Learning Similarity Measure for Natural Image Retrieval With Relevance Feedback

Guo-Dong Guo, Anil K. Jain, *Fellow, IEEE*, Wei-Ying Ma, *Member, IEEE*, and
Hong-Jiang Zhang, *Senior Member, IEEE*

***Abstract*—A new scheme of learning similarity measure is proposed for content-based image retrieval (CBIR). It learns a boundary that separates the images in the database into two clusters. Images inside the boundary are ranked by their Euclidean distances to the query. The scheme is called constrained similarity measure (CSM), which not only takes into consideration the perceptual similarity between images, but also significantly improves the retrieval performance of the Euclidean distance measure. Two techniques, support vector machine (SVM) and AdaBoost from machine learning, are utilized to learn the boundary. They are compared to see their differences in boundary learning. The positive and negative examples used to learn the boundary are provided by the user with relevance feedback. The CSM metric is evaluated in a large database of 10 009 natural images with an accurate ground truth. Experimental results demonstrate the usefulness and effectiveness of the proposed similarity measure for image retrieval.**

***Index Terms*—AdaBoost, constrained similarity measure, content-based image retrieval, feature selection, learning, relevance feedback, support vector machine (SVM).**

I. Introduction

VISUAL information retrieval has been an active research problem in multimedia applications [7], [8], [12], [14], [19]. One approach is to use keywords or text descriptions for indexing and retrieval of image data. However, there are several problems inherent in systems that are exclusively text-based. First, automatic generation of descriptive keywords or extraction of semantic information for broad varieties of images is beyond the capacity of current computer vision and artificial intelligence technologies. Thus, text descriptors have to be typed in by human operators, which is very time consuming and the results are usually inaccurate and incomplete. Second, certain visual properties, such as textures and color patterns, are often difficult, if not impossible, to describe with text in an objective way for general purpose usage. The old saying "An image says more than a thousand words" definitely still holds.

An alternative to text-based indexing of images is to work with descriptions based on the visual features of an image, such as colors, textures, patterns, and shapes. This scheme is the so called content-based image retrieval (CBIR) [12], [14]. For many applications, such indexing schemes may be either supplemental or preferable to text, and in some other cases they may be indispensable. Moreover, visual queries may be easier to formulate [8].

Image retrieval differs from traditional pattern classification such as face detection and digit recognition. In retrieval, there is a user in the loop. The image retrieval system should take into consideration human perceptual similarity between the query and the retrieved images. Thus the process is subjective in a sense [7]. On the contrary, the job of classification is relatively objective and defined more clearly. The classification results are typically the returned class labels or probabilities that a test example belongs to each class.

Relevance feedback (RF) is a powerful technique for interactive image retrieval [17]. Minka and Picard [11] presented a learning technique for interactive image retrieval. The key idea behind this approach is that each feature model has its own strength in representing a certain aspect of image content, and thus, the best way for effective content-based retrieval is to utilize "a society of models." A typical approach in relevance feedback is to adjust the weights of various levels to accommodate the user's need [16], [17]. Another method is to modify and convert the query into a new representation by using the positive and negative examples provided by the users [16]. In [6], relevance feedback is used to modify the weighted metric for computing the distance between feature vectors. The basic idea is to enhance the importance of those dimensions of an feature that help in retrieving the relevant images and reduce the importance of those dimensions that hinder this process.

In this paper, we proposed a technique that learns a boundary to separate the positive and negative examples provided by relevance feedback. Support vector machine (SVM) and AdaBoost are used to learn the boundary which is utilized to filter the images in the database for Euclidean similarity measure. Another approach to filtering is to classify the images in the database into semantic or high-level categories [25]. In our system, the images inside the boundary are compared with the query based on the Euclidean distance, while the images outside the boundary are ranked by their distances to the boundary. The key idea is to constrain the images used for similarity measure with respect to the query. We first provide the motivation of our approach in Section II. Then, we introduce our scheme for image representation in Section III, and the metric of constrained similarity measure in Section IV. The retrieval performance of the proposed method is presented in Section V. Finally, we discuss future research directions and give the conclusions.

Manuscript received April 15, 2001; revised October 30, 2001.

G.-D. Guo was with Microsoft Research China, Beijing 100080, P.R. China. He is now at the Computer Sciences Department, University of Wisconsin, Madison WI 53706 USA (e-mail: gdguo@cs.wisc.edu).

A. K. Jain is with the Department of Computer Science and Engineering, Michigan State University, East Lansing MI 48823 USA.

W.-Y. Ma and H.-J. Zhang are with Microsoft Research China, Beijing 100080, P.R. China.

Publisher Item Identifier S 1045-9227(02)04424-7.

II. Motivation of Our Approach

Similarity measure is a key component in image retrieval. Traditionally, Euclidean distances are used to measure the similarity between the query and the images in the database. The smaller the distance, the more similar the pattern to the query. However, this metric is sensitive to the sample topology, which can be illustrated in Fig. 1(a). Assume the point "A" is the query, the Euclidean distance-based similarity measure can be viewed as drawing a hyper-sphere in the high-dimensional feature space (or a circle in two dimensions), centered at point "A." The larger the radius of the hyper-sphere, the more images are enclosed in the hyper-sphere, as shown in Fig. 1(a). The radius is determined indirectly by the number of retrieved images. For different queries, the centers move accordingly. As a result, the retrieved images enclosed by the hyper-sphere are different although these query images are perceptual similar. Furthermore, many irrelevant images could be enclosed by the regular hyper-sphere and retrieved to the user. To solve these problems, we propose to use an "irregular" nonsphere boundary to enclose the similar images inside and the Euclidean distance measure is applied only to those images inside the boundary, as shown in Fig. 1(a). For query "A," the Euclidean similarity measure are only used to rank images inside the boundary. In this case, the relevant images can be retrieved in the top matches. Our approach is capable of learning the boundary from both positive and negative examples.

The boundary is used to discriminate between the similar images and the others in the database. One possible scheme to identify the boundary is Bayesian decision function. However, the Bayesian classifier usually needs a large number of examples to estimate the model parameters, which is impractical for image retrieval because we can not expect the users to submit many positive and negative examples in the interactive process. Instead, we decided to use learning techniques that are nonparametric and do not need a large number of examples to learn a decision boundary. Large margin classifiers, such as SVM [26] and AdaBoost [4], can be used for such purpose. Duin [3] explained and illustrated why the SVM can work for small training sets. Here we use both SVM and AdaBoost to learn the boundary and compare their performance in image retrieval.

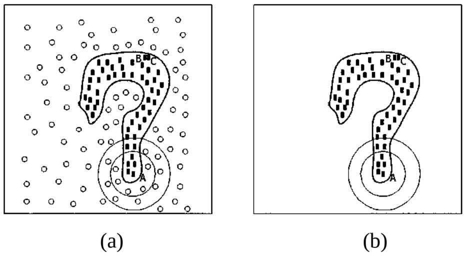


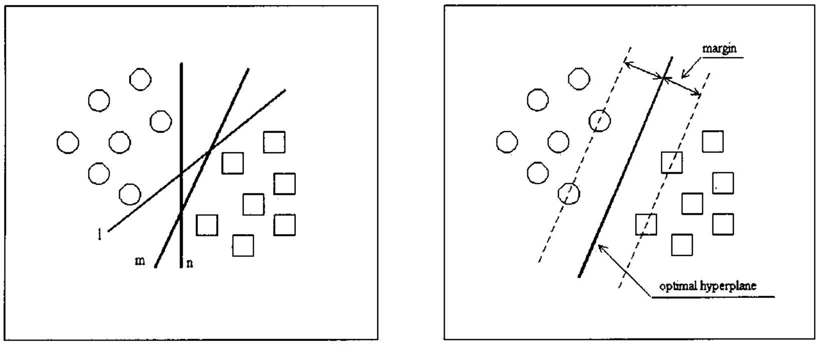
Fig. 1. Perceptually similar images denoted as rectangle patterns and all the others denoted by circle patterns. For query "A," similarity measure based on traditional Euclidean distance can be viewed as drawing circles centered at "A." A boundary can be constructed to separate these two groups of patterns as shown in (a), then the Euclidean distance measures only focus on the rectangle patterns, as if the circle patterns do not exit, as shown in (b). Note that patterns "B" and "C" are very similar and close to the boundary.

One issue should be noticed. That is, can we directly use the distances of the images to the boundary for the similarity measure? The answer is "no." Suppose a query image "B" is given by the user, which is very similar to image "C," as shown in Fig. 1(a) and (b). Both "B" and "C" are located in the positive side of the boundary, and yet close to the boundary. In such a case, other images with large distances to the boundary will always be ranked in the top matches when the distance-from-boundary metric is used for similarity measure, while image "C" can only be retrieved for example after top 20 matches or even more. In an extreme case, image "C" is the same as "B," but can not be retrieved in the top one or two matches. On the contrary, if we use Euclidean distance measure for the small number of images filtered by the boundary, the image "C" can usually be retrieved in the top one or two matches. Furthermore, there is no evidence to prove that the distance-from-boundary measure is a good metric for image similarity measure. To sum up, image similarity is very subjective [7] and different from the task of classification.

III. Image Representation

Color information is the most intensively used feature for image retrieval because of its strong correlation with the underlying image objects or scenes. Compared to other low-level visual information, color is more robust with respect to scaling, orientation, perspective, and occlusion of images [22]. Two issues are essential for color features. The first is to select a proper color space for representing color content of images, and the second is to choose a color quantization scheme to reduce the dimensions of a color feature. We use the hue, saturation, and value (HSV) color space because it provides the best retrieval performance for color histograms [8]. In our approach, the color histogram [22] is quantized with 256 levels, which results in 256 features for each image. Color moments of an image is another set of color features, which are very simple yet very effective feature for color-based image retrieval [21]. It does not require color quantization. The mathematical foundation of this feature is that any color distribution of images can be characterized by its moments. The first-, second-, and third-order moments of images are calculated in the HSV space of each image, resulting in a feature vector of dimension nine. Because color histograms and moments lack information about spatial distribution of colors in an image, another feature called color coherence vector (CCV) is proposed to incorporate spatial information into color histogram representation [13]. We calculate the CCV features of images with 64 quantization, which results in feature vectors of 128 dimensions.

Texture is another type of low-level image features that has been used extensively for content-based image retrieval. The Tamura features are designed based on the psychological studies in human visual perceptions of textures [23]. We select to compute the coarseness histogram with ten quantization, and the histogram of directionality with eight quantization. Other texture features used in our approach are the wavelet coefficients. Wavelet transforms refer to the decomposition of a signal with a family of basis functions obtained through translation and dilation of a mother wavelet. The pyramidal wavelet transform (PWT) [10] is used and the mean and standard deviation of the



(a) (b)

Fig. 2. Classification between two classes using hyperplanes: (a) arbitrary hyperplanes l, m, and n and (b) the optimal separating hyperplane with the largest margin identified by the dashed lines, passing the support vectors.

energy distribution are calculated corresponding to each of the subbands at each decomposition level. For three-level decomposition, PWT results in a feature vector of 3 × 4 × 2 components.

Various features have their own strength in representing a certain aspect of image content. We concatenate above color and texture features into one feature vector of dimension 435 to represent each image in the database. The features are normalized into a normal distribution in each dimension, separately.

IV. Constrained Similarity Measure

In our image representation scheme, each image is transformed into a feature point in the feature space. In retrieval, we use the constrained similarity measure (CSM) with the constraints imposed by the boundary between the positive and negative examples.

A. Providing Examples to Learn the Boundary

How to provide the system with some positive and negative examples? One way is to present a set of preselected positive and negative examples for each query class as in [24]. However, when a new query is given, it may be far away from the positive examples, and thus located outside the prelearned boundary (may be far away from the boundary, too). The retrieval results may be strange to the user. Another way to provide examples to the system is to use relevance feedback technique, which is natural and adaptive. The learned boundary is adapted to each query in our scheme. The system learns the boundary iteratively through the user's relevance feedback in the interactive process.

B. Learning the Boundary With SVM

We first describe the basic theory of SVM which is used to learn the boundary.

1) Basic Theory of Support Vector Machines: Given a set of training vectors belonging to two separate classes, $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_l, y_l)$, where $\mathbf{x}_i \in R^n$ and $y_i \in \{-1, +1\}$, one wants to find a hyperplane $\mathbf{wx} + b = 0$ to separate the data. In Fig. 2(a), there are many possible hyperplanes, but there is only one [shown in Fig. 2(b)] that maximizes the margin (the distance between the hyperplane and the nearest data point of each class). This linear classifier is termed the optimal separating hyperplane (OSH).

The solution to the optimization problem of the SVM is given by the saddle point of the Lagrange functional

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \parallel \mathbf{w} \parallel^2 - \sum_{i=1}^{l} \alpha_i \left\{ y_i \left[(\mathbf{w} \cdot \mathbf{x}_i) + b \right] - 1 \right\} \quad (1)$$

where α_i are the Lagrange multipliers. Classical Lagrangian duality enables the *primal* problem (1) to be transformed to its *dual* problem, which is easier to solve. The solution is given by

$$\bar{\mathbf{w}} = \sum_{i=1}^{l} \bar{\alpha}_i y_i \mathbf{x}_i, \quad \bar{b} = -\frac{1}{2} \bar{\mathbf{w}} \cdot [\mathbf{x}_r + \mathbf{x}_s] \quad (2)$$

where $\mathbf{x}_r$ and $\mathbf{x}_s$ are any two support vectors with $\bar{\alpha}_r, \bar{\alpha}_s > 0$, $y_r = 1$, and $y_s = -1$.

To solve the nonseparable problem, Cortes and Vapnik [2] introduced slack variables $\xi_i \geq 0$ and a penalty function, $F(\xi) = \sum_{i=1}^{l} \xi_i$, where the ξ are a measure of the misclassification error. The solution is identical to the separable case except for a modification of the Lagrange multipliers as $0 \leq \alpha_i \leq C$, $i = 1, \ldots, l$. The choice of C is not strict in practice, and we set $C = 100$ in all our experiments. We refer to [26] for more details on the nonseparable case.

The SVM can realize nonlinear discrimination by kernel mapping [26]. When the samples in the input space can not be separated by any linear hyperplane, they may be linearly separated in the nonlinear mapped feature space. Note that here the feature space of the SVMs is different from the image feature space.

There are many kernel functions for nonlinear mapping [26]. We choose to use the Gaussian radial basis function (GRBF) as the kernel function in our experiments, which has the form, $K(\mathbf{x}, \mathbf{y}) = \exp\left(-(\mathbf{x} - \mathbf{y})^2 / \sigma^2\right)$, where parameter σ is the width of the Gaussian function.

For a given kernel function, the SVM classifier is given by

$$f(\mathbf{x}) = \operatorname{sign}\left(\sum_{i=1}^{l} \bar{\alpha}_i y_i K(\mathbf{x}_i, \mathbf{x}) + \bar{b} \right) \quad (3)$$

and the decision boundary is

$$\sum_{i=1}^{l} \bar{\alpha}_i y_i K(\mathbf{x}_i, \mathbf{x}) + \bar{b} = 0. \quad (4)$$

C. Learning the Boundary With AdaBoost

Boosting is a method to combine a collection of weak classification functions (weak learner) to form a stronger classifier. AdaBoost is an adaptive algorithm to boost a sequence of classifiers, in that the weights are updated dynamically according to the errors in previous learning [4]. AdaBoost is a kind of large margin classifier. Tieu and Viola [24] adapted the AdaBoost algorithm for natural image retrieval. They let the weak learner work in a single feature each time. So after T rounds of boosting, T features are selected together with the T weak classifiers. Tieu and Viola's AdaBoost algorithm [24] is briefly described as follows.

```
AdaBoost Algorithm
  Input: 1) $n$ training examples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ with $y_i = 1$ or 0; 2) the number of iterations $T$.
  Initialize weights $w_{1,i} = 1/2l$ or $1/2m$ for $y_i = 1$ or 0, respectively, with $l + m = n$.
  Do for $t = 1, \ldots, T$:
  1) Train one hypothesis $h_j$ for each
     feature $j$ with $w_t$, and error
     $\epsilon_j = \Pr_i^{w_t}[h_j(x_i) \neq y_i]$.
  2) Choose $h_t(\cdot) = h_k(\cdot)$ such that $\forall\ j \neq k$,
     $\epsilon_k < \epsilon_j$. Let $\epsilon_t = \epsilon_k$.
  3) Update: $w_{t+1,i} = w_{t,i}\beta_t^{e_i}$, where $e_i = 1$
     or 0 for example $x_i$ classified correctly or incorrectly respectively,
     with $\beta_t = \epsilon_t/(1 - \epsilon_t)$ and $\alpha_t = \log 1/\beta_t$.
  4) Normalize the weights so
     that they are a distribution,
     $w_{t+1,i} \longleftarrow (w_{(t+1,i)}/\sum_{j=1}^{n} w_{t+1,j})$.
  Output the final hypothesis
```

$$h_f(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

In [24], image retrieval is realized by using the AdaBoost classifier. However, the authors did not consider the perceptual similarity between the images. In fact, there is no evidence to show that the distance of images to the decision boundaries can be used as a measure of perceptual similarities. Here, we use the AdaBoost [24] algorithm to learn a boundary for a given query and do a comparison with the SVM based boundary learning approach. Further more, in [24], only a set of preselected images are used to represent each class to learn a fixed decision boundary. In our experiments, we learn the boundary adaptive to each query in the interactive process, and give a comprehensive evaluation on a large image database. The goal of our scheme is to learn a boundary to filter the images for late stage Euclidean distance measure.

D. Similarity Measure Constrained by the Boundary

For a query, after the boundary is learned based on the user's feedback, the images in the database are filtered by the boundary. The images inside and outside the boundary are treated differently. For the images inside the boundary, we rank them based on their Euclidean distances to the query. It is well known that in the CIE $L^*a^*b^*$ and $L^*u^*v^*$ color space [28], the Euclidean distance between two colors is proportional to their perceptual dissimilarity [15]. Thus the Euclidean distance can be used as a similarity measure for color images. Currently, there are no texture features where the Euclidean distance corresponds to human perceptual dissimilarity, yet the Euclidean distance can be used intuitively for a texture image similarity measure [9]. On the contrary, the images outside the boundary are ranked only based on their distances to the boundary. There is no evidence that this kind of distance can be used as a similarity or dissimilarity measure. An intuition is that the images similar to the query may be outside the boundary because there is no guarantee that the similar images are always enclosed inside the boundary, but typically they are not far away from the boundary. So, these images can be retrieved after the positive images (located inside the boundary, with positive distances to the boundary) if they are ranked by their distances to the boundary. For this reason, we use the distance-from-boundary (DFB) measure to deal with the negative images (located outside the boundary). Why we need to rank negative images? There are two considerations. The first one is some perceptual similar images may not be enclosed by the learned boundary. If we discard these negative images, they may not be retrieved forever. The second is sometimes the user would like to browse many images beyond the number of images enclosed by the boundary. If the images outside the boundary are discarded, the number of images to retrieve for the user is insufficient. Our strategy for the similarity measure is called constrained similarity measure (CSM). Note that the DFB measure is only used for the images outside the boundary.

The restricted similarity measure can be formulated as follows:

$$S(\mathbf{x}, q) = \begin{cases} ED(\mathbf{x}, q), & \text{if } D(\mathbf{x}, \Theta) \geq 0 \\ M - D(\mathbf{x}, \Theta), & \text{otherwise} \end{cases} \tag{6}$$

where $S(\mathbf{x}, q)$ denotes the similarity measure of the image $\mathbf{x}$ with respect to the query q, and $D(\mathbf{x}, \Theta)$ represents the distance of $\mathbf{x}$ to the boundary characterized by a parameter set Θ. The distance of the image $\mathbf{x}$ to the boundary $D(\mathbf{x}, \Theta) = 0$ is calculated by

$$D(\mathbf{x}, \Theta) = \sum_{i=1}^{S} \bar{\alpha}_i y_i K(\mathbf{x}_i, \mathbf{x}) + \bar{b} \tag{7}$$

for the SVM learned boundary, and computed by

$$D(\mathbf{x}, \Theta) = \sum_{t=1}^{T} \alpha_t \left(h_t(x) - 0.5\right) \tag{8}$$

for AdaBoost learned boundary. In addition

$$ED(\mathbf{x}, q) = \|\mathbf{x} - q\|_2 \tag{9}$$

is the Euclidean distance between image $\mathbf{x}$ and the query q. While M is the maximum Euclidean distance among the positive images to the query

$$M = \max_{\mathbf{x}} ED(\mathbf{x}, q), \quad \forall\, D(\mathbf{x}, \Theta) \geq 0. \tag{10}$$

$M - D(\mathbf{x}, \Theta)$ can be viewed as a kind of *pseudo-Euclidean distance* measure for ranking any negative image $\mathbf{x}$.

V. Retrieval Performance

Our constrained similarity measure is evaluated using a subset of the Corel photo Gallery image database. We select 10 009 images with ground truth of 79 concepts or classes. There are two goals in our evaluation. First, we want to see if the retrieval performance can be improved based on the CSM scheme. Second, we want to find out which method is better for learning the boundary, i.e., with better generalization capability.

Recall and precision are used to evaluate the retrieval performance. Recall is the ratio of the number of relevant images returned to the total number of relevant images. Precision is the ratio of the number of relevant images returned to the total number of images returned. We calculate precision and recall with respect to the number of relevance feedback for evaluation.

The results of the traditional Euclidean distance measure are given as a baseline in the evaluation. Note that although retrieval results based on the Euclidean distance measure are shown in the same figure in the following experiments, there is no feedback (on learning or we call no constraint) to it. The curves for the Euclidean distance measure are just drawn with respect to the number of images to display and used to show the improvements after learning.

A. Image Database

The image database is a selected subset from Corel Gallery 1 000 000 of 14 CDs, which is a collection of clipart, professional photos, Web images, animations, sounds, videos, and fonts. We selected about 10 000 photos from Corel Gallery as our natural image database. It should be noted that Corel Photo Gallery use semantic concepts to group the photos, each with 100 images. However, there are two problems if we directly use the Corel's division as the ground truth. First, some images have the same or similar content but divided into different directories. For instance, the images in directories of "Ballon1" and "Ballon2," "Fruit1" and "Fruit2," "Cuisine" and "Cuisines," and so on. Considering them as different concepts may cause some problems in performance evaluation. To avoid this problem, we put these images into the same concept or group. Second, some "concepts" are very abstract and the images within the same concept can be largely varied in content. For instance, the concepts of "Spring," "Winter," "Hong Kong," and "Montreal" are very abstract and the variations of image content within these concepts are very large, hence it is very difficult for the current algorithms to measure image similarities for these concepts. Therefore we do not include these images in our image database.

Based on above considerations, we construct an image database of 10 009 images of 79 groups. Each group has a semantic name. The number of images within each group ranges from 100 to 300. For "Race Car," we add some images from the "Speed" directory in the original Corel Gallery CDs (474 007–474 015), in addition to the images in "Car_perf" and "Car_Race," hence there are 209 images belonging to our "Race Car" concept. As for other concepts, there are usually 100–300 images.

B. Experimental Results

There are two goals in our experiments. The first goal is to evaluate whether the constrained similarity measure can deliver a better retrieval results; The second is to compare the two methods for learning boundaries to see which one has a better retrieval performance.

We select nine concepts out of 79 to evaluate the retrieval performance based on our constrained similarity measure. They are "flower" (200), "leopard" (100), "model" (300), "mountain" (200), "plane" (200), "race car" (209), "sunsets" (200), "tiger" (100), and "waterfall" (100), as shown in Fig. 3. The numbers indicate how many images belong to each "concept" in our image database.

In relevance feedback, the retrieved images are shown in the screen each time, while the remaining images are left for another round of feedback if the user actually has some responses. The user clicks on similar images as positive examples while leaving the unclicked ones as negative examples. All these responses are taken from the user's interaction with the system. In order to get the performance evaluation curves, we simulate the user's behavior as follows: 40 images are shown to the user each time, and the user clicks all the similar images to submit positive examples. The number of negative examples typically become very large with the number of interactions. The users usually do not like to click on negative examples frequently. For this concern, we just select the negative examples submitted by the user in the first round, and keep them unchanged. In later stage, the users only submit positive examples to the system. We assume the users know the ground truth in the interaction. In the precision and recall curves, the feedback times are nine, and the zero feedback refers to the retrieval results based on the starting Euclidean distance measure without any learning, and the first 40 images are displayed to the users. After that, the system learns a boundary, and the boundary is updated iteratively in response to user's interaction in later steps.

For each concept, the precision and recall are averaged over all query images belonging to that concept, instead of just averaged over several random selected queries. The computation of whole average should be a more objective evaluation.

Because of space limits, we do not show the images in the retrieval process. The precision and recall curves calculated for the nine concepts (illustrated in Fig. 3) are shown in Figs. 4–6, separately. In [5], we gave the total average over the nine concepts. However, we believe that the retrieval performance shown for each individual concept is more specific and illustrating. From these figures, it is obvious that both precision and recall are explicitly improved by using the boundary constrained similarity measure. Even only after one or two iterations of relevance feedback, the performance has dramatically improved. Another observation is that using boundaries learned by SVMs can usually deliver a better

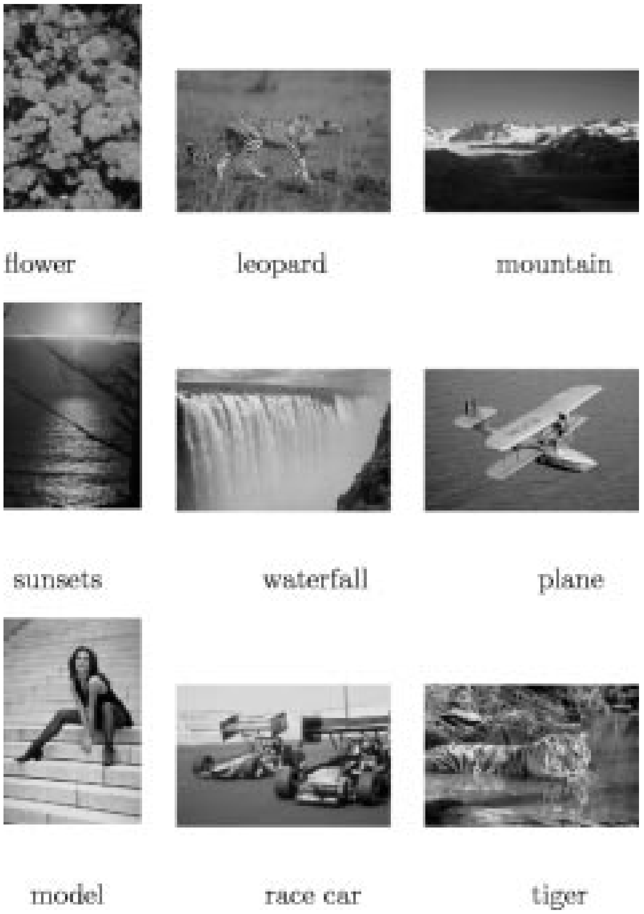


Fig. 3. Some images of the selected nine concepts in our experimental evaluation.

result in comparison with that learned by AdaBoost, such as in retrieval of “flower,” “leopard,” “mountain,” “waterfall,” “plane,” “race car,” and “tiger.” The AdaBoost based boundary learning can only present performance close to the SVM based approach for “sunsets” and “model,” as shown in Figs. 4(d1), (d2), and 5(f1), but still worse than that based on SVM. Further more, the worst cases for AdaBoost approach are in the retrieval of “race car” and “tiger,” in which the boundary constraints do not improve (in Fig. 6) or only improve marginally [in Fig. 5(h1) and (h2)] over the Euclidean distance measure. In summary, the retrieval performance can always be improved by the SVM-based constrained similarity measure, while the AdaBoost based CSM can improve the retrieval performance in most cases, but sometimes may have no improvement.

In addition, we just set $\sigma = 1$ for the SVM GRBF kernel function in all our experiments. Even better results can be obtained if the kernel parameter is selected more carefully and changed adaptively to different queries.

It should be noted that with relevance feedback, the recall curves go up no matter what methods are used, since more and more relevant or perceptually similar images are retrieved when the user interacts with the system and browses more and more images, while the number of relevant images in the database in fixed(assume we know the ground truth in the evaluation). However, the precision curves usually go down with respect to the times of relevance feedback, since the number of relevant images returned to the user becomes small in the later rounds of feedback. From the definition of precision in the second paragraph of Section V, it is not difficult to understand this behavior. The readers should not confuse with the precision and recall curves versus the times of relevance feedback. One interesting observation is that the precision curves of CSM (especially with SVM) go up in the first one or two rounds of feedback, and then go down gradually, but still explicitly above the Euclidean distance-based retrieval without constraint. This is because a large number of similar images is retrieved for the user in the first one or two feedback, which is a very useful property, as usually the user may not like to do many times (e.g., five) of relevance feedback in practice. The reason that we show here nine times of feedback is mostly for the goal of evaluation.

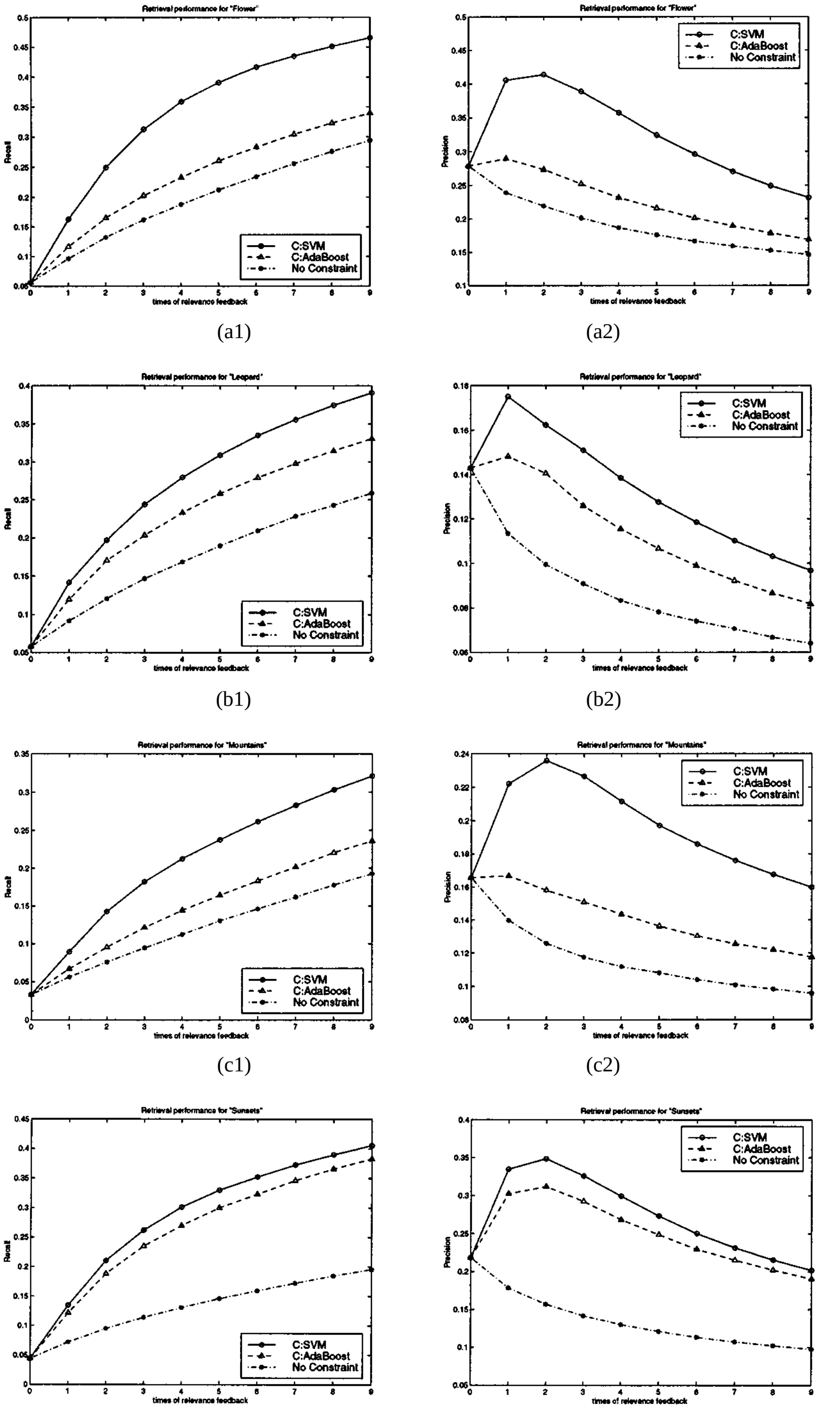
VI. Discussion and Future Research Efforts

Selecting a small set of features and reducing the number of support vectors can largely improve the speed for SVM-based boundary constraints. In addition, two methods for boundary learning can be supplemented in some cases or combined together to further improve the overall retrieval performance.

A. Feature Selection

In our constrained similarity measure with SVM to learn the boundary, we use all 435 features. A further consideration is to reduce the feature dimensions so as to speed up the retrieval process. In AdaBoost [24], feature selection is incorporated into the learning stage. Usually, 20 rounds of boosting is enough to learn the boundary in image retrieval, and hence 20 features are used for retrieval. We like to see if a similar method can be used for SVM to simply select a small number of features. For this purpose, we tried a simple method for feature selection for SVM, as that used in [24] for AdaBoost. That is, the features are ranked independently based on their discriminative power given the feedback examples. Then the first m features are used for SVM to learn the boundary. In Fig. 7, we show the averaged precision and recall performance over 200 images of the “flower” concept, with $m = 20$ features selected and used for SVM, denoted as “C:rSVM-20” for simplicity. It is obvious that its performance is worse than the traditional Euclidean distance metric. To see whether it is because the number of features is too small, we let $m = 50$ and $m = 100$ and show the results in the same figure. The performance of “C:rSVM-50” and “C:rSVM-100” are still worse than the AdaBoost-based approach, even though so many features are selected and used, which indicates the major problem is not the number of features to select. The simple feature selection method similar to that in AdaBoost [24] can not be used for SVM. A more elaborate algorithm should be used to select features for SVM. This also indirectly indicates the different mechanism for SVM and AdaBoost, although both of them are termed as large margin classifiers.

A novel method for feature selection for SVM has been recently proposed by Weston [27]. Currently, we are trying to evaluate that method to see if it can be used for feature selection in our image retrieval problem.



(a1) (a2)

(b1) (b2)

(c1) (c2)

(d1) (d2)

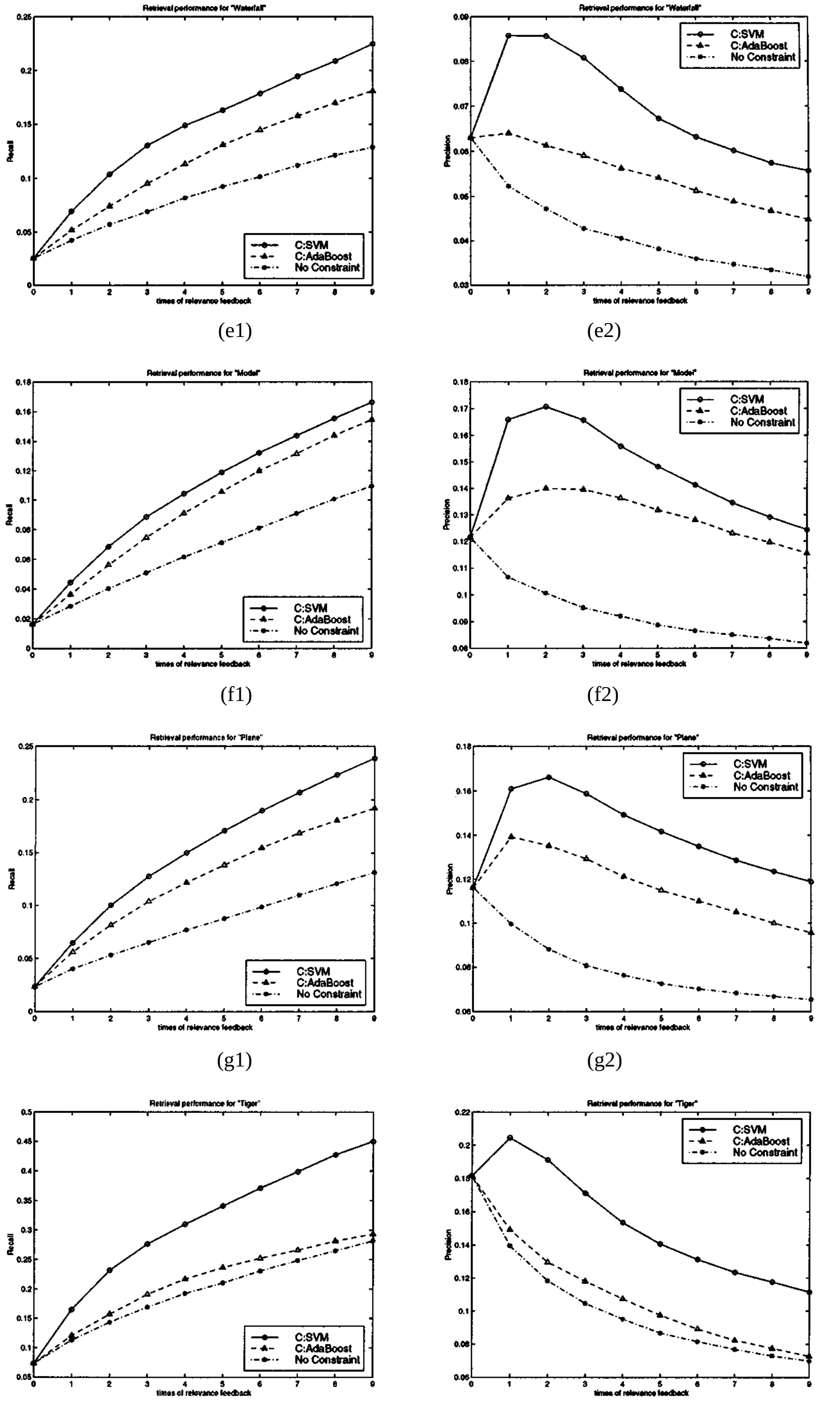
Fig. 4. Averaged precision and recall versus the number of relevance feedback of C:SVM, C:AdaBoost, and the traditional Euclidean distance measure (No Constraint), for concepts of "Flower" in (a1) and (a2), "Leopard" in (b1) and (b2), "Mountains" in (c1) and (c2), and "Sunsets" in (d1) and (d2).

B. Reducing the Number of Support Vectors

The number of support vectors are determined automatically in SVM learning. If there are too many examples submitted by the user (although it is usually not the case in practice), and the boundary is very complex, the SVM decision boundary will be constructed by too many support vectors. Thus the filtering process using the SVM will be slow. Burges [1] and Scholkopf [20] have introduced two methods to reduce the support vectors without explicitly losing the classification accuracy. These methods are expected to largely speed up the retrieval process, which are currently under evaluation for our image retrieval problem.

C. Learning Method Selection and Combination

It is obvious that the average retrieval performance of constrained similarity measure (CSM) with SVM-based boundary learning (noted as "C:SVM") is much better than the CSM



(e1) (e2)

(f1) (f2)

(g1) (g2)

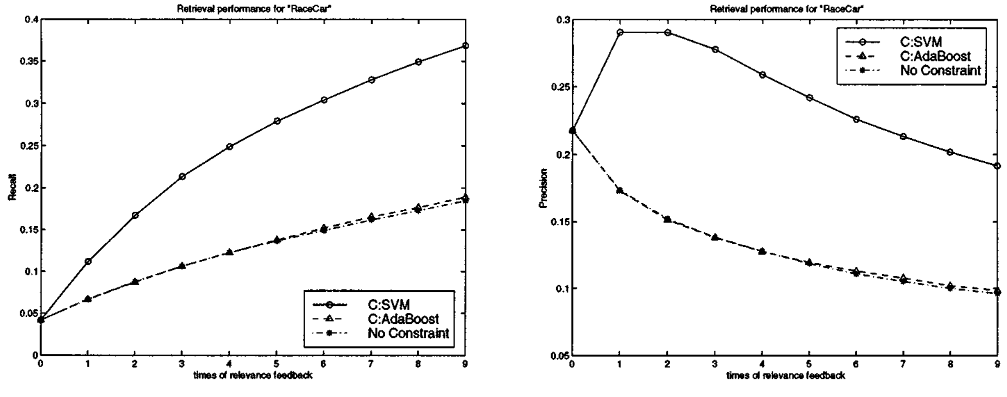
(h1) (h2)

Fig. 5. Averaged precision and recall for concepts of "Waterfalls" in (e1) and (e2), "Model" in (f1) and (f2), "Plane" in (g1) and (g2), and "Tiger" in (h1) and (h2).

with AdaBoost (noted as "C:AdaBoost"). However, it does not necessarily mean that for each query, the performance of "C:SVM" is superior to the "C:AdaBoost." For instance, in Fig. 8, we show the precision and recall curves versus the number of relevance feedback for query "360 001" in the "plane" class. The performance of "C:AdaBoost" is much better than "C:SVM." This demonstrates that sometimes "C:AdaBoost" can be a better choice than "C:SVM." So, for a given query, how to select a better method, or combine above two approaches in order to deliver a more satisfactory result is an open issue for future research.

VII. Conclusion

We have presented a constrained similarity measure for content based image retrieval. This measure takes into consideration the perceptual similarity between images and improves the retrieval performance. Two techniques are used to learn the boundary, and the experimental results indicate that the SVM-based method is better than the AdaBoost-based approach. Hence more research is needed for AdaBoost to improve its generalization capability in learning a decision boundary. As for SVM, further research is needed for selecting



(i1) (i2)

Fig. 6. Averaged precision and recall for concept "Race car" in (i1) and (i2).

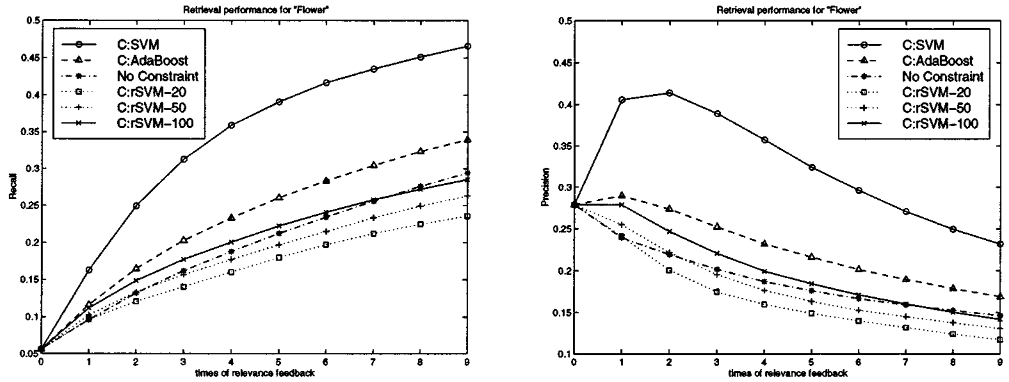


Fig. 7. Retrieval performance averaged over 200 queries of "flower" images to evaluate the simple scheme of feature selection for SVM (denoted as C:rSVM), as compared with the C:SVM, C:AdaBoost, and the Euclidean distance measure. The number of selected features is $m = 20$, 50, and 100, denoted as C:rSVM-20, C:rSVM-50, and C:rSVM-100, respectively.

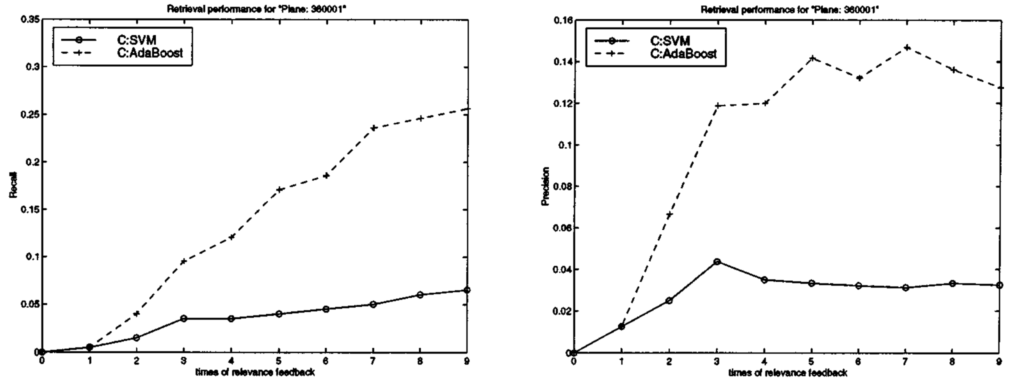


Fig. 8. Retrieval performance comparison between C:SVM and C:AdaBoost, for the query of "360 001" in "plane" group. The latter has better performance in this case.

a small set of features and reduce the number of support vectors to speed up the filtering process.

Acknowledgment

The authors are grateful to the anonymous reviewers who gave many suggestions to improve the paper. Many thanks to K. Tieu and G. Ratsch for helpful discussions on AdaBoost algorithm.

References

[1] C. J. C. Burges, "Simplified support vector decision rules," in *Proc. 13th Int. Conf. Machine Learning*, L. Saitta, Ed., 1996, pp. 71–77.

[2] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[3] R. P. W. Duin, "Classifiers in almost empty spaces," in *Proc. Int. Pattern Recognition*, vol. 2, 2000, pp. 1–7.

[4] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of online learning and an application to boosting," *J. Comp. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[5] *Learning Similarity Measure for Natural Image Retrieval With Relevance Feedback*. Kauai, HI, 2001, pp. 731–736.

[6] J. Huang, S. R. Kumar, and M. Metra, "Combining supervised learning with color coorelograms for content-based image retrieval," in *Proc. ACM Multimedia'95*, 1997, pp. 325–334.

[7] B. Johansson. (2000) A Survey on: Contents Based Search in Image Databases. [Online]. Available: http://www.isy.liu.se/cvl/Projects/VISIT-bjojo/

[8] W. Y. Ma and H. J. Zhang, "Content-based image indexing and retrieval," in *Handbook of Multimedia Computing*, B. Furht, Ed. Boca Raton, FL: CRC, 1998, ch. 11.

[9] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Pattern. Anal. Machine Intell.*, vol. 18, pp. 837–842, Aug. 1996.

[10] S. G. Mallat, "A theory for mutiersolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989.

[11] T. P. Minka and R. W. Picard, "Interactive Learning Using a Society of Models," MIT Media Lab., Cambridge, MA, 349, 1995.

[12] W. Niblack *et al.*, "The QBIC project: Querying images by content using color, texture, and shape," in *Proc. SPIE*, vol. 1908, Storage and Retrieval for Image and Video Databases-II, San Jose, CA, Feb. 1993, pp. 173–187.

[13] G. Pass and R. Zabih, "Histogram refinement for content-based image retrieval," in *IEEE Workshop Applicat. Comput. Vision*, 1996, pp. 96–102.

[14] A. Pentland, R. W. Picard, and S. Sclaroff, "Photobook: Tools for content based manipulation of image databases," in *Proc. SPIE*, vol. 2185, Storage and Retrieval for Image and Video Databases-II, San Jose, CA, Feb. 1994, pp. 34–47.

[15] J. Puzicha, J. M. Buhmann, Y. Rubner, and C. Tomasi, "Emperical evaluation of dissimilarity measures for color and texture," in *Proc. ICCV*, vol. II, 1999, pp. 1165–1172.

[16] Y. Rui *et al.*, "A relevance feedback architecture in content-based multimedia information retrieval systems," in Proc. IEEE Workshop Content-Based Access of Image and Video Libraries, 1997.

[17] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: A powerful tool for interactive content-based image retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, pp. 644–655, Sept. 1998.

[18] S. Santini and R. Jain, "Similarity measures," *IEEE Pattern Anal. Machine Intell.*, vol. 21, pp. 871–883, Sept. 1999.

[19] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 1349–1380, Dec. 2000.

[20] B. Scholkopf *et al.*, "Input space versus feature space in kernel-based methods," *IEEE Trans. Neural Networks*, vol. 10, pp. 1000–1017, Sept. 1999.

[21] M. Stricker and M. Orengo, "Similarity of color images," *SPIE Storage Retrieval Image Video Databases III*, vol. 2185, pp. 381–392, Feb. 1995.

[22] M. J. Swain and B. H. Ballard, "Color indexing," *Int. J. Comput. Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[23] H. Tamura, S. Mori, and T. Yamawaki, "Texture features corresponding to visual perception," *IEEE Trans. Syst., Man, Cybern.*, vol. 8, 1978.

[24] K. Tieu and P. Viola, "Boosting image retrieval," *Proc. Comput. Vision Pattern Recognition*, vol. 1, pp. 228–235, 2000.

[25] A. Vailaya, M. A. T. Figueiredo, A. K. Jain, and H. J. Zhang, "Image classification for content-based indexing," *IEEE Trans. Image Processing*, vol. 10, pp. 117–130, 2001.

[26] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[27] J. Weston, "Feature selection for SVM's," in *Advances in Neural Information Processing Systems*, 2000, vol. 13.

[28] G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*. New York: Wiley, 1982.

Guo-Dong Guo received the B.E. degree in automation from Tsinghua University, P. R. China, in 1991 and the Ph.D. degree in pattern recognition and intelligent control from the Institute of Automation, Chinese Academy of Sciences, P. R. China, in 1998.

He is currently a Research Assistant in Computer Sciences Department at the University of Wisconsin-Madison, Madison. He has been a Visiting Researcher at INRIA, Sophia Antipolis, France in 1997, Ritsumeikan University, Japan, in 1998, and Nanyang Technological University, Singapore, 2000-2001. He also worked at Microsoft Research, China, for about one year. His research interests include computer vision, machine learning, multimedia information retrieval, face recognition, and image analysis. He has published about 30 technical papers in these areas.

Anil K. Jain (S'70–M'72–SM'86–F'91) is a University Distinguished Professor in the Department of Computer Science and Engineering at Michigan State University. He was the Department Chair between 1995-99. He has made significant contributions and published a large number of papers on the following topics: statistical pattern recognition, exploratory pattern analysis, neural networks, Markov random fields, texture analysis, interpretation of range images, 3D object recognition, document image analysis and biometric authentication. Several of his papers have been reprinted in edited volumes on image processing and pattern recognition.

Dr. Jain received the best paper awards in 1987 and 1991, and received certificates from the Pattern Recognition Society for outstanding contributions in 1976, 1979, 1992, 1997, and 1998. He also received the 1996 IEEE TRANSACTIONS ON NEURAL NETWORKS OUTSTANDING PAPER AWARD. He is a Fellow of the International Association of Pattern Recognition (IAPR). He received a Fulbright Research Award in 1998 and a Guggenheim Fellowship in 2001.

Wei-Ying Ma (S'94–M'96) received the B.S. degree in electrical engineering from the national Tsing Hua University, Taiwan in 1990, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California at Santa Barbara, Santa Barbara, in 1994 and 1997, respectively.

He was with Hewlett-Packard Laboratories at Palo Alto, California, where he was a researcher in the Internet Mobile and Systems Lab. From 1994 to 199, he was with the Alexandria Digital Library (ADL) project in University of California at Santa Barbara while completing his Ph.D. In April 2001, he joined Microsoft Research Asia as the Research Manager of the Media Management Group. He has served on the organizing and program committees of many international conferences and has published four book chapters. His research interests include image and video analysis, content based image and video search and retrieval, machine learning techniques, intelligent information systems, adaptive content delivery, content distribution and services networks, and media delivery and caching.

Dr. Ma serves as an Associate Editor for the *Journal of Multimedia Tools and Applications*.

Hong-Jiang Zhang (S'90–M'91–SM'97) received the BS from Zhengzhou University, China, and the Ph.D degree from the Technical University of Denmark, both in electrical engineering, in 1982 and 1991, respectively.

From 1992 to 1995, he was with the Institute of Systems Science, National University of Singapore, where he led several projects in video and image content analysis and retrieval and computer vision. He also worked at MIT Media Lab in 1994 as a Visiting Researcher. From 1995 to 1999, he was a Research Manager at Hewlett-Packard Labs, where he was responsible for research and technology transfers in the areas of multimedia management; intelligent image processing and Internet media. In 1999, he joined Microsoft Research Asia, where he is currently a Senior Researcher and Assistant Managing Director in charge of media computing and information processing research. He has authored three books, more than 170 referreed papers and book chapters, seven special issues of international journals on multimedia processing, content-based media retrieval, and Internet media. He has numerous patents or pending applications.

Dr. Zhang is a Member of ACM. He currently serves on the editorial boards of five professional journals and a dozen committees of international conferences.