Image Management Using the XML Techniques

Shyi-Chyi Cheng* and Zong-Han Lee
Department of Computer and Communication Engineering, National Kaohsiung First University of Science and Technology, 1 University Road, Yenchao, Kaohsiung 824, Taiwan, ROC.
e-mail: csc@ccms.nkfust.edu.tw

Abstract

As the advances of the Internet technologies, more and more books and multimedia documents have been digitized and shared over the Internet. To design an effective and efficient information retrieval scheme, and hence, becomes an important issue for the purpose of querying multimedia data through the Internet.

In this paper, we proposed a hybrid feature measure which combines the spatial structure and color features of an image based on a preprocessed 2DC-string technique for recognizing rotation invariants. Based on the segmented objects in an image, a hierarchical image model consisting of spatial relationships among objects and low-level features associated to objects is proposed. The proposed system is described and managed in XML, which has been proved to be useful in interchanging text information. We use it to achieve the goal of interchanging multimedia data.

An image database consisting of 110 flag images is constructed to verify the performance of the system. Test results of the proposed scheme for retrieval of images using a set of rotated and distorted query images are presented. The efficiency of retrieval is found to be very high and the experimental results are promising for practical applications.

Keywords: Image database, XML, Object, Content-based image retrieval.

1. Introduction

As the advances of the Internet, the demand of storing multimedia information (such as text, image, audio, and video) has increased. With such databases comes the need for a richer set of search facilities that include keywords, sounds, examples, shape, color, texture, spatial structure and motion. With an increasing amount of image data being accumulated, there is a strong demand on coherently organizing the information of an image that consists of color, shape, texture, and spatial features. Many content-based image retrieval systems have been proposed in the literature [1-4]. Faloutsos et al. [1] described the details of the QBIC system especially the querying by image content, wherein they support queries based on image features such as color, shape, texture, and sketch. However, they use these individual features separately and do not use it for combined retrieval. The technique for content-based similarity retrieval of images using multiple kinds of features in an integrated manner is another choice [4]. Besides, the types and extraction methods of features derived from multimedia content vary from one system to another, and thus, limit the exchangeability of the indexes of multimedia data among databases.

Common multimedia indexing models are constructed in attributes, which are valued by a list of meaningful features. For instance, color images can be described in global and local color histograms, the mean and higher order moments. The objects of a media can be defined and referred to at different abstraction levels [5], as described below:

(1) Raw data: Generally speaking, the cost to index a media in this abstraction level is high. One can compare two images pixel-by-pixel and aggregate the differences together as the similarity measure.
(2) Feature: To characterize a media by a set of features, which is extracted from raw data. Instead of raw data of media, the features are used to index/retrieve the object. Similarity search in the feature space thus consisting of comparing the target features with the features stored in the database.
(3) Object: Objects (or regions) of an image can be identified via the features extracted from the object content. The spatial relationships among objects are useful in filtering out dissimilar images from databases.
(4) Semantic: This is the highest abstraction level to index a media. Semantic information usually extracted from a learning process or supplied through human interpretations represents an interpretation (a concept) to the media. Different applications might have

* To whom all correspondences should be sent.

different points of views to the same object. That is, the semantic information is actually dependent on the domain knowledge of applications.

Item 1 above is characterized as the physical databases, on the other hand, Items 2 to 4 are characterized as the logical databases. For most content-based image database systems, image matching and retrieval is based on some characteristic features of an image class under consideration. The image database consists of plenty of images stored along with their characteristic features. A query image is analyzed to extract the characteristic features, which are matched against the features of the images in the database for similarity retrieval. Since exact match is usually not the goal, a set of closely related matching images are brought out as the results of search output. The user is expected to iteratively refine the query result using relevant feedback if this output is not satisfactory. Usually, the querying process requires a huge amount of computation in matching the characteristic features which are represented as high-dimensional vectors. This limits the content-based image databases for practical applications.

In this paper, we propose a hierarchical image model which essential elements are objects of images. In this model, the spatial relationships represented as 2DC strings [6-10] among objects in a query image are used as the filter to skip dissimilar images from databases. The information associated to objects including spatial composition and color features is then used to rank those match images in the previous step according to a similarity measure. Instead of doing a complex feature measure, a general string matching method can be used as the filter. And hence, the performance of the proposed scheme is much better than that of past content-based image database systems. High performance is absolutely essential to retrieve large image databases.

Basically, the parsed structure of an XML document is a tree and searchable by the XSL pattern technique. Given a meaningful tag to each element of the proposed image model, the image content can be simply represented in the XML structure. Images are then interpretable by software and thus can be automatically stored, searched, and manipulated from databases.

The XML techniques have been proved to be useful in interchanging text information over the Internet. Many projects using XML techniques for different purposes of applications including web site management [11], electronic commerce [12,13], heterogeneous database systems [14], and information classification [15] are on going. We use it to achieve the goal of interchanging multimedia data. The ISO (International Standardization Organization) has begun to clarify the scope and objectives of a potential multimedia content description interface standard, known as MPEG-7. The standard is intended to facilitate the future development of audiovisual content-based search engines [18]. The research results of the paper show that the XML technique is a possible choice as the basis of the MPEG-7 standard.

The paper is organized as follows: object-based image representation and retrieval is introduced in Session 2. The proposed image model and the structure of self-defined XML tags are defined in Session 3. Session 4 contains the implementation of the XML-based image retrieval system followed by test results in Session 5. Brief conclusions are given in Session 6.

2. Object-Based Image Representation And Retrieval

For the purpose of matching and similarity computation, an image I can be separated into several objects. Each object O in an image or a query can be characterized by a feature vector f_O, which captures different aspects of image content such as color information, texture features, shape information of objects in the image and any other content which may be of interest for the purpose of querying. The spatial structure S_I of the image I can be described by the spatial relationships among objects in I. Thus the image I assumed to consist of a set of n objects is represented as:

$$F_I = \{(O_i, f_O) \mid 1 \le i \le n\}, \text{ and}$$

$$S_I = \{(O_i, O_j, r_{ij}) \mid 1 \le i \le n, 1 \le j \le n\}$$

where r_{ij} is the spatial relationship between objects O_i and O_j. The feature vector f_O is defined as:

$$f_O = (i_1, i_1, \ldots, i_m)$$

where m is the number of content features. The query of image is of more general applicability by derived attributes involving some degree of logical inference about the identity of the objects segmented from the image.

Image objects can be represented in various forms. One of them is to store significant points such as the centroid or the corners of the minimum bounding rectangle (MBR) which is the minimum size of rectangle that completely enclose a given object. The MBR is storage efficient and useful in dealing with image objects that are arbitrarily complex in terms of their boundary shapes. MBR representation is efficient in constructing the spatial relationships such as whether or not two objects are overlapped or whether one object is completely contained in another one among objects in an image. On the other hand, using MBR as an object representation may lead to inaccuracies in the representation and spatial reasoning of other topological relationships. Figure 1 shows one example of MBR representation for image objects. If two objects in an image are overlapped (or disjoint), their MBRs are overlapped (or disjoint). On the other hand, two MBRs are overlapped (or disjoint) do not mean their corresponding objects overlapped (or disjoint). For this reason, spatial queries often involve a two-stage search. First, a filter step based on MBRs is used to rapidly filter out the images that cannot possibly satisfy the query. Then, a refinement step is followed to examine each candidate image by a similarity measure for the purpose of detecting and eliminating false hits [10].

Based on the MBRs, many 2D-string like algorithms have been proposed [6-10]. One of them called 2DC-string method [8] used thirteen spatial relationships defined according to the x-axis (or y-axis) projection of any two objects. And hence, there are one hundred and sixty-nine spatial relationships while considering both the x- and y- axis projection of two objects in an image or a query. All methods based on 2DC strings can only recognize translation and scaling image variants but not rotation variants.

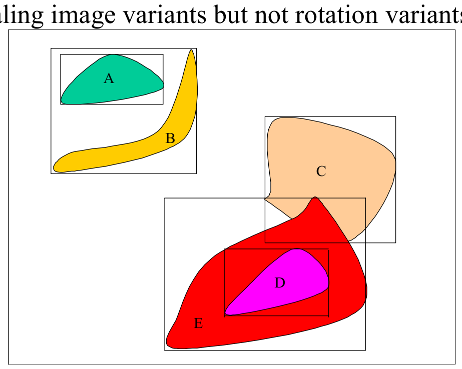


Figure 1. An image contains five objects and the corresponding MBR representation.

Instead of directly using 2DC-string method for recording the spatial relationships among objects in an image or a query, we induce a preprocessing stage to recognize the image rotation variants. The algorithm is as follows.

Algorithm. Rotation Invariant 2DC-String Method.

Input: An image *I*.

Output: Spatial relationships among objects in I.

Method:

1. Compute the coordinates of the centroid of each individual object.
2. For each pair of objects, O_i and O_j in the image, does the following steps:

2.1 Compute the angle θ between the line l joining the contorid points of the two objects and the x-axis.

2.2 Translate the origin of the x-y coordinate system to the point

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(Centroid_x^{O_j} - Centroid_x^{O_i}) \\ \frac{1}{2}(Centroid_y^{O_j} - Centroid_y^{O_i}) \end{bmatrix},$$

where $Centroid_x^O$ and $Centroid_y^O$ represent the x and y coordinates of the centroid of an object *O*, respectively.

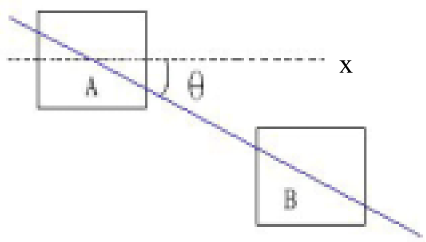
2.3 Rotate the *x*- and *y*-axis by the angle θ, and this will align the new *x*-axis (*x'*-axis) with *l*. The relationship between *x (y)* and *x' (y')* can simply get as follows:

$$x' = x*cos(\theta) - y*sin(\theta), \text{ and}$$

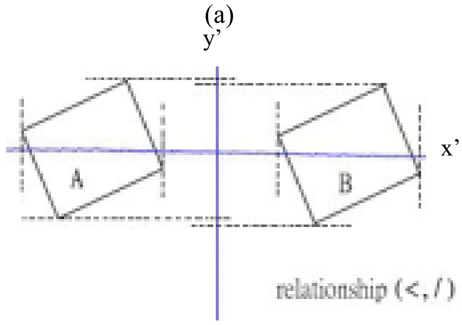
$$y' = x*sin(\theta) + y*cos(\theta).$$

2.4 Follow the definition of the 2DC-string method, we record the spatial relationships according to the *x'*- and *y'*- axis projection of the objects O_i and O_j.

Figure 2 shows an example of such a process. The information associated to the spatial relationship r_{ij} between objects O_i and O_j then includes the left and right object ids, the angle θ, and any other information such as left-to-right distance for the purpose of image query.



(a)



(b)

Figure 2. An example of the rotation processing for a pair of objects A and B; (a) original relationship, (b) rotated relationship.

As mentioned above, the spatial structure of a query is firstly used as the filter step. Note that many objects and hence spatial relationships existed in an image or a query. Those images in a database match the query with at least one spatial relationship are collected as a candidate set which is further ranked by a similarity measure. Figure 3 shows the process of filtering out the dissimilarity images based on the spatial relationships among objects in a query image.

A question arises naturally–how to label objects of images? Without loss of generality we can assume that an object is a uniform region in an image. For the purpose of identifying an object, the object content such as color information, shape features, texture information, or any other content may be of interest can be used. Object clustering based on the associated features is a way for labeling an object in an image. In this paper, we collect all color vectors from the database images and quantize them into several color clusters--with each cluster corresponds to one of the dominant colors. The representative color of each color cluster is its mean, and all means are collected to form a system color palette. The pixels belongs to the same cluster tend to cluster as a region (or an object). And thus, the color index in the system color palette for the cluster can be used to label the object.

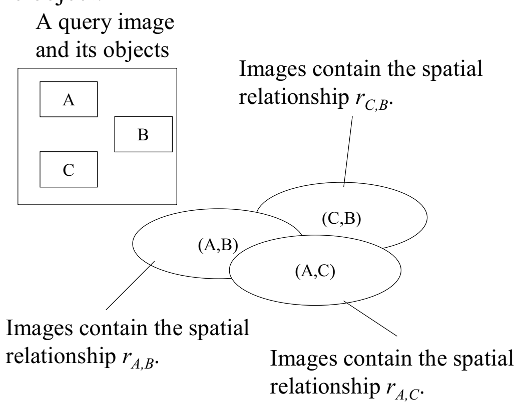


Figure 3. The filter step based on the spatial relationships among objects in a query image.

In the second step, a similarity measure defined according to the low-level features of each object is applied to all images in the candidate set for the purpose of ranking process. Those images with high correlation with the query image are scored with high similarity. On the other hand, some images would be deleted from the match set if their values of similarity are small. The similarity measure between a pair of images is defined as the summation of the individual similarity measure of object pairs.

If a query image Q contains n objects, then the number of spatial relationship within Q is $\frac{n(n-1)}{2}$. The spatial relationships in Q are used to create a candidate image set of size N from a database, and an image P with m objects is one of the images in the set. Let Q_i (P_j) denote the ith (jth) pair of objects in Q (P), and Q_i and P_j both consist of object A and B. If the object pairs Q_i and P_j construct the same spatial relationship (as shown in Figure 4), two aspects of difference between Q_i and P_j can be found. One is the centroid coordinates of the two objects A and B in Q_i might be different from those in P_j. The other is the local color values of A and B in Q_i might be different from those in P_j.

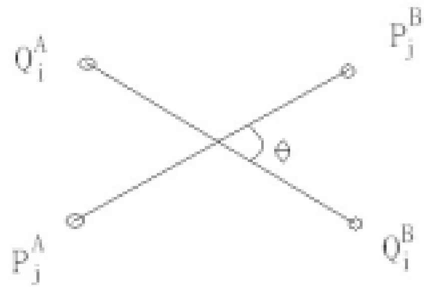


Figure 4. A pair of objects A and B constructs the same spatial relationship with variation in feature details for both images P and Q.

Let $G(O)$ be the centroid coordinates of object O, and $F(O)$ the feature vector associated to O. The spatially similarity between the object pairs Q_i and P_j can be defined as

$$Cos\theta_{Geometry} = \frac{\left[G(Q_i^B) - G(Q_i^A)\right] \bullet \left[G(P_j^B) - G(P_j^A)\right]}{\left|\left[G(Q_i^B) - G(Q_i^A)\right]\right| \times \left|\left[G(P_j^B) - G(P_j^A)\right]\right|}.$$

The method to induce the directional relations in computing the spatial difference between two object pairs corresponding to the images Q and P is first proposed by Gudivada and Raghavan [10].

The difference between the object pairs Q_i and P_j based on the feature vectors associated to objects A and B can also be defined as

$$Cos\theta_{Feature} = \frac{\left[F(Q_i^B) - F(Q_i^A)\right] \bullet \left[F(P_j^B) - F(P_j^A)\right]}{\left|\left[F(Q_i^B) - F(Q_i^A)\right]\right| \times \left|\left[F(P_j^B) - F(P_j^A)\right]\right|}.$$

Combine the two factors above, the similarity between Q_i and P_j can be computed from the following equation:

$$Similarity\_Pair(Q_i, P_j) = W_1\left(\frac{1 + Cos\theta_{Geometry}}{2}\right) + W_2\left(\frac{1 + Cos\theta_{Feature}}{2}\right)$$

, $W_1 + W_2 = 1$.

The value of $W_1(W_2)$ is used to adjust the weighting of the spatial difference (feature variant) and determined according to the domain of application under consideration. For the purpose of reducing the system complexity, only the color information is involved in the feature vector. The value of the measure $Similarity\_Pair(Q_i, P_j)$ is within the interval [0,1]-- 0 means dissimilar and 1 the same.

The measure $Similarity\_Pair(Q_i, P_j)$ should be modified due to the possibility of one object pair in Q matches multiple pairs in P. Only one pair of objects in P is considered in this case for the purpose of similarity computing, and hence the measure $Similarity\_Pair(Q_i, P_j)$ should be modified as

$$Modified\_Similarity\_Pair(Q_i, P) = \underset{P_j \in S_p}{Max}(Similarity\_Pair(Q_i, P_j)),$$

where $S_p = \{P_j \mid P_j \text{ matches } Q_i, 1 \leq j \leq \frac{m(m-1)}{2}\}$. Using the above equation, we can now define the similarity measure between two images *Q* and *P* as follows.

$$Similarity(Q, P) = \frac{2Q_{match}}{m(m-1)} \sum_{i=1}^{Q_{match}} Modified\_Similarity\_Pair(Q_i, P),$$

where Q_{match} denotes the number of match pairs of objects corresponding to the query image *Q* and a database image *P*. Although the pairs of objects in *Q* that match P are not necessary numbered from 1 to Q_{match}. Without loss of generality the similarity measure *Similarity(Q,P)* could be used for all cases if we properly renumber the match pairs of objects in *Q* and *P*.

3. Image Object Model and XML Tags

As mentioned in the above session, objects of an image are the essential entities to construct the spatial structure of the image and the features of objects represent the detail image content. Figure 5 shows the object hierarchy of an image. Each image has a local color palette, which is the collection of representative colors of the image. The value of the attribute "identifier" of "colors" of "Local color palette" in the model refers to the index of the system color palette which indicates the cluster of the corresponding local color.

An object of an image is valued by a list of attributes including the "identifier", "index of system color palette", and "boundary". Among them, "index of system color palette" is used to label the object and the actual color can be found by matching the value of "index of system color palette" with that of "identifier" of "local color palette" for the whole of the image. The collection "relationships" in Figure 5 is a set of spatial relationships among objects, and represents the spatial structure of the image. The value of "code" attribute in "relationships" denotes the cluster to which the spatial relationship of an object pair belongs. Clusters of the spatial relationships among objects follow the definition of the modified 2DC-string method mentioned in the previous session.

In order to describe the image content in XML, we define a set of XML tags. Based on those tags, an index database involving two XML documents is constructed. One of them is an inverted-list file, which records the correspondences between each cluster of spatial relationships among objects and each image. The other describes the content of each image in detail. Figures 6 and 7 show the document type

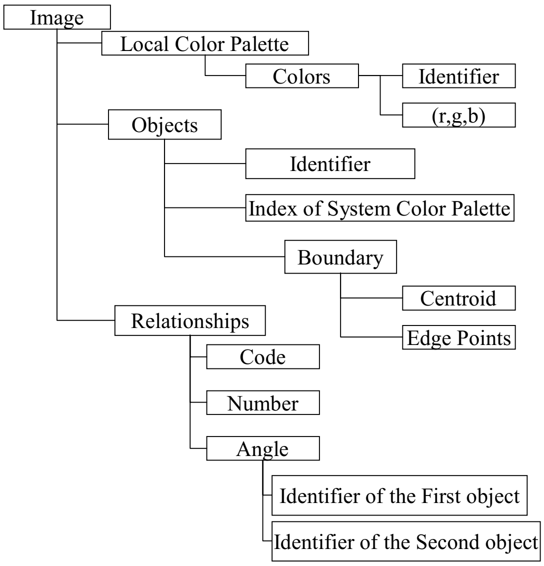


Figure 5. Proposed image object model.

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE spatial-relationship [
<!ELEMENT spatial-relationship (relationship+)>
<!ATTLIST relationship code CDATA #REQUIRED>
<!ELEMENT relationship (graphic*)>
<!ELEMENT graphic EMPTY>
<!ATTLIST graphic gid CDATA #REQUIRED>
]>
```

Figure 6. The part of the XML DTD for indexing the spatial relationships.

definition (DTD) and the index structure of the spatial relationships of the former XML document, respectively. Three tags--"spatial-relationship", "relationship", and "graphic" are defined in the DTD. The tag "spatial-relationship" is the root of the XML document and is required to traverse the XML tree. The values of the tags "relationship" and "graphic" denote the cluster of a spatial relationship and an image identifier, respectively. Those "graphic" child nodes branched from a "relationship" node in the XML document tree belong to the same cluster of spatial relationship. Therefore, one can easily find out all images containing common spatial relationships by traversing all "graphic" nodes under "relationship" nodes.

Each XML document can be parsed into a tree structure and searchable by the XSL pattern technique. For example, the following XSL pattern statement can be used to query all "graphic" nodes for a specified "relationship" node under consideration.

/spatial-relationship/
relationship[code='10']/ graphic.

Given a query image, our system

transforms the spatial relationships of the query into several XSL pattern statements and unites the results of each statement to form a candidate image set. Combine the information contained in

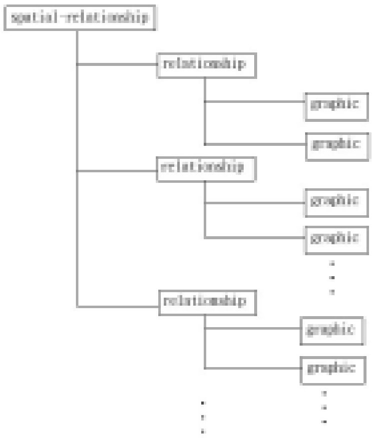


Figure 7. Indexing structure based on spatial relationships.

the XML document that describes each image in detail, the images in the candidate set are ranked according to the similarity measure mentioned in the previous session.

Figures 8 and 9 show the DTD and the structure of the XML document about image content. Basically, most of the tags in the DTD shown the Figure 8 have correspondences to the subtree rooted by "objects" in the image object model shown in Figure 5. Some tags are added into the DTD for the purpose of system implementation. For example, the tag "location" is used to locate the path of the image file.

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE graphicdatabase
[
<!ELEMENT graphicdatabase (graphic*)>
<!ELEMENT graphic (location,local-palette,signature)>
<!ATTLIST graphic gid CDATA #REQUIRE>
<!ELEMENT location (#PCDATA)>
<!ELEMENT local-palette (global+)>
<!ELEMENT global (rgb)>
<!ATTLIST global index CDATA #REQUIRE>
<!ELEMENT rgb EMPTY>
<!ATTLIST rgb r CDATA #REQUIRE
              g CDATA #REQUIRE
              b CDATA #REQUIRE
>
<!ELEMENT signature (relationship+)>
<!ELEMENT relationship (angle+)>
<!ATTLIST relationship code CDATA #REQUIRE
                       number CDATA #REQUIRE>
<!ELEMENT angle (obj+)>
<!ATTLIST angle degree CDATA #REQUIRE>
<!ELEMENT obj EMPTY>
<!ATTLIST obj strobj CDATA #REQUIRE
              endobj CDATA #REQUIRE
>
]>
```

Figure 8. The DTD of the XML document used to describe the content of each image.

5. Implementation of the XML-Based Image Retrieval System

Based on the proposed object-based image retrieval method and the XML tags mentioned in the precious session, the block diagram of the XML-based image retrieval system is shown in Figure 10. The XML tags are used to organize the image content in structure and let the content of images be searchable via the XSL pattern technique. Given an input image, we first analyze the image for the purpose of extracting the spatial structure and the color distribution of the query. The information is then sent to the server to match the images in the database.

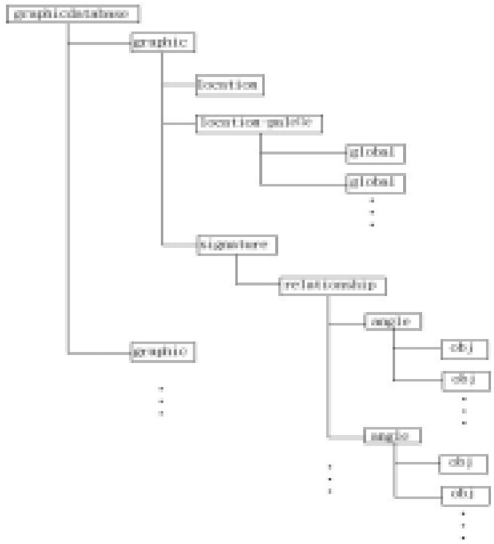


Figure 9. The tree structure of the XML document describes the content of each image.

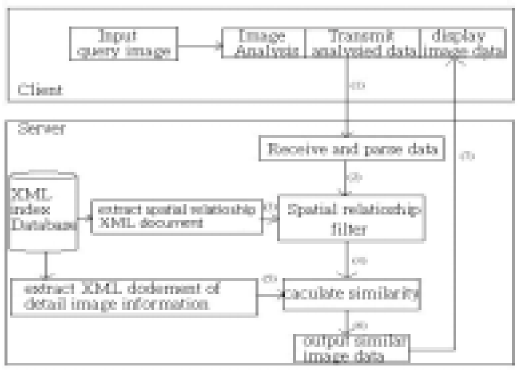


Figure 10. Block diagram of the proposed XML-based image retrieval system.

The system can be divided into three parts. The first one is the analysis module including object segmentation and labeling, spatial relationships construction, and color features extraction. The analysis module represents the image content of a query as a tree (as shown in Figure 9), which can be used to query the database via the XSL pattern technique. The second one is the filtering module that filters out all dissimilar images from the database based on the spatial relationships among objects of a query followed by a similar measure to rank the match images. The last module is used to display the query results in the browser which a user specified the query.

. Without loss of generality, we implement the system on the platform of Microsoft. The programming languages used here include Jscript and active server pages (ASPs) of Microsoft because they well support the XML and XSL pattern techniques. Other platforms or programming languages such as JAVA might be possible to implement the system.

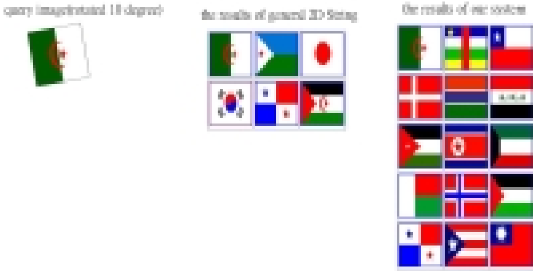
5. Test Results and Discussions

We have tested the XML-based image retrieval system on a "flag" image collection of size 110. The purpose of using this database is to demonstrate the effectiveness of our approach. How to judge the outcome of the test is an important issue. Some systems proposed before compared the efficiency of their systems to that of humans, which means a benchmarking database was first created through the assist of human assessment. Unfortunately, such a benchmarking database depends on the application under consideration and hence a hard work to construct. Instead of comparing the test results with that of humans, we compare our results with that of using the 2DC-string method.

A set of query images is randomly selected from the database. Each of them including its rotated or distorted versions is tested both on the proposed system and the 2DC-string method. As the experimental results shown, the proposed method is rotation invariant but the 2DC-string method is not. Figure 11 shows one test example on both methods by rotating a query image by 10, 30, 50, and 70 degrees. According to the experimental results, our technique can recognize translation, scaling, and rotation invariants. On the other hand, the 2DC-string method cannot recognize rotation invariants.

As compare the performance of our system to that of general content-based image database systems, our system uses the MBRs to filter out all dissimilar images from databases before using the low-level feature vectors to rank match images. This results in a better performance in our technique and promotes the applicability of the proposed method to real-case applications. In order to avoid missing data during retrieval, the number of colors in the system palette used to label objects in images is not large. This limits the efficiency of the proposed filtering scheme. If the shape information of objects is clustered and properly added to label objects, the number of images needs to compute the similarity values in the second query stage could be reduced. However, the recall rate of a query might also be lower due to the number of cluster which is equal to the number of color cluster multiplies the number of shape cluster becomes larger. This fact increases the ambiguity of clusters to specify objects in images or a query.

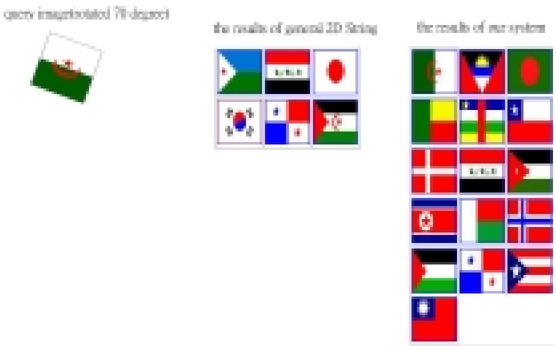
Although we test our system to a small database, we do not see theoretical difficulties in applying our technique to large databases and the integrated spatial structure and color features approach should work equally well for large databases.



(a)

(b)

(c)



(d)

Figure 11. Test results of a query image by different rotation degrees on both the general 2DC-string and the proposed methods. (a) 10 degrees, (b) 30 degrees, (c) 50 degrees, and (d) 70 degrees.

Furthermore, we exchange images among clients and servers in the XML formats which are platform portable. If we maintain the self-defined XML tags for images in a dedicated resource description server, the proposed system can be used to integrate heterogeneous content-based image databases. The XML tags proposed in this work transform free-structured images into well-defined formats and let images be semantically understandable. The information of images to share over the Internet is now not limited to the raw data which meaning is assumed to be interpreted by human vision.

6. Conclusions

In this paper, we have described an XML-based image retrieval method using a hybrid spatial structure-color approach. The proposed technique has been tested on a “flag” image database. The proposed method recognizes translation, scaling, and rotation invariants. These characteristics are important in image retrieval systems. Instead of ranking images from databases for a query by computing the similarity among high-dimensional feature vectors, our two-stage approach has better performance and thus can apply to large databases. The retrieval outputs based on this technique give images that are ranked in the order of similarity, taken into account of both the spatial and color information.

Although the type of media to be processed in our system is only images, we can not find the theoretical limitation for applying our method to format other types of media, i.e. videos, graphics, and audio in XML.

7.References

[1] C. Faloutsos et. al., 1994, "Efficient and Effective Querying by Image Content," Journal of Intelligent Systems, 1, pp. 95-108.

[2] M. Flickner et. al., 1995, "Query by image and video content: The QBIC System," IEEE Computer, 28(9), pp. 23-32.

[3] A. Pentland, R. Picard, and S. Scalroff, 1994, "Photobooks: Tools for Content-Based Manipulation of Image Databases," SPIE Conf. On Storage and Retrieval of Image and Video Databases II, pp. 33-47.

[4] B. M. Mether, M. S. Kankanhall, and W. F. Lee, 1998, "Content-Based Image Retrieval Using A Composite Color-Shape Approach," Information Processing and Management, 34(1) pp. 109-120.

[5] C. S. Li, V. Castelli, and L. Bergman, “Progressive Content-Based Retrieval From Distributed Image/Video Databases,” *Proceedings of IEEE International Symposium on Circuits and Systems*, HongKong June, 1997.

[6] S. K. Chang, Q. Y. Shi, and C. W. Yan, 1984, "Iconic Indexing by 2-D Strings", IEEE Trans. Pattern Anal. Mach. Intell., vol PAMI-6, pp.475-484.

[7] P.W Huang, and Y.R Jean, 1994,"Using 2D c+-String as spatial knowledge representation for image database systems", Pattern Recognit, 27, 9, pp.1249-1257.

[8] S.Y Lee and F.J Hsu, 1990, "2D C-string：A new spatial Knowledge representation for image database systems", Pattern Recognition, 23, pp. 1077-1087.

[9] Ye-In Chang, Bi-Yen Yang, 1997, "Efficient access method for image database", Information Processing Letters 64 , pp.95-105.

[10] Venkat N. Gudivada, and Vijay V. Raghavan, 1995, "Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity", Transactions on Information Systems, Vol 13, NO 2, ACM ,pp.114-144,April.

[11] Aaron Weiss, 1999, “XML gets down to business,” Mixed Media, ACM, pp36-43, Sep.

[12] Robert J. Glushko, Jay M. Tenenbaum, and Bart Meltzer, 1999, “An XML framework for agent-based E-commerce,” Communication of ACM, Vol. 42, No .3, ACM, pp.106-114, March.

[13] Steve Widergren, Amold deVod, and JunZhu, , 1999, “XML for Data Exchange,” IEEE Information System from the Web,” IEEE Proc. 31st Annual Hawaii International Conference on System Sciences, pp.14-23.

[14] Aaron Weiss, 1999, “XML gets down to business,” Mixed Media, ACM, pp36-43, Sep.

[15] Robert J. Glushko, Jay M. Tenenbaum, and Bart Meltzer, 1999, “An XML framework for agent-based E-commerce,” Communication of ACM, Vol. 42, No .3, ACM, pp.106-114, March.

[16] Steve Widergren, Amold deVod, and JunZhu, , 1999, “XML for Data Exchange,” IEEE Data Engineering Bulletin, Vol. 22 No. 3, pp.840-842.

[17] Andrew V.Royappa, 1998, "Implementing Catalog Clearinghouse with XML and XSL,” in Proc. Of SAC’98, ACM, pp. 616-623.

[18] CARLSON, B. 1997. “MPEG-7 for multimedia searching: Room for video content recognition?.” *Adv. Imaging 12*, 3 (Mar.), 21~ 22.