An invariant scheme for exact match retrieval of symbolic images: Triangular spatial relationship based approach

P. Punitha *, D.S. Guru *

Department of Studies in Computer Science, University of Mysore, Manasagangotri, Mysore 570006, Karnataka, India

Received 17 September 2004

Abstract

In this paper, a novel method of representing symbolic images in a symbolic image database (SID) invariant to image transformations, useful for exact match retrieval is presented. The proposed model is based on Triangular Spatial Relationship (TSR) [Guru, D.S., Nagabhushan, P., 2001. Triangular spatial relationship: A new approach for spatial knowledge representation, Pattern Recognition Lett. 22, 999–1006]. The proposed model preserves TSR among the components in a symbolic image by the use of quadruples. A distinct and unique key called TSR key is computed for each distinct quadruple. The mean and standard deviation of the set of TSR keys computed for a symbolic image are stored along with the total number of TSR keys as the representatives of the symbolic image. An exact match retrieval scheme based on the modified binary search technique [Guru, D.S., Raghavendra, H.J., Suraj, M.G., 2000. An adaptive binary search based sorting by insertion: An efficient and simple algorithm, Statist. Appl., 2, 85–96] is also presented in this paper. The presented retrieval scheme requires O(log n) search time in the worst case, where n is the total number of symbolic images in the SID. An extensive experimentation on a large database of 13,680 symbolic images is conducted to corroborate the superiority of the model.

© 2004 Elsevier B.V. All rights reserved.

Keywords: Symbolic image; Triangular spatial relationship; Symbolic image database; Modified binary search; Exact match retrieval

1. Introduction

Retrieval of images with the desired content from a symbolic image database (SID) is a challenging and motivating research issue. However, to effectively represent/retrieve an image in/from a SID, the attributes such as symbols/icons and
their relationships which are rich enough to
describe the corresponding symbolic image are
necessary. Thus, many researchers (Chang et al., 1987,
1989; Chang and Wu, 1992; Wu and Chang, 1994;
Huang and Jean, 1994; Zhou and Ang, 1997; Zhou et al., 2001) have
highlighted the importance of perceiving spatial relationships
existing among the components of an image for
efficient representation/retrieval of symbolic
images in/from a SID. In fact, the perception of
spatial relationships preserves the reality being
embedded in physical images besides making the
system intelligent, fast and flexible.

Basically, there are two types of retrieval: simi-
nlarity retrieval and exact match retrieval. In simi-
nlarity retrieval, the task is to retrieve all those
images that are similar to a given query image,
while the exact match retrieval, retrieves only the
image exactly identical (100% similar) to a query
image from the SID.

In fact, exact match retrieval is a special case of
similarity retrieval and more precisely is an image
recognition problem. Exact match retrieval is
widely used in professional applications on indus-
trial automation, biomedicine, social security,
crime prevention and many more robotics/multi-
spectral/computer vision applications.

There have been several attempts made by the
research community to scatter the demands in
the design of efficient, invariant, flexible and intel-
ligent image archival and retrieval systems based
on the perception of spatial relationships. To make
image retrieval, visualization, and traditional image
database operations more flexible and faster, the
data structure should be object oriented. The de-
sign of such object oriented data structures began
with the discovery of 2D string (Chang et al., 1987)
representation. Based on the 2D string representa-
tion, many algorithms were proposed (Chang and
Li, 1988; Chang et al., 1989; Lee and Hsu, 1990,
1991; Chang and Lin, 1996; Chang and Ann,
1999) to represent symbolic images in a SID.

Using the concept of 2D string, in order to re-
trieve similar symbolic images from a SID, algo-
rithms (Lee et al., 1989; Lee and Shan, 1990; Lee
and Hsu, 1992) based on the longest common sub-
sequence matching were also proposed. Though,
these iconic image representation schemes offer
many advantages, the linear string representation
given to the spatial relations existing among the
components takes non-deterministic-polynomial
time complexity during the process of string match-
ing, in addition to being not invariant to image
transformations, especially to rotation. In order
to reduce the search time and to avoid string match-
ing, hash oriented methodologies for similarity re-
trieval based upon the variations of 2D string
were explored (Chang and Wu, 1992; Wu and
Chang, 1994; Bhatia and Sabharwal, 1994; Sabhar-
wal and Bhatia, 1995, 1997). However, hash func-
tion based algorithms require $O(m^2)$ search time
in the worst case for retrieval of symbolic images,
where $m$ is the number of iconic objects.

Chang (1991) proposed a symbolic indexing ap-
proach called nine directional lower triangular
(9DLT) matrix to encode symbolic images. Based
on 9DLT matrix, few models for image archival
and retrieval were developed (Chang and Wu,
1995; Zhou and Ang, 1997). In the work proposed
by Chang and Wu (1995), the pair-wise spatial rela-
tionships existing between iconic objects were
preserved with the help of nine directional codes and
were then represented in a 9DLT matrix. The first
principal component direction of the set of triplets
representing the 9DLT matrix of a symbolic image
was computed and stored as the representative of
the symbolic image in the SID. The first principal
component direction of all symbolic images were
stored in a sorted sequence, thereby enabling the
retrieval process to consume $O(\log n)$ search time
in the worst case, with the help of the binary search
 technique, where $n$ is the number of symbolic
images stored in the SID. Despite its incomparable
efficiency, the method is not robust to take care of
image transformations especially rotation.

However, one can find the existence of a few
invariant models (Petraglia et al., 1996; Zhou
et al., 2001; Guru et al., 2003) in the literature
for similarity retrieval. Although, these invariant
models proposed for similarity retrieval can be
used for exact match retrieval, it is not advisable
due to the fact that the exact match retrieval can
be achieved more efficiently and more effectively
with less computational effort and less resource
investment when compared to that of similarity
match retrieval. Hence, design of an efficient, effec-
tive and invariant model for exact match retrieval still remains as an open issue in the field of image databases.

In this paper, we present a novel scheme for representing symbolic images in a SID invariant to image transformations, useful for exact match retrieval. The proposed model is based on Triangular Spatial Relationship (TSR) (Guru and Nagabhushan, 2001). The proposed model preserves TSR among the components in a symbolic image by the use of quadruples. A distinct and unique key called TSR key is computed for each distinct quadruple. The mean and standard deviation of the set of TSR keys computed for a symbolic image are stored along with the total number of the distinct quadruple. The mean and standard deviation is the total number of symbolic images in the SID.

The remaining part of the paper is organized as follows. An overview of triangular spatial relationship (Guru and Nagabhushan, 2001) for the sake of readers is given in Section 2. Section 3 briefly about 9DLT matrix based exact match retrieval scheme and also explores a major problem with the 9DLT matrix based approaches. In Section 4, a novel invariant methodology for exact match retrieval is proposed. The results of the experiments conducted to establish the efficacy of the proposed methodology are given in Section 5. Section 6 follows with discussion and conclusion.

2. The concept of triangular spatial relationship: An overview

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

Let \( L_1, L_2, \ldots, L_m \) be the ordered sequence of the labels of components present in a symbolic image. Let \( A, B \) and \( C \) be any three non-collinear components of the 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 \( \theta_1, \theta_2, \) and \( \theta_3 \) be the smaller angles (measured in degrees) subtended at \( M_1, M_2 \) and \( M_3 \) respectively and are shown in Fig. 1. The triangular spatial relationship 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 \((L_{i1}, L_{i2}, L_{i3}, 0)\) 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
\[ L_{i2} > L_{i1} > L_{i3}. \]
2. \( L_{i1} = L_{i2} \) and \( L_{i3} < L_{i1} \).
3. \( L_{i1} > L_{i2} \) and \( L_{i2} = L_{i3} \) and \( \text{Dist}(\text{Comp}(L_{i1}), \text{Comp}(L_{i2})) \geq \text{Dist}(\text{Comp}(L_{i1}), \text{Comp}(L_{i3})). \)
4. \( L_{i1} = L_{i2} = L_{i3} \) and \( \text{Dist}(\text{Comp}(L_{i1}), \text{Comp}(L_{i2})) \geq M, \)

where
\[ M = \text{Max}(\text{Dist}(\text{Comp}(L_{i1}), \text{Comp}(L_{i3})), \text{Dist}(\text{Comp}(L_{i2}), \text{Comp}(L_{i3}))). \]

Here,
\[ \text{Dist}(A, B) \] is a function which computes the Euclidean distance between the midpoints of the components \( A \) and \( B \). \( \text{Max}(a, b) \) is a function denoting the maximum among \( a \) and \( b \) and \( \text{Comp}(L) \) indicates the component, the label of which is \( L \).
It is guaranteed that, even if more than one possible permutation of the components satisfies the above-stated conditions, 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 \((L_{11}, L_{12}, L_{13}, \theta)\), where the sequence of \( L \)'s satisfies one of the above stated conditions and \( \theta \) is the smaller angle (measured in degrees) subtended at the midpoint of the components, the labels of which are \( L_{11} \) and \( L_{12} \) due to the line joining that midpoint and the centroid of the remaining component, the label of which is \( L_{13} \). The \( \theta \) is given by

\[
\theta = \begin{cases} 
\theta_1 & \text{if } \theta_1 \leq 90^\circ \\
180 - \theta_1 & \text{otherwise}
\end{cases}
\]

where

\[
\theta_1 = \cos^{-1}\left(\left(S_1^2 - S_2^2 - S_3^2\right)/\left(2 * S_2 * S_3\right)\right),
\]

shown in Fig. 2. We may use nine directional codes shown in Fig. 3 to represent the pair-wise spatial relationships between \( x \), a referenced component and \( y \), a contrasted component. The directional code say \( r = 0 \), represents that \( y \) is to the east of \( x \), \( r = 1 \) represents that \( y \) is to the north-east of \( x \), and so on. Thus, the 9DLT matrix \( T \) for the symbolic image of Fig. 2 is as shown in Fig. 4. Since each relationship is represented by a single triplet \((x, y, r)\) the 9DLT matrix is a lower triangular matrix.

The 9DLT matrix can now be formally defined as follows (Chang, 1991). Let \( V = \{v_1, v_2, v_3, v_4, \ldots, v_m\} \) be a set of \( m \) distinct components/objects. Let \( Z \) consist of ordered components \( z_1, z_2, z_3, \ldots, z_s \) such that, \( \forall i = 1, 2, \ldots, s, z_i \in V \). Let \( C \) be the set of nine directional codes as defined in Fig. 3. Each directional code is used to specify the spatial relationship between two components. So, a 9DLT matrix \( T \) is an \( s \times s \) matrix over \( C \) in which \( t_{ij} \), the \( i \)th row and \( j \)th column element of

3. A review of 9DLT matrix based exact match retrieval scheme

This section reviews the exact match retrieval scheme proposed by Chang and Wu (1995) and explores a major problem with the 9DLT matrix based approaches.

3.1. Nine directional lower triangular (9DLT) matrix: definition

Consider a symbolic image consisting of four components with labels \( L1, L2, L3 \) and \( L4 \) as shown in Fig. 2. We may use nine directional codes shown in Fig. 3 to represent the pair-wise spatial relationships between \( x \), a referenced component and \( y \), a contrasted component. The directional code say \( r = 0 \), represents that \( y \) is to the east of \( x \), \( r = 1 \) represents that \( y \) is to the north-east of \( x \), and so on. Thus, the 9DLT matrix \( T \) for the symbolic image of Fig. 2 is as shown in Fig. 4. Since each relationship is represented by a single triplet \((x, y, r)\) the 9DLT matrix is a lower triangular matrix.

The 9DLT matrix can now be formally defined as follows (Chang, 1991). Let \( V = \{v_1, v_2, v_3, v_4, \ldots, v_m\} \) be a set of \( m \) distinct components/objects. Let \( Z \) consist of ordered components \( z_1, z_2, z_3, \ldots, z_s \) such that, \( \forall i = 1, 2, \ldots, s, z_i \in V \). Let \( C \) be the set of nine directional codes as defined in Fig. 3. Each directional code is used to specify the spatial relationship between two components. So, a 9DLT matrix \( T \) is an \( s \times s \) matrix over \( C \) in which \( t_{ij} \), the \( i \)th row and \( j \)th column element of

![Fig. 2. A symbolic image.](image1)

![Fig. 3. The nine directional codes.](image2)

![Fig. 4. The 9DLT matrix of Fig. 2.](image3)
T is the directional code of Zj to Zi if j < 1, and undefined otherwise. The matrix T is a 9DLT matrix according to the ordered set Z.

3.2. Exact match retrieval scheme (Chang and Wu, 1995)

Using the concept of 9DLT matrix, Chang and Wu (1995) proposed an exact match retrieval scheme based upon Principal Component Analysis (PCA). The 9DLT matrix (Fig. 4) is represented by a set of triplets, \{(L1, L2, 7), (L1, L3, 7), (L1, L4, 7), (L2, L3, 6), (L2, L4, 0), (L3, L4, 1)\} or simply \{(1, 2, 7), (1, 3, 7), (1, 4, 7), (2, 3, 6), (2, 4, 0), (3, 4, 1)\}. The first Principal Component Vector (PCV), \((-0.1977, -0.1568, 0.9676)\), of the above set of triplets was found and stored in SID as the representative of the symbolic image.

Thus, the retrieval of a symbolic image requires one to:

- construct the 9DLT matrix for the given symbolic image;
- find out the first PCV (say D) of the set of triplets representing the 9DLT matrix;
- search for D in SID;
- extract the image index associated with D.

3.3. A problem in 9DLT matrix based approaches

Let us assume that a rotated version (Fig. 5) of the symbolic image shown in Fig. 2 is given as input during the retrieval phase. For the sake of simplicity, the angle of rotation is taken, in this example, as \(-90^\circ\). The corresponding 9DLT matrix of the rotated symbolic image is shown in Fig. 6 and the corresponding set of triplets is \{(1, 2, 5), (1, 3, 5), (1, 4, 5), (2, 3, 4), (2, 4, 6), (3, 4, 7)\}. The associated first PCV is \((0.5048, 0.4705, 0.7237)\). It is clearly observed that the first PCVs associated with 9DLT matrices shown in Figs. 4 and 6 are not identical as their corresponding sets of triplets are totally different even though they represent the same physical image. This problem is due to the fact that the directional codes are not invariant to rotation. Instead of considering the pair wise spatial relationships directly between two components independent of other components present in the image, if the triangular spatial relationship (TSR) is perceived, then the problem of getting different interpretations for the symbolic images representing the same physical image in different orientations can be resolved. Thus, an invariant scheme for exact match retrieval based on TSR is proposed and explained in the following section.

4. The proposed methodology

The proposed scheme has two stages. The first stage proposes a novel method of representing symbolic images invariant to image transformations in the SID, while the second stage suggests a corresponding exact match retrieval scheme for a given query image invariant to image transformations.

4.1. Representation of symbolic images in SID

The proposed representation scheme computes for each image, a set of TSR keys, through the perception of triangular spatial relationship. Subsequently, the mean and standard deviation of the set of keys are stored in SID along with the total number of keys generated for the image, as the image representatives. Thus, the following are the...
major steps involved in the proposed representation scheme.

4.1.1. Computation of TSR keys

Let \( \{S_1, S_2, S_3, \ldots, S_n\} \) be a set of \( n \) symbolic images to be archived in the SID. Let \( L_1, L_2, L_3, \ldots, L_m \) be the labels of \( m \) distinct iconic objects called components. Here each \( L_i \) is an integer and \( 1 \leq L_i \leq m \). 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 \( m_i \leq m \) number of labels. However, transformation of a physical image into its corresponding symbolic image is intentionally kept 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 Section 2. 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. But, storing the quadruples themselves is unwieldy and makes retrieval cumbersome. Hence, we recommend to compute a unique and distinct real number called TSR key for each distinct quadruple. Computation of unique TSR key for a quadruple, not only makes the task of retrieval easier, but also, minimizes the space complexity from \( O(4m^3) \) to \( O(m^3) \).

If \( (L_a, L_b, L_c, 0) \) is a quadruple then the key \( K \) corresponding to the quadruple is computed as

\[
K = D_0(L_a - 1)m^2 + D_0(L_b - 1)m + D_0(L_c - 1) + \theta,
\]

where \( D_0 \) is the allowable maximum value for \( \theta \) and here \( D_0 = 90 \). It should be noticed that Eq. (1) associates a quadruple with exactly one key (unique) and hence it is a mapping from the set of TSR quadruples to the set of TSR keys. In addition, it can also be noticed that the mapping defined by Eq. (1) is one-one. That is, the TSR keys associated with two different quadruples are distinct and unique.

**Statement:** Given two integers \( m \) and \( D_0 \), the mapping defined by Eq. (1) is one to one from the set of quadruples \( \{(L_a, L_b, L_c, 0)|L_a, L_b, L_c \text{ are non-zero positive integers less than or equal to } m \text{ and } 0 < \theta \leq D_0\} \) to the set of TSR keys.

**Proof.** Let \( (L_a, L_b, L_c, 0) \) and \( (L'_a, L'_b, L'_c, \theta^1) \) be two distinct quadruples associated with the same key generated by Eq. (1).

\[
m^2D_0(L_a - 1) + mD_0(L_b - 1) + D_0(L_c - 1) + \theta = m^2D_0(L'_a - 1) + mD_0(L'_b - 1) + D_0(L'_c - 1) + \theta^1,
\]

i.e.,

\[
m^2D_0(L_a - L'_a) + mD_0(L_b - L'_b) + D_0(L_c - L'_c) + (\theta - \theta^1) = 0.
\]

Let \( L_a - L'_a = X_1; L_b - L'_b = X_2; L_c - L'_c = X_3; \) and \( \theta - \theta^1 = X_4; \) now (3) becomes,

\[
m^2D_0X_1 + mD_0X_2 + D_0X_3 + X_4 = 0,
\]

Here, \( X_1, X_2, X_3 \) are integers and \( |X_1|, |X_2|, |X_3| \leq (m - 1),
\]

\[
(\because L_a, L_b, L_c, 1 \text{ are integers and } 1 \leq L_a, L_b, L_c, X_1, X_2, X_3 \leq m \text{ and } |X_4| < D_0).
\]

\[
|X_4| < D_0,
\]

\[
(\because \theta \text{ and } \theta^1 \text{ are real numbers and } 0 < \theta, \theta^1 \leq D_0).
\]

By rewriting (4) we get,

\[
D_0(m^2X_1 + mX_2 + X_3) = -X_4.
\]

Notice here that \( X_4 \) must be an integer and a multiple of \( D_0 \).

Therefore, because of (6), we must have,

\[
X_4 = 0.
\]

Thus, (7) reduces to

\[
m^2X_1 + mX_2 + X_3 = 0,
\]

i.e.,

\[
m(mX_1 + X_2) = -X_3.
\]
From (10) and (5), it is understood that \( X_3 \) is a multiple of \( m \) and \(- (m - 1) \leq X_3 \leq (m - 1)\). Hence,

\[
X_3 = 0.
\]

Hence, (10) reduces to

\[
mX_1 + X_2 = 0, \tag{12}
\]

i.e.,

\[
mX_1 = -X_2. \tag{13}
\]

From (13) and (5), it is understood that \( X_2 \) is also a multiple of \( m \) and \(- (m - 1) \leq X_2 \leq (m - 1)\). Therefore,

\[
X_2 = 0. \tag{14}
\]

From (13) and (14) we get,

\[
X_1 = 0. \tag{15}
\]

From (8), (11), (14) and (15), we have,

\[
L_a - L_a^1 = 0; \quad L_b - L_b^1 = 0; \quad L_c - L_c^1 = 0; \quad \text{and} \quad \theta - \theta^1 = 0,
\]

i.e., \( L_a = L_a^1; \quad L_b = L_b^1; \quad L_c = L_c^1; \quad \theta = \theta^1 \).

This contradicts our assumption that the quadruples \( (L_a, L_b, L_c, \theta) \) and \( (L_a^1, L_b^1, L_c^1, \theta^1) \) are distinct. Hence the proof. \( \square \)

### 4.1.2. Creation of symbolic image database system

Let \( P_q = \{q_1, q_2, \ldots, q_3\} \) be the set of \( N \) distinct quadruples and \( P_k = \{K_1, K_2, K_3, \ldots, K_N\} \) be the set of corresponding TSR keys generated for a symbolic image \( S \). The set \( P_k \) can itself be stored in the SID for the matching process at the time of retrieval. However, it could still be unwieldy as the size of the set \( P_k \) is \( O(m^3) \) in the worst case. Therefore, to further reduce the storage requirement, we suggest to compute the mean \( \mu \) and the standard deviation \( \sigma \) of the set \( P_k \) and then to store the triplet \( (N, \mu, \sigma) \) as the representative vector of the symbolic image \( S \) in the SID. Thus, the storage requirement for a symbolic image has further been reduced to only three real numbers (i.e., \( O(3) \)). Hence, for all \( n \) images to be archived in the SID, the triplets \( (N, \mu, \sigma) \) are computed and stored in a sorted sequence so that binary search can be employed during retrieval. The proposed representation scheme not only reduces the size of the SID, but also enhances the efficacy of the retrieval process requiring only \( O(\log n) \) search time in the worst case.

In spite the method is theoretically claimed to be invariant, due to the limitations of the computing system in handling floating point numbers and also because of rotation errors, the components \( \mu \) and \( \sigma \) of the representative vector \( (N, \mu, \sigma) \) of a symbolic image cannot be expected to remain entirely invariant but to lie within a certain range. Since the representative vector has three values, each rotated instance of a symbolic image can be looked upon as a point in 3-dimensional Euclidean space \( R^3 \). Therefore, the set of all rotated instances of a symbolic image defines in this manner a subspace of \( R^3 \) and the centroid of that subspace is chosen as the actual representative vector of the symbolic image in SID.

The following algorithm has thus been devised to create a SID for a given set of symbolic images useful for exact match retrieval.

**Algorithm.** Creation_of_SID  
**Input:** Set of symbolic images  
**Output:** SID, Symbolic Image Database  
**Method:**

**Step 1:** For each symbolic image \( S \) to be archived in the SID do  
For each rotated instance \( R_i \) of \( S \) do  
(i) Apply TSR as explained in Section 2 and obtain a set of quadruples \( P_q \) preserving TSR among the components present in \( R_i \).  
(ii) For each quadruple in \( P_q \), compute a unique TSR key using Eq. (1).  
(iii) Compute the vector \( D = (N, \mu, \sigma) \) where \( N \) is the total number of TSR keys, \( \mu \) is the mean and \( \sigma \) is the standard deviation of the TSR keys generated.
Compute the representative vector $C_s$ (for the symbolic image $S$) which is the centroid of all $D_s$ computed for $S$.

**Step 2:** Store the centroids obtained for all images in a sorted sequence.

**Creation of SID ends.**

It should be noticed that creation of a SID is an offline process and consideration of several instances of the same symbolic image in different orientations helps in recording the possible variations in the components of its representative vector, so that the centroid can be chosen as the best representative vector of the image. Indeed, consideration of several instances at the time of SID creation does neither increase the storage requirement (still it takes $O(3)$) nor the retrieval time.

### 4.2. Exact match retrieval of symbolic images from SID

Exact match retrieval is an image retrieval process where a symbolic image $S$ is retrieved as an exact match to a given query image $Q$, if and only if both $S$ and $Q$ are identical. Since each symbolic image $S$ is represented in terms of a vector, which is the average vector of the vectors computed for all rotated instances of $S$, the retrieval process reduces to a problem of searching for, if not an exact, at least a nearest neighbor for the computed vector $D_q$ of the query image $Q$ in the SID. Since the vectors in SID are stored in a sorted order, the desired image can be retrieved in $O(\log n)$ search time, by employing the modified binary search technique (Guru et al., 2000) where, $n$ is the number of images stored in the SID. The modified binary search technique searches for two successive vectors which bound $D_q$ in the SID. Once such two vectors are found, their distances to $D_q$ are computed and the image corresponding to the vector which is nearer to $D_q$ is retrieved from the SID as the desired image.

### Algorithm.

**Exact match retrieval**

**Input:** $Q$, a symbolic query image

**SID, Symbolic Image Database**

**Output:** Desired image

**Method:**

1. **Step 1:** Preserve TSR existing among the components of $Q$ by the use of quadruples.
2. **Step 2:** For each quadruple, compute the corresponding TSR key using Eq. (1).
3. **Step 3:** Compute the vector $D_q = (N, \mu, \sigma)$ as explained in Section 4.1.2.
4. **Step 4:** Employ the modified binary search technique to find out the two adjacent vectors $D_i$ and $D_{i+1}$, such that $D_i \leq D_q \leq D_{i+1}$.
5. **Step 5:** Find out the distances, $d_1$ and $d_2$ of $D_q$ to $D_i$ and $D_{i+1}$ respectively.
6. **Step 6:** Retrieve the symbolic image corresponding to the index $\begin{cases} i & \text{if } d_1 < d_2 \\ i + 1 & \text{otherwise} \end{cases}$.

**Exact match retrieval ends.**

### 5. Experimental results

To corroborate the efficacy of the proposed methodology, we have conducted several experiments on various symbolic images of both model and real images. Out of them we present only five experiments here. For all these experiments the number of distinct components $m$ is fixed up to be 8 and $D_0$ is fixed up to be 90.

#### 5.1. Experimentation 1

We have conducted an experiment on the symbolic images (see Fig. 7f1–f4) considered by Chang and Wu (1995). During representation, each symbolic image is considered in 76 different orientations, out of which 72 are rotated instances (at $5^\circ$ regular interval) and 4 are scaled versions with $\pm10\%$ and $\pm20\%$ scaling factors. For each instance of a symbolic image, the triangular spatial relationships existing among the components are perceived and the TSR keys are generated.
For example, the set of quadruples preserving the triangular spatial relationship among the components of the symbolic image shown in Fig. 7f1 is $P_{q0} = \{(4, 2, 1, 26.021120), (3, 2, 1, 26.021120), (4, 3, 1, 90.000000), (4, 3, 2, 90.000000)\}$. As there are four components, the set $P_{q0}$ has got $4C_3 = 4$ quadruples. The set $P_{q0}$ represents logically the symbolic image (Fig. 7f1). The set $P_{q0}$ could itself be stored in SID, but storing the set $P_{q0}$ itself is unwieldy as the size of $P_{q0}$, in general, is $O(4^n)$ in the worst case. Hence, for each quadruple in $P_{q0}$, a TSR key is computed as explained in Section 4.1.1 and the set of computed TSR keys is $P_{k0} = \{18026.021120, 12266.021120, 18810.000000, 18900.000000\}$. Though these TSR keys can be directly stored, it is not advisable due to the fact that, the corresponding retrieval takes $O(nm^3)$ search time in the worst case, where $n$ is the number of symbolic images. Therefore, a vector $D_0 = (4, 17000.511, 3180.6380)$ is computed as explained in Section 4.1.2. Likewise, the sets $P_{k1}, P_{k2}, \ldots, P_{k75}$ corresponding to the remaining 75 instances of the symbolic image (Fig. 7f1) are generated and the corresponding vectors $D_1, D_2, \ldots, D_{75}$ are computed. The span due to variations in the three components of these $D$’s is (4–4, 17000.152–17000.637, 3179.4776–3180.9431). As suggested in Section 4.1.2, the centroid (4, 17000.3945, 3180.21035) is computed and stored as the representative vector of the image (Fig. 7f1). It has to be noticed here that, only this centroid vector is stored as a representative of all instances of the symbolic image Fig. 7f1. Table 1 gives the span in vector components computed in a similar manner for all four symbolic images shown in Fig. 7 and Table 2 shows the centroids (representative vectors) of the symbolic images stored in a sorted sequence. One can notice that the centroids are distinct.

Let the symbolic image shown in Fig. 8Q be the query image. This query image is an arbitrarily rotated instance, which is not considered during representation, of the symbolic image shown in Fig. 7f1. The set of generated quadruples preserving the triangular spatial relationship existing among the components is {$(4, 2, 1, 25.783132), (3, 2, 1, 26.617002), (4, 3, 1, 89.863744), (4, 3, 2, 89.555855)$} and the set of corresponding TSR keys is {$(18025.783132, 12266.617002, 18809.863744, 18899.000000)$}.

Table 1
Span in vectors $(N, l, r)$ for the symbolic images shown in Fig. 7

<table>
<thead>
<tr>
<th>Image index</th>
<th>First component</th>
<th>Second component</th>
<th>Third component</th>
</tr>
</thead>
<tbody>
<tr>
<td>f1</td>
<td>4</td>
<td>17000.152–17000.637</td>
<td>3179.4776–3180.9431</td>
</tr>
<tr>
<td>f2</td>
<td>4</td>
<td>17000.260–17001.057</td>
<td>3171.8722–3172.6999</td>
</tr>
<tr>
<td>f3</td>
<td>4</td>
<td>16976.272–16977.643</td>
<td>3150.8453</td>
</tr>
<tr>
<td>f4</td>
<td>1</td>
<td>18809.308–18810.000</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 2
Representative vectors of the symbolic images shown in Fig. 7 in a sorted sequence

<table>
<thead>
<tr>
<th>Image index</th>
<th>First component</th>
<th>Second component</th>
<th>Third component</th>
</tr>
</thead>
<tbody>
<tr>
<td>f4</td>
<td>1</td>
<td>18809.654</td>
<td>0</td>
</tr>
<tr>
<td>f3</td>
<td>4</td>
<td>16976.9575</td>
<td>3150.8453</td>
</tr>
<tr>
<td>f1</td>
<td>4</td>
<td>17000.3945</td>
<td>3180.21035</td>
</tr>
<tr>
<td>f2</td>
<td>4</td>
<td>17000.6585</td>
<td>3172.28605</td>
</tr>
</tbody>
</table>

Fig. 7. Images taken from Chang and Wu (1995).
Thus, the representative vector computed for the query is $D_q = (4, 17000.00, 3180.20)$. The modified binary search technique is employed to search two successive vectors bounding $D_q$. It is found that the vector $D_q$ lies in between that of the images $f_3$ and $f_1$ and their distances from $D_q$ are respectively 6.10887 and 0.6282. Since the distance between $D_q$ and $f_1$ is less than the distance between $D_q$ and $f_3$, the image $f_1$ is retrieved and hence $f_1$ is the desired image.

Consider the symbolic image shown in Fig. 8 which is a rotated instance of Fig. 7f1. The vector $D_q' = (4, 16997.61523, 3177.81909)$ is the vector computed for the symbolic image (Fig. 8Q'), through the perception of the TSR among its components. The application of the modified binary search technique gives us the representative vectors of the symbolic images $f_3$ and $f_1$ as the vectors bounding the vector $D_q'$. Since the distance between $D_q'$ and $f_1$ is less than the distance between $D_q'$ and $f_3$, the image $f_1$ is retrieved and hence $f_1$ is the desired image.

It should be noticed that the vectors computed for the images Fig. 8Q and Q', though are not identical, the retrieved symbolic image (Fig. 7f1) is one and the same as the images (Fig. 8Q and Q') are the different instances of Fig. 7f1. This is the advantage of the centroid based representation and the application of the modified binary search technique.

5.2. Experimentation 2

We have considered the symbolic images (Fig. 9) of four real keys extracted from (Guru, 2000) and computed the representative vectors shown in Table 3. Similar to that of the previous experimentation, we have considered 76 different instances of each image during representation. We have intentionally chosen this set of symbolic vectors.

<table>
<thead>
<tr>
<th>Image index</th>
<th>First component</th>
<th>Second component</th>
<th>Third component</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>918</td>
<td>26522.6515</td>
<td>3033.6706</td>
</tr>
<tr>
<td>a</td>
<td>926</td>
<td>26582.0875</td>
<td>2911.6274</td>
</tr>
<tr>
<td>d</td>
<td>1070</td>
<td>26573.1025</td>
<td>3007.9421</td>
</tr>
<tr>
<td>c</td>
<td>1079</td>
<td>26584.7115</td>
<td>3003.7666</td>
</tr>
</tbody>
</table>

Fig. 8. (Q) A rotated instance of Fig. 7f1. (Q') A 5° rotation combined with 10% scaled down instance of Fig. 7f1.

Fig. 9. Symbolic images of four real keys taken from Guru (2000).
images to establish the high reliability of the proposed method in retrieving the desired image even though the symbolic images (Fig. 9) almost look alike or are made up of same number of iconic objects with almost same spatial scattering. In fact, sometimes, even our vision system fails in distinguishing some of the rotated instances of these symbolic images.

The robustness of the proposed retrieval scheme is validated by conducting several experiments on rotated instances of the symbolic images (Fig. 9a–d) and is observed that the desired images are retrieved for all query images.

5.3. Experimentation 3

Symbolic images in Fig. 10 are extracted from (Guru, 2000). They are the symbolic images obtained for 10 planar objects. In this experiment also, we have generated 76 different instances for each image during representation. The computed representative vectors of the images (Fig. 10P0–P9) are as shown in Table 4. One should note that all vectors in Table 4 are distinct except (P6) and (P9) as (P6) is a rotated instance of (P9). The SID is created by eliminating such duplicates and storing them in a sorted sequence.

![Fig. 10. Symbolic images of 10 planar objects taken from Guru (2000).](image-url)
In this experimentation also, we have tested the efficacy of our retrieval scheme with several rotated instances of the symbolic images (Fig. 10) and obtained the desired results.

5.4. Experimentation 4

Symbolic images in Fig. 11 are extracted from (Guru et al., 2003). Similar to the other experiments here also, 76 different instances are generated for each image during representation. The representative vectors computed for the images (Fig. 11S1–S5) are as shown in Table 5. In this experimentation also, it is found that the retrieval scheme is robust and yields desired results.

5.5. Experimentation 5

The efficacy of the proposed methodology in retrieving an appropriate symbolic image has been further examined by storing the representative vectors of all the images (union of all images considered for experimentations 1, 2, 3 and 4) in a single database. All the representative vectors are stored in a sorted sequence (see Table 6). This database is created by considering a total of 1748 instances due to all symbolic images (Figs. 7, 9, 10 and 11). The representative vector of P9 is eliminated as it is same as that of P6. It can be noticed that the representative vectors are distinct and unique. It is also observed that the performance of the proposed retrieval scheme is accurate in retrieving the appropriate symbolic image of interest, for a given query symbolic image.

6. Discussion and conclusion

Perception and representation of invariant spatial relationships existing among iconic objects present in a symbolic image indeed helps in preserving the reality being embedded in the symbolic image. Devising schemes which are invariant, fast, flexible and good in preserving the reality being embedded in images is a challenging task in the

---

**Table 5**

Representative vectors for the symbolic images shown in Fig. 11 in a sorted sequence

<table>
<thead>
<tr>
<th>Image index</th>
<th>First component</th>
<th>Second component</th>
<th>Third component</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>4</td>
<td>17006.272</td>
<td>3154.86835</td>
</tr>
<tr>
<td>S2</td>
<td>10</td>
<td>21701.464</td>
<td>4457.7141</td>
</tr>
<tr>
<td>S3</td>
<td>10</td>
<td>27973.784</td>
<td>4685.3539</td>
</tr>
<tr>
<td>S4</td>
<td>19</td>
<td>39889.422</td>
<td>6346.5395</td>
</tr>
<tr>
<td>S5</td>
<td>19</td>
<td>33017.6235</td>
<td>5903.57815</td>
</tr>
</tbody>
</table>

---

Fig. 11. Symbolic images taken from Guru et al. (2003).
field of image databases. In fact, these are the shortcomings existing in almost all the methodologies proposed so far.

Similarity match and exact match retrieval are the two major issues related to any image database. Similarity retrieval task deals with retrieving all those images that are similar to the given query image from the SID, while exact match retrieval process retrieves from SID, only those images, exactly identical to the query image and is more likely an image recognition problem.

Though, many models were devised for similarity retrieval, only a few of them are invariant to image transformations. The similarity retrieval models claimed as invariant to image transformations are less efficient from the point of view of their usage for exact match retrieval as exact match retrieval can be achieved more efficiently and more effectively with less computational effort and less resource investment when compared to that of similarity retrieval. To the best of our knowledge only one model (Chang and Wu, 1995) has been proposed for exact match retrieval but it is not invariant to image transformations specifically to rotation.

In view of this, in this paper, we have made a successful attempt in exploring a model which overcomes the aforementioned shortcomings and best suits exact match retrieval. The paper presents a novel way of representing a symbolic image in SID invariant to image transformations through the perception of triangular spatial relationships. The triangular spatial relationship is preserved in terms of quadruples which are then mapped to unique TSR keys. The mean and standard deviation of the set of TSR keys are computed and stored in the SID along with the total number of TSR keys as the representative vector of the image in a sorted sequence. This representation not only makes the task of retrieval easier but also achieves reduction in memory requirement at two levels: at the level of TSR key computation and also at the level of computation of representative triplets. Since the proposed retrieval scheme works based on the modified binary search technique, it requires only $O(\log n)$ search time in the worst case.

The proposed model integrates the representation of an image with the retrieval of an image. Unlike, other models, our model automatically takes care of additional information such as angles and is invariant to image transformations as it is based on triangular spatial relationship. In addition, it takes care of multiple instances of objects, which is considered to be a major problem in most of the existing methodologies.

Further, as the 9DLT matrix based approach (Chang and Wu, 1995) is based on principal component analysis (PCA) and the transformation of a set of triplets to the first principal component vector (PCV) for different set of triplets is not biunivocal, there may be the same PCV generated for different symbolic images. Hence, two entirely different symbolic images may have the same PCV. Under such conflicting situations, one has to employ second PCV for resolution, and go up to the third PCV. If all the PCVs associated with two or more symbolic images are same then the conflict in discriminating such symbolic images could be resolved.

<table>
<thead>
<tr>
<th>Table 6</th>
<th>Representative vectors computed for all the symbolic images of Figs. 7, 9, 10 and 11</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Image Index</strong></td>
<td><strong>First component</strong></td>
</tr>
<tr>
<td>f4</td>
<td>1</td>
</tr>
<tr>
<td>f3</td>
<td>4</td>
</tr>
<tr>
<td>f1</td>
<td>4</td>
</tr>
<tr>
<td>f2</td>
<td>4</td>
</tr>
<tr>
<td>S1</td>
<td>10</td>
</tr>
<tr>
<td>S2</td>
<td>10</td>
</tr>
<tr>
<td>S3</td>
<td>10</td>
</tr>
<tr>
<td>S4</td>
<td>19</td>
</tr>
<tr>
<td>P1</td>
<td>76</td>
</tr>
<tr>
<td>P3</td>
<td>76</td>
</tr>
<tr>
<td>P7</td>
<td>76</td>
</tr>
<tr>
<td>P4</td>
<td>76</td>
</tr>
<tr>
<td>P5</td>
<td>76</td>
</tr>
<tr>
<td>P0</td>
<td>76</td>
</tr>
<tr>
<td>P2</td>
<td>76</td>
</tr>
<tr>
<td>P6</td>
<td>76</td>
</tr>
<tr>
<td>P8</td>
<td>76</td>
</tr>
<tr>
<td>b</td>
<td>918</td>
</tr>
<tr>
<td>a</td>
<td>926</td>
</tr>
<tr>
<td>d</td>
<td>1070</td>
</tr>
<tr>
<td>c</td>
<td>1079</td>
</tr>
</tbody>
</table>
by storing the associated triplets themselves which inevitably entails additional memory. This drawback is overcome in our methodology as the vector \((N, l, r)\) uniquely represents a given image (set of TSR keys in our case). In addition, the usage of modified binary search algorithm during retrieval, requires only \(O(\log n)\) search time even in the worst case to retrieve an exact matched symbolic image from the SID.

A comparison of the proposed model with some of the other models is given in Table 7.

It should be noticed that the proposed methodology concentrates only on exact match retrieval of symbolic images from a SID. The task of transforming a physical image into its corresponding symbolic image is itself a research topic (Chang and Wu, 1995). However, a few interesting attempts towards the transformation of physical image to a corresponding symbolic image can be found in (Guru, 2000).

In summary, an invariant model for exact match retrieval of symbolic images from a SID is proposed in this paper. The major problem in the 9DLT matrix based approaches is discussed. Unlike Chang and Wu’s (1995) method, the proposed model is invariant to image transformations and the beauty of our scheme lies in its efficiency from the point of view of retrieval time (as it is of logarithmic time complexity). The efficacy of the proposed methodology is experimentally established by considering a large database of 13,680 symbolic images.

### References


