$6^{th}$ International Conference on Visual Information Systems (VISUAL 2003), Miami, Florida, Sep. 24-26, 2003, pp. 467-474.
Held in conjunction with $9^{th}$ International Conference on Distributed Multimedia Systems (DMS' 03).

# Feature Integration, Multi-image Queries and Relevance Feedback in Image Retrieval

Qasim Iqbal* and J. K. Aggarwal†
Computer and Vision Research Center
Department of Electrical and Computer Engineering
The University of Texas at Austin
Austin, Texas 78712, USA

**ABSTRACT**

*In this paper, we explore the effect of feature integration, multi-image queries, and relevance feedback in enhancing the performance of an image retrieval system. Weighted integration of structure, color and texture features is studied. In addition, we propose a methodology of retrieval consisting of multiple query images, as opposed to the traditionally-used model of a single query image. Two different mechanism of relevance feedback are also proposed and analyzed. Integration of features and feedback significantly improves the performance of the retrieval system.*

## 1 INTRODUCTION

Over the years various techniques have been proposed for retrieval in digital image databases. Many of these techniques operate under specific constraints about the type of image data they operate upon. Frequently, such constrains have been put on the retrieval of images containing certain objects. For example, early work used user-defined sketches [1] for retrieval, whereas some recent approaches perform content-based search and retrieval of images using object models [2]. Such methods are known as "model-based" approaches, because they exploit a higher-level shape representation of objects present in an image for retrieval. Other techniques have also been proposed that use higher-level inferencing, but do not require detailed object models for representation and retrieval. They include retrieval of images containing manmade structures [3] and buildings [4, 5], retrieval using edge/structural features [6], and differentiating city and landscape images based upon the orientation of edges [7].

Techniques that are collectively referred to as "view-based" approaches tend to characterize what is really observed in an image instead of making some underlying assumptions about the model-based representations of the objects. These techniques essentially exploit the color, intensity or luminance information in an image. The basic idea is to use color / intensity information, which does not depend on the geometric shape of the objects. Most of the current view-based retrieval techniques analyze image data at a lower level on a strictly quantitative basis for color [8] and texture features [9]. These techniques are geared towards retrieval on overall image similarity, especially for images containing natural objects such as trees, vegetation, water and sky.

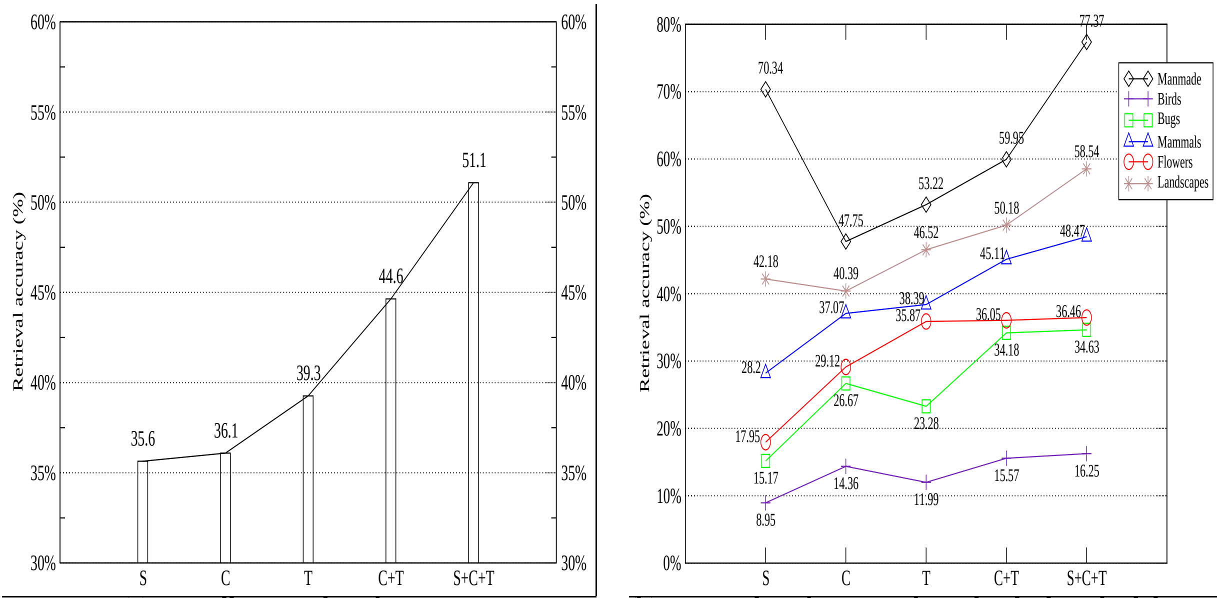
There is no clear consensus among researchers about which technique to use for a general image retrieval system. The answer to this problem depends on many factors, such as the number and complexity of objects present in an image, the amount of *a priori* information about the scene, and in case of retrieval based upon shape, the number of objects present in the model database. The last factor is a challenging issue because occlusion can severely degrade the retrieval of images based upon object models. A general solution to the problem of image retrieval is not well-settled, because it is not possible to preempt the type of image data as provided by a user in queries.

In this paper, we explore how user involvement with an image retrieval system can improve retrieval performance. In particular, we investigate the effect of feature integration, multiple query images, and relevance feedback. As mentioned above, features extracted from different techniques emphasize image attributes in different domains. We integrate a number of features for boosting retrieval performance. Further, user specification of the importance of different features based upon visual inspection of image content is also analyzed. In addition, the effect of supplying more than one query image is considered. Relevance feedback is a mechanism of providing input regarding the quality of retrieval to the image retrieval system after a query is performed. The system uses the input to rerun the query and, hopefully, get a better retrieval. The input is typically provided by specifying which of the retrieved images agree with the query, and which do not. We propose two different mechanism of feedback, namely, Cluster feedback, and Multi-class feedback. The combination of weighted feature integration and feedback is also proposed.

We use our image retrieval system CIRES – Content-based Image REtrieval System [10] – for the experiments performed. CIRES is a robust image retrieval system that serves queries ranging from images containing conspicuous structure, such as buildings, bridges and towers, to images containing purely natural objects, such as vegetation, water, sky and clouds. It is available online at `http://amazon.ece.utexas.edu/~qasim/research.htm`

The rest of the paper is organized as follows. Section 2 details the feature integration, Section 3 describes multi-image query, Section 4 explains the mechanism of

*qasim@ece.utexas.edu
†aggarwaljk@mail.utexas.edu




(a) Overall retrieval performance. (b) Retrieval performance for individual methodologies.

Figure 1: Retrieval performance using five different methodologies: structure only (S), color only (C), texture only (T), color and texture (C+T), and structure, color and texture (S+C+T). Retrieval accuracy increases sharply with feature integration. Best results are obtained for the integration of structure, color and texture.

the feedback process, Section 5 outlines the combination of weighted feature integration and feedback, and finally, Section 6 presents the conclusions.

## 2 Feature Integration

Automatic analysis of image content is a challenging problem. The ability to extract and describe distinct objects in a complex scene is crucial for image understanding. Even human observers have difficulty in describing the content of some images using only limited keywords. For example, an image depicting a bridge on water and vegetation may be assigned to both a structural scene, because of the presence of a manmade object, the bridge, and a natural scene because of the presence of water and vegetation. Presence of both manmade and natural objects exacerbates the computer perception of images. Natural objects, such as trees and other vegetation, rivers, rocks and clouds, coexist with manmade objects and are unspecified; their appearance is unpredictable. Very few natural objects have compact shape descriptions, and it is difficult to establish complete boundaries between the objects of interest and the background objects [11].

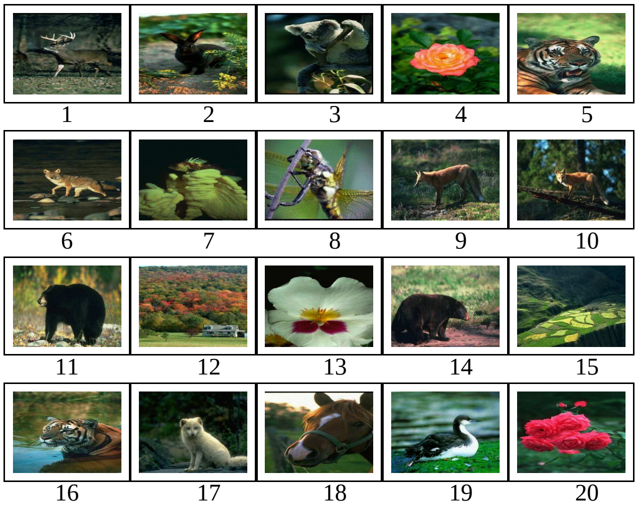
The difficulty in establishing boundaries between manmade and natural objects stems from the fact that automatic segmentation is a difficult problem. Until a complete automatic solution to the segmentation problem is achieved, a recourse seems to be the integration of various methodologies. Indeed recent trend in image retrieval has been to use information obtained from multiple cues such as color, texture and shape [12, 13].

We study the integration of structure features, which are particularly suitable for the retrieval of manmade objects, and color and texture features, which are geared towards the retrieval of natural images in general. Specifically, we restrict our attention to five different methodologies of feature analysis: structure only, color only, texture only, color and texture only, and structure, color and texture. The aim is to explore if feature integration using (i) color and texture, and (ii) structure, color, and texture results in better performance than using structure, color, and texture individually. Structure, color, and texture features are extracted from each image by the process described in [13].

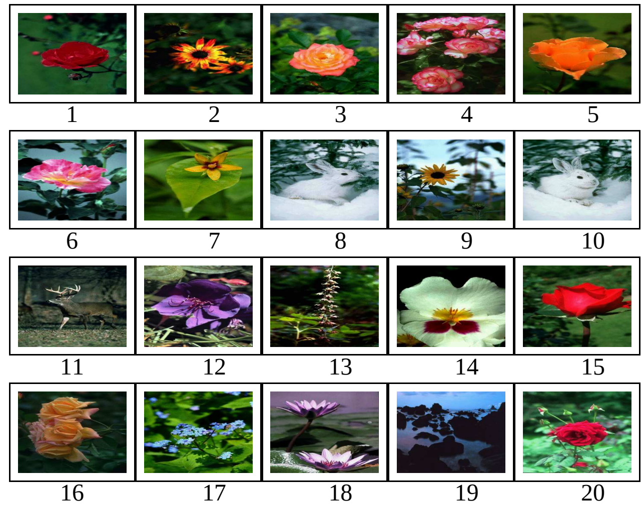
We use a linear (convex) combination of the distances in the product space of structure, color and texture for retrieval. Distance between a query image and a test image in the database in the structure and texture feature spaces are calculated using the $\ell_2$ norm. Histogram intersection measure [8], which is a variant of the $\ell_1$ norm, is used for color. Distances are properly normalized to take into account the difference in image size, and to make sure that the histogram intersection measure is symmetric. Weights are associated with distances, which assign the degree of importance attached to feature extracted from different methodologies. The distances in these spaces were pre-normalized in the range $[0, 1]$. However, it may be possible that a relatively larger value in a feature space biases the calculation of the weighted distance. To overcome this problem, we have used the following Gaussian normalization that puts equal emphasis on the distances in the each of the three feature spaces [14], before taking a linear combination.

Let $d = (d_i)$ be a sequence of distances in any of the above-mentioned three feature spaces. Gaussian normalization results in the mapping: $d_i \rightarrow (d_i - \mu)/3\sigma$. where $\mu$ and $\sigma$ represent the mean and the standard deviation of $d_i$. Let $\acute{d_i} = (d_i - \mu)/3\sigma$. This normalization ensures that probability of the normalized value, $\acute{d_i}$, being in the range $[-1, 1]$, is 99%. Values outside this range may be forced to map to either -1 or 1. Finally, the mapping, $\acute{d_i} \rightarrow (\acute{d_i} + 1)/2$, normalizes distances in $[0, 1]$.

(a) Query image: Flower and vegetation




(b) Weights: S = 0.33, C = 0.33, T = 0.33.




(c) Weights: S = 0.05, C = 0.05, T = 0.9.

Figure 2: Adjusting weights corresponding to structure, color and texture in feature integration increases retrieval performance. Precision increases from 15% in (b) to 80% in (c). S = Structure, C = Color, T = Texture.

### 2.1 *Results Obtained*

We performed our experiments on an image database of 10,221 images. The database consists of six classes: 1908 images of manmade objects, 811 images of birds, 1134 images of bugs, 2496 images of mammals, 1161 images of flowers, and 2711 images of landscapes. The experiments measured the overall retrieval accuracy and the accuracy of retrieval for a particular class.

Let $X$ denote the database and let $X^i_j$ denote the $j^{th}$ image belonging to $i^{th}$ class, $X^i$. For ease of notation, throughout this paper we shall use the notation that $X^i$ represents both a class and the set of images falling in that class. Let $C$ denote the number of classes, and $|X^i|$ denote the number of images in the $X^i$ class. We treated each image in each class in the database as a query image and retrieved the corresponding first $H$ images as the set $R$, where $R_k$ denotes the $k^{th}$ image in the retrieved set for a particular query. Then, we defined the retrieval accuracy to be

$$\mathcal{A}_i = \frac{1}{|X^i| * H} \sum_{j=0}^{|X^i|} \sum_{k=0}^{H} h(X^i_j, R_k) \tag{1}$$

where, $\mathcal{A}_i$ denotes the accuracy of retrieval for a certain class (for a particular methodology), and $h$ is defined by the following function

$$h(X^i_j, R_k) = \begin{cases} 1 & R_k \in X^i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

i.e., $h$ is equal to one if the image $R_k$ belongs to the same class $X^i$ to which the image $X^i_j$ belongs, and zero otherwise. In other words, for a query the number of images belonging to the same class as the query image in the first $H$ images retrieved is counted. Each image in a particular class was treated as the query image and the count was updated. Retrieval accuracy is obtained by dividing the final count by $|X^i| * H$, which is the total number of correct images that can possibly be retrieved for that class.

Equation 1 is used for the computation of class-specific retrieval accuracy. For the computation of overall retrieval performance the following measure is used

$$\mathcal{A} = \frac{1}{|X| * H} \sum_{i=0}^{C} \sum_{j=0}^{|X^i|} \sum_{k=0}^{H} h(X^i_j, R_k) \tag{3}$$

where, $\mathcal{A}$ denotes the accuracy of retrieval (for a particular methodology), and $|X| = \sum_i^C |X^i|$, is the number of images in the database. It may be noted that $\mathcal{A}$ is actually a weighted linear combination of $\mathcal{A}_i$'s

$$\mathcal{A} = \sum_i^C w_i \mathcal{A}_i \tag{4}$$

where, $w_i = \frac{|X^i|}{|X|}$ is the weight corresponding to $\mathcal{A}_i$ such that $\sum_i^C w_i = 1$. The value of $H$ was chosen to be 20 in all experiments.

Figure 1(a) displays the result of overall retrieval performance using the whole database for five methodologies: structure only, color only, texture only, color and texture, and structure, color and texture. For queries using color and texture, and structure, color and texture, equal emphasis was


(a) Singe-image query.

(b) Multi-image query.

Figure 3: Single-image query vs. multi-image query. Much better retrieval is obtained using three query images compared to one query image. Precision increases from 35% in (a) to 95% in (b). Equals weights for structure, color, and texture were used for (a) and (b).

placed on each methodology. It is observed that the the retrieval performance increases sharply with the feature integration, and the best retrieval is obtained when features from all three methodologies are combined.

Figure 1(b) shows the result of class-specific retrieval performance. It is seen that for each of the six classes the performance also increases with feature integration, and again, the best retrieval is obtained when features from all three methodologies are combined.

### 2.2 *Effect of weights on feature integration*

Different query images have different content. CIRES provides an option where a user can change weights to fine tune retrieval. The effect is the same as assigning more weight to certain methodologies, or certain features.

As an example, Figure 2 presents a query of a flower in vegetation. Figure 2(a) displays the query image. Figure 2(b) shows the images retrieved using equal weights (default) for structure, color and texture. The aim of the query was to find images of flowers without regard to the color of the flower. The retrieved result set contains only 3 images containing flowers, in positions 4, 13, and 20. The precision of a retrieved set of images is defined as the ratio of the images that are perceived to be correct, i.e., similar to the query image, to the number of all images in the set. Therefore, Figure 2(b) has a retrieval precision of 3/20 = 15%.

When weights were changed to structure = 0.05, color = 0.05, and texture = 0.9, the retrieved set of images shown in Figure 2(c) contained 16 images with flowers, a precision of 16/20 = 80%. The 4 images that are not flowers are in positions 8, 10, 11 and 19. A lower weight for color, and a high weight for texture, ensured the retrieval of a large number of flowers of different colors. In our experiments with CIRES, we have observed large increases in precision, similar to the example given in Figure 2, by supplying a judicious set of feature weights.

## 3 Multi-Image Queries

Most of the image retrieval systems support the single-query model only. In the single-query model a database of images is searched to find images similar to a given query image. However, it may be desirable to query an image database using more than one query images for detailed knowledge representation. An aim of this paper is to develop a paradigm that supports the multi-query model. An advantage of the multi-query model is that it overcomes the limitation on the specification of image content using a single-query model.

Let $X_j$ denote the $j^{th}$ image in an image database $X$. Let $S$ denote a set of query images provided by the user. The distance of $X_j$ to the set is defined by

$$D(X_j, S) = \min_k \; d(X_j, S_k) \tag{5}$$

where, $D$ represents the distance of the image $X_j$ to the set of images, $S$, and $d$ is the distance of $X_j$ from an image $S_k$, which is contained in $S$. Equation 5 essentially defines the distance of a test image in the database to a set of query images to be the distance between the test image and its nearest neighbor in the query set. The distances are sorted in ascending order to obtain a set of images that are decreasingly similar to the query image.

Our paradigm is similar to the works presented in [15], where instead of the $\min$ distance between a test image and the query image set, a linear combination of the distances of a test image to all images in the query image set was used, and [16] where a modified form of Fisher linear discriminant analysis was used for distance calculation.

### 3.1 Results Obtained

Figure 3(a) displays an image query of a building generated using equals weights for color, texture and structure. The retrieved result set of 20 images is shown below. Of these images, 7 are those that are predominantly buildings (in positions 2, 3, 5, 8, 12, 13 and 18), and the remaining 13 are automobiles. We impose a conservative criterion for query performance that though the retrieved images that are automobiles are manmade objects, and some of these images have large buildings in the background, we will consider them as false matches. In this sense, the result set has a precision of 7/20 = 35%.

To improve performance, two more query images of buildings were added to the original query image, as shown in Figure 3(b). The weights for color, texture and structure were held the same. It is observed that now 19 of the retrieved images are buildings, and only one image (in position 4) has a foreground bus and a much larger background consisting of buildings. Still, we consider this image to be a false match. The query result precision is now 19/20 = 95%. The large increase in precision is an empirical justification of the proposed notion of using more than one query image. In our experiments we have seen that generally multi-image queries tend to provide better retrieval than single image queries.

## 4 Relevance Feedback

In previous work in relevance feedback [14, 17, 18] usually weights are associated with different features (inter-feature), and feature components (intra-feature). Once a query is done, a user provides samples for positive images and negative images from the retrieved set of images. Positive images are those images that a user considers similar to the query image. Negative images are those images that are considered to be dissimilar. The weights are automatically adjusted by using the feedback images to redo a query and obtain a better result. Both positive and negative images are used in traditional approaches to adjust the weights.

We propose two different mechanisms of feedback, Cluster feedback, and Multi-class feedback. Cluster feedback uses only positive images in a modified framework of multi-image query. Multi-class feedback is proposed as a classification problem. The positive images are assumed to fall in one class, where as, the negative images are considered to fall in as many classes as the user deems necessary. Similar to the case of multi-image query, the sorted distance is used after feedback calculations to render images. The feedback mechanisms are described below.

### 4.1 Cluster Feedback

Cluster feedback is essentially based upon the multi-image query paradigm. In a multi-image query a user selects a number of query images *before* performing a query. In the case of cluster feedback, a user starts with a single or a multi-image query, and *after* the query selects a number of images from the retrieved set of images as feedback images. The selection of these feedback images corresponds to the user's judgment that he or she is not fully satisfied with the all the results of a particular query, and now wants to use some returned images, which are judged by the user to be similar to the set of query images, to further refine the query. These images may or may not be added to the original set of query images for another query on the database. In our implementation, CIRES always groups the feedback images selected by the user with the original query images to make a larger set for another query. The process may be repeated any number of times until the user is satisfied with the results of the query.

### 4.2 Multi-class Feedback

Multi-class feedback builds upon the concept of Cluster feedback by providing more than one cluster (sets) of feedback images. These sets typically represent images belonging to different classes that a user has envisaged. Multi-class feedback is basically the usual nearest-neighbor classification that is used frequently in pattern recognition. Let $S = \{S^m\}$ represent the set of all feedback images, where $S^m = \{S_k^m\}$ represents the $m^{th}$ set of feedback images provided by the user as training samples, $S_k^m$ is the $k^{th}$ image in $S^m$, and $m$, and $k$ are indices. The image database $X$ is partitioned into various disjoint sets of images, where each of these sets $X^p$ is such that

$$X = \bigcup_p X^p, \quad X^p \cap X^q = \emptyset, \quad p \neq q, \tag{6}$$

where, $p$ and $q$ are indices. The partition $X^l = \{X_j^l\}$ represents images classified to the same class as the feedback set $S^l$ by the proximity of distances. That is, a test image $X_j$ in the database is classified to the partition $X^l$ if

$$D(X_j, S^l) = \min_m D(X_j, S^m) = \min_{m,k} d(X_j, S_k^m), \tag{7}$$

where, $m$ ranges over all indices of such sets.

The classification problem then reduces to the scenario that there is one relevant class, and all the remaining classes are irrelevant. The work presented in [19] is based on a similar premise, and a recent extension of their work generalizes to the use of multiple relevant classes [16].

### 4.3 Results Obtained

Figure 4 presents a query of horses. Figure 4(a) displays the query image. Figure 4(b) shows the results obtained using just the query image. The results include 10 images containing horses (in positions 1, 2, 4, 5, 6, 7, 8, 13, 17 and 20), a precision of 10/20 = 50%.

Figure 4(c) shows the result of cluster feedback when the 10 images of horses retrieved in Figure 4(b) are used as feedback images in conjunction with the original query image shown in Figure 4(a). It is observed that the number of retrieved images containing horses in Figure 4(c) increases to 16 (in positions 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 16, 18 and 20), a precision of 16/20 = 80%.

Figure 4(d) presents the results of Multi-class feedback formulated as a two-class feedback problem using CIRES on the result set of Figure 4(b). The same set of 10 images of horses retrieved in Figure 4(b) together with the query image shown in Figure 4(a) are used in one class, which is considered as the positive class. The remaining 10 images in Figure 4(b) are considered as the negative class. All of the retrieved images shown in 4(d) contain horses except one image (in position 14), resulting in a precision of 19/20 = 95%.

It is observed that higher precision is obtained using two-class feedback compared to Cluster feedback. In general, we have observed in our experimentation with CIRES that Multi-class feedback tends to give slightly better results than Cluster feedback. The happens because of the discriminating effect of other classes in the pattern space besides the class of positive images.

## 5 Combination of Feature Integration and Relevance feedback

Feature integration may be used in conjunction with relevance feedback for the further boosting of retrieval performance. Figure 5 displays a query for finding yellow automobiles. The query image is shown in Figure 5(a). The initial results obtained at the default setting of equal weights for structure, color and texture are shown in Figure 5(b). It is observed that the result set contains 6 yellow automobiles (in positions 1, 2, 10, 11, 14 and 16), 11 automobiles of a different color (in positions 3, 4, 5, 6, 7, 8, 9, 13, 15, 18 and 20), and 3 images that are not automobiles (in positions 12, 17, and 19). The precision is 6/20 = 30%. It is interesting to note that had our query paradigm been only to retrieve automobiles of any color, then the precision would have been (6 + 11)/20 = 85%.

We partitioned the 20 images in the result set into 3 classes. The positive class consisted of the 6 yellow automobiles. The 11 images that are automobiles, which are not yellow, were put in a different class of “Not Sure” images, and the 3 images that are not automobiles at all were put in the third class of negative images. Refer to Figures 5(c) - 5(e). The number in parenthesis below each image is the position of the image in Figure 5(b). These images were used as feedback images by using CIRES in a three-class mode. Color weight was increased to 0.5, and the weights of structure and texture were adjusted to 0.3 and 0.2, respectively. In addition, we also changed the mode of texture analysis from the default of using all three channels of CIE LAB space, used in Figure 5(b), to using just the L channel only. Roughly, this would correspond to using the “grayscale texture” only.

Figure 5(f) presents the result set obtained using a combination of weighted feature integration and three-class feedback. It is observed that returned images contained 15 images of automobiles that are yellow in color (in positions 1, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 15, 18, 19, 20). The resulting precision is 15/20 = 75%. We did not include an image of an automobile that is partially yellow (in position 17), and an image of a yellow taxi that is mostly hidden behind a dark colored car (in position 10).

The large increase in precision, form 30% to 75%, is consistent with our general observation that a combination of weighted linear feature integration and Multi-class feedback tends to give better retrieval than using just equal weights for structure, color, and texture, and using no feedback.

## 6 Conclusions

Automatic analysis and retrieval of images from a database is a challenging task. The difficulty arises from a number of issues, such as the complex mix of manmade and natural objects in an image, *a priori* information about the image content, and the availability of the structural representations of objects present in an image. User interaction may be used to provide some information to an image retrieval system that is difficult to obtain automatically. We studied the effect of integration of features extracted from three different techniques, and the proposed mechanism of multi-image queries, and two different techniques for relevance feedback, on the system performance. Structure, color, and texture features were extracted from an image. It was observed that feature integration enhanced retrieval accuracy in general. Weighting of different features by a user based upon images content also improved retrieval. In addition, the proposed mechanisms of multi-image queries and relevance feedback boosted system performance considerably. In particular, the combination of weighted feature integration and feedback was seen to be efficient.

## References


[1] A. D. Bimbo, P. Pala, and S. Santini, "Visual image retrieval by elastic deformation of object sketches," in *IEEE Symposium on Visual Languages, St. Louis, MO*, pp. 216–223, 1994.

[2] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs, "A search engine for 3D models," *ACM Transactions on Graphics*, vol. 22, pp. 83–105, January 2003.

[3] Q. Iqbal and J. K. Aggarwal, "Retrieval by classification of images containing large manmade objects using perceptual grouping," *Pattern Recognition*, vol. 35, pp. 1463–1479, July 2002.

[4] Q. Iqbal and J. K. Aggarwal, "Applying perceptual grouping to content-based image retrieval: Building images," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), Fort Collins, Colorado*, vol. 1, pp. 42–48, June 1999.

[5] Y. Li and L. G. Shapiro, "Consistent line clusters for building recognition in cbir," in *IEEE International Conference on Pattern Recognition (ICPR)*, vol. 3, pp. 952–956, 2002.

[6] X. S. Zhou and T. S. Huang, "Edge/structural features for content based image retrieval," *Pattern Recognition Letters*, vol. 22, pp. 457–468, April 2001.

[7] A. Vailaya, A. K. Jain, and H.-J. Zhang, "On image classification: City images vs. landscapes," *Pattern Recognition*, vol. 31, pp. 1921–1935, December 1998.

[8] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[9] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[10] Q. Iqbal and J. K. Aggarwal, "CIRES: A system for content-based retrieval in digital image libraries," in *Invited Session on Content-based Image Retrieval: Techniques and Applications,* $7^{th}$ *International Conference on Control Automation, Robotics and Vision (ICARCV)*, pp. 205–210, December 2002.

[11] M. A. Fischler and T. M. Strat, "Recognizing objects in a natural environment: A contextual vision system (CVS)," in *DARPA Image Understanding Workshop, Palo Alto, California*, pp. 774–796, 1989.

[12] E. Saber and A. M. Tekalp, "Integration of color, edge, shape and texture features for automatic region-based image annotation and retrieval," *Electronic Imaging*, vol. 7, pp. 684–700, July 1998.

[13] Q. Iqbal and J. K. Aggarwal, "Combining structure, color and texture for image retrieval: A performance evaluation," in $16^{th}$ *International Conference on Pattern Recognition (ICPR)*, vol. 2, pp. 438–443, August 2002.

[14] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A powerful tool for interactive content-based image retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 644–655, 1998.

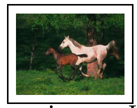
[15] L. Zhu and A. Zhang, "Supporting multi-example image queries in image databases," in *International Conference on Multimedia and Expo (ICME 2000)*, pp. 697–700, July 30 - August 2 2000.

[16] M. Nakazato and T. S. Huang, "Extending image retrieval with group-oriented interface," in *IEEE International Conference on Multimedia and Expo (ICME)*, vol. 1, pp. 201–204, 2002.

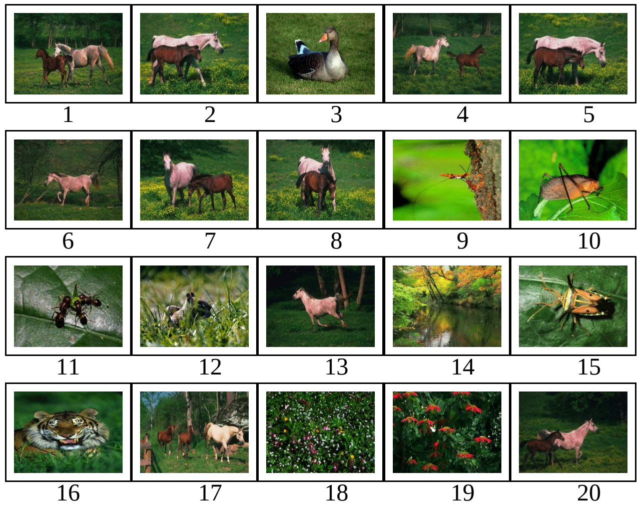
[17] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos, "The Bayesian image retrieval system, PicHunter: Theroy,implementation, and psychophysical experiments," *IEEE Transactions on Image Processing*, vol. 9, pp. 20–37, Januray 2000.

[18] R. Schettini, G. Ciocca, and I. Gagliardi, "Content-based color image retrieval with relevance feedback," in *IEEE International Conference on Image Processing (ICIP)*, pp. 75–79, 1999.

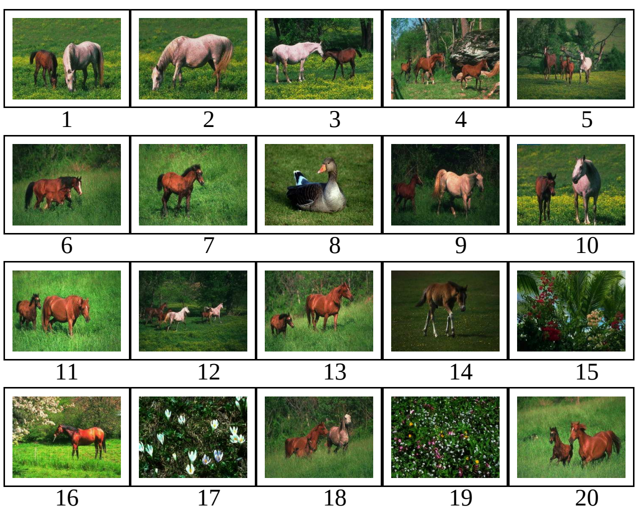
[19] X. S. Zhou and T. S. Huang, "Comparing discriminating transformations and SVM for learning during multimedia retrieval," in *ACM Multimedia*, pp. 137–146, 2001.



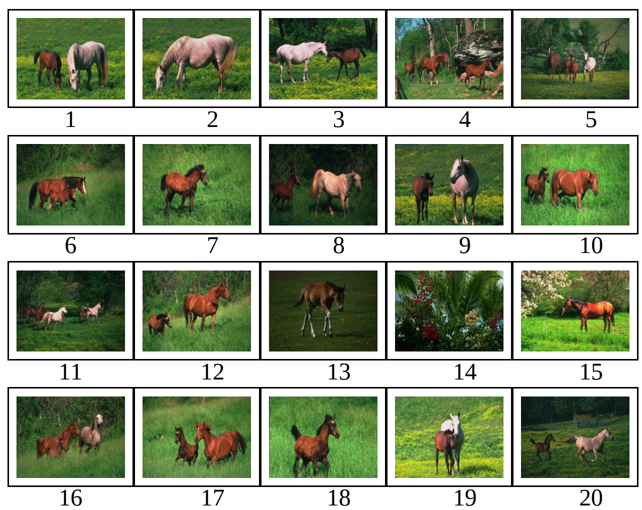
(a) Query image: Horses



(b) Without feedback.



(c) Cluster feedback.



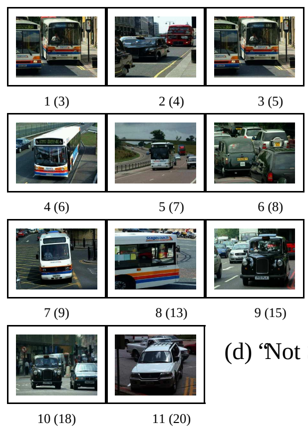
(d) Two-class feedback.

Figure 4: A query of horses. Default precision in (b) is 50%. Cluster feedback increases precision to 80% in (c). Two-class feedback further increases the precision to 95% in (d). Equal weights for structure, color and texture are used.

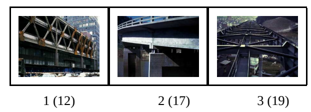
(a) Query image:
Yellow automobile


(b) Initial results.
Weights: S = 0.33, C = 0.33, T = 0.33
L, A, and B channels used for texture.

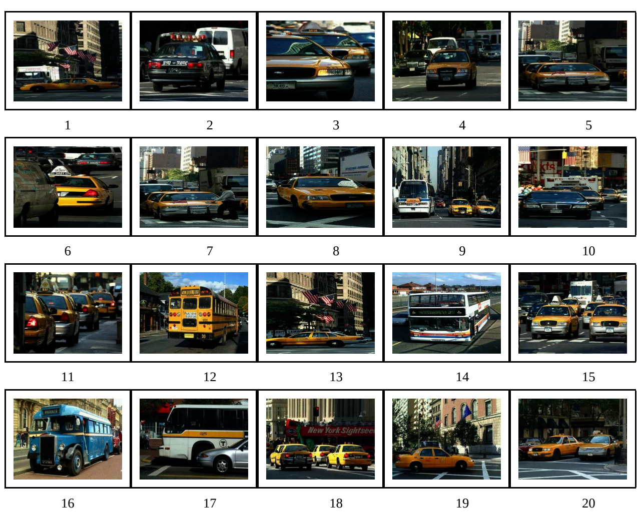

(c) Positive images.




(d) "Not Sure" images.




(e) Negative images.




(f) Weighed feature integration
and three-class feedback.
Weights: S = 0.3, C = 0.5, T = 0.2
L channel used only for texture.

Figure 5: Combination of weighted feature integration and three-class feedback. Default setting in (b) has a precision of 30%. Precision increases to 75% in (f). S = Structure, C = Color, T = Texture.