Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Pattern Recognition Letters 24 (2003) 2397–2408

Pattern Recognition Letters

www.elsevier.com/locate/patrec

Archival and retrieval of symbolic images: An invariant scheme based on triangular spatial relationship

D.S. Guru *, P. Punitha, P. Nagabhushan

Department of studies in Computer Science, University of Mysore, Manasagangothri, Mysore 570006, Karnataka, India

Received 22 October 2002; received in revised form 22 March 2003

Abstract

In this paper, a novel scheme for representing symbolic images in a symbolic image database (SID) is proposed. The proposed scheme is based on triangular spatial relationship (TSR) [Pattern Recognition Lett. 22 (2001) 999]. The scheme preserves TSR among the components in a symbolic image by the use of quadruples. A SID is created through the construction of B-tree, an efficient multilevel indexing structure. A methodology to retrieve similar images for a given query image is also presented. The presented retrieval model has logarithmic search time complexity. The study made in this work reveals that the model bears various advantages when compared to other existing models and could be extended towards dynamic databases.

Keywords: Symbolic images; 2D string; 9DLT matrix; Iconic indexing; Triangular spatial relationship; B-tree; Symbolic image database; Similarity retrieval

1. Introduction

The widespread availability of images in digital form has created a growing interest in methods that can search and retrieve images of desired content from a symbolic image database (SID). A SID is a system, which consists of a large amount of image data and their related information represented by both symbolic and physical images (Guru and Nagabhushan, 2001). The symbolic image can be regarded as an abstract physical image, while the physical image is the real image itself. However, to represent/retrieve a physical image in/from SID, the attributes such as symbols/icons and their relationships that describe the corresponding symbolic image, are needed. These attributes should be sufficient enough to speed up the process of retrieving similar symbolic images from SID in addition to being rich enough to represent symbolic images in SID.

The perception of topological relationships (Zhou and Ang, 1997; Zhou et al., 2001), especially spatial relationships existing among the components/iconic objects of a symbolic image, helps in making the SID system more intelligent, fast and flexible (Chang et al., 1988). Indeed, the

* Corresponding author. Fax: +91-821-510789.
E-mail addresses: guruds@mailcity.com, guruds@lycos.com (D.S. Guru), punithaswamy@yahoo.com (P. Punitha), pnagabhushan@hotmail.com (P. Nagabhushan).

0167-8655/03/$ - see front matter
doi:10.1016/S0167-8655(03)00069-2

perception of spatial relationships among the components of a symbolic image preserves the reality being embedded in the respective physical images.

Thus, the gist of the problem lies in designing an effective scheme to represent symbolic images in SID so that the corresponding retrieval process would take less time. The representation scheme must support the retrieval process in retrieving similar images irrespective of the transformation through which the query image has undergone.

In view of this, there have been continuous attempts made by the research community to come out with such an invariant, flexible and intelligent SID system based on the perception of spatial relationships. Among the insinuated methodologies, the value oriented viz., pixel based (Chock et al., 1984), quadtree based (Samet, 1984), R-trees based (Guttman, 1984), vector based (Jungert, 1985), are shown to be insufficient to deal with complicated operations in an intelligent, fast and flexible manner (Chang and Lin, 1996), while other alternative object oriented models receive considerable attention. Chang et al. (1987) introduced a data structure called 2D-string, which is an ordered pair (u, v), where u and v denote a one-dimensional string containing spatial relations between symbolic objects along the x- and y-axes respectively. The 2D-string uses a set of relational operators to show the spatial relationship between two objects along each axis. In the original formulation (Chang et al., 1987), objects are assimilated to points (usually centroids) and the two-dimensional arrangement of set of objects is encoded into 2D-string, a sequential structure. Chang and Li (1988) presented 2D-H string, which can be viewed as a combination of quadtree (Samet, 1984) and 2D-strings. Though, symbolic images can be efficiently represented in a hierarchical manner, Chang and Lin (1996) discovered some redundancies existing in 2D-H representations and put forth an alternative scheme called adaptive 2D-H string for representing the relationships among the objects in an image more efficiently. Chang and Ann (1999) suggested an improvement on the adaptive 2D-H string by specifying how to avoid ambiguities that can occur in adaptive 2D-H string representation. Jungert (1988) introduced an edge-to-edge operator, to segment the components into connected sub-components and then to represent the relationships existing among the sub-components present in an image in a realistic way. However, the usage of edge-to-edge operator fails to represent all feasible relationships. As a remedy, Chang et al. (1989) recommended the concept of generalized 2D-string (2D-G string) by cutting mechanism. However, 2D-G string representation is not space efficient.

Later, many attempts were made to reduce the space complexity of 2D-G string and also to expand the number of operators in order to represent all possible relationships through incorporation of new spatial operators. As a result of this, the concept of 2D-C string (Lee and Hsu, 1990) was emerged.

Using the concept of 2D-string, in order to retrieve similar symbolic images from SID, algorithms (Lee et al., 1989; Lee and Hsu, 1992) based on longest common subsequence matching were proposed. However, Drakopoulos and Constantopoulos (1989) reported that the subsequence matching fails for some instances. To overcome such false drops, Petrakis (1993) devised an alternative algorithm. Though, these iconic image representation schemes offer many advantages, the linear string representation given to the spatial relations existing among the components is not capable of representing the semantics and as well the structures of the images to be archived in the SID. In addition, the process of string matching takes non-deterministic-polynomial time complexity.

Based upon the variations of 2D-string, hash-oriented algorithms for similarity retrieval were explored (Chang and Wu, 1992; Wu and Chang, 1994). An image retrieval scheme making use of the concept of perfect hash table was designed (Chang and Lee, 1991). Though the hash-oriented approaches are efficient from the point of view of retrieval time, construction of hash table takes exponential time complexity. Insertion or deletion of a component into or from an image database requires the reconstruction of the hash table along with the re-computation of iconic/symbol values (Zhou and Ang, 1997). Hence, these methods best suit static database applications. Sabharwal and

Bhatia (1995, 1997) explored the suitability of hash tables towards dynamic database applications and proposed a near perfect hash table for image retrieval. However, this scheme is not completely free from collisions, thereby making the search mechanism itself a bottleneck.

On the other hand, Chang (1991) formulated an indexing structure called nine-directional lower triangular (9DLT) matrix to logically represent symbolic images with pairwise spatial relationships among the components in a symbolic image with the help of nine-directional codes. Chang and Wu (1995) have proposed an exact match retrieval scheme based upon principal component analysis to retrieve symbolic images from SID by the use of the 9DLT matrix. However, two or more different images may be associated with the same principal component direction (Guru, 2000). In order to alleviate this problem, an improved 9DLT matrix has been suggested by defining component to component distances (Guru, 2000). But, this improved model is not invariant even to scaling since the component to component distance is considered as a feature. Zhou and Ang (1997) suggested a better scheme based on the 9DLT matrix for retrieving symbolic images by an improved hashing table. However, the computation of iconic values for iconic objects such that no two different triplets will have the same hash address, takes exponential time.

An alternative approach based on G-tree, a region decomposition (Samet, 1984) method, was propounded (Wu and Cheng, 1997) as an efficient accessing method for retrieving similar pictures from iconic image database. According to this method, two pictures are similar to each other if they have the same objects that are enclosed in the same predefined regions. However, the method is not invariant even to translation. El-Geresy and Abdelmoty (1997) formulated a general approach, for spatial reasoning, consisting of a set with two general constraints and two general rules to respectively govern the spatial relationships between objects in space and to propagate two definite relationships between objects in space. The formalism is based on representing objects and their space topology by an adjacency matrix, where a partition strategy of objects and space is carried out to reflect the specific decomposition of interest in different applications. An intersection matrix between different objects and space parts then represents the topological relations. This approach preserves the adjacency relationships among the objects in an image, but it fails in the complete perception of the spatial relationships among the components as it keeps no information about the relative positions of the objects.

Zhou et al. (2001) introduced an image retrieval scheme based on object's orientation spatial relationship, which preserves the information of partial and total orientation relationships among objects. For an image with 'm' distinct components, the orientation spatial relationship is preserved in a matrix of order $m \times m$. Unlike other methods this method is invariant to image transformations, but, even for considerably small number of images, the space complexity is very high and hence retrieval takes longer.

Petraglia et al. (1996) discovered 2D-R string representation based on 2D-C string mechanism (Lee and Hsu, 1990) as a rotation invariant concept. 2D-R strings are defined by considering cutting lines at the borders of the objects along the radial segments, and the concentric circles' tangent to the object extremes (Chang and Jungert, 1996; Bimbo, 1999). The concentric circles are drawn with the center being the center of the object nearer to the center of the image. The sectors and rings obtained in this way are ordered from left to right and from the center to the outer circle. Rotation invariance is achieved as both the beginning and ending points of any object do not vary under image rotation. But, change in the center of an image results with different 2D-R string representation for the image. This situation is not uncommon as in reality it is seldom possible to have a query image with the same scene coverage as that of the images stored in SID.

It is clear from the above discussion that all the aforementioned methods appear to be efficient in one or the other sense. But, the models which are robust and simple seem to be inefficient in addition to being not invariant to image transformations and the methods which are invariant to image transformations are either sensitive to the reference point or not effective from the point of view of storage requirement. The problem therefore is to devise an efficient methodology, which is smart enough to take care of image transformations and can rejuvenate to be acceptable for real pragmatic situations.

In view of this, a problem associated with retrieval of symbolic images from SID invariant to image transformations based on spatial relationship is addressed in this paper and a scheme based on triangular spatial relationship (TSR) (Guru and Nagabhushan, 2001) is proposed. The proposed model preserves TSR among the components in a symbolic image by the use of quadruples. A SID is created through the construction of an efficient indexing structure called B-tree (Wirth, 1999). Since B-tree does not support multidimensional data (4D data in our case), a distinct and unique key is computed for each of the distinct quadruple and then the computed keys are stored in B-tree as representatives of the corresponding quadruples. Each key value stored in B-tree is attached with a list of images, which have the corresponding quadruple as one of their associated quadruples. An efficient methodology to retrieve similar images (images containing a query image as their subimage) from SID is presented. The presented model takes $O(\log_r n)$ search time in worst case, where, n is the total number of symbolic images stored in the B-tree and r is the order/rank of the B-tree. The designed model can be adapted easily for maintaining a dynamic image database. Unlike other models, the proposed model does not require any additional computational time when applied on dynamic databases.

The remaining part of the paper is organized as follows. Section 2 briefs about the concept of TSR. Section 3 proposes a novel system for image archival and retrieval while Section 4 illustrates the proposed system with a complete example. The efficacy of the proposed model is brought out clearly in Section 5 and Section 6 follows with conclusion.

2. The concept of triangular spatial relationship: An overview

A TSR is formally defined (Guru and Nagabhushan, 2001) by connecting three non-collinear components in a symbolic image as follows.

Let A, B, and C be any three non-collinear components of a symbolic image. Let L_a, L_b and L_c be the labels of A, B and C respectively. Connecting the centroids of these components mutually forms a triangle as shown in Fig. 1. Let M_1, M_2, and M_3 be the midpoints of the sides of the triangle as shown in Fig. 1. Let θ_1, θ_2, and θ_3 be the smaller angles subtended at M_1, M_2, and M_3 respectively and are shown in Fig. 1. The TSR among the components A, B and C is represented by a set of quadruples $\{(L_a,L_b,L_c,\theta_3),(L_a,L_c,L_b,\theta_2),(L_b,L_a,L_c,\theta_3),(L_b,L_c,L_a,\theta_1),(L_c,L_a,L_b,\theta_2),(L_c,L_b,L_a,\theta_1)\}$. This representation is unwieldy, as there are six possible quadruples for every three non-collinear components. Thus, it was recommended to choose only one of those, which satisfies the following criteria.

If $(L_{i1},L_{i2},L_{i3},\theta)$ is the quadruple to be chosen, then the labels L_{i1}, L_{i2}, and L_{i3} must satisfy one of the following conditions:

1. The labels L_{i1}, L_{i2}, and L_{i3} are distinct and $L_{i1} > L_{i2} > L_{i3}$.
2. $L_{i1} = L_{i2}$ and $L_{i3} < L_{i1}$.
3. $L_{i1} > L_{i2}$ and $L_{i2} = L_{i3}$ and $\mathrm{Dist}(\mathrm{Comp}(L_{i1}), \mathrm{Comp}(L_{i2})) \geqslant \mathrm{Dist}(\mathrm{Comp}(L_{i1}), \mathrm{Comp}(L_{i3}))$.
4. $L_{i1} = L_{i2} = L_{i3}$ and $\mathrm{Dist}(\mathrm{Comp}(L_{i1}), \mathrm{Comp}(L_{i2})) \geqslant M$, where
$$M = \mathrm{Max}(\mathrm{Dist}(\mathrm{Comp}(L_{i1}),\mathrm{Comp}(L_{i3})), \mathrm{Dist}(\mathrm{Comp}(L_{i2}),\mathrm{Comp}(L_{i3}))).$$
here $\mathrm{Dist}(A,B)$ is a function which computes the Euclidean distance between the midpoints of the components A and B. $\mathrm{Max}(a,b)$ is a function denoting the maximum among a and b and $\mathrm{Comp}(L)$ indicates the component, the label of which is L.

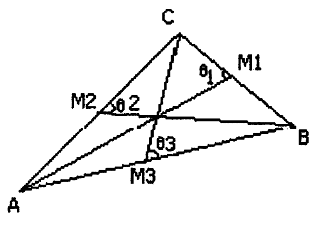

Fig. 1. Triangular spatial relationship.

It is guaranteed that, even if more than one possible permutation of the components satisfies the above-stated condition, the corresponding quadruples are one and the same.

In other words, the TSR among any three non-collinear components A, B and C is defined by a quadruple $(L_{i1}, L_{i2}, L_{i3}, \theta)$, where the sequence of L's satisfies one of the above stated conditions and θ is the smaller angle subtended at the midpoint of the components, the labels of which are L_{i1} and L_{i2}, due to the line joining that midpoint and the centroid of the remaining component, the label of which is L_{i3}. The θ is given by

$$\theta = \begin{cases} \theta_1 & \text{if } \theta_1 \leqslant 90^\circ, \\ 180 - \theta_1 & \text{otherwise.} \end{cases}$$

here

$$\theta_1 = \cos^{-1}((S_1^2 - S_2^2 - S_3^2)/(2 * S_2 * S_3)),$$

where $S_1 = \mathrm{Dist}(\mathrm{Comp}(L_{i1}),\ \mathrm{Comp}(L_{i3}))$, $S_2 = \mathrm{Dist}(\mathrm{Comp}(L_{i1}),\ \mathrm{Comp}(L_{i2}))/2$, $S_3 = \mathrm{Dist}(\mathrm{Mid}(\mathrm{Comp}(L_{i1}),\ \mathrm{Comp}(L_{i2})),\ \mathrm{Comp}(L_{i3}))$.

Here $\mathrm{Mid}(X, Y)$ denotes the midpoint of the line joining the centroids of the components X and Y.

The concept of TSR is proved to be invariant to image transformations viz., translation, rotation, scaling and flipping. For more details on the invariant properties of TSR, the readers are directed to refer (Guru and Nagabhushan, 2001).

3. The proposed symbolic image archival and retrieval system

This section describes a novel scheme to represent symbolic images in a SID and also a corresponding retrieval scheme.

3.1. Representation of symbolic images in SID

Let $\{S_1, S_2, S_3, \ldots, S_n\}$ be a set of n symbolic images to be archived in a SID. Let $L_1, L_2, L_3, \ldots, L_m$ be the labels of m distinct generic iconic objects called components. These iconic objects are generic in the sense that all physical images can be described as different combinations of these iconic objects. Encoding each iconic object present in a physical image by the respective label produces the corresponding symbolic image (Guru, 2000). Therefore, each symbolic image S_i $\forall_i = 1, 2, 3, \ldots, n$ is said to contain $m_i \leqslant m$ number of labels. However, the transformation of a physical image into its corresponding symbolic image is beyond the scope of the current study.

In order to make the representation scheme invariant to image transformations, we recommend to perceive the TSR existing among all components present in a symbolic image and then to preserve the TSR by the use of quadruples as explained in the previous section. Thus, the problem of symbolic image representation is reduced to the problem of storing those quadruples such that the retrieval task becomes effective and efficient. Hence, it is advised to store the quadruples in the database through creation of B-tree, an efficient multilevel indexing structure, which outperforms any method based on hashing technique. Another advantage of B-tree is that it supports easy insertion and deletion. However, B-tree does not support storage of multivalued data such as quadruples. Thus, a unique and distinct number called key is generated for each of the distinct quadruples to be stored in B-tree and then the generated keys are stored in the B-tree as representatives of the corresponding quadruples.

If (L_a, L_b, L_c, θ) is a quadruple to be stored in B-tree then the key K corresponding to the quadruple is computed as

$$K = D_\theta(L_a - 1)m^2 + D_\theta(L_b - 1)m + D_\theta(L_c - 1) + (C_\theta - 1), \quad (1)$$

where D_θ is the number of slices/classes, the continuous interval type domain $[0 \cdots 90]$ associated with θ is split into and C_θ is the class number to which a specific value of θ belongs. The discretization of the continuous domain of θ is suggested to take care of the possible errors that can occur during the computation of θ value due to the limitation of the computing system in handling floating point numbers.

It can be noticed that the associated keys for any two different quadruples are distinct and unique. Let N be the total number of distinct quadruples generated due to all n symbolic images and let $\{K_1, K_2, K_3, \ldots, K_N\}$ be the set of corresponding keys. All these N keys are stored in a B-tree. Each key value is then attached with a list of image indices. The image indices that are attached to a key value (K) are the indices of images, which have the key K as one of the keys in their corresponding key set.

Therefore, following is the algorithm proposed for representing symbolic images in SID.

Algorithm: Proposed representation scheme
Input: $S_1, S_2, S_3, \ldots, S_n$—Set of symbolic images
Output: Symbolic Image Database (B-tree)
Method:

Step 1: For each symbolic image S_i do
- (i) Apply TSR and obtain a set of quadruples preserving TSR among the components present in S_i.
- (ii) For each obtained quadruple compute a unique key using Eq. (1) and update the list of image indices associated with that key by inserting i as a new index.

For end

Step 2: Obtain the set of all keys computed in step 1.

Step 3: Create a B-tree of rank r containing all the keys present in the set obtained in step 2.

Step 4: Attach each key, present in the B-tree, with the respective list of image indices.

Algorithm ends.

The task of retrieving symbolic images from SID has thus become as trivial as follows.

3.2. Retrieval of symbolic images from SID

There are two kinds of image retrieval: exact match retrieval and similarity retrieval, with the former a special case of the latter. In similarity retrieval the task is to retrieve from a SID those images that are similar to the given query image. Since, each symbolic image is just a collection of TSR quadruples, the image matching problem becomes the problem of subset matching between the TSR quadruples of the given query image and all the TSR quadruples of all images stored in the database. If q_{Q} is the number of quadruples generated for the query image and q_{M} is the average number of quadruples that each model image is associated with, then the conventional matching algorithm requires $\mathrm{O}(nq_{\mathrm{M}}q_{\mathrm{Q}})$ search time. Instead of searching through all images one by one, the proposed novel B-tree based image representation scheme allows us to identify as quickly as possible the matched images as follows.

Let Q be a query image given by a user and q_{Q} be the number of TSR quadruples of the query image Q and hence there are q_{Q} number of keys generated. Accessing through the B-tree to obtain the list of image indices corresponding to each query key and then computing the intersection of all those lists would produce the list of indices of images similar to the query image Q. Accessing through the B-tree in search of a key requires $\mathrm{O}(\log_r N)$ search time, where N is the total number of keys stored in the B-tree and r is the order of the tree and hence the proposed retrieval scheme requires $\mathrm{O}(q_{\mathrm{Q}} \log_r N)$ search time.

The following algorithm is therefore, devised for retrieving similar symbolic images from SID for a given query symbolic image.

Algorithm: Proposed retrieval scheme
Input: Q, a symbolic query image
Output: List of symbolic images containing Q
Method:

Step 1: Obtain a set of quadruples preserving TSR among the components of Q.

Step 2: Compute a unique key for each quadruple using Eq. (1).

Step 3: Access through B-tree in search of each key and extract the list of image indices associated with that key.

Step 4: Compute the list of indices of similar images through intersection of those extracted lists.

Algorithm ends.

The search time complexity of the proposed retrieval scheme is stated as follows.

Statement. The proposed retrieval scheme takes $\mathrm{O}(\log_r n)$ search time for each retrieval based on a single TSR key where n is the number of symbolic images present in SID.

Proof. Let n be the total number of images present in the SID.

Let m denote the total number of iconic objects present in the SID. Thus, any symbolic image can have atmost $m(m-1)(m-2)/6 = O(m^3)$ number of distinct quadruples in the worst case and hence $O(m^3)$ distinct TSR keys.

Let N be the total number of distinct TSR keys due to all n symbolic images.

Therefore, the value of N will be atmost nm^3, i.e.,

$$N = nm^3. \tag{2}$$

The time T_{PM} required by the proposed method to retrieve a list of image indices for a given single TSR key is given by,

$$\begin{aligned} T_{PM} \text{ (for a TSR key)} &= O(\log_r N) \\ &= O(\log_r (m^3 n)) \quad \text{from (2)} \\ &= O(3\log_r m + \log_r n). \end{aligned}$$

Since, in any SID, $m \ll n$,

$$T_{PM} \text{ (for a TSR key)} = O(\log_r n).$$

Thus, the proposed retrieval scheme requires $O(\log_r n)$ search time for each TSR key.

4. A complete illustration with an example

In this section we illustrate the proposed scheme for symbolic image representation in SID and demonstrate how the proposed similarity retrieval takes place.

4.1. Representation of symbolic images in SID

Let us consider $n = 5$ symbolic images shown in Fig. 2. Let {1, 2, 3, 4, 5, 6, 7, 8} be the set of symbolic labels of eight distinct iconic objects. As explained in Section 2, the concept of TSR is employed on the image S_1 and the TSR existing among the components of S_1 is preserved by the use of quadruples. Since S_1 has got four components and no three of them are collinear, the number of quadruples generated is ${}^4C_3 = 4$ and they are,

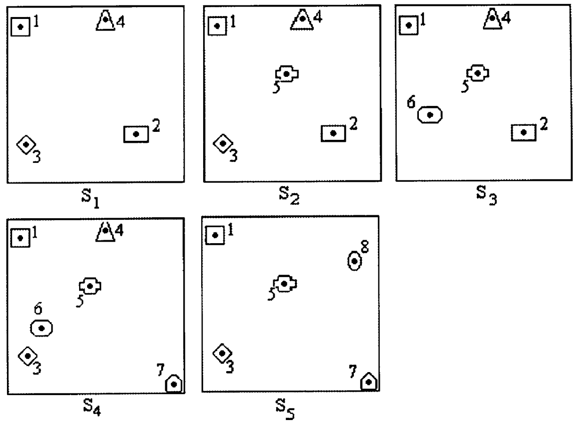

Fig. 2. Five symbolic images as example.

$$\{(3,2,1,67.238292),(4,3,1,71.400327),$$
$$(4,3,2,84.925637),(4,2,1,48.009121)\}.$$

Now the fourth component, which is a real value, of each quadruple is mapped into its class index (as suggested in Section 3.1). In this example, the θ-domain $[0^\circ \cdots 90^\circ]$ has been split into 18 classes/intervals of size 5°. Thus, the quadruples become, $\{(3,2,1,14),(4,3,1,15),(4,3,2,17),(4,2,1,10)\}$.

Similarly, the sets of quadruples preserving TSR among the components of the remaining symbolic images S_2, S_3, S_4 and S_5 are generated and respectively they are;

$$\begin{aligned} S_2 = \{&(3,2,1,14),(5,3,1,14),(5,3,2,13), \\ &(5,4,1,18),(5,4,3,5),(5,4,2,8),(5,2,1,4), \\ &(4,3,1,15),(4,3,2,17),(4,2,1,10)\}, \end{aligned}$$

$$\begin{aligned} S_3 = \{&(6,5,1,17),(6,5,4,4),(6,5,2,14), \\ &(6,4,1,17),(6,4,2,16),(6,2,1,9),(5,4,1,18), \\ &(5,4,2,8),(5,2,1,4),(4,2,1,10)\}, \end{aligned}$$

$$\begin{aligned} S_4 = \{&(6,3,1,7),(6,5,1,7),(6,5,3,2),(6,5,4,4), \\ &(6,4,1,17),(6,4,3,1),(5,3,1,14),(5,4,1,18), \\ &(5,4,3,5),(4,3,1,15),(7,3,1,10),(7,6,1,6), \\ &(7,6,3,4),(7,6,5,11),(7,6,4,13),(7,5,1,1), \\ &(7,5,3,13),(7,5,4,5),(7,4,1,7),(7,4,3,8)\}, \end{aligned}$$

and

$$S_5 = \{(5,3,1,14),(8,3,1,16),(8,5,1,8),(8,5,3,5),(8,7,1,11),(8,7,3,17),(8,7,5,12),(7,3,1,10),(7,5,1,1),(7,5,3,13)\}.$$

For each distinct quadruple, a distinct and unique key is generated. Hence, each symbolic image can be described by a set of TSR key values as given below.

$$\text{TSR Key Set of } S_1 = \{2461, 3758, 3778, 3609\},$$

$$\text{TSR Key Set of } S_2 = \{2461, 4909, 4926, 5057, 5080, 5065, 4755, 3758, 3778, 3609\},$$

$$\text{TSR Key Set of } S_3 = \{6352, 6393, 6367, 6208, 6225, 5912, 5057, 5065, 4755, 3609\},$$

$$\text{TSR Key Set of } S_4 = \{6054, 6352, 6373, 6393, 6208, 6228, 4909, 5057, 5080, 3758, 7209, 7637, 7671, 7714, 7698, 7488, 7536, 7546, 7350, 7397\},$$

and

$$\text{TSR Key Set of } S_5 = \{4909, 8367, 8647, 8680, 8938, 8980, 9011, 7209, 7488, 7536\}.$$

A B-tree of order $r = 4$ is constructed to store the distinct keys. For each distinct key K, a list of indices of images, the key sets of which contain K is worked out and then attached to K in the B-tree for later retrieval. For example, for the key $K = 3758$, the list obtained is the list containing only the image indices 1, 2 and 4. The B-tree constructed for the example is as shown in Fig. 3.

4.2. Retrieval of similar symbolic images from SID

Consider a query image consisting four iconic objects as shown in Fig. 4. According to the proposed retrieval algorithm (Section 3.2), the TSR existing among the components of Q is preserved by the set of quadruples $\{(4,3,1,15),(5,4,1,18),(5,4,3,5),(5,3,1,14)\}$ with their fourth component being the respective θ-class index and the corresponding key set is $\{3758, 5057, 5080, 4909\}$. This set, thus describes the query image Q.

In order to retrieve the images similar to Q, we access through the B-tree in search of each key and

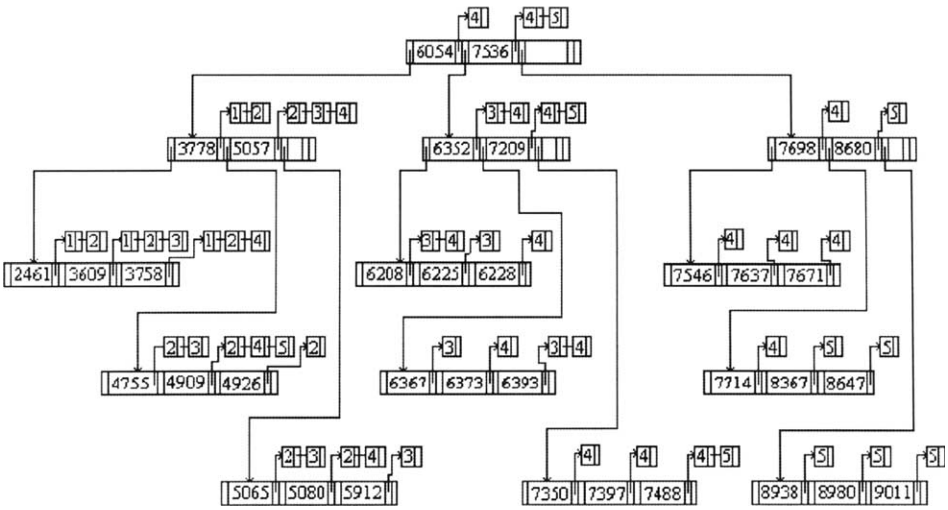

Fig. 3. B-tree representation.

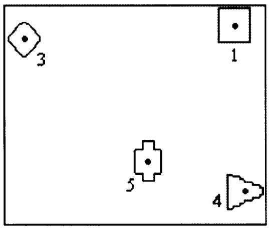

Fig. 4. *Q–A* query image.

then we extract the list of image indices attached to them. The lists of image indices extracted are,

corresponding to the key 3758: $\{1,2,4\}$,
corresponding to the key 5057: $\{2,3,4\}$,
corresponding to the key 5080: $\{2,4\}$,
corresponding to the key 4909: $\{2,4,5\}$.

Presence of a common index in the extracted lists of image indices indicates that the image corresponding to such an index has got all the keys of the query image in its key set and hence, it is said to be a similar image. Therefore, the intersection of the extracted lists gives us the list of indices of the similar images.

So, only images S_2 and S_4 that match the query image Q will be retrieved from the database. It can be noticed that the query image (Fig. 4) is a rotated and scaled up version of sub-image of S_2 and S_4 of Fig. 2. Although the database can be very large and the corresponding B-tree can be very large too, we only pull out four lists of image indices (each corresponding to one key in the key set of the query image) from which the list of similar images can be obtained.

5. Discussion

The effectiveness of image database retrieval depends on the correctness and types of image feature representation. Perception of invariant spatial relationships existing among the generic components present in images indeed helps in preserving the reality being embedded in images. Avoiding ambiguity problem in visualization/representation and the need to use complex rules or algorithms in retrieval are the two important issues yet to be addressed.

In addition, devising a fast and effective retrieval scheme which is invariant to image transformations is still an open and challenging issue in the field of SID retrieval. Invariant property of a system is, in fact, essential to best suit the real pragmatic applications. Indeed, these are the shortcomings exhibited in almost all existing representation schemes.

Only few attempts were made to design invariant models. The model (Petraglia et al., 1996) being sensitive to the reference point cannot be the model of choice, despite it is claimed as invariant to image transformations. The other invariant scheme proposed by Zhou et al. (2001) is not so efficient because it is a model based methodology. The major problem with such an approach is that as the number of images increases, the computational time required to find the similarity match becomes very high.

In this paper, we have made a successful attempt in exploring a model which can overcome the aforementioned shortcomings. The paper presents a way to represent a symbolic image in SID invariant to image transformations by a set of key values, each representing a quadruple which preserves TSR among three iconic objects present in the image. Compared with other methods, the TSR keys can be obtained easily to encode three iconic objects along with their spatial relationships and they can be used directly to represent images in a multilevel indexing structure called B-tree.

The proposed model integrates the representation of an image with the retrieval of an image. It avoids the ambiguity problem in representation/visualization and the need to use complex rules exhibited in other representation schemes in addition to being invariant to image transformations. The proposed system can be easily extended towards dynamic databases as it is very easy to include new iconic objects' labels without relabeling of existing iconic objects. In addition insertion of new symbolic objects results with different TSR quadruples and hence distinct TSR keys.

Thus, the TSR quadruples of all the images that do not contain the new symbol are not affected at all. Therefore, insertion of a new symbol is just a task of inserting a new TSR key into the existing B-tree without requiring modification/reconstruction of any part of B-tree structure. Nevertheless, unlike other models, our model automatically takes care of additional information such as angles since it is based on TSR concept.

The models that seem to be more efficient than our model could be the models which are based on hash tables (Chang and Wu, 1992; Wu and Chang, 1994; Chang and Lee, 1991; Zhou and Ang, 1997; Sabharwal and Bhatia, 1995, 1997). The tuples representing images were used to index an entire image database based on a perfect hash table (Zhou and Ang, 1997; Sabharwal and Bhatia, 1995) leading to O(1) retrieval time of desired images. A perfect hash table does not permit any collisions on any of its addresses. Therefore, retrieval process is simplified to simply computing the hash addresses from the specified spatial constraints in the image depicting the query (Zhou and Ang, 1997). The hash address contains a pointer to a linked list of images that are described by the respective tuple. Thus, each retrieval based on a constraint that can be specified in a single tuple can be performed in O(1) time whereas, the retrieval of images based on q_{Q} tuples can be performed in $\mathrm{O}(q_{\mathrm{Q}})$ time. But, even the best known algorithm for the construction of perfect hash table itself is of exponential time (Cook and Oldehoeft, 1982) and hence even a slight modification to the database is a bottleneck as it requires reconstruction of the entire hash table. An attempt (Sabharwal and Bhatia, 1997) to relax the perfect hash table constraint to near perfect hash table without losing the computations spent in the construction of the existing hash table has been made and that technique allows for a maximum of 10% of its addresses to resolve collisions and hence suits for dynamic databases. However, due to conflict in addresses it requires some more additional memory and is of linear time complexity. Thus the overall search time is of $\mathrm{O}(q_{\mathrm{Q}}N)$ where N is the total number of keys stored. In addition, the model (Sabharwal and Bhatia, 1997) is not invariant to image transformations and makes use of a set of heuristic rules to represent and retrieve images.

On the other hand, the logarithmic time complexity model proposed by Chang and Wu (1995) preserves spatial relationship by the use of 9DLT matrix and then employs the principal component analysis to represent each image by the first principal component direction. The method best suits exact match retrieval but not similarity retrieval and it is not invariant to image transformations. The scheme proposed by Wu and Cheng (1997) also offers an efficient scheme to retrieve images in logarithmic search time complexity. But, this model being based on Grid-tree fails to perceive relative spatial relationships among iconic objects in an image and hence it is not invariant even to translation, a simple and unavoidable transformation.

It is evident from the above discussion that the models that are invariant to image transformations are not efficient from the point of view of retrieval time and the models which are of logarithmic time complexity are not invariant to image transformations. It is also clear that an indexing structure based technique would be a better technique from the point of view of retrieval time than any hash table based technique since each hash table based technique suffers either from computational bottleneck due to exponential time complexity task of hash table construction (Cook and Oldehoeft, 1982) or from linear retrieval time complexity.

Thus, the beauty of our scheme lies in its efficiency from the point of view of retrieval time (as it is of logarithmic time complexity) as well as its property of invariance to image transformations. In addition, it could be extended easily towards dynamic databases. Nevertheless, it automatically takes care of additional information such as angles, an aspect which is not yet addressed so far. Therefore, our technique offers not only the advantages of efficient retrieval and convenience of maintenance but also the advantages of image transformation invariant property. However, the only drawback of our algorithm is its size. Though the space requirement of the index structure is of $\mathrm{O}(m^3)$ where m is the number of iconic objects in the database, the impact of the drawback is negligible when compared to the advantage of the invariant property of the model.

Table 1
Comparison of the proposed method with other previously proposed methodologies

Representation and retrieval schemes	Adopted data structure	Extension towards dynamic database	Invariant to image transformation	Retrieval time complexity
Chang et al. (1987)	2D string	Not suitable	Not invariant	Non-polynomial
Chang et al. (1988)	2D string	Not suitable	Not invariant	Non-polynomial
Lee et al. (1989)	2D string	Not suitable	Not invariant	Non-polynomial
Lee and Hsu (1990)	2D string	Not suitable	Not invariant	Non-polynomial
Chang and Lee (1991)	9DLT matrix + hashing	Not suitable	Not invariant	$O(n)$
Chang and Wu (1992)	9DLT matrix + hashing	Not suitable	Not invariant	$O(n)$
Lee and Hsu (1992)	2D-C string	Not suitable	Not invariant	$O(n)$
Wu and Chang (1994)	9DLT matrix + hashing	Extendable	Not invariant	$O(n)$
Chang and Wu (1995)	9DLT matrix + PCA	Not suitable	Not invariant	$O(\log n)$
Sabharwal and Bhatia (1995)	Perfect hash table	Not suitable	Not invariant	$O(1)$
Petraglia et al. (1996)	2D-R string	Not suitable	Claimed as invariant but, sensitive to the reference point	Exponential
Sabharwal and Bhatia (1997)	2D string + hashing	Extendable	Not invariant	$O(n)$
Wu and Cheng (1997)	G-tree	Extendable	Not invariant	$O(\log n)$
Zhou and Ang (1997)	9DLT matrix + hashing	Not suitable	Not invariant	$O(1)$
Zhou et al. (2001)	A square matrix	Not suitable	Invariant	$O(n^2)$
The proposed scheme	TSR + B-tree	Extendable	Invariant	$O(\log_r n)$

A comparison of the proposed model with some of the other models is given in Table 1.

6. Conclusion

This paper proposes a novel scheme for symbolic image representation invariant to image transformations. The proposed scheme is based on TSR. A SID is created through the construction of B-tree an advanced, multilevel, efficient indexing structure. The corresponding retrieval scheme is also devised. The proposed retrieval scheme requires $O(q_Q \log_r n)$ search time where q_Q is the number of TSR keys of the query image and n is total number of symbolic images stored in the B-tree. The study made in this paper reveals that the method could be easily extended towards dynamic databases and hence it is our future research.

References

Bimbo, A.D., 1999. Visual Information Retrieval. Morgan Kaufmann Publishers, San Francisco, California.

Chang, C.C., 1991. Spatial match retrieval of symbolic pictures. J. Inform. Sci. Eng. 7, 405–422.

Chang, C.C., Lee, S.Y., 1991. Retrieval of similar pictures on pictorial database. Pattern Recognit. 24 (7), 675–680.

Chang, C.C., Lin, D.C., 1996. A spatial data representation: An adaptive 2D-H string. Pattern Recognit. Lett. 17, 175–185.

Chang, C.C., Wu, T.C., 1992. Retrieving the most similar symbolic pictures from pictorial databases. Inform. Process. Manage. 28 (5), 581–588.

Chang, C.C., Wu, T.C., 1995. An exact match retrieval scheme based upon principal component analysis. Pattern Recognit. Lett. 16, 465–470.

Chang, S.K., Jungert, E., 1996. Symbolic Projection for Image Information Retrieval and Spatial Reasoning. Academic Press, San Diego, CA.

Chang, S.K., Li, Y., 1988. Representation of multi-resolution symbolic and binary pictures using 2D-H strings. In: Proceedings of IEEE Workshop on Languages for Automata, pp. 190–195.

Chang, S.K., Jungert, E., Li, Y., 1989. Representation and retrieval of symbolic pictures using generalized 2D string. In: SPIE Proceedings on Visual Communications and Image Processing, Philadelphia, pp. 1360–1372.

Chang, S.K., Shi, Q.Y., Yan, C.W., 1987. Iconic indexing by 2D-strings. IEEE Trans. Pattern Anal. Mach. Intell. 9 (5), 413–418.

Chang, S.K., Yan, C.W., Dimitroff, D.C., Arndt, T., 1988. An intelligent image database system. IEEE Trans. Software Eng. 14 (5), 681–688.

Chang, Y.I., Ann, H.Y., 1999. A note on adaptive 2D-H strings. Pattern Recognit. Lett. 20, 15–20.

Chock, M., Cardenas, A.F., Klinger, A., 1984. Database structure and manipulation capabilities of picture database management system (PICDMS). IEEE Trans. Pattern Anal. Mach. Intell. 9 (3), 413–428.

Cook, C.R., Oldehoeft, R., 1982. A letter oriented minimal perfect hashing function. ACM SIGPlan Notices 17 (9), 18–27.

Drakopoulos, J., Constantopoulos, P., 1989. An Exact Algorithm for 2D String matching. Technical Report 21, Institute of Computer Science, Foundation for Research and Technology, Heraklion, Greece.

El-Geresy, B.A., Abdelmoty, A.I., 1997. Order in space: A general formalism for spatial reasoning. Int. J. Artificial Intell. Tools 6 (4), 423–450.

Guru, D.S., Nagabhushan, P., 2001. Triangular spatial relationship: A new approach for spatial knowledge representation. Pattern Recognit. Lett. 22, 999–1006.

Guru, D.S., 2000. Towards accurate recognition of objects employing a partial knowledge base: Some new approaches. Ph.D. Thesis. Department of Studies in Computer Science, University of Mysore, Manasagangothri, Mysore, India.

Guttman, A., 1984. R-trees: A dynamic index structure for spatial searching. In: Proceedings of ACM-SIGMOD International Conference on Management of Data, pp. 47–57.

Jungert, E. et al., 1985. Vega—a geographical information system. In: Proceeding of Scandinavian Res. Conference on Geographical Information Systems, Linkoping, Sweden.

Jungert, E., 1988. Extended symbolic projection used in a knowledge structure for spatial reasoning. In: Proceedings of 4th BPRA Conference on Pattern Recognition. Springer, Verlag.

Lee, S.Y., Hsu, F.J., 1990. Picture algebra for spatial knowledge representation for image database systems. Pattern Recognit. 23 (10), 1077–1087.

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

Lee, S.Y., Shan, M.K., Yang, W.P., 1989. Similarity retrieval of iconic image databases. Pattern Recognit. 22 (6), 675–682.

Petraglia, G., Sebillo, M., Tucci, M., Tortora, G., 1996. A normalized index for image databases. In: Chang, S.K., Jungert, E., Tortora, G. (Eds.), Intelligent Image Database Systems. World Scientific, Singapore.

Petrakis, E.G.M., 1993. Image representation, indexing and retrieval based on spatial relationships and properties of objects. Ph.D. Thesis. Technical Report FORTH-ICS/TR-075, Institute of Computer Science, Foundation for Research and Technology, Heraklion, Greece.

Sabharwal, C.L., Bhatia, S.K., 1995. Perfect hash table algorithm for image databases using negative associated values. Pattern Recognit. 28 (7), 1091–1101.

Sabharwal, C.L., Bhatia, S.K., 1997. Image databases and near perfect hash table. Pattern Recognit. 30 (11), 1867–1876.

Samet, H., 1984. The quadtree and related data structures. ACM Comput. Surveys 16 (2), 187–260.

Wirth, N., 1999. Algorithms + Data Structures = Programs. Prentice Hall series.

Wu, T.C., Chang, C.C., 1994. Application of geometric hashing to iconic database retrieval. Pattern Recognit. Lett. 15, 871–876.

Wu, T.C., Cheng, J., 1997. Retrieving similar pictures from iconic database using G-Tree. Pattern Recognit. Lett. 18, 595–603.

Zhou, X.M., Ang, C.H., 1997. Retrieving similar pictures from pictorial database by an improved hashing table. Pattern Recognit. Lett. 18, 751–758.

Zhou, X.M., Ang, C.H., Ling, T.W., 2001. Image retrieval based on object's orientation spatial relationship. Pattern Recognit. Lett. 22, 469–477.