Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

ELSEVIER

Image and Vision Computing 23 (2005) 1170–1180

image AND vision COMPUTING

www.elsevier.com/locate/imavis

A histogram-based approach for object-based query-by-shape-and-color in image and video databases

Ediz Şaykol*, Uĝur Güdükbay, Özgür Ulusoy

Department of Computer Engineering, Bilkent University 06800 Bilkent, Ankara, Turkey

Received 19 April 2004; received in revised form 22 July 2005; accepted 26 July 2005

Abstract

Considering the fact that querying by low-level object features is essential in image and video data, an efficient approach for querying and retrieval by shape and color is proposed. The approach employs three specialized histograms, (i.e. distance, angle, and color histograms) to store feature-based information that is extracted from objects. The objects can be extracted from images or video frames. The proposed histogram-based approach is used as a component in the query-by-feature subsystem of a video database management system. The color and shape information is handled together to enrich the querying capabilities for content-based retrieval. The evaluation of the retrieval effectiveness and the robustness of the proposed approach is presented via performance experiments.

Keywords: Histogram-based approach; Object-based querying; Query-by-shape; Query-by-color; Distance histogram; Angle histogram; Distance-weighted color histogram

1. Introduction

Advances in multimedia technology accelerate the amount of digitized information so as the data stored as image and video content. Both of these data types require application-dependent processing strategies, easy-to-handle storage and indexing methods as well as sophisticated querying mechanisms. Mainly, in the querying process of image and video databases, spatial (for both image and video data) and spatio-temporal (for video data only) information are considered. Besides, the semantic information captured in images and/or videos can be queried through database management systems. Not only the images and videos, but also the objects in these data types can be enrolled in the queries with the corresponding object features (e.g. color, shape, texture, motion). Specific tools are developed in order to extract the object features from images and/or videos. These tools may require user assistance for the object extraction process in order not to be domain-specific.

As far as the querying process is concerned in multimedia databases, the content of the image and/or video described by a set of feature descriptors to represent the visual content is taken into consideration. This type of querying and retrieval is known as content-based retrieval (CBR). Most of the CBR systems include color as the major querying feature. Color histograms [1,2] and color sets [3,4] are some of the methods that are used within these systems. CBR systems are intended to query the color content of image and video data as a whole. Region-based color queries and the queries specifying spatial layout of the color regions are supported in some of these systems [5,6]. Shape is another feature that is used for querying the content of images and videos. Various pattern-matching methods (e.g. moments [7,8], Fourier descriptors [9], generic Fourier descriptors [10], grid descriptors [11], turning angles [12], Haussdorf distance [13]) are used to handle shape queries and CBR systems generally deal with images as a whole. For object-based shape queries, edge-detection can be applied to the image and then the polygonal object boundaries can be compared.

In this paper, we propose a histogram-based approach (HBA) for querying image and video data according to the shape and color content. The approach is implemented as a part of the query-by-feature component of the BilVideo

* Corresponding author. Tel.: +90 3122901776; fax: +90 3122664047
E-mail address: ediz@cs.bilkent.edu.tr (E. Şaykol).

0262-8856/$ - see front matter © 2005 Elsevier Ltd All rights reserved.
doi:10.1016/j.imavis.2005.07.015

video database management system [14–16]. The BilVideo system provides full support for spatio-temporal queries that contain any combination of spatial, temporal, object-appearance, external-predicate, trajectory-projection and similarity-based object-trajectory conditions by a rule-based system built on a knowledge-base, while utilizing an object-relational database to respond to semantic (keyword, event/activity and category-based), color, shape and texture video queries. The video data is pre-processed with our Fact Extractor tool that extracts the salient objects and the relations between these objects in video frames. The feature descriptors representing the shape and color content of the objects are extracted for each keyframe. Thus, HBA treats videos as a sequence of images so that each keyframe can be considered as a single image. Another tool, Object Extractor [17], processes the images and videos—through their keyframes—and extracts the object regions, as well as the corresponding object features. HBA provides a simple yet effective solution for querying by low-level object features (color, shape) without imposing complex data structures and indexing schemes.

In HBA, three histograms-distance, angle and color—are used to represent the shape and color content of the salient objects in image and video data. To the best of our knowledge, this is the first approach that employs these three histograms for object-based querying of image and video data by shape and color. The retrieval effectiveness of our histogram-based approach is shown through a comprehensive set of performance experiments. The impact and benefits of the proposed histogram-based approach can be summarized as follows:

1.1. Object-based query-by-shape

The shape content of the salient objects are represented by distance and angle histograms. HBA relies on the basic principles of the human visual system. Both the boundary and the interior regions of the objects are visualized by the human eye, and the regional distribution is captured as the shape content [27]. The data stored in these histograms are gathered with respect to the center of mass of objects, thus the distribution of the pixels in the boundary and the interior regions of the object is considered. The center of mass is selected as a pivot to model the distribution, and it is computed by taking each pixel as a unit mass. This approach is compared with both contour-based and region-based techniques, especially with the ones that have been shown to be effective in their domain, to demonstrate the retrieval effectiveness of these two histograms.

1.2. Object-based query-by-color

An improved version of color histograms is used for querying the salient objects with respect to their color content. The difference of HBA from the traditional color histograms is that it employs a probabilistic measure to weigh the colors of the pixels with respect to some neighborhood of pixels. The neighborhood can be either square or octagonal regions, which provides flexibility for assigning weights to pixels, thus the color histograms can be modified. While computing the weight of a pixel, the pixels in the neighborhood do not contribute equally. A distance-weighting scheme is embedded to the probabilistic measure. By the help of this probabilistic measure, the retrieval effectiveness of the color histograms is improved without imposing any computational complexity.

1.3. Object-based query-by-shape-and-color

An integration scheme is presented for using the distance, angle, and color histograms together to achieve better retrieval accuracy. The similarity can be obtained by using the linear combination of three partial similarity values corresponding to these histograms. A possible set of appropriate coefficients to use in the linear combination can be estimated by performing separate executions for each histogram. The average-case similarity values obtained from these three separate runs can be used as the partial similarity coefficients, which possibly reflects the characteristics of the dataset as much as possible. This integration scheme not only improves the retrieval accuracy but also helps reducing the impact of the object extraction phase.

The organization of the paper is as follows. Section 2 presents a summary of the existing methods for querying by shape and color contents. The histogram-based approach is discussed in Section 3. In Section 4, two approaches for integrating the shape and color features are described. Results of the performance experiments conducted to verify the retrieval effectiveness are presented in Section 5. Finally, Section 6 concludes the paper.

2. Related work

2.1. Query-by-shape approaches

Shape is an important feature that identifies objects in images. In order to retrieve objects according to their features, color is inadequate in most of the circumstances. Shape strengthens the image retrieval since, it encodes the object-based information in a way different from color. Thus, query-by-shape is supported by CBR systems that allow object feature-based queries by query-by-example (e.g. [18]) and query-by-visual sketch (e.g. [3,4]). Taxonomy of shape description techniques can be found in [19].

In QBIC system [1], area, circularity, eccentricity, major-axis direction, object moments, tangent angles and perimeter identify the objects in an image database. Specification of these parameters requires specific analysis during the database population phase. In [20] a method is proposed for image-based shape retrieval. This method requires edge-detection on the image, which is done by using the well-known Canny edge-detection algorithm [21]. The identified edges are stored in a histogram for an image. Chang et al. [3] explain a process for gathering object-based shape information. They use eigenvalue analysis for the generation of the first and second order moments. In Photobook [22], the method used for shape representation is based on eigenvectors of the object's physical model, that is constructed by the help of a finite element method.

The shape descriptors in the literature are basically in two forms: contour-based (e.g. Fourier descriptors [9] and Turning Angle representation [12]) and region-based (e.g. moment descriptors [7], generic Fourier descriptors [10], and grid descriptors [11]). The former type of descriptors deals with the object's boundary only and has some major limitations. The latter type of descriptors deals with both the interior and the boundary of the objects, thus can be employed in more general shape retrieval applications. In a recent work, generic Fourier descriptors [10] are shown to be a more effective shape descriptor than the other contour-based and region-based descriptors.

2.2. Query-by-color approaches

Color is the most frequently used feature in querying and retrieval of multimedia data. In most of the existing CBR systems, specification of a color query is generally query-by-example [23]. A sample query image is specified and the database images that are similar enough to the query image are retrieved. Not only the images as a whole but also the color regions that exist in the image can be queried. Since there may exist more than one region in an image, the spatial relations among the color regions can also be queried.

The color histograms [1,2,24] are used to represent the color distribution in an image or a video frame. Mainly, the color histogram approach counts the number of occurrences of each unique color on a sample image. Since, an image is composed of pixels and each pixel has a color, the color histogram of an image can be computed easily by visiting every pixel once. By examining the color histogram of an image, the colors existing on the image can be identified with the size of the areas containing the color as the number of pixels. This method is very simple, easy-to-handle and computationally cheap.

In [25], colorsets are proposed instead of color histograms. The colorsets are binary masks on color histograms and they store the presence of colors as 1 without considering their amounts. For the absent colors, the colorsets store 0 in the corresponding bins. The colorsets reduce the computational complexity of the distance between two images. Besides, region-based color queries are possible to some extent by employing colorsets. On the other hand, processing regions with more than two or three colors is quite complex.

Another image content storage and indexing mechanism is color correlograms [6]. This mechanism includes not only the spatial correlation of color regions but also the global distribution of local spatial correlation of colors. In fact, a color correlogram is a table each row of which is for a specific color pair of an image. The kth entry in a row for color pair (i, j) is the probability of finding a pixel of color j at a distance k from a pixel of color i. The method resolves the drawbacks of the pure local and pure global color indexing methods since it includes local spatial color information as well as the global distribution of color information.

In order to avoid the high computational cost of color correlograms, an approach is proposed [26], which improves the effectiveness of color histograms without the computational complexity of correlograms. The contributions of the pixels to the color histogram are weighted according to the Laplacian, probabilistic, and fuzzy measures. The weighting is related to a local measure of color non-uniformity (or color activity) determined by some neighborhood of pixels. This weighting approach, without changing the sizes of the histograms, is found to be more effective and its computational complexity is not excessive. The histogram-based approach we propose in this paper is more flexible than this approach since the neighborhood of pixels are restricted to squares here, whereas they can be both rectangular and octagonal in our histogram-based approach. Including octagonal neighborhoods does not increase computational complexity since by ignoring the corner pixels of a rectangular neighborhood, octagonal neighborhoods can be obtained. However, using octagonal neighborhoods seems more intuitive while considering the pixels far from a specific distance for a fixed pixel. Moreover, a distance-weighting scheme is employed in the histogram-based approach to increase the effect of closer pixels within the neighborhood while calculating the weights of the pixels.

3. Histogram-based approach for query-by-shape-and-color

Our histogram-based querying approach is basically motivated with the importance of the interrelation among pixels, thus objects are processed in such a way that each pixel provides a piece of information about both the shape and color content of the objects. The human vision system identifies objects with the edges they contain, both on the boundary and in the interior based on the intensity differentiations among pixels [27]. These intensity differentiations are captured with respect to the center of mass of these pixels. Hence, the histogram-based approach processes the shape and color content of the objects similar to the human vision system via considering the pixels in the interior and on the boundary of the objects with respect to the center of mass.

As shown in Fig. 1, the histogram-based querying process can be divided into three phases: object extraction, histogram construction, and querying. In the object

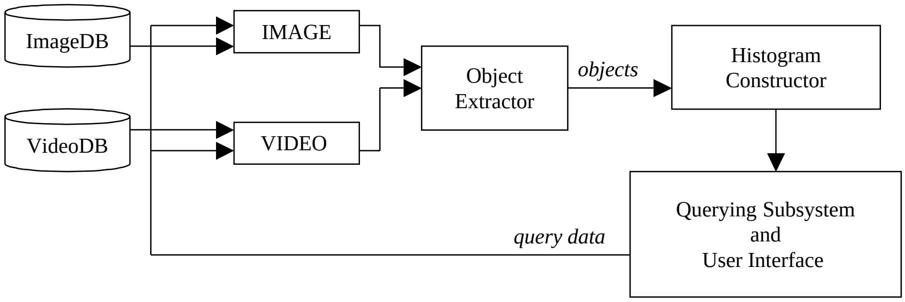

Fig. 1. The overall architecture of the query-by-feature component of BilVideo [14].

extraction phase, the color information is encoded in HSV color space by transforming RGB color vectors for each pixel to HSV. Then, color quantization and color median filtering are applied. The next step is the histogram construction phase where the color information of the filtered image is stored in one color histogram and the shape information is stored in two specialized histograms. The last step, the querying phase, involves an integrated use of the interface and the querying sub-system, in which the user specifies an example for the query. The users may also build up queries by sketching geometric primitives such as rectangles, circles, triangles, etc. The specified query is also passed through the object extraction and histogram construction phases. The results of the query are the 'most similar' objects to the given query object. The three histograms used for storing shape and color information can be described as follows:

Distance histogram stores the Euclidean distance between the center of mass of an object (c_m) and all of the pixels within the object. The distance between a pixel p and c_m is re-scaled with respect to the maximum distance d_max among the pixels (i.e. the distance of the farthest pair of pixels). The scaled distance accumulates into the corresponding bin in the distance histogram. In other words, instead of a constant scale for the objects, an object-specific scale metric is employed, which eases to satisfy the scale invariance by keeping the total number of bins in the distance histogram constant for all of the objects. We set the dimensionality λ of the distance histogram to 10 for storing pixel-wise distance information w.r.t. 10 levels.

Angle histogram stores the counter-clockwise angle between pixel vectors and the unit vector on the x-axis (e_x). The pixel vector v_p for a pixel p is a vector directed from c_m to p. The unit vector e_x is translated to c_m. As illustrated in Fig. 2, α is the counter-clockwise angle, polar angle, for p_i and increments the corresponding bin in the angle histogram. This type of information storage is novel since it provides an easy and intuitive way to capture angular distribution of the pixels around a fixed object point (c_m). We set the dimensionality λ of the angle histogram to 36 with a swapping angle of 10°.

Color histogram stores the three-dimensional vector $c_i = (h_i,\ s_i,\ v_i)$ in HSV color space for all of the pixels p_i belonging to an object. We have quantized the HSV color space as follows: a circular quantization of 20° steps sufficiently separates the hues such that the six primary colors are represented with three sub-divisions. Saturation and intensity are quantized to 3 levels leading to a proper perceptual tolerance along these dimensions. Hence, 18 hues, three saturations, three intensities, and four gray levels are en-coded yielding 166 unique colors, i.e. a color feature vector of $\lambda = 166$ dimensions (see [25] for details). To improve the effectiveness of the histogram, the contributions of the pixels are weighted according to a probabilistic measure in a way similar to the one proposed in [26]. A distance weighting-scheme is employed in the calculation of the pixel weights from the neighborhood, in which the effects of the closer pixels axe increased. The probability of the occurrence of a color within its some neighborhood, P, is a good estimate for the local color activity within some neighborhood. If the neighborhood of a color is uniform, then P is higher, and vice versa. In this approach, the contribution of uniform regions is decreased and that of singular points is increased. The computations are based on the following formula:

$$w_\mathrm{p} = \frac{w^c}{N_\mathrm{p}(c)}, \tag{1}$$

where w_p is the weight of pixel p, and $N_\mathrm{p}(c)$ is the number of pixels having color c as p within the neighborhood of p, w^c

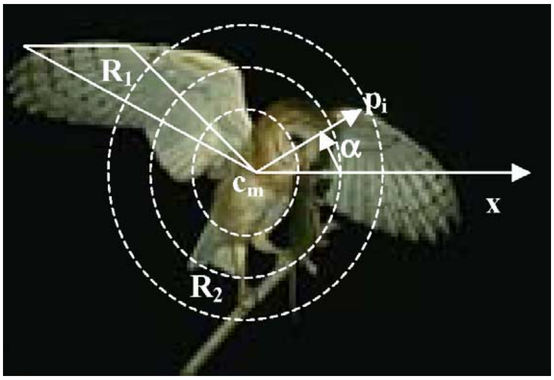

Fig. 2. Angle histogram calculations. α is the polar angle for object pixel p_i. R_1 region is an equivalence pixel class for angle histogram, and R_2 region is an equivalence pixel class for distance histogram.

is the weight assigned to the unique color c. The color weights can be set to 1, but to embed global color information to the formula, w^c can be computed as:

$$w^c = \frac{N_\mathrm{c}}{N_\mathrm{T}}, \tag{2}$$

where N_c is the number of pixels having color c, and N_T is the area of the object.

In HBA, the pixel weights axe also scaled with respect to the distances of the pixels within the neighborhood of a pixel. Rectangular or octagonal $n \times n$ neighborhood settings are valid for our approach. Having performed a reasonable amount of experiments, the neighborhood of the pixels is chosen as 5×5 octagonal windows, since the use of octagonal neighborhoods is more intuitive from the distance point of view.

The two histograms that store shape information are constructed with respect to the center of mass (c_m) of the object. The center of mass of an object is computed based on the assumption that the pixels have equal masses. The center of mass is a physical property of the objects in nature, and it is a geometric property of objects in images. Thus, it is favorable to choose c_m as a reference instead of other points. Obviously, we have more reasons for selecting c_m as a pivot as explained below:

Since the coordinates of c_m is computable in one pass on the pixels of the objects, the color and shape information is correctly referenced by c_m of the object. For an object O, $c_\mathrm{m} = (x_{c_\mathrm{m}}, y_{c_\mathrm{m}})$ computed as follows:

$$x_{c_\mathrm{m}} = \frac{\sum_i^n x_i}{n}, \qquad y_{c_\mathrm{m}} = \frac{\sum_i^n y_i}{n}, \tag{3}$$

where n is the number of pixels in O.

The upper bound on the number of bins of the distance histogram, d_u, is limited $0 \le d_\mathrm{u} < \sqrt{w^2 + h^2}$, where w and h are the width and height of the minimum bounding rectangle (MBR) enclosing the object. However, the upper bound on the size of the distance histogram becomes smaller when it is filled with respect to c_m. For most of the objects, c_m is not far from the center of the MBR and the upper bound of d_u becomes very close to $\sqrt{w^2 + h^2}/2$. As far as the storage of this histogram for the objects is concerned, c_m is a proper choice as a pivot.

The polar angle α_i for a pixel p_i is stored in the angle histogram and since object pixels surround c_m, the angular distribution of pixels within object can be gathered. The number of bins of the angle histogram is constant ($0 \ge \alpha_\mathrm{i} \ge 2\pi$) and with a proper quantization scheme on the angle histogram, this number becomes smaller.

Definition 1. (Equivalent Pixel Class) The set of pixels $S_{H[i]}$ forms an equivalent pixel class for an histogram bin $H[i]$ such that for each $p_i \in H[i]$, $p_i \in S_{H[i]}$.

Since, we have three histograms, a pixel p_i participates in three different equivalent pixel classes. The equivalent pixel classes for angle and distance histograms form contiguous regions. However, the equivalent pixel classes for color histograms contain the pixels painted with the same color in the object region. As illustrated in Fig. 2, the triangular region R_1 is one of the angle histogram equivalent pixel classes. Similarly, the donut-like region R_2 is an equivalent pixel class for the distance histogram. Fig. 3 shows the histogram content of an elephant object. The object is shown in Fig. 3(a) through its MBR, and the distance, angle, and color histograms are shown in Fig. 3(b)–(d), respectively.

The three histograms are scale, rotation and translation invariant. For distance histogram, re-scaling with respect to the maximum object distance (i.e. the distance of the farthest pair of pixels in the object), for angle histogram, dividing each bin by the total number of pixels, for color histogram, the weighted sum of the pixels provide scale invariance. The translation invariance of the histograms is quite obvious and the rotation invariance is an issue for the angle histogram only. For rotation normalization, first the major axis of the object is found by traversing the boundary of the object. The major axis is the line segment connecting the farthest pair of pixels [10]. Having found the major axis of the object, the angle α_MA between the major axis and the x-axis is computed (For the object in Fig. 2, $\alpha_\mathrm{MA} = 150°$). It can be observed that $0° \le \alpha_\mathrm{MA} \le 180°$. The object is then rotated clockwise so that the major axis becomes horizontal, i.e. the object is rotated $-\alpha_\mathrm{MA}$ degrees. Since the angle histogram is filled according to this rotated form of the object, the angle histogram is rotation invariant.

As an application, the two shape histograms are employed in a content-based retrieval system for historical Ottoman documents [28] to retrieve similar database documents to a query image. These two feature histograms, distance and angle, are extracted for a symbol in a document image corresponding to a page in a historical Ottoman document. The ottoman script is connected, and the symbols are extracted based on the similarities to the symbols in the symbol library. The similarities between the symbols are determined by comparing the feature histograms. Once the similar symbols are identified in document images, their locations are stored and the symbols are inserted to the symbol library.

4. Integrating object-based shape and color information

The histogram-based approach extracts color and shape information of images. Color and shape, as being the most frequent descriptors, are used in most of the existing systems in a combined way. The main reason for this integration is to improve the retrieval effectiveness of the systems. To satisfy this goal, the color and shape information can be linearly combined with proper weights. The integration in the histogram-based approach is quite different since, shape information is stored in two histograms.

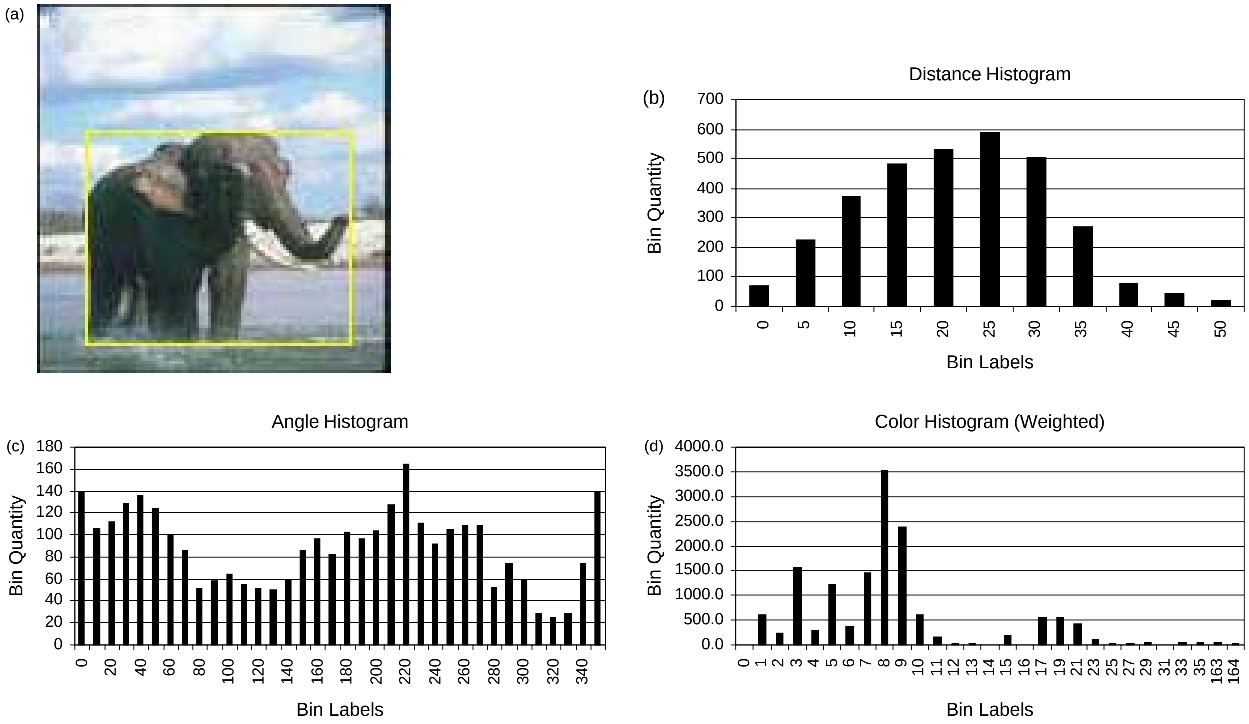

Fig. 3. (a) Elephant image where the elephant object is shown through its MBR; (b) Distance histogram of elephant object; (c) Angle histogram of elephant object (without major axis orientation); (d) Color histogram of elephant object.

For similarity calculations between database objects and query object, histogram intersection technique is employed. This technique is used by Swain and Ballard [2] for color feature, and now used by most of the existing content-based retrieval systems (e.g. [20]). Besides, there exist many histogram comparison techniques in the literature that axe employed for shape comparison. A wide range of these techniques is detailed in [29].

4.1. Histogram intersection

In the histogram intersection method, two normalized histograms are intersected as a whole to end up with a similarity value. Both of the histograms must be of the same size. The similarity between the histograms is a floating-point number between 0 and 1. Equivalence is designated with similarity value I and the similarity between two histograms decreases when the similarity value approaches to 0.

Let $H_1[1...n]$ and $H_2[1...n]$ denote two normalized histograms of size n, and S_{H_1,H_2} denote the similarity value between H_1 and H_2. Then, S_{H_1,H_2} can be expressed as:

$$S_{H_1,H_2} = \frac{\sum_{i}^{n} \min(H_1[i], H_2[i])}{\min(|H_1|, |H_2|)}, \tag{4}$$

where $|H|$ denotes the L_1-norm (i.e. length) of an histogram H.

4.2. Approaches for shape and color integration

In the histogram-based approach, since the object content to compare is composed of three histograms, the total similarity between the query object and a database object is determined as follows:

$$S_T = \frac{S_C \times w_C + S_D \times w_D + S_A \times w_A}{w_C + w_D + w_A}, \tag{5}$$

where S_C, S_D, and S_A denote similarities in color, distance, and shape histograms, respectively. In this formula, color, distance, and angle histograms are linearly combined with pre-specified weights w_C, w_D, and w_A. Based on this discussion, two approaches for integrating shape and color can be summarized as follows:

(i) *Assigning equal weights to color, distance, and angle histograms*. This approach is simple and fast but not suitable for most of the datasets. Treating color and shape features equally may lead to loss of information in some domains.

(ii) *Integrating features via assigning different weights*. A global distance can be obtained by linear combination of three partial distances with appropriate weights [20]. A possible set of weights can be determined by performing similarity calculations for each histogram separately. The average-case retrieval accuracies as a result of these three similarity calculations are found for the three histograms. Then, the results are normalized and used as weights. This pre-computed weight assignment provides more effective results for similarity queries since, this approach reflects the characteristics for the datasets more than the first one.

5. Performance experiments

5.1. Evaluating retrieval effectiveness

In the first set of performance experiments, the histogram-based approach (HBA) is evaluated in terms of retrieval effectiveness. Our testbed has three sets: The first one is a set of 200 images from various domains including nature photographs, portraits, animal figures, etc. are first passed through Object Extractor. For the experiments, 5×5 octagonal color median filters are applied three times to have appropriate filtered images. 267 objects are extracted from the images, and the database is expanded further to test the robustness of the HBA method for rotation and scale invariance. Each object is stored with its four scaled version as well as five rotated versions (rotation by 60 degrees). The second set is holds 1490 Corel images [30]. These images generally contain patterns from various domains. We have extracted one object from each image in the set. The third set is constructed to demonstrate the use of HBA with video frames. Sixteen video clips from various TV channels are processed and some salient objects axe extracted from the video frames. The extraction is performed by treating the video frames as images. The details of the video clips are presented in Table 1. The frame size is 384×288 for all of the video clips. Fig. 4 shows nine database images, three from each set.

Having extracted the objects, the shape and color features are integrated for retrieval in two different ways as mentioned before. First, the histograms are treated as equally-weighted. Second, the average-case retrieval accuracy of individual histograms, distance, angle, and color, is assigned to each of these histograms as weights. For the image domain that we experimented on, three separate executions are performed to determine the weights of the histograms. The average-case retrieval accuracies at the end of these three executions are found as follows: for color histogram, 0.66, for distance histogram, 0.82, and for angle histogram, 0.72. By normalizing these values, the weights are found to be 0.3, 0.37 and 0.33 for the color, distance and angle histograms, respectively.

Table 1
The detailed information on the video clips used in the third set of the experiment testbed

TV Channel	Domain	Duration
BBC Prime	Documentary	0.26.0
BBC World	News	2.26.16
CNBCE	Film	2.11.00
CNN	News	1.48.83
EuroSport	Sport	2.17.41
HistoryChannel	Documentary	0.53.50
NationalGeographic	Documentary	1.49.41
Number One	Music clip	1.06.25
NTV	News	2.28.16
Star	Film	0.50.33
TRT1	Film	1.51.08
TRT2	News	2.08.16
TRT2	Documentary	1.05.50
TRT3	Sport	2.41.16
TV5	Film	1.38.25
TV8	Music clip	3.38.00

In order to evaluate effectiveness of retrieval systems, two well-known metrics, precision and recall, are used. Precision is the ratio of the number of retrieved images that are relevant to the number of retrieved images. Recall is the ratio of the number of retrieved images that are relevant to the total number of relevant images [31]. In the experiments, 100 query objects are randomly picked from the extracted objects. The effectiveness is evaluated as the average of the results calculated for each query separately. As mentioned in most of the information retrieval systems, the individual precision values are interpolated to a set of 11 standard recall levels (0, 0.1, 0.2, …, 1) to facilitate the computation of average of precision and recall values. Three relevance degrees (0, 0.5, 1) are used to signify the relevance of an object to the query object for each query. Since objects are from various domains, assigning partial relevance to the extracted objects is meaningful. These relevance values are assigned to each query object-database object pair.

Another experiment is conducted for evaluating the behavior of HBA to noisy objects. The noisy forms of each object are added to the object database. These noisy forms are constructed by simply changing the locations of k% of the pixels in a test image randomly. For the experiment, k is set to 7. It has been observed that HBA retrieves the noisy forms of an object before any other object, which is expected and desired. Moreover, the similarity values between objects and their noisy forms vary between 0.92 and 0.98. The retrieval effectiveness results of HBA are shown in Fig. 5. In these experiments, the first 20 retrievals are evaluated for randomly picked query objects to be fair.

5.2. Evaluating color effectiveness

In this section, three different color information storage techniques are compared in terms of retrieval effectiveness. As discussed before, color histograms are the first attempt for color retrieval. In order to improve the effectiveness of color histograms, various techniques and adaptations are proposed. Here, the comparisons are carried out between three methods: the traditional color histograms without any weighting, probabilistic weighting scheme [26], and the distance-weighting scheme in the histogram-based approach. The evaluation of color retrieval effectiveness is performed similarly by precision-recall analysis. The results of the first 20 retrievals are presented in Fig. 6. The use of

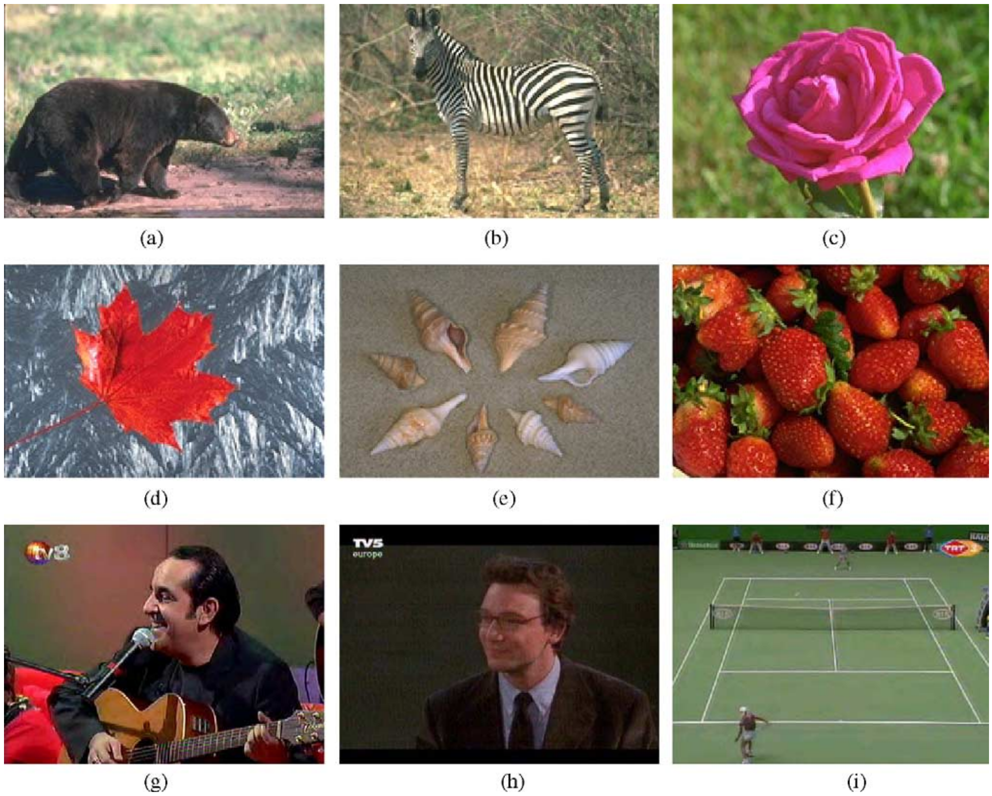

Fig. 4. (a), (b), and (c) are examples for the first set; (d), (e), and (f) are for the second set; and (g), (h), and (i) are for the third set.

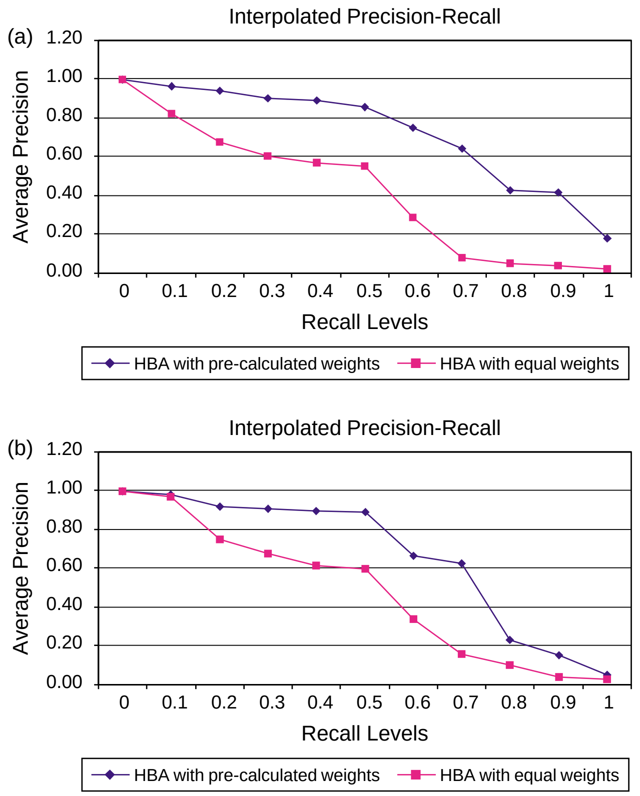

Fig. 5. Retrieval effectiveness results, (a) without noise; (b) with noise when $k=7$. The pre-calculated weights are 0.3, 0.37 and 0.33 for the color, distance and angle histograms, respectively.

distance-weighted color histograms for color information is found to be more effective than the other methods.

5.3. Evaluating shape effectiveness

5.3.1. Comparison with turning angle representation

In this section, HBA is compared with the well-known shape matching method, the Turning Angle (TA) method [12]. The reason of choosing the TA method is the fact that it behaves better in most of the cases than the other methods (e.g. moments [8], Hausdorrf distance [13], sign of curvature [32]) due to the human perceptual judgments, as shown in [32]. In order to compare TA and HBA, the output of Object Extractor should be converted to an input to TA.

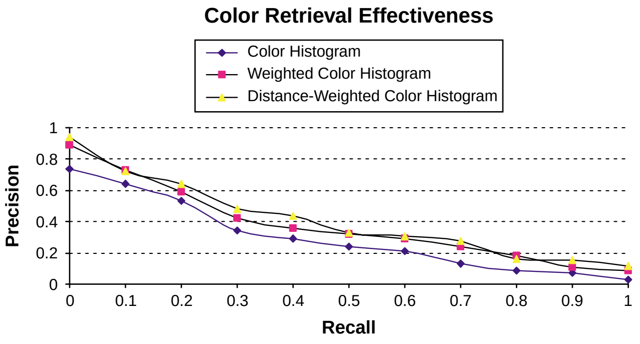

Fig. 6. Comparison of color retrieval effectiveness.

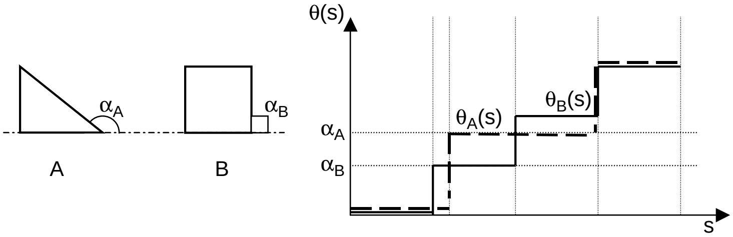

Fig. 7. Turning angle representation.

Thus, the filtering steps applied before the histogram-based approach also take place before the TA method. Besides, a model for converting the output of Object Extractor to an input to TA, called KiMPA [33], is designed. In KiMPA, the 'most important' points of the object boundaries are preserved. In this manner, the experiments compare the approaches accurately and fairly.

TA is a contour-based method, which processes two polygonal shapes and produces turning angle representations of the shapes. Then, L_2 distance metric is employed to get a similarity distance from the representations. Formally, the turning angle representation $\Theta_P(s)$ of a polygonal shape P is a collection of turning angles as a function of the arc length s. A turning angle at a point is the counter-clockwise angle between the tangent of P at that point and the x-axis [12]. Fig. 7. illustrates the turning angle representation of two shapes initiated from the bottom-left vertex. In order to satisfy invariance under rotation, $\Theta_P(s)$ can be shifted with an angular amount θ. The arc length s is scaled to 1 for all of the input shapes, which satisfies scale invariance. The method is also invariant under translation.

Let $\Theta_A(s)$ and $\Theta_B(s)$ be turning angle representations for polygonal shapes A and B, respectively. According to the L_p metric, the distance between two polygonal shapes is given as follows:

$$d_T(A,B) = \left(\int |\Theta_A(s) - \Theta_B(s)|^p \, ds \right)^{1/p}. \quad (6)$$

In the experiments, the main point is to demonstrate the effectiveness of HBA with the objects having noise on the boundaries. Retrieving similar objects even if their boundaries are slightly distorted is important for shape retrieval. Table 2 presents the average retrieval accuracy of single object queries such that each query object (in original version) is queried with all of the objects in the noisy version. The noisy versions of the test objects are created by performing changes on the boundary pixels only. The weight of the color histogram is set to 0 in order to be fair. The two shape histograms are equally weighted for the experiment. The experiments show that HBA yields promising results such that the embedded noise on the object boundaries can distort the similarity at most 2.7%.

Table 2
Average retrieval accuracy of the single object queries

	$N=1$	$N<4$	$N<6$	$N<10$	$N<20$
HBA	1.00	1.00	1.00	1.00	1.00
TA	0.86	0.95	1.00	1.00	1.00

HBA is an abbreviation of the histogram-based approach, N denotes the retrieval rank of the noisy version of the query object.

5.3.2. Comparison with generic fourier descriptors

In order to test the retrieval effectiveness of HBA as compared to Generic Fourier Descriptors (GFD), another set of experiments is conducted. GFD is derived by applying a modified polar Fourier transform on the shape of an object [10,34]. The polar image of the shape of an object is a rectangular image, thus 2D Fourier transform can be applied on the polar image. For a given shape image $f(x, y)$, the modified polar Fourier transform is obtained as

$$pf(\rho,\phi) = \sum_r \sum_i f(r,\theta_i) \exp\left[j2\pi \left(\frac{r}{R}\rho + \frac{2\pi}{T}\phi \right) \right], \quad (7)$$

where $0 \le r = [(x-x_c)^2 + (y-y_c)^2]^{1/2}$ and $\theta_i = \frac{2\pi i}{T}$, $(0 \le i \le T)$; $0 \le \rho < R$, $0 \le \phi < T$, (x_c, y_c) is the center of mass of the shape; R and T are the radial and angular resolutions. GFD is translation invariant. For rotation normalization, an optimized version of the major axis normalization algorithm is used. To achieve scale invariance, the following normalization is employed:

$$\text{GFD} = \left\{ \frac{|pf(0,0)|}{\text{area}}, \frac{|pf(0,1)|}{|pf(0,0)|}, \ldots, \frac{|pf(m,0)|}{|pf(0,0)|}, \ldots, \frac{|pf(m,n)|}{|pf(0,0)|} \right\}, \quad (8)$$

where area is the area of the bounding circle in which the polar image of shape resides. m is the maximum number of the radial frequencies, and n is the maximum number of angular frequencies. In the experiments, five radial frequencies and 12 angular frequencies are selected to obtain 60 GFD features. The following datasets are used:

MPEG-7 region shape database (CE-2) is used to test the robustness of the methods including perspective transformation, scaling, and rotation invariance. The dataset contains 3621 shapes, 651 of which are organized into 31 categories. Each category consists of 21 similar shapes generated by rotating, scaling, and perspectively transforming the shapes.

This dataset is also used in [10,34] to test the retrieval effectiveness of the GFDs.

In [35], 216 shapes are selected from MPEG-7 set. The shape database can be grouped into 18 categories, each of which has 12 similar shapes.

The shapes are extracted automatically from images, since the images are binary. The retrieval is also performed automatically since the categorization (i.e. relevance) is known in advance. The experiment is performed for randomly picked 100 query objects, and the results are averaged. Fig. 8 presents the retrieval effectiveness results of this set of experiment. The retrieval analysis is performed for the first 20 retrievals and the weights are 0.0, 0.5, and 0.5 for color, distance, and angle histograms, respectively in order to be fair since GFD operates on the shape content only. The results in (a) demonstrate that the performance of GFD is generally better than HBA. The main reason behind this is the fact that the dataset in (a) has perspectively transformed objects. However in (b), the performance of HBA is better for most of the recall levels where there are not too many perspectively transformed objects.

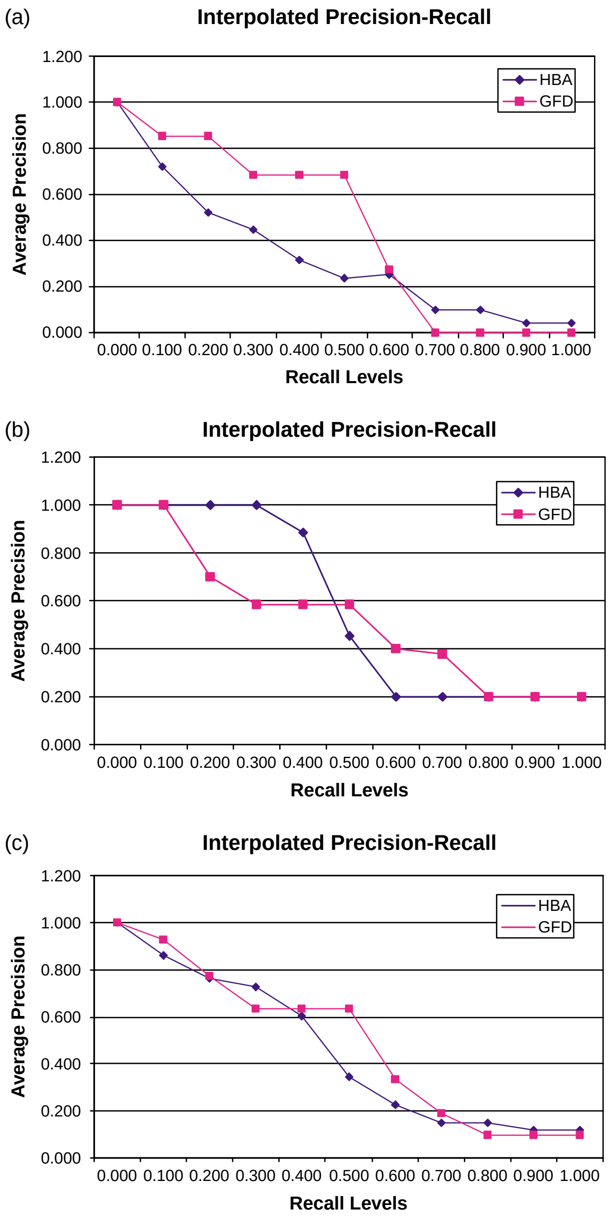

Fig. 8. Retrieval effectiveness results, (a) with the dataset used in [10,34]; (b) with the dataset used in [35]; and (c) with the two datasets mixed.

The average extraction time for GFD is 95.6 ms, whereas that of HBA is 11.6 ms on a PC of dual P-III 800 processors with 1GB RAM. HBA is 8.24 times faster than GFD in feature extraction. Euclidean distance is used to obtain the similarity of two shapes in generic Fourier descriptors, whereas histogram intersection is employed in HBA. The feature extraction phase of HBA is much simpler and faster than GFD, while its retrieval effectiveness is comparable with GFD.

6. Conclusion

In this paper, a novel approach is proposed for object-based querying by shape and color features. The approach is histogram-based, in which one color histogram and two shape histograms are stored for an object. The main idea of employing histogram-based approach is to include the contributions of each pixel in the object region to the color and shape information of the object. The histogram-based approach is invariant under rotation, translation and scaling.

The main principles of the human visual system are taken into account while constructing the feature histograms of the objects. The contributions of the pixels to the color histogram are weighted with respect to the color activity estimated within a local neighborhood. The pixels also contribute to shape feature histograms based on their distance and angle with respect to the center of mass of the object, which considers the object interior as well as the object boundary.

The precision-recall analysis is performed to evaluate the retrieval effectiveness of the histogram-based approach. This approach generally outperforms the contour-based methods, and is comparable to region-based methods. Moreover, the feature extraction and similarity computation steps are easier and faster.

The object-based information can be queried without any pre-processing phase such as edge-detection since the output of an extraction tool, Object Extractor, is directly used. The output contains the necessary data to construct the histograms for the object. The approach is also scalable in the sense that the images having more than one object can be processed without any difficulty.

Acknowledgements

This work was supported by Scientific and Technical Research Council of Turkey (TÜBİTAK) under the grant number EEEAG-199E025, and is supported in part by Turkish State Planning Organization (DPT) under grant number 2004K120720, and European Commission 6th Framework Program MUSCLE Network of Excellence Project with grant number FP6-507752.

References

[1] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petrovic, D. Steele, P. Yanker, Query by image and video content: the QBIC system, IEEE Computer Magazine 28 (9) (1995) 23–32.

[2] M. Swain, D. Ballard, Color indexing, International Journal of Computer Vision 7 (1) (1991) 11–32.

[3] S. Chang, W. Chen, H. Meng, H. Sundaram, D. Zhong, VideoQ: an automated content based video search system using visual cues, in: ACM Multimedia'97 Conference Proceedings, Seattle, Washington, USA, 1997, pp. 313–324.

[4] J. Smith, S. Chang, VisualSEEk: A fully automated content-based image query system, in: ACM Multimedia'96 Conference Proceedings, Boston, MA, USA, 1996, pp. 87–98.

[5] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, J. Malik, in: D. Huijsmans, A. Smeulders (Eds.), Blobworld: A System for Region-Based Image Indexing and Retrieval Proceedings of Third International Conference on Visual Information Systems, VISUAL'99 vol. 1614 of LNCS, Springer, New York, 1999, pp. 509–516.

[6] J. Huang, S. Kumar, M. Mitra, W. Zhu, R. Zabih, Image indexing using color correlograms, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, 1997, pp. 762–768.

[7] H. Kim, Region-based shape descriptor invariant to rotation, scale and translation, Signal Processing: Image Communication 16 (1,2) (2000) 87–93.

[8] G. Taubin, D. Cooper, in: J. Mundy, A. Zisserman (Eds.), Object Recognition Based on Moment (or Algebraic) Invariants Geometric Invariance in Computer Vision, MIT Press, Cambridge, MA, 1992, pp. 375–397.

[9] C. Zahn, R. Roskies, Fourier descriptors for plane closed curves, IEEE Tfansactions on Computer C-21 (3) (1972) 269–281.

[10] D. Zhang, G. Lu, Shape based image retrieval using generic fourier descriptors, Signal Processing: Image Communication 17 (10) (2002) 825–848.

[11] G. Lu, A. Sajjanhar, Region-based shape representation and similarity measure suitable for content-based image retrieval, Multimedia Systems 7 (2) (1999) 165–174.

[12] E. Arkin, P. Chew, D. Huttenlocher, K. Kedem, J. Mitchel, An efficiently computable metric for comparing polygonal shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (3) (1991) 209–215.

[13] D. Huttenlocher, G. Klanderman, W. Rucklidge, Comparing images using the Haussdorff distance, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (9) (1993) 850–863.

[14] M. Dönderler, E. Şaykol, Ö. Ulusoy, U. Güdükbay, BilVideo: a video database management system, IEEE Multimedia 10 (1) (2003) 66–70.

[15] M. Dönderler, E. Şaykol, U. Arslan, Ö. Ulusoy, U. Güdükbay, BilVideo: Design and implementation of a video database management system, Multimedia Tools and Applications 27(1) (2005) 79–104.

[16] M. Dönderler, Ö. Ulusoy, U. Güdükbay, A rule-based video database system architecture, Information Sciences 143 (1–4) (2002) 13–45.

[17] E. Şaykol, U. Güdükbay, Ö. Ulusoy, in: S. Adali, S. Tripathi (Eds.), A Semi-Automatic Object Extraction Tool for Querying in Multimedia Databases Proceedings of Multimedia Information Systems, MIS'01, Capri, Italy, 2001, pp. 11–20.

[18] B. Günsel, A. Tekalp, Shape similarity matching for query-by-example, Pattern Recognition 31 (7) (1998) 931–944.

[19] M. Safar, C. Shahabi, X. Sun, Image retrieval by shape: a comperative study, in: Proceedings of IEEE International Conference on Multimedia and Expo, New York, NY, USA, 2000, pp. 141–144.

[20] A. Jain, A. Vailaya, Image retrieval using color and shape, Pattern Recognition 29 (8) (1996) 1233–1244.

[21] J. Canny, A computational approach to edge-detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 8 (6) (1986) 679–698.

[22] A. Pentland, R. Picard, S. Scarloff, Photobook: content-based manipulation of image databases, in: Proceedings of Storage and Retrieval for Image and Video Databases II, SPIE, number 2185, Vol. 2, 1994, pp. 34–47.

[23] M. Zloof, Query-by-example: a database language, IBM System Journal 16 (4).

[24] J. Hafner, H. Sawhney, W. Equitz, M. Flickner, W. Niblack, Efficient color histogram indexing for quadratic form distance functions, IEEE Transactions on Pattern Analysis and Machine Intelligence 17 (7) (1995) 729–736.

[25] J. Smith, S. Chang, in: I. Sethi, R. Jain (Eds.), Tools and Techniques for Color Image Retrieval Proceedings of Storage and Retrieval for Image and Video Databases IV, IS&T/SPIE vol. 2670 (1996), pp. 426–437.

[26] N. Boujemaa, C. Vertan, Integrated color texture signature for image retrieval, in: Proceedings of International Conference on Image and Signal Processing, Agadir, Morocco, 2001, pp. 404–411.

[27] P. Buser, M. Imbert, Vision, MIT Press, Cambridge, Massachusetts, 1992.

[28] E. Şaykol, A. Sinop, U. Güdükbay, Ö. Ulusoy, A. Çetin, Content-based retrieval of historical Ottoman documents stored as textual images, IEEE Transcations on Image Processing 13 (3) (2004) 314–325.

[29] R. Veltkamp, Shape matching: Similarity measures and algorithms, Tech. Rep. UU-CS-2001-03, Utrecht University (January 2001).

[30] Corel, Image Library, University of California, Berkeley, http://elib.cs.berkeley.edu/photos/corel/.

[31] K. Jones, Information Retrieval Experiment, Butterworth and Co., London, 1981.

[32] B. Scassellati, S. Alexopoulos, M. Flickner, Retrieving images by 2D shape: a comparison of computation methods with human perceptual judgments, in: Proceedings of Storage and Retrieval for Image and Video Databases II, SPIE, number 2185, Vol. 2, 1994, pp. 2–14.

[33] E. Şaykol, G. Güleşir, U. Güdükbay, Ö. Ulusoy, in: T. Yakhno (Ed.), KiMPA: A kinematics-based method for polygon approximation International Conference on Advances in Information Systems (ADVIS'02) vol. 2457 of LNCS, İzmir, Turkey, 2002, pp. 186–194.

[34] D. Zhang, G. Lu, Enhanced generic fourier descriptors for object-based image retrieval, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, Orlando, Florida, USA, 2002, pp. 3668–3671.

[35] T. Sebastian, P. Klein, B. Kimia, Recognition of shapes by editing shock graphs, in: IEEE International Conference on Computer Vision, Vancouver, Canada, 2001, pp. 755–762.