Available online at www.sciencedirect.com
ScienceDirect
Journal of Visual Languages and Computing 19 (2008) 637–651
Journal of Visual Languages & Computing
www.elsevier.com/locate/jvlc

Spatial inference and similarity retrieval of an intelligent image database system based on object's spanning representation

Po-Whei Huang[a,*], Lipin Hsu[b], Yan-Wei Su[a], Phen-Lan Lin[c]

[a]*Department of Computer Science, National Chung Hsing University, Taichung, Taiwan*
[b]*Department of Applied Mathematics, National Chung Hsing University, Taichung, Taiwan*
[c]*Department of Computer Science and Information Management, Providence University, Salu, Taiwan*

Received 12 October 2006; received in revised form 7 August 2007; accepted 25 September 2007

Abstract

In this paper, we presented a novel image representation method to capture the information about spatial relationships between objects in a picture. Our method is more powerful than all other previous methods in terms of accuracy, flexibility, and capability of discriminating pictures. In addition, our method also provides different degrees of granularity for reasoning about directional relations in both 8- and 16-direction reference frames. In similarity retrieval, our system provides twelve types of similarity measures to support flexible matching between the query picture and the database pictures. By exercising a database containing 3600 pictures, we successfully demonstrated the effectiveness of our image retrieval system. Experiment result showed that 97.8% precision rate can be achieved while maintaining 62.5% recall rate; and 97.9% recall rate can be achieved while maintaining 51.7% precision rate. On an average, 86.1% precision rate and 81.2% recall rate can be achieved simultaneously if the threshold is set to 0.5 or 0.6. This performance is considered to be very good as an information retrieval system.

Keywords: Image database; Image representation; Spatial inference; Similarity retrieval

1. Introduction

Pictorial databases have been used more widely in recent years. In applications such as computer-aided design, office automation, medical image archiving, geographic information, and trademark picture registration, retrieving pictures from pictorial databases is frequently required. Content-based image retrieval is the current trend of designing image database systems as opposed to text-based image retrieval [1–9]. Rather than proceeding via a manually generated text-based description, the content-based image retrieval works by matching the query image against a database image according to the contents of images.

The features used in content-based image retrieval can be roughly divided into two categories: the low-level visual features such as color, shape, and texture and the high-level features such as spatial relationships among the objects in a picture. Examples of content-based image retrieval systems include QBIC [10], VisualSEEK [11], and WaveGuide [12], etc. They allow users to retrieve similar

*Corresponding author. Tel.: +886 4 2235 0813.
E-mail address: pwhuang@cs.nchu.edu.tw (P.-W. Huang).

1045-926X/$ - see front matter
doi:10.1016/j.jvlc.2007.09.001

pictures from a large database based on low-level visual features.

Retrieving pictures that satisfy high-level spatial queries is also an important issue in image database systems [13–20]. For example, to answer the query "find all pictures having a swimming pool to the left of a house", we need to keep at least the directional relationship between the swimming pool and the house for all pictures.

In this paper, we present a spatial knowledge representation method for pictures containing nonzero-sized objects. The spatial relations we considered are both topological and directional relations between nonzero-sized objects. A set of similarity measures for handling different types of queries that will satisfy various kinds of users' requirements is also proposed. We use some examples to demonstrate that our image representation method is more powerful than many well-known methods. In particular, we built a database of 3600 pictures, with each picture containing four to six objects, to demonstrate the performance of our system in terms of recall and precision. Experiment results show that, in the two extreme cases, 97.8% precision rate can be achieved with 62.5% recall rate, and 97.9% recall rate can be achieved with 51.7% precision rate. On an average, a very good performance of 86.1% precision rate and 81.2% recall rate can be achieved simultaneously with threshold $T_h = 0.5$ or 0.6.

2. Overview

Spatial relationships between objects have been identified as one of the most important features for describing the contents of images. Unlike topological relations which are well defined and less disputable, directional relations can be viewed and modeled in different ways. Previous methods for modeling directional relations include direction between Minimum Bounding Rectangles (MBRs) [13,18–20], 2D-strings [16,17], 2D-PIR [22], triangular model [23], and symbolic arrays [24]. A detailed analysis for comparing these methods can be found in Ref. [24]. Except the above methods, two other direction models, direction–relation matrix [21] and 9D-SPA [14], based on rectangle-shaped partition were also proposed. They partition the whole space around a reference object and record into which direction tiles an target object falls. They can provide better approximations for spatial relations between objects with complex structures including shapes such as concave regions or objects with holes. The 9D-SPA method can even support an efficient indexing structure to facilitate search in similarity retrieval. In this paper, we only concentrate on iconic picture retrieval based on spatial relations where the set of icons are known and each object in a picture must match an icon [25].

Different from 9D-SPA which uses rectangle-shaped partition, this paper presents an image representation method for capturing the relations among nonzero-sized objects based on a triangular partition approach. Triangular partition is more consistent with human perception and is better than rectangle-shaped partition from an observer's point of view, if the observer is placed at the position where the reference object is located. Such a phenomenon is clearly demonstrated in Fig. 1, where object A is to the east of object B in triangular partition while object A is to the northeast of object B in rectangle-shaped partition. However, directions based on triangular partition approach may still be misleading and not suitable for direction queries in spatial databases if objects are overlapping, intertwined, or horseshoe-shaped. This paper improves

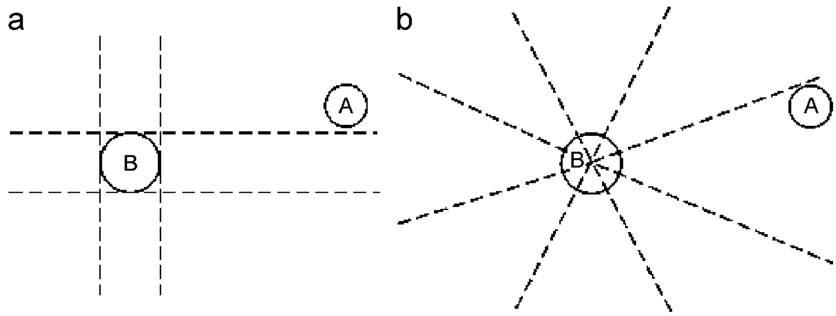

Fig. 1. A is the target object and B is the reference object. (a) Rectangle-shaped partition: A is to the northeast of B. (b) Classic triangular partition: A is to the east of B.

the previous triangular model to overcome the above problems.

3. Image representation

To represent a picture using our method, the picture has to be preprocessed first. 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 [26].

Once the objects of a picture are segmented and recognized, the next step is to define the directional relation between any two objects. Each directional relation involves one reference object and one target object. For each directional relation, the whole space is divided into eight non-overlapping neighborhood areas with respect to the reference object. These eight neighborhood areas are namely east, northeast, north, northwest, west, southwest, south, southeast; and each area is assigned an integer from 1 to 8, respectively. The areas in which the target object is located indicate the directional relation of the target object with respect to the reference object.

There are two different types for partitioning a space.

- Type-1 partition: there is no overlapping between the MBR of the reference object and the target object as shown in Fig. 2(a). Given a reference object, the procedure for segmenting the whole space into eight direction areas is as follows:
  (1) Find the MBR M of the reference object.
  (2) Find the center of M and treat it as the origin of a Cartesian coordinate system.
  (3) For each side s of M
  If s is parallel to the y-axis (x-axis), draw an angle A such that the following three conditions are satisfied
    (i) The vertex of angle A is on the x-axis (y-axis).
    (ii) Each side of angle A must go through the respective terminal point of s.
    (iii) The size of angle A is 45°.
- Type-2 partition: There is some overlapping between the reference object's MBR and the target object as shown in Fig. 2(b). The space is equally divided into eight areas by the partition lines emitting from the centroid of the reference object.

No matter which type is used to partition a space, a direction area can be further divided into two sections, called the *first* and the *second section* in counterclockwise order, by the middle line of each area.

Type-1 partition modifies the classic triangular model by taking sizes and orientations of both objects (the reference and the target) into account. It is essential to consider sizes and orientations of objects when determining the binary directional relationship between them. For example, in Fig. 3(a), the classic triangular partition method (or type-2 partition) uses the centroid of reference object B as the center for partitioning and obtains the result "object A is to the northeast of object B."

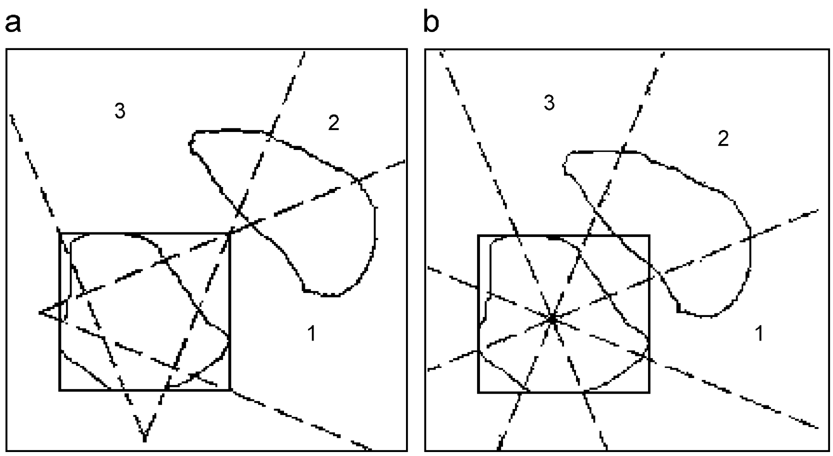

Fig. 2. (a) Type-1 space partitioning, (b) Type-2 space partitioning.

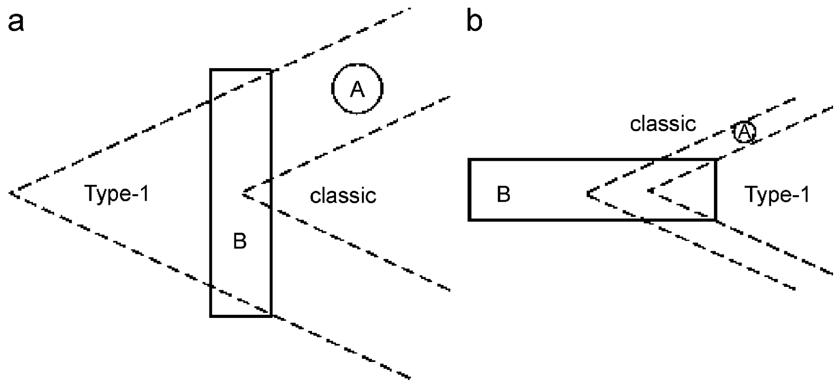

Fig. 3. Our type-1 partition method is better than the classic triangular partition method. (a) "Object A is to the east of object B" is more correct. (b) "Object A is to the northeast of object B" is more correct.

However, "A is to the east of B" can be obtained if type-1 partition is applied and this result is more consistent with human perception. On the other hand, "A is northeast of B" is obtained by the type-1 partition method while "A is to the east of B" is obtained by the classic triangular model as shown in Fig. 3(b). From this example, we can easily see that type-1 partition method is better than the classic triangular partition method because the former considers the size and orientation of the reference object.

In type-1 partition, the "area of acceptance" for any direction with respect to a reference object is defined as the area inside the partition triangle of that direction and outside the MBR of the reference object. If a target object overlaps with the MBR of a reference object, the area of acceptance for any direction in relation to the reference object mentioned above is inappropriate and type-1 partition becomes not applicable. Thus, type-2 partition (i.e. the classic triangular partition method) will be applied in this case. Indeed, if two objects are close to each other, size and orientation of the reference object become unimportant for determining the directional relationship among objects.

Now we are ready to define the representation of a picture. Suppose that a picture P contains n objects (O_1, O_2, …, O_n). Then, the image representation of P can be encoded as a set of 4-tuples: $R = \{(O_{ij}, T_{ij}, D_{ij}, D_{ji}) |\ \forall O_i, O_j \in P$, and $1 \leqslant i < j \leqslant n\}$, where O_{ij} is the code of object-pair (O_i, O_j); T_{ij} is the code for the topological relation between O_i and O_j while D_{ij} is the code for the directional relation between objects O_i and O_j with O_j as the reference object; D_{ji} is the code for the directional relation between objects O_i and O_j with O_i as the reference object. It is obvious that the total number of 4-tuples in R is $\frac{n(n-1)}{2}$.

Let O_i be the ith object in the image database ($1 \leqslant i \leqslant N$). 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 proposed in [14]:

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

To obtain the two object numbers i and j from O_{ij} (or to decode O_{ij}), we use the formula $i = O_{ij} - \frac{a(a+1)}{2}$, where a is the largest non-negative integer such that $\frac{a(a+1)}{2} < O_{ij}$ and $j = a+2$.

T_{ij} indicates the topological relationship between objects O_i and O_j with O_j as the reference object. Possible values assigned to T_{ij} are: 0 (stands for "disjoint"), 1 (stands for "meet"), 2 (stands for "overlap"), 3 (stands for "cover"), 4 (stands for "covered-by"), 5 (stands for "contain"), 6 (stands for "inside"), 7 (stands for "equal"). The precise definitions for the above topological relations are based on the four-intersection model which can be found in ref. [27].

D_{ij} represents the directional relation between target object O_i and reference object O_j. The directional relation D_{ij} is a 5-tuple (C_{ij}, f_{ij}, b_{ij}, s_{ij}, e_{ij}), where

$C_{ij} \in \{k | 1 \leqslant k \leqslant 8\}$ indicates in which direction area the centroid of object O_i is located with O_j as the reference object.

f_{ij} (or b_{ij}) indicates the number of additional direction areas that object O_i spans forward in counterclockwise direction (or backward in clockwise direction) starting from the direction area in which the centroid of O_j is located. It is obvious that $0 \leqslant f_{ij}, b_{ij} \leqslant 8$;

s_{ij} (or e_{ij}) indicates the start-point (or end-point) information about target object O_i. Let the centroid of the reference object be the origin of a Cartesian coordinate system. Imagine that a radial segment moves counterclockwise from the positive x-direction (or zero-degree direction) until the first tangent point to the target object is met and this point is called the start-point of the target object. The second point at which the radial segment is tangent to the target object is called the end-point of the target object. The information about s_{ij} (e_{ij}) related to the start-point (end-point) of a target object is interpreted as follows (see Fig. 4):

(a) In the case of $f_{ij} = b_{ij} = 0$:
If $s_{ij} = e_{ij} = 0$, then the whole object O_i is located in the first section of the direction area of object O_j.
If $s_{ij} = e_{ij} = 1$, then the whole object O_i is located in the second section of the direction area of object O_j.
If $s_{ij} = 0$ and $e_{ij} = 1$, then object O_i is located over the middle line of the direction area of object O_j.

(b) In the case of $b_{ij} \neq 0$ or $f_{ij} \neq 0$:
If $s_{ij} = 1$, then object O_i spans over the middle line of the direction area in which the start-point of O_i is located; otherwise, object O_i does not span over the middle line of the direction area in which the start-point of O_i is located.
If $e_{ij} = 1$, then object O_i spans over the middle line of the direction area in which the end-point of O_i is located; otherwise, object O_i does not span over the middle line of the direction area in which the end-point of O_i is located.

Notice that middle lines are needed only in the direction area that a target object O_i begins and in the direction area that the same object O_i ends in counterclockwise direction.

Let us look at the example shown in Fig. 5. Assume that object B is the reference object. Since the MBR of object B overlaps with object A, we divide the space using type-2 partition. In Fig. 5(a), the centroid of A is located in the "east" direction area of object B, therefore, $C_{AB} = 1$. Since the whole object A falls in the second section of the "east" direction area of B, we have $f_{AB} = b_{AB} = 0$ and $s_{AB} = e_{AB} = 1$. Consequently, the 5-tuple for D_{AB} is (1, 0, 0, 1, 1). In Fig. 5(b), the centroid of A is located in the "west" direction area of object B, therefore, $C_{AB} = 5$. Since the whole object A falls in the first section of the "west" direction area of B, we have $f_{AB} = b_{AB} = 0$ and $s_{AB} = e_{AB} = 0$. As a result, the 5-tuple for D_{AB} is (5, 0, 0, 0, 0). Using our representation, we can easily tell the difference between the image in Fig. 5(a) and the image in Fig. 5(b). In particular, we can easily see that A is to the east of B in Fig. 5(a) because $C_{AB} = 1$ while A is to the west of B in Fig. 5(b) because $C_{AB} = 5$. In 2D*-string representations, there is no way to distinguish these two images because they have the

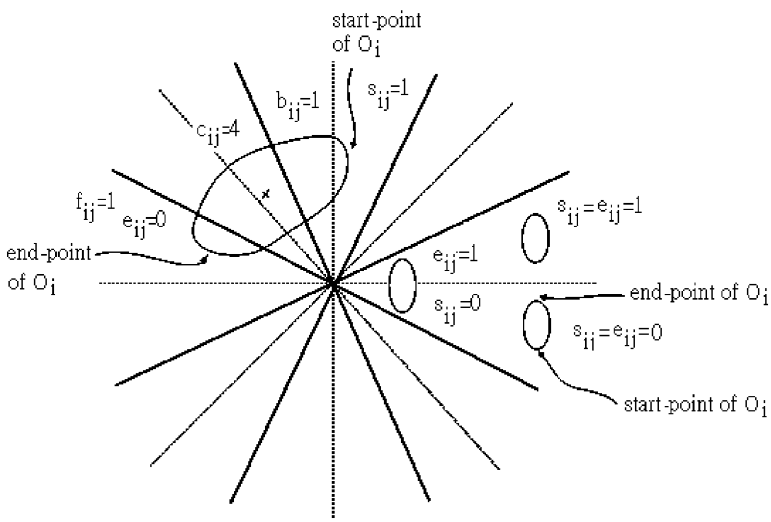

Fig. 4. An example for interpreting f_{ij}, b_{ij}, s_{ij}, and e_{ij}.

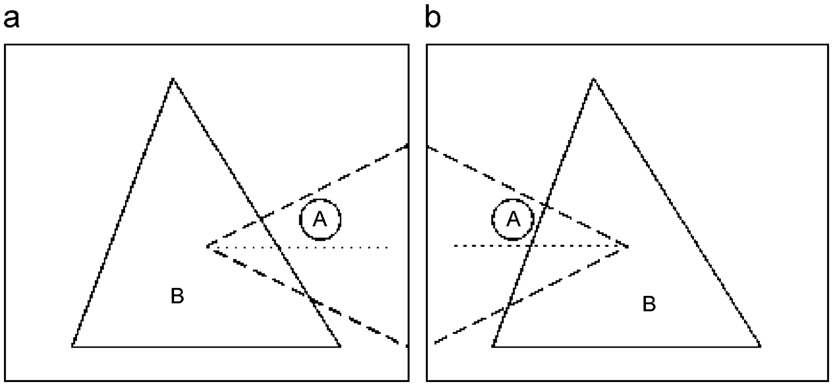

Fig. 5. Pictures (a) and (b) are not distinguishable in all 2D*-string representations. It is also impossible to infer the directional relation between objects A and B.

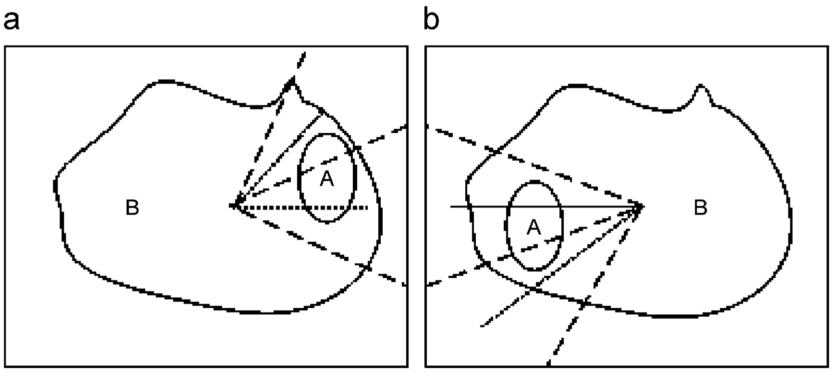

Fig. 6. Pictures (a) and (b) are not distinguishable in all 2D*-string representations, 2D-PIR, and 9D-SPA representation. However, the difference can be easily identified by our proposed method.

same spatial image representation: ($B\%A$, $B\%A$). It is also impossible to infer the directional relation between objects A and B.

As a second example, let us look at the pictures shown in Figs. 6(a) and (b). For all image representations based on topological relations, there is no way to differentiate these two pictures because the only information we can get is "object A is contained in object B" or "object B contains object A." In all 2D*-string representations [13,16–20], both pictures have the same representation ($B\%A$, $B\%A$) based on the projection interval relation along the x- and y-axis between A and B. Since 2D-PIR representation [22] combines topological relations with the idea of projection interval relations, it still has the disadvantage of being not able to distinguish pictures such as the ones shown in Fig. 6(a) and (b). In 9D-SPA representation, although it has many powerful functions, the directional relations between objects A and B are also the same in both pictures (i.e. $D_{AB} = 00000000$ and $D_{BA} = 11111111$), therefore, it still can not distinguish Fig. 6(a) from Fig. 6(b). However, in our representation method, D_{AB} is (1, 1, 0, 1, 0) in Fig. 6(a) and (5, 1, 0, 1, 0) in Fig. 6(b). It is obvious that these two pictures are easily discriminated because image representations are different. Moreover, we have $T_{AB} = 5$ in both pictures, therefore, they all imply that "object B contains object A." In Fig. 6(a), we may conclude that A is to the east of centroid of B and its size spans from the east direction area to the northeast direction area with respect to B because $C_{AB} = 1$, $f_{AB} = 1$, $b_{AB} = 0$. In Fig. 6(b), we may conclude that A is to the west of the centroid of B and its size spans from the west direction area to the southwest direction area with respect to B because $C_{AB} = 5$, $f_{AB} = 1$, $b_{AB} = 0$. If we look at more details in the representation, we can easily find that the size of object A spans over the middle line of the east direction area (because $s_{AB} = 1$), however, does not span over the middle line of the northeast direction area (because

$e_{AB} = 0$) with respect to the centroid of object B as shown in Fig. 6(a). Similarly, from the representation for Fig. 6(b), we can easily conclude that the size of object A spans over the middle line of the west direction area (because $s_{AB} = 1$); however, does not span over the middle line of the southwest direction area (because $e_{AB} = 0$) with respect to the centroid of object B. This example clearly reveals the fact that our method supports not only similarity retrieval but also the capability of spatial reasoning to make the system become cleverer as desired by the intelligent image database systems.

The direction areas of the above two examples are defined by type-2 partition because the target object overlaps with the MBR of the reference object in the pictures shown in Figs. 5 and 6. Let us look at the third example where the MBR of object B is disjoint from object A and type-1 partition is used to define the direction areas as shown in Fig. 7. The two pictures in Figs. 7(a) and (b) are exactly the same except the orientation of object A. In Fig. 7(a), the centroid of object A is located at the north direction area of object B, so $C_{AB} = 3$. Since object A spans over two extra direction areas (the northeast and the east) in backward (clockwise) direction and no extra direction area in forward (counterclockwise) direction, so $b_{AB} = 2$ and $f_{AB} = 0$. Moreover, because the start-point of object A does not cross over the middle line of the east direction area and its end-point does not cross over the middle line of the north direction area neither, so $s_{AB} = 0$ and $e_{AB} = 0$. Therefore, $D_{AB} = (3, 0, 2, 0, 0)$ for Fig. 7(a). Similarly, in Fig. 7(b), the centroid of object A is located at the northeast direction area of object B, so $C_{AB} = 2$. Since object A spans over one extra direction area (the east) in backward (clockwise) direction and one extra direction area (the north) in forward (counterclockwise) direction, so $b_{AB} = 1$ and $f_{AB} = 1$. Moreover, because the start-point of object A does not cross over the middle line of the east direction area and its end-point does not cross over the middle line of the north direction area neither, so $s_{AB} = 0$ and $e_{AB} = 0$. Therefore, $D_{AB} = (2, 1, 1, 0, 0)$ for Fig. 7(b). As a result, the 5-tuple D_{AB} in Fig. 7(a) is (3, 0, 2, 0, 0) which is different from the 5-tuple $D_{AB} = (2, 1, 1, 0, 0)$ in Fig. 7(b). This example demonstrates that our method can even discriminate pictures containing objects with the same locations but in different orientations.

4. Spatial Inference on multiple reference frames

As mentioned before, a picture P with n objects $(O_1, O_2, \ldots, O_n)$ can be transformed into a set of 4-tuples $R = \{(O_{ij}, T_{ij}, D_{ij}, D_{ji}) | \forall O_i, O_j \in P$, and $1 \leqslant i < j \leqslant n\}$. Given a 4-tuple $(O_{ij}, T_{ij}, D_{ij}, D_{ji})$ in R, we can easily find the topological relation between O_i and O_j through T_{ij} as well as the directional relation between O_i and O_j through D_{ij} and D_{ji}. One of the advantages of our approach is that we can use the same representation to interpret the relationships between two objects on different reference frames. For example, we may need finer granularity for describing a directional relation such as "A is to the north of northeast of B" which requires to partition the space into 16 direction areas instead of 8 with respect to the reference object. By cutting each one of the 8 direction areas into equal halves, we can easily obtain the 16 direction areas as shown in Fig. 8. In other words, the user needs an 8-direction reference frame for coarse spatial

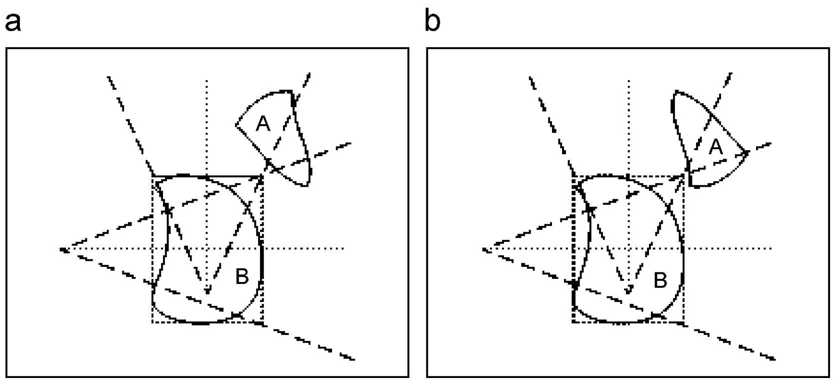

Fig. 7. Pictures (a) and (b) are not distinguishable in all 2D*-string representations or any direction model based on rectangle-shape partitioning. However, the difference can be easily determined by our image representation method.

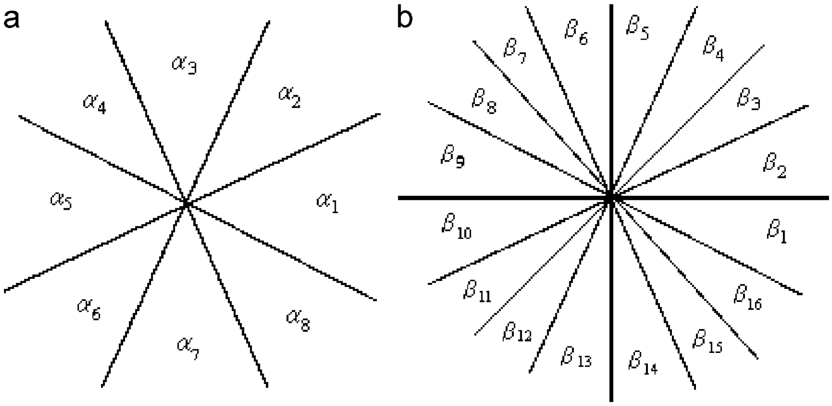

Fig. 8. (a) The reference frame with eight direction areas. (b) The reference frame with sixteen direction areas.

Table 1
Interpretation of direction areas in 8-direction/16-direction reference frames

8-direction reference frame		16-direction reference frame	
α_1	East	β_1	South of east
		β_2	North of east
α_2	Northeast	β_3	East of northeast
		β_4	North of northeast
α_3	North	β_5	East of north
		β_6	West of north
α_4	Northwest	β_7	North of northwest
		β_8	West of northwest
α_5	West	β_9	North of west
		β_{10}	South of west
α_6	Southwest	β_{11}	West of southwest
		β_{12}	South of southwest
α_7	South	β_{13}	West of south
		β_{14}	East of south
α_8	Southeast	β_{15}	South of southeast
		β_{16}	East of southeast

reasoning while requires a 16-direction reference frame for finer spatial reasoning.

The relationship between the 8-direction reference frame and the 16-direction reference frame, as well as the interpretation for all direction areas in these two reference frames is shown in Table 1. In our system, we can transform the reference frame of 8 direction areas into the reference frame of sixteen direction areas very easily. The converting method is as follows.

Given $D_{ij} = (C_{ij}, f_{ij}, b_{ij}, s_{ij}, e_{ij})$, we assume that target object O_i spans from the starting direction area b to the ending direction area e with respect to reference object O_j in the eight-direction reference frame. Let b' and e' be the starting and ending direction areas, respectively, for target object O_i in the 16-direction reference frame. Then, b and e can be calculated by the following equations:

Table 2
Look-up table for δ_1 and δ_2

	(s_{ij}, e_{ij})	(δ_1, δ_2)
$f_{ij} = b_{ij} = 0$	(0, 0)	(−1, −1)
	(0, 1)	(−1, 0)
	(1, 0)	–
	(1, 1)	(0, 0)
$f_{ij} \neq 0$ or $b_{ij} \neq 0$	(0, 0)	(0, −1)
	(0, 1)	(0, 0)
	(1, 0)	(−1, −1)
	(1, 1)	(−1, 0)

$$b = \begin{cases} C_{ij} - b_{ij} + 8 & \text{if } C_{ij} - b_{ij} \leqslant 0, \\ C_{ij} - b_{ij} & \text{otherwise,} \end{cases}$$

$$e = \begin{cases} C_{ij} + f_{ij} - 8 & \text{if } C_{ij} + f_{ij} > 8, \\ C_{ij} + f_{ij} & \text{otherwise,} \end{cases} \quad (1)$$

In the 16-direction reference frame, b' and e' can be easily obtained from the corresponding b and e as follows:

$$b' = b * 2 + \delta_1,$$
$$e' = e * 2 + \delta_2, \quad (2)$$

where δ_1 and δ_2 can be determined by Table 2 without extra computation.

To illustrate the above concept, let's look at the picture shown in Fig. 5(a) again. We have $D_{AB} = (1, 0, 0, 1, 1)$ as explained in Section 3. From formula (1), we can get $b = 1$ and $e = 1$. Therefore, we say that object A is located in the α_1 direction area of object B in the 8-direction reference frame (or object A is to the east direction area of object B). From formula (2) and Table 2, we can get $b' = 2$ and $e' = 2$ in the 16-direction reference frame. So we say that object A is located in the β_2 direction area with respect to object B in the 16-direction reference frame (or object A is in the second section of the east direction area of object B) from a more detailed observer's point of view. For another example, let's see the picture shown in Fig. 6(a). As already explained in Section 3, we have $D_{AB} = (1, 1, 0, 1, 0)$. Therefore, we can get $b = 1$ and $e = 2$ by using formula (1). Since the code for T_{AB} is 6, so we say that object A is inside of object B and spans two direction areas α_1 and α_2 (the first and second sections of east direction area) in the 8-direction reference frame. On the other hand, in the 16-direction reference frame, we get $b' = 1$ and $e' = 3$ by using formula (2). As a result, we may also say that object A spans three direction areas β_1 (the first section of east direction area), β_2 (the second section of east direction area), and β_3 (the first section of NE direction area) in the 16-direction reference frame.

5. Similarity retrieval

In similarity retrieval, the user usually submits a sketch picture to the system for searching similar pictures in the database and this method is called the "query-by-pictorial-example" approach [25,28]. Since the user may not remember the exact spatial relationships among the objects in a desired picture, the system should provide the user with a set of coarse-to-fine similarity measures to flexibly evaluate the difference between the query picture and the database pictures for satisfying user's different requirements.

There are 12 types of similarity measures provided by our system as shown in Table 3 and their semantics are described below:

- Type-00: matching is performed based on IDs of objects.
- Type-10: matching is performed based on topological relations between objects.
- Type-20: matching is performed based on directional relations between centroids of objects.
- Type-30: matching is performed based on both topological relations between objects and directional relations between centroids of objects.
- Type-01: matching is performed based on type-00 criterion and the spanning ranges (in terms of direction areas) of all other objects viewed from every object.
- Type-11: matching is performed based on type-10 criterion and the spanning ranges of all other objects viewed from every object.
- Type-21: matching is performed based on type-20 criterion and the spanning ranges of all other objects viewed from every object.
- Type-31: matching is performed based on type-30 criterion and the spanning ranges of all other objects viewed from every object.
- Type-02: matching is performed based on type-01 criterion and the condition whether the start-point and the end-point of each target object cross through the middle line of the direction area in which start-point (or end-point) is located.
- Type-12: matching is performed based on type-11 criterion and the second condition specified in type-02.
- Type-22: matching is performed based on type-21 criterion and the second condition specified in type-02.
- Type-32: matching is performed based on type-31 criterion and the second condition specified in type-02.

Table 3
The 12 types of similarity measures

Type-uv	$v = 0$, without considering object's span	$v = 1$, consider object's span using f_{ij}, b_{ij}	$v = 2$, consider object's span using f_{ij}, b_{ij}, s_{ij}, e_{ij}
$u = 0$, without considering any spatial relation	Type-00	Type-01	Type-02
$u = 1$, consider topological relation between objects (T_{ij})	Type-10	Type-11	Type-12
$u = 2$, consider direction relation between centroids of objects (C_{ij})	Type-20	Type-21	Type-22
$u = 3$, consider both T_{ij} and C_{ij}	Type-30	Type-31	Type-32

For convenience of explanation, let us introduce the following notations before giving the precise definitions for similarity measures.

- R_p (or R_q): the image representation for picture p (or q).
- W_p (or W_q): the set of objects in picture p (or q).
- t^p (or t^q): a tuple in R_p (or R_q).
- $t.C_1$: the centroid's location-code, with O_j as the reference object, of tuple t.
- $t.C_2$: the centroid's location-code, with O_i as the reference object, of tuple t.
- $t.D_1$: the directional relation-code, with O_j as the reference object, of tuple t.
- $t.D_2$: the directional relation-code, with O_i as the reference object, of tuple t.
- $t.T$: the topological relation-code of tuple t.
- $a(k)$ (or $b(k)$): the kth bit of the binary code a (or b).

Let $R_q = \{t^{q1}, t^{q2}, ..., t^{qm}\}$ and $R_p = \{t^{p1}, t^{p2}, ..., t^{pn}\}$ be the image representations for query picture q and database picture p, respectively. Let K be an ordered set to record the same code for object-pair in these two image representations in an ascending order; $|K|$ is the number of elements in K; $K(i)$ is the ith object-pair code in K. We define the object-similarity measure between R_p and R_q as

$$S_O(p,q) = \frac{|W_p \cap W_q|}{|W_p \cup W_q|}. \tag{3}$$

For defining the directional-similarity measure, we use a transformation function

$f_r : t.D \to a_{r-1}a_{r-2}\ldots a_1a_0$,

to transform D_{ij} (or D_{ji}) to a binary code $a_{r-1}a_{r-2}\ldots a_1a_0$ with $r = 8$ or 16. Then the directional-similarity measure between R_p and R_q is defined as

where

$$S'_{Dr}(a,b) = \frac{\sum_{k=0}^{r-1} a(k) \wedge b(k)}{\sum_{k=0}^{r-1} a(k) \vee b(k)}. \tag{5}$$

Similarly, the topological-similarity measure between R_p and R_q is defined as

$$S_T(R_p,R_q) = \frac{\sum_{i=1}^{|K|} S'_T(t^p_{K(i)}.T, t^q_{K(i)}.T)}{|K|}, \tag{6}$$

where

$$S'_T(r_1,r_2) = 1 - \frac{dis(r_1,r_2)}{4}. \tag{7}$$

In the above equation, r_1 and r_2 are the codes for two topological relations, and $dis(r_1, r_2)$ represents the shortest distance between r_1 and r_2 (the length of the shortest path connecting nodes r_1 and r_2) on the topological relationship neighborhood graph [22] as shown in Fig. 9.

Finally, we define the centroid-location-similarity measure between R_p and R_q as

$$S_C(R_p,R_q) = \frac{\sum_{i=1}^{|K|} S'_C(t^p_{K(i)}.C_1, t^q_{K(i)}.C_1) + \sum_{i=1}^{|K|} S'_C(t^p_{K(i)}.C_2, t^q_{K(i)}.C_2)}{2 \times |K|}, \tag{8}$$

where

$$S'_C(\alpha,\beta) = 1 - \frac{diff(\alpha,\beta)}{4}, \tag{9}$$

with $diff(\alpha,\beta) = \min\{|\alpha-\beta|, 8-|\alpha-\beta|\}$; α and β are the numbers assigned to the direction areas in which the two centroids of objects are located, respectively.

Based on the above similarity measuring equations and the rationale that S_O, S_{Dr} ,S_T, and S_C are not orthogonal to one another, the twelve types of similarity measures S_{ij} ($0 \leqslant i \leqslant 3$, $0 \leqslant j \leqslant 2$) can be defined as the product result of some subset selected from $\{S_O, S_T, S_C, S_{Dr}\}$.

1. Type-00: $S_{00} = S_O$.
2. Type-01: $S_{01} = S_O \times S_{D8}$.
3. Type-02: $S_{02} = S_O \times S_{D16}$.
4. Type-10: $S_{10} = S_O \times S_T$.
5. Type-11: $S_{11} = S_O \times S_T \times S_{D8}$.
6. Type-12: $S_{12} = S_O \times S_T \times S_{D16}$.

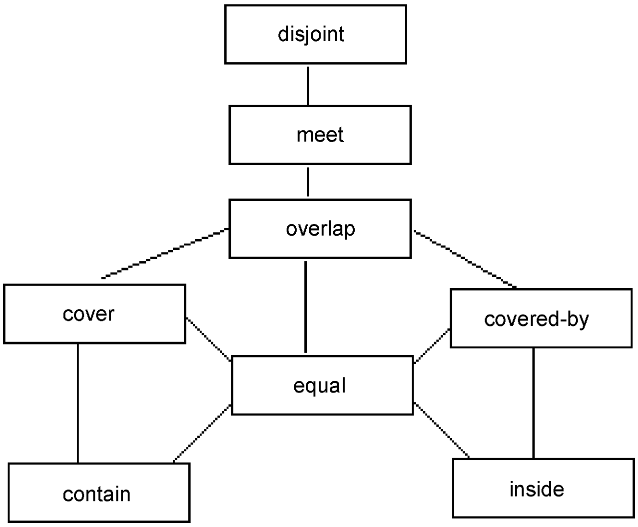

Fig. 9. Topological neighborhood graph.

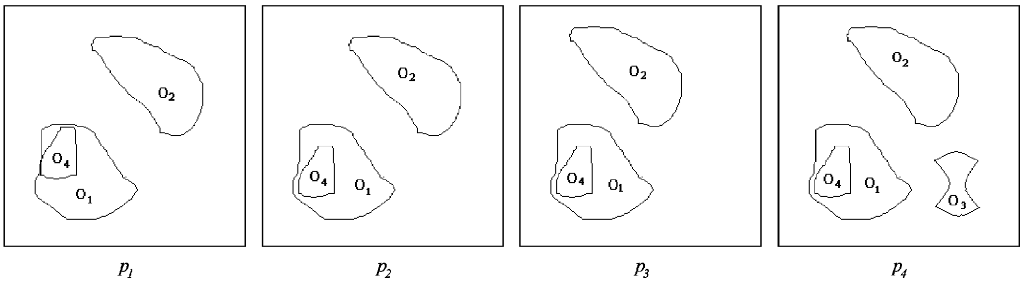

Fig. 10. An example for similarity measurement.

7. Type-20: $S_{20} = S_O \times S_C$.
8. Type-21: $S_{21} = S_O \times S_C \times S_{D8}$.
9. Type-22: $S_{22} = S_O \times S_C \times S_{D16}$.
10. Type-30: $S_{30} = S_O \times S_T \times S_C$.
11. Type-31: $S_{31} = S_O \times S_T \times S_C \times S_{D8}$.
12. Type-32: $S_{32} = S_O \times S_T \times S_C \times S_{D16}$.

Now we use the pictures shown in Fig. 10 as an example to illustrate the effectiveness of our similarity measures. Let R_{p1}, R_{p2}, R_{p3} and R_{p4} be the image representations for pictures p_1, p_2, p_3, and p_4, respectively. Then

$R_{p1} = \{(1,0,\ (6,1,0,1,0),\ (3,0,2,0,0)),\ (4,4,\ (8,8,8,1,1),\ (4,1,1,0,1)),\ (5,0,\ (2,1,1,0,0),\ (6,1,1,0,0))\}$,

$R_{p2} = \{(1,0,\ (6,1,0,1,0),\ (3,0,2,0,0)),\ (4,6,\ (1,8,8,1,1),\ (5,2,2,0,0)),\ (5,0,\ (2,1,1,0,0),\ (6,1,1,0,0))\}$,

$R_{p3} = \{(1,0,\ (7,0,1,1,1),\ (3,0,1,1,1)),\ (4,6,\ (1,8,8,1,1),\ (5,2,2,0,0)),\ (5,0,\ (2,1,1,0,1),\ (7,0,0,0,0))\}$,

$R_{p4} = \{(1,0,\ (6,1,0,1,1),\ (3,0,2,0,1)),\ (2,0,\ (5,0,0,0,1),\ (1,0,0,0,1)),\ (3,0,\ (3,1,0,1,0),\ (7,1,0,0,0)),\ (4,6,\ (1,8,8,1,1),\ (5,2,2,0,0)),\ (5,0,\ (2,1,1,0,0),\ (7,0,1,0,0)),\ (6,0,\ (1,0,0,0,1),\ (5,0,0,0,1))\}$.

Since Type-32 is the most precise measure, we use it to calculate the similarity scores for database pictures $p2$, $p3$, $p4$ with respect to the query picture $p1$. Because $p1$ has objects O_1, O_2, O_4 and $p4$ has objects O_1, O_2, O_3, O_4, therefore, the object-similarity measure $S_O(R_{p4}, R_{p1})$ is

$$\frac{|W_{p4} \cap W_{p1}|}{|W_{p4} \cup W_{p1}|} = \frac{3}{4} = 0.75.$$

In topological-similarity measure or directional-similarity measure, the tuples used for comparison must have the same object-pair code. For this purpose, we use notation t_i^p to represent a tuple whose object-pair code is i in image representation R_p for picture p. So we have

$$S'_T(t_1^{p4}.T, t_1^{p1}.T) = 1, \quad S'_T(t_4^{p4}.T, t_4^{p1}.T) = 3/4,$$
$$S'_T(t_5^{p4}.T, t_5^{p1}.T) = 1.$$

As a consequence, the topological-similarity measure $S_T(R_{p4}, R_{p1})$ is

$$\frac{\sum_{i=1}^{3} S'_T(t^{p4}_{K(i)}.T, t^{p1}_{K(i)}.T)}{3} = \frac{11}{12} = 0.917.$$

By applying Eq. (5) to R_{p1} and R_{p4}, we obtain the following results:

$$S'_{D16}(f_{16}(t^{p4}_1.D_1), f_{16}(t^{p1}_1.D_1)) = \tfrac{3}{4},$$
$$S'_{D16}(f_{16}(t^{p4}_4.D_1), f_{16}(t^{p1}_4.D_1)) = 1,$$
$$S'_{D16}(f_{16}(t^{p4}_5.D_1), f_{16}(t^{p1}_5.D_1)) = 1,$$
$$S'_{D16}(f_{16}(t^{p4}_1.D_2), f_{16}(t^{p1}_1.D_2)) = \tfrac{4}{5},$$
$$S'_{D16}(f_{16}(t^{p4}_4.D_2), f_{16}(t^{p1}_4.D_2)) = \tfrac{5}{8},$$
$$S'_{D16}(f_{16}(t^{p4}_5.D_2), f_{16}(t^{p1}_5.D_2)) = \tfrac{1}{2}.$$

Thus, the direction-similarity measure $S_{D16}(R_{p4}, R_{p1})$ is

By applying Eq. (9) to R_{p1} and R_{p4}, we obtain the results:

$$S'_C(t^{p4}_1.C_1, t^{p1}_1.C_1) = 1, \quad S'_C(t^{p4}_4.C_1, t^{p1}_4.C_1) = \tfrac{3}{4},$$
$$S'_C(t^{p4}_5.C_1, t^{p1}_5.C_1) = 1, \quad S'_C(t^{p4}_1.C_2, t^{p1}_1.C_2) = 1,$$
$$S'_C(t^{p4}_4.C_2, t^{p1}_4.C_2) = \tfrac{3}{4}, \quad S'_C(t^{p4}_5.C_2, t^{p1}_5.C_2) = \tfrac{3}{4}.$$

Thus, the centroid location-similarity measure $S_C(R_{p4}, R_{p1})$ is

$$\frac{\sum_{i=1}^{3} S'_C(t^{p4}_{K(i)}.C_1, t^{p1}_{K(i)}.C_1) + \sum_{i=1}^{3} S'_C(t^{p4}_{K(i)}.C_2, t^{p1}_{K(i)}.C_2)}{2 \times 3}$$
$$= \frac{21}{24} = 0.875.$$

As a result, the type-32 similarity measure between R_{p1} and R_{p4} is

$$S_{32}(R_{p4}, R_{p1}) = S_O \times S_T \times S_{D16} \times S_C = 0.469.$$

Similarly, we can obtain the type-32 similarity measures for (p_1, p_2) and (p_1, p_3) as follows:

$$S_{32}(R_{p2}, R_{p1}) = 0.787, \quad S_{32}(R_{p3}, R_{p1}) = 0.624.$$

The above result implies that picture p_2 is more similar to picture p_1 than picture p_3. This result is due to the fact that the directional relations between objects O_1 and O_2 in p_1 and p_2 are the same while the directional relations between objects O_1 and O_2 in p_1 and p_3 are different as shown in Fig. 10. Among p_2, p_3, and p_4, picture p_4 is most dissimilar to p_1 because p_4 has an additional object O_3 as compared to the other three pictures. The results obtained from our similarity measures defined in this section are consistent with the human visual system.

6. Experimental results

In this section, we present the simulation results to demonstrate the effectiveness of our similarity retrieval system based on the image representation scheme proposed in Section 3 and the similarity measures described in Section 5. In our experimental system, the image database contains 3600 pictures with each picture contains four to six objects. The pictures were randomly generated by considering all possible directional and topological relations between objects.

The similarity requirement of a query process is denoted by a pair (*Type-ij*, T_h), where *Type-ij* is one of the 12 types of similarity measures defined in Section 5, and $T_h \in (0, 1)$ is a threshold for *Type-ij* similarity. The performance of our image retrieval system was evaluated in terms of recall and precision. For a given query picture, let a be the number of all relevant pictures, b be the number of pictures that are relevant and retrieved, and c be the number of pictures retrieved. Then, $recall = \frac{b}{a} \times 100\%$ and $precision = \frac{b}{c} \times 100\%$.

The experimental results are shown in Tables 4–7, respectively, according to each of the following four criteria: (1) no spatial relation is considered; (2) only topological relations are considered; (3) only directional relations between centroids of objects are considered; (4) both topological relations and directional relations are considered. In each table, the recall and precision of image retrieval for the following three situations are listed: (1) without considering object's span; (2) considering object's

Table 4
Recall and precision for type-00, type-01 and type-02 similarity

T_h	Type-00		Type-01		Type-02	
	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)
0.3	100	8.3	100	35.0	99.6	44.7
0.4	100	8.3	99.9	43.9	97.0	54.8
0.5	100	8.3	98.4	52.2	91.9	68.3
0.6	100	8.3	93.4	64.1	83.4	81.8
0.7	100	13.8	84.3	79.9	71.6	93.6

Table 5
Recall and precision for type-10, type-11 and type-12 similarity

T_h	Type-10		Type-11		Type-12	
	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)
0.3	100	8.3	99.9	35.7	99.3	45.5
0.4	100	8.3	99.7	44.6	96.1	56.0
0.5	100	8.3	97.2	53.4	90.2	70.0
0.6	100	8.6	91.1	66.1	81.2	83.9
0.7	99.9	14.1	81.2	82.8	69.0	95.1

Table 6
Recall and precision for type-20, type-21 and type-22 similarity

T_h	Type-20		Type-21		Type-22	
	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)
0.3	100	14.0	99.9	42.2	98.5	50.9
0.4	100	21.3	99.1	51.4	94.3	64.9
0.5	100	29.8	95.4	64.3	87.7	78.1
0.6	100	33.9	88.2	78.7	78.8	90.5
0.7	100	40.0	77.1	90.9	64.8	97.2

Table 7
Recall and precision for type-30, type-31 and type-32 similarity

T_h	Type-30		Type-31		Type-32	
	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)	Rec. (%)	Pre. (%)
0.3	100	14.4	99.8	42.8	97.9	51.7
0.4	100	21.9	98.3	52.3	92.9	66.2
0.5	100	30.4	93.6	65.7	86.2	80.2
0.6	99.9	35.1	86.1	81.0	76.1	92.0
0.7	99.3	41.6	74.2	92.4	62.5	97.8

span in 8-direction areas; (3) considering object's span in 16-direction areas. Since all similarity measures are normalized within the range from 0 to 1, we recorded the image retrieval results using thresholds T_h from 0.3 to 0.7 with increment of 0.1.

In these four tables, we can easily see that they have the same following trend: when T_h is increasing, recall is decreasing and precision is increasing too except the recall rate in the type-00 case. These results are consistent with the information retrieval principle that a lower threshold implies a higher recall rate and a lower precision rate while a higher threshold implies a lower recall rate and a higher precision rate. The reason why the recall rates stay at 100% for all thresholds in the type-00 case is because two pictures are considered as similar as long as they have the same set of objects and this criterion for measuring similarity is too loose. We can also see another trend in each of these four tables, that is, the recall rate is decreasing and the precision rate is increasing from type-00 to type-01, then from type-01 to type-02 for any given T_h. The occurrence of this trend is quite reasonable because the similarity criterion of type-02 is more rigid than that of type-01 which in turn is more rigid than type-00. The third trend that we can see from Tables 4–7 is that, for any given T_h, the recall rate is decreasing and the precision rate is increasing from type-0i to type-1i, from-type 1i to type-2i, then from type-2i to type 3i (for $i = 0$ or $i = 1$ or $i = 2$). Such a trend is due to the fact that type-3i must consider both topological and directional relations, type-2i only considers directional relations (which are more sensitive than topological relations from human's perception), type-1i only considers topological relations, and type-0i considers no spatial relations at all.

As shown in the last two columns in Table 7, our system can achieve a 97.8% precision rate while still maintaining a 62.5% recall rate in one extreme case. In the other extreme case, our system can achieve 97.9% recall rate while maintaining a 51.7% precision rate. On an average, a good performance with 81.2% recall rate and 86.1% precision rate can be simultaneously achieved if a threshold is well adjusted.

7. Conclusions

Content-based image retrieval is the current trend of designing image database systems and the binary spatial relationships between objects are the important features reflecting the content of a picture. Thus, an appropriate knowledge representation for spatial relations plays an important role in designing a CBIR system. In this paper, we presented a novel image representation method based on a triangular partition model to capture the information about the spatial relationships between any two objects in a picture. The capability of representing a picture by our proposed method is more powerful than all other previous methods in terms of accuracy, flexibility, the discriminating power of differentiating pictures, and the ability of handling special relations which are rarely discussed in the literature. Our method also provides different degrees of granularity for reasoning about directional relations between objects. In particular, our method can use the same image representation of a picture to infer the directional relationships between objects in that picture either in the 8-direction reference frame or in the 16-direction reference frame depending on user's different requirements. In similarity retrieval, our system provides 12 types of similarity measures to support flexible matching between the query picture and the database pictures and this capability is highly desired for meeting user's different requirements. By exercising a database containing 3600 pictures, we successfully demonstrated the effectiveness of our image retrieval system based on the image representation method proposed in this paper. Experiment results show that 97.8% precision rate can be achieved while still maintaining 62.5% recall rate; and 97.9% recall rate can be achieved while still maintaining 51.7% precision rate. On an average, 86.1% precision rate and 81.2% recall rate can be achieved simultaneously, and this result is considered as a very good performance to any information retrieval system including the image database system.

References

[1] Y. Chen, J.Z. Wang, A region-based fuzzy feature matching approach to content-based image retrieval, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (9) (2002) 1252–1267.

[2] P.W. Huang, S.K. Dai, Image retrieval by texture similarity, Pattern Recognition 36 (2003) 665–679.

[3] C. Hennig, L.J. Latecki, The choice of vantage objects for image retrieval, Pattern Recognition 36 (2003) 2187–2196.

[4] D. Zhang, G. Lu, Review of shape representation and description techniques, Pattern Recognition 37 (2004) 1–19.

[5] D. Zhang, G. Lu, Study and evaluation of different Fourier methods for image retrieval, Image and Vision Computing 23 (2005) 33–49.

[6] A.K. Majumdar, I. Bhattacharya, A.K. Saha, An object oriented fuzzy data model for similarity detection in image databases, IEEE Transactions on Knowledge and Data Engineering 14 (5) (2002) 1186–1189.

[7] E. Petrakis, C. Faloutsos, K.I. Lin, ImageMap: an image indexing method based on spatial similarity, IEEE Transactions on Knowledge and Data Engineering 14 (5) (2002) 979–987.

[8] Y. Rui, T.S. Huang, Image retrieval: current techniques, promising directions, and open issues, Journal of Visual Communication and Image Representation 10 (1999) 39–62.

[9] P. Punitha, D.S. Guru, An invariant scheme for exact match retrieval of symbolic images: triangular spatial relationship based approach, Pattern Recognition Letters 26 (2005) 893–907.

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

[11] J.R. Smith, S.F. Chang, VisualSEEK: a full automated content-based image query system, in: *Proceedings of the Fourth ACM International Multimedia Conference*, 1996, pp. 87–98.

[12] K.C. Liang, C.C. Jay Kuo, WaveGuide: a joint wavelet-based image representation and description system, IEEE Transactions on Image Processing 8 (11) (1999) 1619–1629.

[13] P.W. Huang, Y.R. Jean, Using 2D C+- string as spatial knowledge representation for image database systems, Pattern recognition 27 (9) (1994) 1249–1257.

[14] P.-W. Huang, C.-H. Lee, Image database design based on 9D-SPA representation for spatial relations, IEEE Transactions on Knowledge and Data Engineering 16 (11) (2004) 1486–1496.

[15] M.J. Egenhofer, Query processing in spatial-query-by-sketch, Journal of Visual Languages and Computing 8 (1997) 403–424.

[16] S.K. Chang, Q.Y. Shi, C.W. Yan, Iconic indexing by 2D strings, IEEE Transactions on Pattern Analysis and Machine Intelligence 9 (3) (1987) 413–428.

[17] S.K. Chang, E. Jungert, Y. Li, Representation and retrieval of symbolic pictures using generalized 2D strings, *Technical Report, University of Pittsburg*, 1988.

[18] S.Y. Lee, F.J. Hsu, 2D C-string: a new spatial knowledge representation for image database systems, Pattern Recognition 23 (10) (1990) 1077–1087.

[19] A.J.T. Lee, H.-P. Chiu, 2D Z-string: A new spatial knowledge representation for image databases, Pattern Recognition Letters 24 (2003) 3015–3026.

[20] S.-Y. Lee, F.-J. Hsu, Spatial reasoning and similarity retrieval of images using 2D C-string knowledge representation, Pattern Recognition 25 (3) (1992) 305–318.

[21] R. Goyal, M.J. Egenhofer, Similarity of cardinal directions, Lecture Notes in Computer Science, vol. 2121, Springer, Berlin, 2001, pp. 36–55.

[22] M. Nabil, A.H.H. Ngu, J. Shepherd, Picture similarity retrieval using the 2D projection interval representation, IEEE Transactions on Knowledge and Data Engineering 8 (4) (1996) 533–539.

[23] D.J. Peuquet, Z. Ci-Xiang, An algorithm to determine the directional relationship between arbitrarily-shaped polygons in the plane, Pattern Recognition 20 (1) (1987) 65–74.

[24] R.K. Goyal, Similarity assessment for cardinal directions between extended spatial objects, PhD Thesis, Departmen of

Spatial Information Science and Engineering, University of Maine, May 2000.
[25] S.K. Chang, Principles of Pictorial Information Systems Design, Prentice-Hall, Englewood Cliffs, NJ, 1989.
[26] B. Bhanu, S. Lee, Genetic Learning for Adaptive Image Segmentation, Kluwer Academic, Norwell, 1994.
[27] M. Egenhofer, R. Franzosa, Point-set topological spatial relations, International Journal of Geographic Information Systems 5 (1991) 161–174.
[28] N.-S. Chang, K.-S. Fu, Query-by-pictorial-example, IEEE Trans. On Software Engineering, 6 (6) (1980) 519–524.