-level features for content description, i.e. color and shape. The color space of choice was HSV, since it has been designed to model human perception. We represent the color content of the image by Color Histograms and first two Color Moments [13], while we extract Fourier Descriptors to represent the properties of the shape [3, 4, 2, 1, 14]. To evaluate similarity, Histogram intersection [12, 1] has been used as a measure for Color Histograms, simple Euclidean distance for Color Moments and a weighted one for Fourier Descriptors [2].

For updating the weights, we also implement the approach outlined in [1]; in Relevance Feedback terminology, this is a *one-class* (considering only the positive feedback examples), *re-weighting* (using standard deviation of some feature among the relevant images as an indicator of the user's need) algorithm.

The whole graphical user interface has been designed in MATLAB programming environment, using GUI design, or GUIDE.

The system has not been tested extensively, with more difficult images (whose segmentation would be more difficult), or against some existing working prototype. However, with the subset of images that we were working on, we achieved the biggest convergence ratio (came "closer" to what we were looking for) after 1 (after feature extraction alone) or 2 (after the first round of feedback) iterations of the algorithm. These results are very desirable, because they mean that the user will be satisfied after a short time and not overwhelmed by repetitious requirements of the relevance feedback interface.

4. Conclusion and Future Work

In this paper, we gave an introduction into relevance feedback methods in CBIR and an overview of a variety of methods used in this area. Since different algorithms have taken different approaches to the problem and were developed under different assumptions, we also tried to classify the systems based on various criteria.

We have also proposed a novel user interface in which, besides being able to mark retrieved images as relevant or not, the user sees the images marked semantically similar by previous users. The implemented design, however, can be improved in many ways. Feature extraction engine could include more features (such as texture, with possible descriptors being a wavelet filter bank and a co-occurrence matrix), or more descriptors for the existing features (such as Chamfer matching or Curvature-Scale Space (CSS) for shape).

For Relevance feedback, we use a one-class similarity search algorithm and consider only positive examples (i.e. those marked as *Visually* similar) in feature-updating; future enhancements of the algorithm could include replacing the option *Neither* by *Neutral* (disregarding the images for the rest of the process) and adding a new option, *Dissimilar*, for which common visual features would be used as negative feedback (making the algorithm a two-class search). We can also include more sophisticated learning and clustering techniques for classification of images into different groups (according to the learned understanding of the user's perception of the query) before retrieving the similar ones.

As mentioned earlier, since the segmentation of images can be a critical issue, the choice of images for our database was such that this process is facilitated; another novelty that can be introduced into the system is to give the user the ability to specify the example image and the region(s) of interest within it (by clicking on it with a mouse). The region-growing algorithm would then determine the boundaries of the object and extract it much easier. With the region of interest specified, feature extraction and relevance feedback algorithm would give much more emphasis to the local features of the extracted object than to the background (which can contain the predominant amount of information in the image) or other objects and refine the query accordingly.

This paper described the ongoing design and implementation aspects of a CBIR system which we envision to be the starting point for a more involved research project and for a more sophisticated Feature Extraction and Relevance Feedback components. With the planned improvements mentioned earlier in this

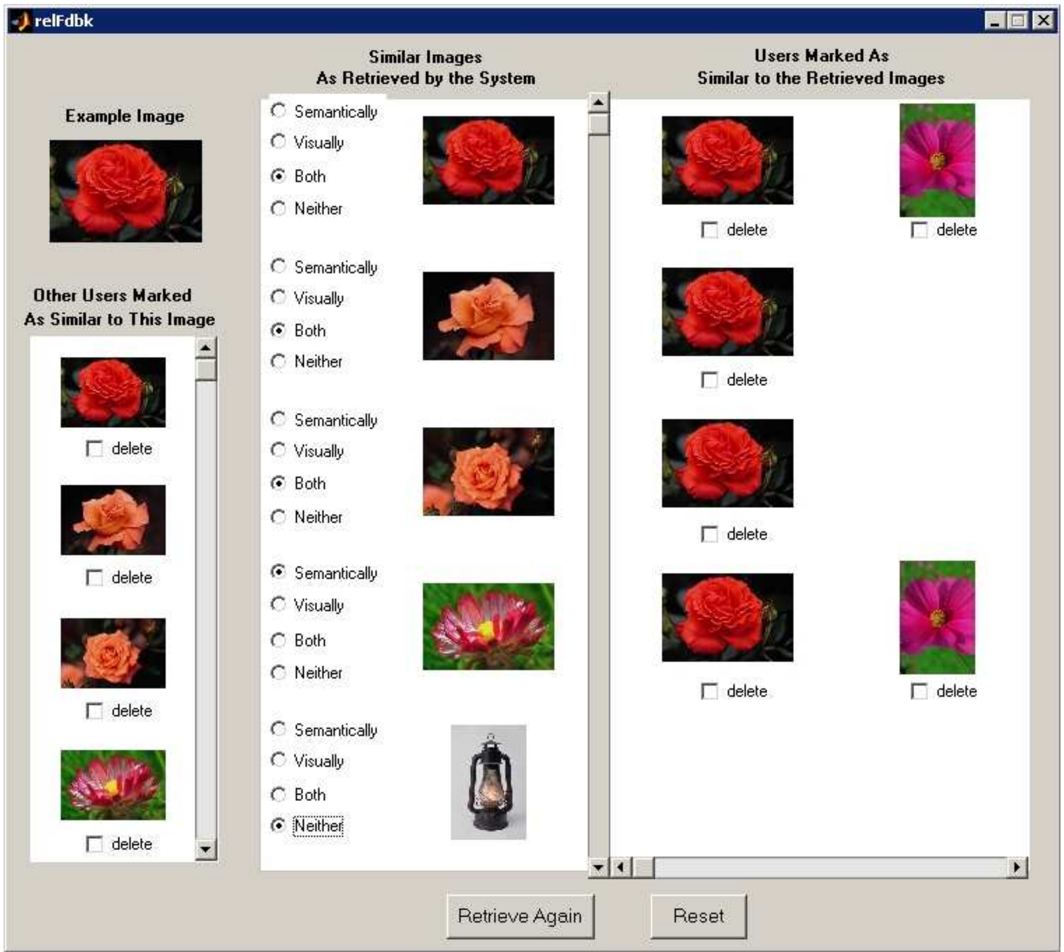

Figure 2: The system's GUI

section, as well as the collaborative contribution of many users to the relevance and classification of images in the database (using the stored results of previous query sessions), we hope to improve our system significantly and to eventually bridge the semantic gap that is currently one of the biggest obstacles for CBIR engines.

References

[1] Yong Rui, Thomas S. Huang, Michael Ortega, and Sharad Mehrotra, "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval", *IEEE Transactions on Circuits and Video Technology*, vol. 8, Sept. 1998.

[2] Yong Rui, Thomas S. Huang, Sharad Mehrotra, and Michael Ortega, "Automatic Matching Tool Selection Using Relevance Feedback in MARS", in *Proc. of 2nd Intl. Conference on Visual Information Systems*, 1997.

[3] Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing", *Second Edition, Prentice-Hall, NJ*, 2002.

[4] Rafael C. Gonzalez, Richard E. Woods, and Steven L. Eddins, "Digital Image Processing Using MATLAB", *First Edition, Prentice-Hall, NJ*, 2003.

[5] Charlie Dagli, Lecture Notes on Content-Based Image Retrieval, University of Illinois – Urbana Champaign's Image Processing Class, Fall 2003.

[6] Yoshiharu Ishikawa, Ravishankar Subramanya, and Christos Faloutsos, "Mindreader: Querying Databases Through Multiple Examples", In *Proc. of the 24th VLDB Conference*, 1998.

[7] Munehiro Nakazato, Ljubomir Manola, Thomas S. Huang, "Group-Based Interface for Content-Based Image Retrieval", in *Proc. of Advanced Visual Interfaces 2002 (AVI 2002)*, 2002.

[8] S. Santini, R. Jain, "Integrated Browsing and Querying for Image Database", *IEEE Multimedia*, vol. 7, 2000.

[9] Yong Rui, Thomas S. Huang, Shih-Fu Chang, "Image Retrieval: Current Techniques, Promising Directions, and Open Issues", *Journal of Visual Communication and Image Representation*, vol.10, April 1999.

[10] Xiang Sean Zhou and Thomas S. Huang, "Exploring the Nature and Variants of Relevance Feedback",

IEEE Workshop on Content-based Access of Image and Video Libraries, Dec. 2001.

[11] Xiang Sean Zhou and Thomas S. Huang, "Comparing Discriminating Transformations and SVM for Learning during Multimedia Retrieval", in *Proc. of ACM Multimedia '01*, 2001.

[12] Annalisa Barla, Francesca Odone, Alessandro Verri, "Histogram Intersection Kernel for Image Classification", *Intl. Conference on Image Processing ICIP 2003*, 2003.

[13] Markus Stricker and Markus Orengo, "Similarity of Color Images", In *Proc. of SPIE, Vol. 2420 Storage and Retrieval of Image and Video Databases III*, 1995.

[14] Yong Rui, A.C. She, and T.S. Huang, "Modified Fourier Descriptors for Shape Representation – A Practical Approach", in *Proc. of First International Workshop on Image Databases and Multi Media Search*, 1996.

[15] Farzin Mokhtarian, Sadegh Abbasi, and Josef Kittler, "Efficient and Robust Retrieval by Shape Content through Curvature Scale Space", In *Proc. Of Intl. Workshop on Image Databases and Multimedia Search*, 1996.

[16] Gunilla Borgefors, "Hierarchical Chamfer Matching: A Parametric Edge Matching Algorithm", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 10, Nov. 1988.

[17] Xiang Sean Zhou and Thomas S. Huang, "Unifying Keywords and Visual Contents in Image Retrieval", *IEEE Multimedia*, vol. 9, April-June 2002.

[18] Yong Rui, Thomas S. Huang, and Sharad Mehrotra, "Relevance Feedback Techniques in Interactive Content-Based Image Retrieval", in *Proc.of IS&T SPIE Storage and Retrieval of Images/Video Datrabases VI, EI '98*, 1998.

[19] Yong Rui, Michael Ortega, Thomas S. Huang, "Information Retrieval Beyond the Text Document", *Library Trends*, vol.48, 1999.

[20] Yong Rui, Thomas S. Huang, and Sharad Mehrotra, "Content-Based Image Retrieval with Relevance Feedback in MARS", in *Proc. of IEEE International. Conference on Image Processing*, 1997.

[21] Zhong Su ad Hongjiang Zhang, "Relevance Feedback in CBIR", *Microsoft Research Asia*, 2003.

[22] Ye Lu, Chunhui Hu, Xingquan Zhu, Hongjiang Zhang, Qiang Yang, "A Unified Framework for Semantics and Feature Based Relevance Feedback in Image Retrieval Systems", *ACM Multimedia 2000*.

[23] C. Faloutsos, R. Barber, M. FLickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and Effective Querying by Image Content*", Journal of Intelligent Information Systems*, July 1994.

[24] Marco La Cascia, Saratendu Sethi, and Stan Sclaroff, "Combining Textual and Visual Cues for Content-Based Image Retrieval on the World Wide Web", *IEEE Workshop on Content-based Access of Image and Video Libraries*, June 1998.

[25] Daniel Heesch and Stefan Ruger, "Performance boosting with three mouse clicks – Relevance feedback for CBIR", In *Proc. of the European Colloquium on IR Research 2003*.

[26] I.J. Cox, M. Miller, T. Minka, T. Papathomas, and P. Yianilos, "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical Experiments", *IEEE Trans. on Image Processing*, vol. 9, Jan. 2000.

[27] K. Tieu and P. Viola, "Boosting Image Retrieval", In Proc. *IEEE Conference on Computer Vision and Pattern Recognition*, June 2000.

[28] Rong Yan, Alexander Hauptmann, and Rong Jin, "Multimedia Search with Pseudo-Relevance Feedback", *Intl. Conference on Image and Video Retrieval*,2003.

[28] Marjo Markkula and Eero Sormounen, "Searching for Photos – Journalists' Practices in Pictorial IR", in *Workshop on Image Retrieval*, Feb. 1998.

[29] Arnold Smeulders et al., "Content-Based Image Retrieval at the End of the Early Years", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, Dec. 2000.

[22] Simon Tong and Edward Chang, "Support Vector Machine Active Learning for Image Retrieval", *ACM Multimedia*, 2001.

[30] Tao Wang, Yong Rui, Shi-min Hu, and Jia-guang Sun, "Adaptive Tree Similarity Learning for Image Retrieval", *ACM Multimedia Systems Journal (Springer)*, vol.9, Aug. 2003.