Image Database Design Based on 9D-SPA Knowledge Representation for Spatial Relations

C. H. Lee[1] and P. W. Huang[2]

[1]Department of Information Management, Chaoyang University of Technology

168 Gifeng E.Rd., Wufeng, Taichung, Taiwan, R.O.C.

[2]Department of Applied Mathematics, National Chung-Hsing University

Taichung 40227, Taiwan, R.O.C.

Abstract

Spatial relationships between objects are important features for designing a content-based image retrieval system. In this paper, we propose a new scheme, called 9D-SPA representation, for encoding the spatial relations in an image. With this representation, important functions of intelligent image database systems such as iconic indexing and similarity retrieval can be easily achieved. The capability of discriminating images based on 9D-SPA representation is much more powerful than any spatial representation method based on Minimum Bounding Rectangles or centroids of objects. The similarity measures using 9D-SPA representation provides a wide range of fuzzy matching capability in similarity retrieval to meet different user's requirements. More importantly, the 9D-SPA representation can be incorporated into a three-level index structure to help reducing the search space of each query processing. The experimental results demonstrated that, on an average, only 0.1254% $\sim$ 1.6829 % of symbolic pictures (depending on various degrees of similarity) were accessed per query in an image database containing 50,000 symbolic pictures. This is a significant improvement over pure "iconic indexing" without using any higher-level index scheme.

Keywords

Image database Spatial relations Similarity retrieval Spatial reasoning 9D-SPA

This paper is intended for the Workshop on Databases and Software Engineering.

Contact author: C. H. Lee

[1]Email: chlee@cyut.edu.tw, Phone: 886-4-23323000 ext 7122, Fax: 886-4-23742337

[2]Email: huang@amath.nchu.edu.tw, Phone: 886-4-22860133 ext 612, Fax: 886-4-22873028

1. INTRODUCTION

A pictorial database plays an important role in many applications including Geographical Information Systems, Computer Aided Design, Office Automation, Medical Image Archiving, and Trademark Picture Registration. The traditional approach to image database design is to use textual descriptions to annotate images and then store annotations in a text-based Data Base Management System. Searching for desired images is equivalent to searching for the associated annotations. This approach is too tedious and labor-intensive. Moreover, the key words used in annotations or query descriptions are too subjective and may become inadequate especially when the number of images in the database increases tremendously.

Content-based image retrieval (CBIR) is the current trend of designing image database systems [19] as opposed to text-based image retrieval. The features used in content-based image retrieval can be roughly divided into two categories: the low-level visual features (such as color, texture, shape) and the high-level features (such as pairwise spatial relationships between objects). Some examples of content-based image retrieval systems are QBIC [7], Virage [1], Retrieval Ware [22], VisualSEEK [20], WaveGuide [14] and Photobook [17]. They allow users to retrieve similar pictures from a large image database based on low-level visual features. On the other hand, there is also a large group of researchers emphasizing image retrieval based on spatial relationships between objects [3], [4], [5], [9], [12], [13], [16], [18], [21]. In this paper, we only concentrate on picture retrieval based on spatial relations.

The method of representing images is one of the major concerns in designing an image database system. The representation method for an image should capture the knowledge about the image's content as much as possible. One way of representing an image is to construct a symbolic picture for that image which in turn is encoded into a 2D-string [5]. The 2D string representation method opened up a new approach to picture indexing and similarity retrieval. There are many follow-up research works based on the concept of 2D string such as 2D C-string [12], [13], and 2D C^{+}-string [8].

An ideal representation method for symbolic pictures should provide image database systems with many important functions such as similarity retrieval and picture indexing. In this paper, we propose a new scheme for encoding spatial relations called 9-Direction SPanning Area (9D-SPA) representation method. Using the 9D-SPA representation, we can easily accomplish the following system design goals: (1) Flexibility and accuracy in similarity retrieval can be achieved at the same time through a set of coarse-to-fine similarity measures; (2) The 9D-SPA representation can be incorporated into an efficient index structure so that the search space will be restricted to a relatively small portion of the database for improving retrieval efficiency.

The remainder of this paper is organized as follows. In Section 2, previous research works about knowledge representations for spatial relations are discussed. In Section 3, the 9D-SPA representation is introduced. In Section 4, we define a set of similarity measures for fuzzy matching. We also introduce the index structure based on 9D-SPA representation to facilitate image retrieval. The similarity retrieval algorithm is presented in this same section. The experimental result to demonstrate the effectiveness of our approach is presented in Section 5. Finally, conclusions are given in the last section.

2. OVERVIEW OF SPATIAL KNOWLEDGE REPRESENTATION

Binary spatial relationships between objects have been identified as one of the most important features for describing the contents of images [6]. For example, a query such as "finding all the pictures containing a house to the east of a tree" relies on spatial relations to retrieve the desired pictures. Different kinds of spatial knowledge representations have been proposed so far. Chang et al. [5] proposed the 2D string as a spatial knowledge representation to capture the spatial information about the content of a picture. The fundamental ideal of 2D string is to project the objects of a picture along the x- and y-directions to form two strings representing the relative positions of objects in the x- and y-axis, respectively. Since a 2D string preserves the spatial relationships between any two objects in a picture, it has the advantage of facilitating spatial reasoning. Moreover, since a query picture [6] can also be represented as a 2D string, the problem of similarity retrieval becomes a problem of 2D string subsequence matching.

Jungert [10], Chang et al. [4], and Jungert and Chang [11] extended the idea of 2D strings to form 2D G-strings by introducing several new spatial operators to represent more relative positional relationships among objects of a picture. The 2D G-string representation embeds more information about spatial relationships between objects, thus facilitates spatial reasoning about sizes and relative positions of objects.

Following the same concept, Lee and Hsu [12] proposed the 2D C-string representation based on a special cutting mechanism. Since the number of subparts generated by this new cutting mechanism is reduced significantly, the lengths of the strings representing pictures are much shorter while still preserving the spatial relationships among objects. The 2D C-string representation is more economical in terms of storage space efficiency and navigation complexity in spatial reasoning. The 2D C^+-string representation [8] extended the 2D C-string representation by adding relative metric information about the picture to the strings. As a consequence, reasoning about relative sizes and locations of objects, as well as the relative distance between objects in a symbolic picture becomes possible.

Chang [3] proposed a structure called 9DLT to encode the spatial relationships between objects in terms of nine directions. Since the 9DLT method uses centroid to represent the position of an object, such a representation is too sensitive in spatial reasoning. For example, the spatial relationships between the two objects shown in Fig. 1(a)-(c) are all different in 9DLT representation; however, they seem not too much different in human visual perception.

The representation of spatial relations proposed by Zhou and Ang [21] combines the nine directional relations proposed in 9DLT with the five topological relations, namely, disjoint, meet, partly_overlap, contain, and inside. The topological relation can record the 2D relationship between any two sized objects with irregular shapes and, therefore, makes spatial reasoning more accurate as compared to using MBR or centroid to represent an object. However, Zhou and Ang's method still has the problem with being too sensitive when reasoning about directional relations.

Instead of combining the nine directional relations with the five topological relations, the 2D-PIR proposed by Nabil et al. [15] combines the 13 projection interval relations with the topological relations. Although 2D-PIR seems particularly useful in similarity retrieval, it did not provide any picture reconstruction mechanism for visualization. Besides, incorporating 2D-PIR into any indexing structure is difficult. Thus, similarity retrieval based on 2D-PIR becomes inefficient if the volume of images in the database increases.

3. The 9D-SPA REPRESENTATION

To represent a picture using our method, the picture has to be preprocessed first. We assume that the objects in a picture can be identified by some image segmentation and object recognition procedures. Various techniques of image segmentation and object recognition can be found in [2].

Suppose that a picture P contains n objects (O_1,O_2,...,O_n). Then, the 9D-SPA representation of P can be encoded as a set of 4-tuples: $R = \{(O_{ij}, D_{ij}, D_{ji}, T_{ij}) | \forall O_i,\ O_j \in P,$ and $1 \leq i < j \leq n\}$, where O_{ij} is the code for object-pair (O_i, O_j), D_{ij} is the code for the direction relation between objects O_i and O_j with O_j as the reference object, D_{ji} is the code for the direction relation between O_i and O_j with O_i as the reference object, and T_{ij} is the code for the topological relation between O_i and O_j. It is obvious that the number of 4-tuples in R is $\frac{n(n-1)}{2}$.

Let O_i be the ith object in the image database ($1 \leq i \leq N$). We assign integer i to object O_i as its object number. Then, O_{ij} is called the *object-pair code* for object-pair (O_i, O_j). Given two objects O_i and O_j, we can easily compute the object-pair code O_{ij} using the following formula:

$$O_{ij} = \frac{(j-1)(j-2)}{2} + i.$$

To obtain the two object numbers from O_{ij} (or to decode O_{ij}), we need to calculate $b = O_{ij} - \frac{a(a+1)}{2}$, where a is the largest integer such that $\frac{a(a+1)}{2} \leq O_{ij}$. Then $i = b$ and $j = a+2$.

D_{ij} represents the value assigned to the directional relationship between objects O_i and O_j with O_j as the reference object. The value of D_{ij} is determined by the following procedure. First, we find the Minimal Bounding Rectangle (MBR) for reference object O_j. Then, we extend the four boundaries of this MBR horizontally and vertically until they cut the whole picture into nine neighborhood areas and assign each area a binary code as shown in Table 1. The value of D_{ij} is determined by the formula $D_{ij} = \sum_{k=0}^{8} b_k w_k$, where w_k is the binary code of neighborhood area k; $b_k = 1$ if object O_i overlaps area k, otherwise, $b_k = 0$. The value of T_{ij} indicates the topological relationship between objects O_i and O_j. The possible values assigned to topological relations are: 0 (stands for "disjoint"), 1 (stands for "meet"), 2 (stands for "partly-overlap"), 3 (stands for "contain" or "inside").

Let us look at the two pictures shown in Fig. 2(a) and 2(b). Assume that object B is the reference object in both pictures. Then, in Fig. 2(a), the code for D_{AB} is $(00001000 + 00010000 + 00100000 + 01000000 + 10000000)_2 = 248$ and the code for T_{AB} is 0. In Fig. 2(b), the code for D_{AB} is $(00000001 + 00000010 + 00100000 + 01000000 + 10000000)_2 =$ 227, and the code for T_{AB} is 0. In 2D *-string representations, the pictures in Fig. 2(a) and 2(b) are not distinguishable because they have the same spatial representation (i.e. A%B in both x- and y-directions). However, we can easily tell the difference between them by using 9D-SPA representation because D_{AB} in Fig.2(a) is 248 while D_{AB} in Fig. 2(b) is 227.

Moreover, from $D_{AB} = 248 = (11111000)_2$, we can easily determine that object A spans five neighborhood areas of object B, namely, the northwest, the west, the southwest, the south, and the southeast neighborhood areas as shown in Fig. 2(a). Similarly, from $D_{AB} = 227 = (11100011)_2$, we can easily determine that object A spans another different five neighborhood areas of object B, namely, the northeast, the east, the southeast, the south, and the southwest neighborhood areas as shown in Fig. 2(b).

4. SIMILARITY RETRIEVAL

In similarity retrieval, the user must present a query picture to be matched with the database images. One convenient way of presenting a query is to draw a sketch diagram called query picture [6]. The task of image retrieval is to measure the similarity between the query picture and the database picture, then retrieve relevant pictures from the database. Usually, the user may not remember the exact spatial relationships among the objects in a desired picture. To accommodate this flexibility, the system should provide the user with a set of coarse-to-fine similarity measures to measure the difference between the query picture and the database picture. With a coarse measure, we allow the retrieved images to be slightly different from the query picture. However, the retrieved images still meet the user's requirements in term of user's visual perception.

4.1. Similarity Measures

Before giving detailed definitions for similarity measures, we introduce the following notations first:

- p: a database picture.
- q: a query picture.
- S_p(or S_q): the set of objects in picture p (or q).
- R_p (or R_q): the 9D-SPA representation for picture p (or q).
- t^p (or t^q): a tuple in R_p (or R_q).
- $t.O$: the object-pair code of tuple t.
- $t.D_1$ (or $t.D_2$): the directional relation-code, with O_j (or O_i) as the reference object, of tuple t.
- $t.T$: the topological relation-code of tuple t.
- $t.D_1(j)$ (or $t.D_2(j)$): the jth bit of the binary code of $t.D_1$ (or $t.D_2$).
- $x(k)$ (or $y(k)$): the kth bit of the binary code of x (or y).

Let $R_p = \{t_1^p, t_2^p, \ldots, t_n^p\}$ and $R_q = \{t_1^q, t_2^q, \ldots, t_m^q\}$ with $n \geq m$. Let ε be a one-to-one function from $\{1, 2, \ldots, m\}$ to $\{1, \ldots, n\}$ such that $t_i^q.O = t_{\varepsilon(i)}^p.O$ for all $1 \leq i \leq m$. Then, we define the directional similarity measure between R_p and R_q as follows:

$$S_D(R_p, R_q) = \frac{\sum_{i=1}^m s_D(t_{\varepsilon(i)}^p.D_1, t_i^q.D_1) + \sum_{i=1}^m s_D(t_{\varepsilon(i)}^p.D_2, t_i^q.D_2)}{2m}, \quad (1)$$

where

$$s_D(x, y) = \begin{cases} 1, & \text{if } \sum_{k=1}^8 x(k) \vee y(k) = 0; \\ \frac{\sum_{k=1}^8 x(k) \wedge y(k)}{\sum_{k=1}^8 x(k) \vee y(k)}, & \text{otherwise.} \end{cases} \quad (2)$$

In the above formula, symbol "$\wedge$" ("$\vee$") represents the logical "AND" ("OR") operation. Similarly, we define the topological similarity measure between R_p and R_q as follows:

$$S_T(R_p, R_q) = \frac{\sum_{i=1}^m s_T(t_{\varepsilon(i)}^p.T, t_i^q.T)}{m}, \quad (3)$$

where

$$s_T(x, y) = 1 - \frac{|x - y|}{3}. \quad (4)$$

Notice that $S_D(R_p, R_q)$ and $S_T(R_p, R_q)$ are undefined if function ε does not exists (i.e. there is some object in query picture q that cannot be found in picture p). Based upon the above two similarity measuring equations for $S_D(R_p, R_q)$ and $S_T(R_p, R_q)$, we provide the following definitions:

Definition 1: A database picture p is directional-similar to a query picture q with a degree of similarity s_1 iff $S_q \subseteq S_p$ and $S_D(R_p, R_q) = s_1$.

Definition 2: A database picture p is topological-similar to a query picture q with a degree of similarity s_2 iff $S_q \subseteq S_p$ and $S_T(R_p, R_q) = s_2$.

We use the pictures q, p_1, and p_2 shown in Fig. 3 to illustrate our similarity measures. The 9D-SPA representations for pictures q, p_1, and p_2 are listed below:

R_q ={(1,255,0,3), (2,14,64,0), (4,24,129,0), (3,2,32,0), (5,24,129,0), (6,48,3,0)}.

R_{p_1}={(1,255,0,3), (2,14,64,0), (4,8,128,0), (7,12,64,0), (3,2,32,0), (5,8,128,0), (8,4,224,0), (6,16,131,0), (9,8,128,0), (10,2,32,0)}.

R_{p_2}={(1,241,0,3), (2,14,64,0), (4,14,64,1), (7,8,128,0), (3,2,32,0), (5,4,224,0), (8,8,128,0), (6,16,131,0), (9,8,128,0), (10,8,128,0)}.

After applying equs. (2) and (4) to each tuple in R_q and the corresponding tuple in R_{p_1} (or R_{p_2}), we obtain the following results:

$s_D(t_1^{p_1}.D_1, t_1^q.D_1) = 1$,	$s_T(t_1^{p_1}.T, t_1^q.T) = 1$,	$s_D(t_1^{p_2}.D_1, t_1^q.D_1) = 5/8$,	$s_T(t_1^{p_2}.T, t_1^q.T) = 1$,
$s_D(t_1^{p_1}.D_2, t_1^q.D_2) = 1$,	$s_T(t_2^{p_1}.T, t_2^q.T) = 1$,	$s_D(t_1^{p_2}.D_2, t_1^q.D_2) = 1$,	$s_T(t_2^{p_2}.T, t_2^q.T) = 1$,
$s_D(t_2^{p_1}.D_1, t_2^q.D_1) = 1$,	$s_T(t_3^{p_1}.T, t_3^q.T) = 1$,	$s_D(t_2^{p_2}.D_1, t_2^q.D_1) = 1$,	$s_T(t_3^{p_2}.T, t_3^q.T) = 2/3$,
$s_D(t_2^{p_1}.D_2, t_2^q.D_2) = 1$,	$s_T(t_5^{p_1}.T, t_4^q.T) = 1$,	$s_D(t_2^{p_2}.D_2, t_2^q.D_2) = 1$,	$s_T(t_5^{p_2}.T, t_4^q.T) = 1$,
$s_D(t_3^{p_1}.D_1, t_3^q.D_1) = 1/2$,	$s_T(t_6^{p_1}.T, t_5^q.T) = 1$,	$s_D(t_3^{p_2}.D_1, t_3^q.D_1) = 1/4$,	$s_T(t_6^{p_2}.T, t_5^q.T) = 1$,
$s_D(t_3^{p_1}.D_2, t_3^q.D_2) = 1/2$,	$s_T(t_8^{p_1}.T, t_6^q.T) = 1$,	$s_D(t_3^{p_2}.D_2, t_3^q.D_2) = 0$,	$s_T(t_8^{p_2}.T, t_6^q.T) = 1$,
$s_D(t_5^{p_1}.D_1, t_4^q.D_1) = 1$,		$s_D(t_5^{p_2}.D_1, t_4^q.D_1) = 1$,	
$s_D(t_5^{p_1}.D_2, t_4^q.D_2) = 1$,		$s_D(t_5^{p_2}.D_2, t_4^q.D_2) = 1$,	
$s_D(t_6^{p_1}.D_1, t_5^q.D_1) = 1/2$,		$s_D(t_6^{p_2}.D_1, t_5^q.D_1) = 0$,	
$s_D(t_6^{p_1}.D_2, t_5^q.D_2) = 1/2$,		$s_D(t_6^{p_2}.D_2, t_5^q.D_2) = 1/4$,	
$s_D(t_8^{p_1}.D_1, t_6^q.D_1) = 1/2$,		$s_D(t_8^{p_2}.D_1, t_6^q.D_1) = 1/2$,	
$s_D(t_8^{p_1}.D_2, t_6^q.D_2) = 2/3$,		$s_D(t_8^{p_2}.D_2, t_6^q.D_2) = 2/3$,	

The resultant similarity measures are obtained by averaging the similarity values in each column:

$$S_D(R_{p_1}, R_q) = 0.76, \quad S_T(R_{p_1}, R_q) = 1, \quad S_D(R_{p_2}, R_q) = 0.61, \quad S_T(R_{p_2}, R_q) = 0.94,$$

Thus, picture p_1 is directional-similar to picture q with a degree of similarity 0.76. Picture p_2 is directional-similar to picture q with a degree of similarity 0.61. Picture p_1 is topological-similar to picture q with a degree of similarity 1. Picture p_2 is topological-similar to picture q with a degree of similarity 0.94. In a general sense, we can say that p_1 is more similar to q than p_2 in both directional and topological relations. This result is consist with what we expected.

4.2. Index Structure

The 9D-SPA representations of database pictures can be incorporated into a three-level index structure to facilitate image retrieved. An example of such an index structure is shown in Fig. 4. There are three levels of indices in this index structure based on 9D-SPA representation. The first-level index, called the *level-1 index array*, is an array of size

$\frac{N(N-1)}{2}$, where N is the number of distinct objects in the database. Each entry in this array may contain a pointer to a list of directional relation-codes, called "direction-code list", which constitutes the second-level index structure. Each item in a direction-code list is an array of four elements: the first element is the directional relation-code D_{ij}; the second element is the directional relation-code D_{ji}; the third element is a pointer to another array of size 4, called "topological relation array", which constitute the third level index structure; the fourth element is a pointer to the next item in the current direction-code list. Each entry in a topological relation array corresponds to one type of topological relations and may contain a pointer to a list of database images. In Fig. 4, the tuple $(9, 3, 48, 1)$ will be mapped to image f_2 according to our indexing scheme. In other words, image f_2 have objects O_3 and O_5 because object-pair code 9 is decoded into object codes 3 and 5; Object O_3 is to the east or northeast of object O_5 because directional relation-code 3 is decoded into "00000001" $\vee$ "00000010"; Object O_5 is to the west or southwest of object O_3 because directional relation-code 48 is decoded into "00010000" $\vee$ "00100000"; and the topological relationship between objects O_3 and O_5 is "meet" because $T_{3,4} = 1$.

4.3. Image Retrieval Algorithm

The index structure helps reducing the search space. There are four types of similarity requirements (S_D, S_T) =(0,0), (1,0), (0,1), and (1,1) which can be used to retrieve images by directly following the index structure. Others similarity requirements need detailed similarity measuring operation to discard unqualified pictures. The algorithm of retrieving similar images based on 9D-SPA representation and the three-level index structure is presented as follows.

Algorithm: Similarity retrieval based on 9D-SPA representation and three-level index structure.

Input: A 9D-SPA representation R_q for query picture q and two thresholds h_D and h_T.

Output: $\{p|p$ is a database picture, $S_q \subseteq S_p$, $S_D(R_p, R_q) \geq h_D$, $S_T(R_p, R_q) \geq h_T\}$.

1. $g_D = \lfloor h_D \rfloor$; $g_T = \lfloor h_T \rfloor$.

2. $\forall$ tuple $(O_{ij}, D_{ij}, D_{ji}, T_{ij}) \in R_q$

   (a) Find the direction-code list L associated with O_{ij} from the level-1 index array.

   (b) Get the set S of pointers to the topological relation arrays associated with the items in L whose two directional relation-code are k_1 and k_2 such that $s_D(k_1, D_{ij}) \geq g_D$ and $s_D(k_2, D_{ji}) \geq g_D$.

(c) $\forall r \in S$

i. Find the topological relation array A pointed by r.

ii. Obtain the set Γ_t of images associated with A[u], $0 \leq u \leq 3$, such that $S_T(u, T_{ij}) \geq g_T$.

3. $\Gamma \leftarrow \cap_{t \in R_q} \Gamma_t$.

4. if $(g_D \neq h_D)$ or $(g_T \neq h_T)$, then
$\forall p \in \Gamma$, if $(S_D(p, q) < h_D)$ or $(S_T(p, q) < h_T)$, then remove p from Γ.

5. Return Γ.

We use the index structure in Fig. 4 to illustrate our similarity retrieval algotithm. Assume that, the 9D-SPA representation for query picture q is $R_q = \{t_1, t_2, t_3\} = \{(6, 14, 64, 0), (9, 16, 129, 1), (10, 48, 3, 0)\}$ and the similarity requirement is (h_D, h_T)=(1,1). In step 1, we get $g_D = \lfloor 1 \rfloor = 1$ and $g_T = \lfloor 1 \rfloor = 1$. Therefore, we obtain $\Gamma_{t_1} = \{f_6, f_7\}$, $\Gamma_{t_2} = \{f_6\}$, and $\Gamma_{t_3} = \{f_6, f_7\}$, respectively. The set of related database pictures are $\Gamma_{t_1} \cap \Gamma_{t_2} \cap \Gamma_{t_3} = \{f_6\}$. Because $g_D = 1 = h_D$ and $g_T = 1 = h_T$, we do not need extra check for the pictures in Γ. So f_6 is the only qualified picture.

Now we change the similarity requirement to $(h_D, h_T) = (0.8, 0.8)$. In step 1, we get $g_D = \lfloor 0.5 \rfloor = 0$ and $g_T = \lfloor 0.5 \rfloor = 0$. So we obtain $\Gamma_{t_1} = \{f_8, f_6, f_7\}$, $\Gamma_{t_2} = \{f_2, f_7, f_6, f_8\}$, and $\Gamma_{t_3} = \{f_8, f_6, f_7, f_{14}, f_4, f_{16}\}$. Thus, the related database pictures are $\Gamma_{t_1} \cap \Gamma_{t_2} \cap \Gamma_{t_3} = \{f_6, f_7, f_8\}$. Because $g_D = 0 \neq 0.8 = h_D$ in this case, we need detailed check to see if the pictures in Γ meet the similarity requirement. The values of (S_D, S_T) for f_6, f_7, and f_8 with respect to q are (1,1), (1,0.89), and (0.63,0.89), respectively. According to the new similarity requirement $(S_D, S_T) = (0.8, 0.8)$, only pictures f_6 and f_7 are qualified and returned.

5. EXPERIMENTAL RESULTS

In this section, we present the simulation results to demonstrate the efficiency of the similarity retrieval algorithm based on the three-level index structure. Without using an index structure, we need to inspect all 9D-SPA representations associated with the database images and compare them with the 9D-SPA representation of the query picture during image retrieval. By using the three-level indexing structure, the search space can be restricted to a small set of images. According to our retrieval algorithm, we can reduce the search space to a set of relevant pictures whose similarity degree with q is $(\lfloor S_D \rfloor, \lfloor S_T \rfloor)$. The four possible values of $(\lfloor S_D \rfloor, \lfloor S_T \rfloor)$ are (0,0), (0,1), (1,0), and (1,1). The similarity requirement $(S_D, S_T) = (0, 0)$ means that a database picture p is similar to the query picture q if each object in q can also be found in p without considering spatial relationships among objects in both pictures. The similarity requirement $(S_D, S_T) = (0, 1)$ means that all topological relationships between objects in the query picture must be fully matched with those in a relevant database picture. The similarity requirement $(S_D, S_T) = (1, 0)$ means that all directional relationships between objects in the query picture are matched with those in a relevant database picture. For $(S_D, S_T) = (1, 1)$, two pictures are considered as similar only if all the spatial relationships (both directional and topological) between objects in the query picture are fully matched with those in the database picture.

In our experiment, we would like to estimate the percentage of the average number of pictures accessed per query. The percentage is 100% for exhaustive search without using any index structure. We expect that this percentage will be significantly reduced if the index structure is used. In our experimental system, there are 50000 pictures and 26 different iconic objects in an image database. There are three versions of database so that a picture in version-1 database contains 4 to 6 objects, a picture in version-2 database contains 6 to 8 objects, and a picture in version-3 database contains 8 to 10 objects. There are three types of query pictures: type-1 contains 2 to 3 objects, type-2 contains 3 to 4 objects, type-3 contains 4 to 5 objects in each query picture. The pictures were randomly generated by considering all possible directional and topological relations between objects.

We use the notation $[Q_l,\ Q_u]$ to represent the case that the number of objects contained in a query picture is between Q_l and Q_u. Similarly, $[P_l,\ P_u]$ denotes that the number of objects contained in a database picture is between P_l and P_u. There are nine test cases in our experiment:

(1) $[Q_l, Q_u]$ = [2,3] and $[P_l, P_u]$ = [4,6],

(2) $[Q_l, Q_u]$ = [2,3] and $[P_l, P_u]$ = [6,8],

(3) $[Q_l, Q_u]$ = [2,3] and $[P_l, P_u]$ = [8,10],

(4) $[Q_l, Q_u]$ = [3,4] and $[P_l, P_u]$ = [4,6],

(5) $[Q_l, Q_u]$ = [3,4] and $[P_l, P_u]$ = [6,8],

(6) $[Q_l, Q_u]$ = [3,4] and $[P_l, P_u]$ = [8,10],

(7) $[Q_l, Q_u]$ = [4,5] and $[P_l, P_u]$ = [4,6],

(8) $[Q_l, Q_u]$ = [4,5] and $[P_l, P_u]$ = [6,8],

(9) $[Q_l, Q_u]$ = [4,5] and $[P_l, P_u]$ = [8,10].

In our experiment, we tested each of the nine cases against each of the four similarity requirements. The four similarity requirements are listed in the first column of Table 2. For each case and each similarity requirement, we randomly generated 100 query pictures and calculated the average number of database pictures accessed per query. The data entry shown in Table 2 is the average number of database pictures accessed per query divided by the total number of database pictures. It can be seen that the percentages are very low for all test cases. For example, in the case of $[Q_l, Q_u] = [2, 3]$ and $[P_l, P_u] = [8, 10]$, the percentage of the average number of database pictures accessed per query with similarity requirement $(S_D, S_T) = (0, 0)$ is 5.8012%. Moreover, similarity requirement (S_D, S_T) = (0,0) represents the fuzziest measure while similarity requirement (S_D, S_T) = (1,1) represents the most precise measure. The average percentages of images accessed per query for all nine cases from $(S_D, S_T) = (0, 0)$ to $(S_D, S_T) = (1, 1)$ are 1.6829%, 0.1337%, 1.4915%, and 0.1254%, respectively. As we can see, only a very small portion of database images are accessed per query based on our three level index structure, the efficiency of similarity retrieval is well demonstrated.

6. CONCLUSIONS

Content-based image retrieval (CBIR) based on visual features of images is the trend of designing a modern image database system. In the past, symbolic pictures are used to approximate segmented images containing recognized objects. Searching for desired images based on symbolic pictures is called "iconic indexing." The most important feature of a symbolic picture is probably the binary spatial relationships between objects. Thus, an appropriate knowledge representation for spatial relations plays an important role in designing a CBIR system.

A novel spatial knowledge representation called 9D-SPA has been presented in this paper to capture the information about the spatial relationships between objects in a segmented picture. The capability of representing a picture using 9D-SPA representation is more powerful than any other representation schemes based on MBR or centroids of objects. The flexibility of similarity measures are provided to support fuzzy matching that is highly desired to meet user's different requirements. More importantly, the 9D-SPA representation can be easily incorporated into a three-level index structure to allow us to access the desired pictures very efficiently. The experimental results have demonstrated that only a very small portion of the image database is accessed per query transaction by using our three-level index structure based on the 9D-SPA representation.

References

[1] J.R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. C. Jain, and C.Shu. "Virage Image Search Engine: an Open Framework for Image Management," *Proc. Symposium on Electronic Imaging: Science and Technology - Storage & Retrieval for Image and Video Database IV*, IS&T/SPIE, pp. 76-87, 1996.

[2] B. Bhanu, S. Lee, *Genetic Learning for Adaptive Image Segmentation*, Kluwer Academic, Norwell, 1994.

[3] C.C. Chang, "Spatial Match Retrieval of Symbolic Pictures," *Journal of Information Science and Engineering*, vol. 7, pp. 405-422, Dec. 1991.

[4] S.K. Chang, E. Jungert and Y. Li,*Representation and Retrieval of Symbolic Pictures Using Generalized 2D Strings,* tech. report, University of Pittsburg, 1988.

[5] S.K. Chang, Q.Y. Shi, and C.W. Yan, "Iconic Indexing by 2-D Strings," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 9, no. 3, pp.413-428, May 1987.

[6] S.K. Chang, *Principles of Pictorial Information Systems Design*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1989.

[7] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Streele, and P. Yanker, "Query by Image and Video Content: The QBIC System," *IEEE Computer*, vol. 28, issue 9, pp. 23-32, Sept. 1995.

[8] P.W. Huang and Y.R. Jean, "Using 2D C^+-Strings as Spatial Knowledge Representation for Image Database Systems," *Pattern Recognition*, vol. 27, no. 9, pp. 1249-1257, Sept. 1994.

[9] P.W. Huang and Y.R. Jean, "Design of Large Intelligent Image Database Systems," *International Journal of Intelligent Systems*, vol. 11, pp. 347-365, 1996.

[10] E. Jungert, "Extended Symbolic Projections as A Knowledge Structure for Spatial Reasoning," *Proc. 4th BPRA Conf. on Pattern Recognition,* Springer, Cambridge, pp. 343-351, 1988.

[11] E. Jungert and S.K. Chang, "An Algebra for Symbolic Image Manipulation and Transformation," *Visual Database Systems,* T.L. Kunii, ed., Elsevier Science Publishers B.V.,North-Holland, 1989.

[12] S.Y. Lee and F.J. Hsu, "2D C-String: A New Spatial Knowledge Representation for Image Database Systems," *Pattern Recognition*, vol. 23, no. 10, pp. 1077-1087, Oct. 1990.

[13] S.Y. Lee and F.J. Hsu, "Spatial Reasoning and Similarity Retrieval of Images using 2D C-String Knowledge Representation," *Pattern Recognition*, vol. 25, no. 3, pp. 305-318, Mar. 1992.

[14] K.C. Liang and C.C. Jay Kuo, "WaveGuide: A Joint Wavelet-Based Image Representation and Description System," *IEEE Trans. Image Processing*, vol. 8, no. 11, pp. 1619-1629, 1999.

[15] M. Nabil, J. Shepherd, and A.H.H. Ngu, "2D Projection Interval Relationships: A Symbolic Representation of Spatial Relationships," *Lecture Notes in Computer Science*, no. 951, pp. 292-309, 1995.

[16] M. Nabil, A.H.H. Ngu, and J. Shepherd, "Picture Similarity Retrieval Using the 2D Projection Interval Representation," *IEEE Trans. on Knowledge and Data Engineering*, vol. 8, no. 4, pp. 533-539, Aug. 1996.

[17] A. Pentland, R. W. Picard, and S. Sclaroff, "Photobook: Tool for Content-Based Manipulation of Image Databases," *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233-254, June 1996.

[18] Gennaro Petraglia, Monica Sebillo, Maurizio Tucci, and Genoveffa Tortora, "Virtual Images for Similarity Retrieval in Image Databases," *IEEE Trans. on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 951-967, Nov. 2001.

[19] Y. Rui and T.S. Huang, "Image Retrieval: Current Techniques, Promising Directions, and Open Issues," *J. Visual Commun. Image Representation*, vol. 10, pp. 39-62, 1999.

[20] J.R. Smith and S.F. Chang, "VisualSEEK: A Full Automated Content-Based Image Query System," *Proceedings of the Fourth ACM International Multimedia Conference*, pp. 87-98, 1996.

[21] X. M. Zhou and C. H. Ang, "Retrieving Similar Pictures from a Pictorial Database by an Improved Hashing Table," *Pattern Recognition Letters*, vol. 18, pp. 751-758, 1997.

[22] http://www.annapolistech.com/retrieva.htm

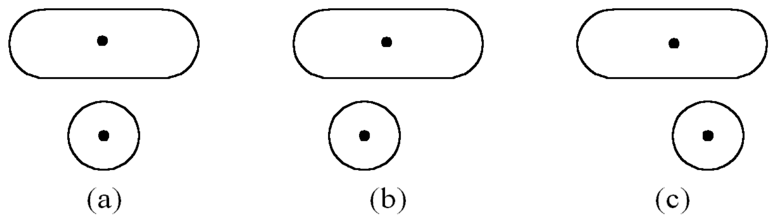

Figure 1: Spatial reasoning is too sensitive in the 9DLT representation

Table 1: The codes for nine neighborhood areas of MBR of O_j

Area 4: $(00001000)_2 = 8$	Area 3: $(00000100)_2 = 4$	Area 2: $(00000010)_2$=2
Area 5: $(00010000)_2 = 16$	MBR of O_j	Area 1: $(00000001)_2 = 1$
Area 6: $(00100000)_2 = 32$	Area 7: $(01000000)_2 = 64$	Area 8: $(10000000)_2 = 128$

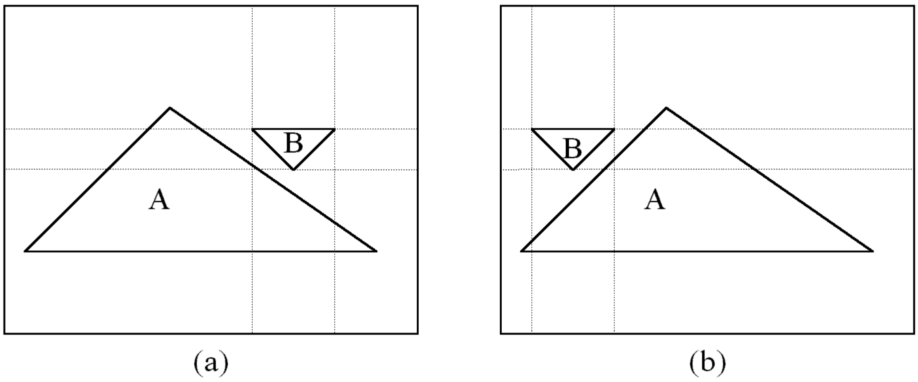

Figure 2: Pictures (a) and (b) are not distinguishable in all 2D *-string representations. However, the difference can be easily determined by the 9D-SPA representation.

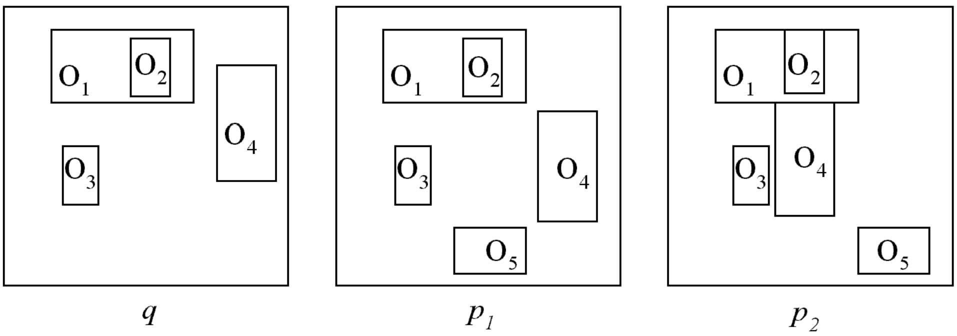

Figure 3: A similarity measurement example

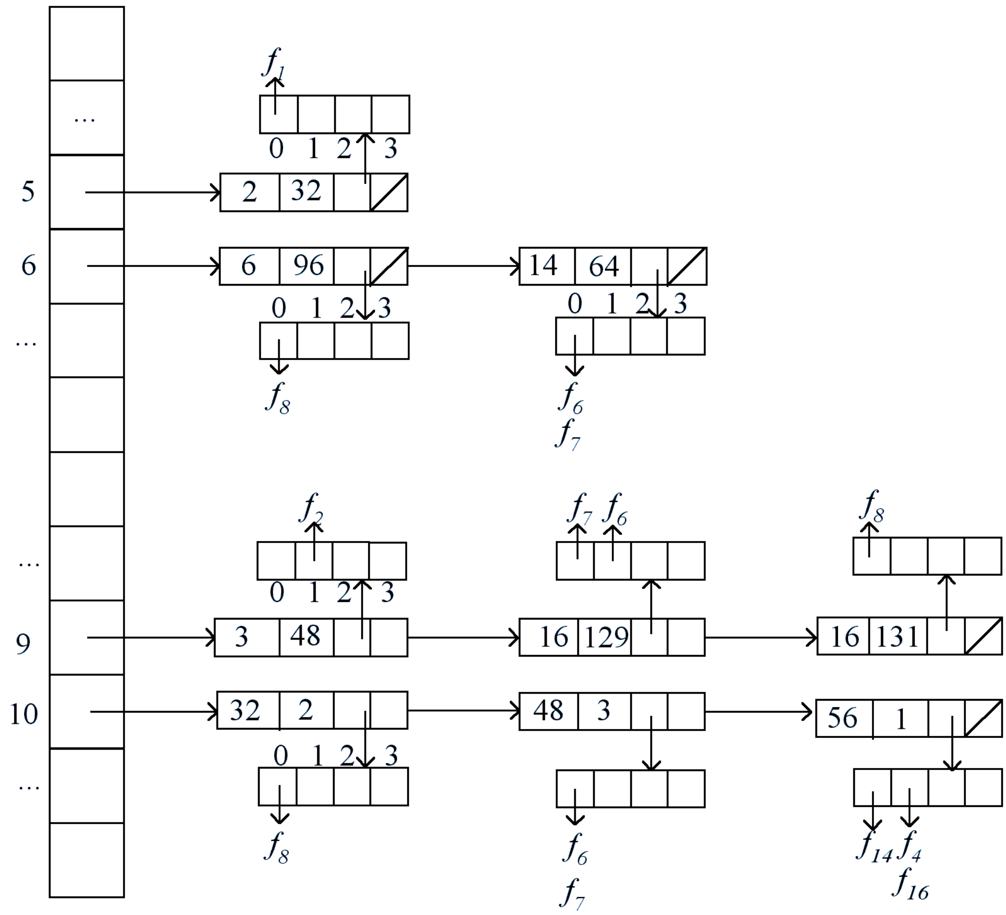

Figure 4: An example of index structure for a pictorial database

Table 2: The percentage of average number of pictures accessed per query(%)

	$[Q_l, Q_u]$=[2,3]			$[Q_l, Q_u]$=[3,4]			$[Q_l, Q_u]$=[4,5]			
	$[P_l, P_u] =$			$[P_l, P_u] =$			$[P_l, P_u] =$			
(S_D, D_T)	[4,6]	[6,8]	[8,10]	[4,6]	[6,8]	[8,10]	[4,6]	[6,8]	[8,10]	Avg.
(0,0)	1.5569	3.1692	5.8012	.2494	.9182	2.0102	.0849	.3000	1.0564	1.6829
(1,0)	.1861	.3971	.5974	.0017	.0078	.0131	0	0	.0004	.1337
(0,1)	1.4210	2.9252	5.3409	.1860	.8175	1.7330	.0678	.2183	.7138	1.4915
(1,1)	.1743	.3745	.5593	.0013	.0070	.0119	0	0	.0002	.1254