Geometry-Based Image Retrieval in Binary Image Databases

Naif Alajlan, Mohamed S. Kamel, *Fellow*, *IEEE*, and George H. Freeman

Abstract—In this paper, a geometry-based image retrieval system is developed for multiobject images. We model both shape and topology of image objects using a structured representation called curvature tree (CT). The hierarchy of the CT reflects the inclusion relationships between the image objects. To facilitate shape-based matching, triangle-area representation (TAR) of each object is stored at the corresponding node in the CT. The similarity between two multiobject images is measured based on the maximum similarity subtree isomorphism (MSSI) between their CTs. For this purpose, we adopt a recursive algorithm to solve the MSSI problem and a very effective dynamic programming algorithm to measure the similarity between the attributed nodes. Our matching scheme agrees with many recent findings in psychology about the human perception of multiobject images. Experiments on a database of 13,500 real and synthesized medical images and the MPEG-7 CE-1 database of 1,400 shape images have shown the effectiveness of the proposed method.

Index Terms—Geometry-based image retrieval, shape matching, attributed tree matching, medical image retrieval.

1 Introduction

The retrieval of images from image databases using generic image features such as shape has become increasingly important due to the rapid advances in imaging technologies. In large image databases, textual annotation of images becomes impractical and inefficient for image description and retrieval. Thus, content-based image retrieval (CBIR) has received considerable attention in recent years from researchers in various fields [40], [45]. For comparing images, CBIR uses generic image features that are traditionally either intensity based (color and texture) or geometry based (shape and topology). Many systems have been proposed for CBIR; among the most popular ones are QBIC from IBM [14], Virage [3], and Photobook from MIT [35]. Geometry-based image retrieval is generally less developed than the intensity-based retrieval; for example, the QBIC system is more successful in the intensity-based than in the geometry-based search [44].

In this paper, a structural representation, called the Culvature tree (CT), is introduced as a tool for multiobject image representation and matching. Our aim is to provide a unified framework for geometry-based image retrieval that includes the shape and topology of objects and holes composing an image.[1] More specifically, the CT is a hierarchical data structure that reflects the inclusion relationships among objects and holes. It consists of nodes and edges, where each node stores shape descriptor of the closed boundary corresponding to an object or hole. For measuring the similarity between two multiobject images, their CTs are matched based on the notion of maximum similarity subtree isomorphism (MSSI). This matching scheme is highly desirable since it handles both shape and topology at once. More importantly, our matching scheme comes in accordance with many recent findings from the visual cognition community about comparing multiobject images (see Section 3). The performance of the proposed approach is evaluated using a database of 13,500 real and synthesized medical images and the MPEG-7 CE-1 database of 1,400 silhouettes.

The main contribution of the work presented in this paper is achieving higher retrieval accuracy than all published methods based on the MPEG-7 CE-1 shape retrieval test and the medical image retrieval test in [36]. In addition, our matching scheme agrees with recent findings in psychology about the human perception of multiobject images. The rest of the paper is organized as follows: Section 2 reviews the related work in the literature. Then, a review of recent psychological findings is presented in Section 3. The CT is introduced in Section 4; Section 5 explains the matching algorithm followed by comments on the computational complexity in Section 6. Section 7 presents experimental results, and finally, we conclude our work in Section 8.

1. In this paper, we assume the objects in an image are already segmented, and their boundaries are well identified. Therefore, binary images are considered.

- *N. Alajlan is with the Department of Electrical Engineering, Engineering College, King Saud University, PO Box 800, Riyadh 11421, Saudi Arabia. E-mail: najlan@ksu.edu.sa.*
- *M.S. Kamel and G.H. Freeman are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1. E-mail: mkamel@uwaterloo.ca, G.Freeman@ece.uwaterloo.ca.*

Manuscript received 6 Mar. 2007; revised 2 Aug. 2007; accepted 14 Jan. 2008; published online 6 Feb. 2008.
Recommended for acceptance by B.S. Manjunath.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-03-0152.
Digital Object Identifier no. 10.1109/TPAMI.2008.37.

2 Related Work

Geometry-based image retrieval has received considerable attention from researchers in many areas in the past few decades. One of the earliest attempts in this field is based on image features extracted from the whole image, i.e., global features, such as invariant moments [17], Zernike moments [19], [21], geometric features [13], combination of invariant moments and edge directions histograms [18], and the angular radial transform (ART) descriptor [20], which is the MPEG-7 region-based shape descriptor. While the global features have the advantages of being compact and efficient for matching, local information and spatial configuration of

image objects are lost. To overcome this problem, another class of methods extracts features for individual objects and then combines these features into a single distance measure for matching. In [34], a binary image is decomposed into a set of closed contours, where each contour is represented as a chain code. To measure the similarity between two images, the distances between each contour in one image and all contours in the other are computed using a string matching technique. Then, a weighted sum of the average and maximum of these distances constitutes the final similarity. Another approach was suggested in [41] using global feature vectors of regions of connected components. Here, the similarity function is the average of the pairwise similarities between each region in one image, and the closest region in the other, which allows many-to-one matching between the regions in the two images. In an attempt to ensure a one-to-one matching, Eakins et al. [10], [11] suggested a predefined ordering of the image objects based on their sizes and positions prior to the matching. Then, the overall distance is a weighted sum of the distances between global image features as well as object-based features following the predefined ordering in matching the objects. While these methods overcome some of the limitations of the global methods by including the contribution of each object in the final similarity function, they still ignore the topological structure of the image and do not account for the unmatched objects in the similarity measure.

A recent class of the geometry-based image retrieval approaches focuses on the spatial relationships between image objects in measuring the similarity [7], [12], [15]. Basically, spatial methods work on symbolic images,[2] which limits their use in fully automated systems. Although these methods have many application domains such as geographical information systems (GIS), spatial methods ignore the shape information. In [4], a graph structure, called concavity graph, represents multiobject images using individual objects and their spatial relationships; however, matching concavity graphs is still an open problem. One of the most successful methods for the spatial image retrieval is based on the attributed relational graph (ARG) [36], [38]. The ARG is a fully connected graph, where nodes represent objects or regions of an image, and edges reflect spatial relationships between the objects. In [36], two ARG-matching methods were applied. The first is a graph edit distance, which is the cost of the least-cost sequence of edit operations on the nodes and the edges that transforms one ARG to the other. The other method formulates the ARG matching as a bipartite matching problem between two sets of nodes and uses the Hungarian method to solve it. The ARG method outperformed the 2D strings method [8] in terms of retrieval accuracy; however, the latter has less computational complexity.

For 2D shape matching, a huge number of approaches has been reported in the literature [24], [47]. One of the most well-researched closed-contour shape representations is the curvature scale space (CSS) method proposed by Mokhtarian et al. [27], [28], which has been recommended by the MPEG-7 community as the standard for boundary-based shape description [32]. In this method, the zero-crossings of the boundary points' curvature function are located at different scales (by gradually smoothing the boundary using a Gaussian kernel until the boundary becomes totally convex). The result is a binary image, called CSS image, that shows the end-points of the concave segments along the contour (the horizontal axis) at each scale level (the vertical axis). For matching, only the maxima of the CSS image contours are used. Adamek and O'Connor proposed a very effective representation that makes use of both concavities and convexities of all contour points [1]. It is called multiscale convexity concavity (MCC) representation, where different scales are obtained by smoothing the boundary with Gaussian kernels of different widths. Then, the curvature of each boundary point is measured based on the relative displacement of a contour point with respect to its position in the preceding scale level. Afterward, the matching is done using a dynamic programming (DP) approach. Belongie et al. [5] developed a very efficient shape descriptor called shape context (SC). In this descriptor, a histogram is attached to each boundary point describing the relative distribution of the remaining points to that point. The matching is then formulated as finding the correspondence in a point-by-point manner. Another recent method was proposed by Ling and Jacobs [23]. The inner-distance (ID), which is the length of the shortest path between two boundary points within the shape boundary, was derived to be invariant to shape articulation. In order to apply the inner distance for shape matching and retrieval, the authors extended the SC method [5] using this distance and called it IDSC. Then, DP is used for matching shapes after calculating the IDSC distances.

In this paper, a geometry-based image retrieval method is proposed for multiobject images and is applied to retrieving medical images. Our matching scheme captures both shape and topology of the image objects efficiently. The matching strategy agrees with recent findings in psychology about the human perception of multiobject images.

3 Image Comparison in Psychology

Advances in signal processing and applied mathematics have enabled researchers to develop sophisticated techniques to analyze the geometric content of images. Although such methods are effective in capturing low-level image features, the gap between these low-level features and high-level semantics is still wide [9]. Humans perform image comparisons efficiently and effortlessly; however, it is not the case for machines. In the following, we review recent studies in the psychological literature about how humans perceive and compare multiobject images. This is particularly important in developing computational models that satisfy the user requirements in retrieving similar images.

Gestalt theory suggests four principles known as Gestalt principles of perception [42] that include proximity, size, and shape similarity, continuity, and closure. Although the mechanisms behind these principles are still unclear, they provide a strong evidence that humans do not perceive a multiobject image as purely the sum of the individual objects, which also has been noted by Lowe [25]. Biederman [6] concluded that different arrangements of the same objects of an image can easily lead to a completely different perceived image.

According to our literature review in visual cognition, the most relevant work to multiobject image representation and matching is due to Markman and Gentner [26], which provides more details about the comparison process in the human cognition system. They concluded that carrying out similarity comparisons involves structured representations and such comparisons work to align these structures. In addition, the similarity comparison possesses the following characteristics:

2. A symbolic image is a logical representation of the original image, where the image objects are uniquely labeled with symbolic names.

- *Consistency*. The similarity comparison is constrained by a one-to-one correspondence, that is, each element in one representation can be matched to at most one element in the other representation.
- *Systematicity*. The similarity comparison is driven by a search for correspondence that preserves connections between the representation elements. In other words, matching connected elements is preferred over matching isolated elements. In relation with our work, the connectivity refers to the object level, i.e., the topological relations between objects and holes, and does not refer to the pixel connectivity.
- *Subjectivity*. Different people might generate different interpretations of the similarity between the same images, so as the same person does at different times. This property justifies the difficulty in evaluating content-based image retrieval systems.

As explained in the rest of this paper, these findings constitute the main motivations for the proposed similarity function between two multiobject images.

4 Curvature Trees

The curvature tree (CT) is a structured representation of a multiobject binary image that describes both shape and topology of objects[3] and holes comprising the image. Formally, the CT is a rooted, directed, unordered, and acyclic graph $\mathcal{T} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of nodes, and $\mathcal{E}$ is a set of edges. The CT has a single root node at level 0 representing the background, the external contours of the primary objects are stored at the first-level nodes, and the contours of possible holes are at the second-level nodes, and so on. Therefore, the tree hierarchy reflects the inclusion relationships between the objects and holes. To facilitate shape-based matching, the triangle-area representation (TAR) of each closed boundary of an object or hole is stored at the corresponding node (Section 4.1). Fig. 1 shows an illustrative example of the CT of a simple multiobject image.

Many researchers have suggested different tree representations for gray-scale images. For example, Salembier and Garrido [39] proposed a binary partition tree (BPT), where initial image partitions are merged following a sequential pairwise fashion. Therefore, the BPT hierarchy reflects the merging sequence, and the topological relationships between the regions cannot be easily identified. In another work, Monasse and Guichard [29], [30] proposed a tree representation of gray-scale images that employs a region-growing algorithm to detect connected components and then organizes them based on their inclusion relationships. In fact, our CT exhibits the same structure as the inclusion tree in [29] for binary images. However, the CT differs in the stored information in the nodes and its application to measure the similarity between images. In the following, TAR is introduced.

4.1 Triangle-Area Representation (TAR)

The shape of each object and hole is represented using the areas of the triangles formed by the boundary points and then stored at the corresponding node to facilitate measuring the similarity based on shape. TAR is invariant to position, scale, and rotation, robust against noise and moderate amounts of deformations, and computationally efficient [2]. In the following, the TAR of an arbitrary closed boundary is briefly explained.

3. In the context of this paper, an object (hole) is a connected set of foreground (background) pixels.

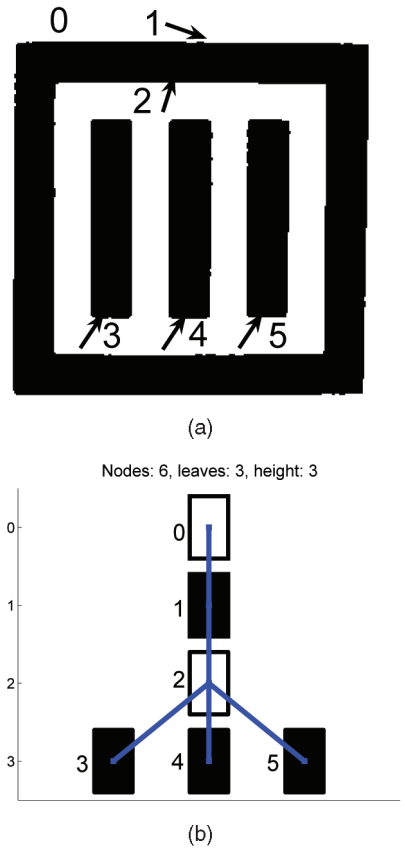


Fig. 1. An example of a (a) multiobject image and (b) its CT.

After extracting the boundary points, separated parameterized contour sequences $x(n)$ and $y(n)$ are obtained and resampled to N points. Then, the curvature of each point is measured as follows: For each three consecutive points (x_{n-t_s}, y_{n-t_s}), (x_n, y_n), and (x_{n+t_s}, y_{n+t_s}), where $n \in \langle 1, N\rangle$ and $t_s \in \langle 1, T_s\rangle$ is the triangle side length. The signed area of the triangle formed by these points is given by

$$TAR(n, t_s) = \frac{1}{2} det \begin{bmatrix} x_{n-t_s} & y_{n-t_s} & 1 \\ x_n & y_n & 1 \\ x_{n+t_s} & y_{n+t_s} & 1 \end{bmatrix}, \quad (1)$$

where det is the matrix determinant. When the contour is traversed in counter clockwise direction, positive, negative, and zero values of TAR mean convex, concave, and straight-line points, respectively. Fig. 2 demonstrates these three types of the triangle area. The triangles at the edge points are formed by considering the periodicity of the closed boundary. By increasing t_s, i.e., considering farther points, the function of (1) represents longer (more global) variations along the boundary. Note that (1) exhibits an odd symmetry with inflection at $t_s = \frac{N}{2}$, i.e., $TAR(n, \frac{N}{2}) = 0$, when N is even. Thus, T_s is chosen to be equal to $\frac{N}{2} - 1$. Note also that TAR values are high at large t_s; therefore, the TAR signatures are normalized by dividing the TAR of all points at each t_s on the maximum TAR value at that t_s. This normalization ensures equal contribution of the TAR signatures at all triangle side lengths. Fig. 3 shows the TAR of the Misk shape.

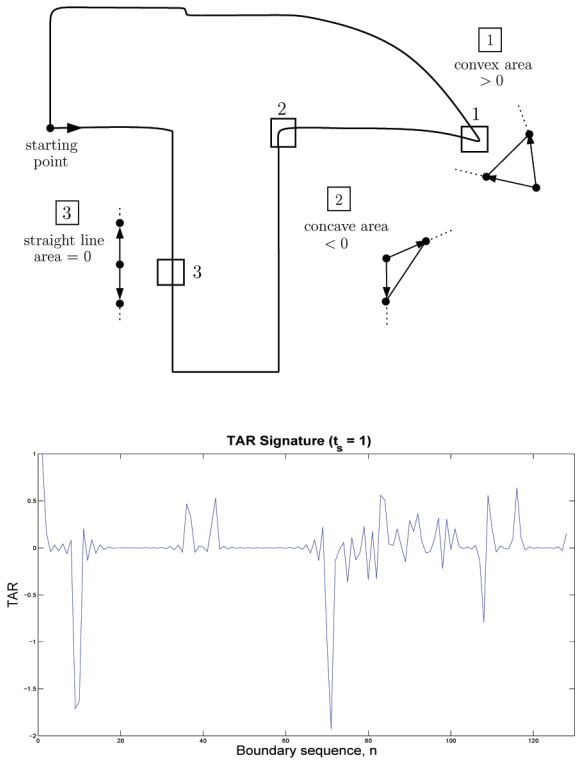


Fig. 2. Three different types of the triangle-area values and the TAR signature for the hammer shape.

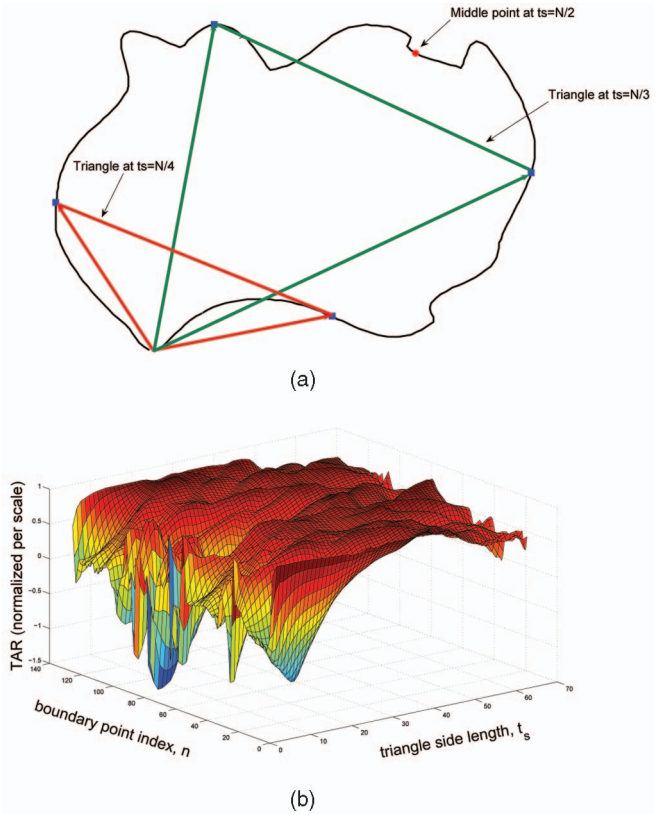


Fig. 3. The (a) Misk shape and (b) its TAR.

5 Attributed Tree Matching

The main aim is to measure the similarity between two multiobject images using their CTs. As discussed in Section 2, interesting recent findings in psychology strongly suggest that such similarity should follow a one-to-one correspondence between structured representations of the two images [26]. Therefore, we propose a similarity function based on the MSSI between the two CTs. This matching scheme is desirable since it allows matching whole or part of one image with whole or part of the other image. In the following, an exact algorithm is introduced to solve the maximum similarity subtree isomorphism problem.

5.1 Notations and Definitions

We first introduce some graph theoretic notations and definitions. More details can be found in the graph theory literature such as [16]. An attributed graph is a triple $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \delta)$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges, and δ is a function that assigns attributes $\delta(u)$ to each node $u \in \mathcal{V}$. The order of $\mathcal{G}$ is the number of nodes, and its size is the number of edges. Two nodes $u, v \in \mathcal{V}$ are said to be adjacent, denoted by $u \sim v$, if they are connected by an edge. A path is a sequence of distinct nodes $u_0 u_1 \ldots u_n$ such that $u_{i-1} \sim u_i$, for $i = 1 \ldots n$. If $u_0 = u_n$, the path is called a cycle. A graph is said to be connected if there is a path between any two nodes.

A tree is a connected graph with no cycles. An attributed tree has attributes stored at its nodes. A weighted tree is the one with weights assigned to its edges. A rooted tree has a distinguished node called the root. The level of a node u in a rooted tree, denoted by $l(u)$, is the length of the path connecting the root node to u. Note that the rooted tree implies a multilevel organization of the tree nodes. Moreover, if $u \sim v$ and $l(u) - l(v) = 1$, then u is the parent of v, and conversely, v is a child of u. Note that any node in a rooted tree has exactly one parent node, except the root, whereas a node can have any number of children; a node that does not have any child node is called a leaf. An ordered tree respects the order of the children nodes at any tree level, whereas in an unordered tree, the children nodes do not follow any order.

Let $\mathcal{T}_1 = (\mathcal{V}_1, \mathcal{E}_1, \delta_1)$ and $\mathcal{T}_2 = (\mathcal{V}_2, \mathcal{E}_2, \delta_2)$ be two rooted and attributed trees, $\mathcal{H}_1 \subset \mathcal{V}_1$ and $\mathcal{H}_2 \subset \mathcal{V}_2$. A subtree isomorphism is a one-to-one mapping $\phi : \mathcal{H}_1 \rightarrow \mathcal{H}_2$ that preserves both adjacency and hierarchical relations between the nodes of each subtree. In matching attributed trees, the isomorphism should also pair similar nodes. Let $\omega(\delta_1(u), \delta_2(v))$ be a similarity function between nodes $u \in \mathcal{T}_1$ and $v \in \mathcal{T}_2$ based on their attributes (their TAR signatures in our case). Then, the overall similarity Ω between the subtrees $\mathcal{H}_1$ and $\mathcal{H}_2$ induced by ϕ can be defined as follows:

$$\Omega(\phi) = \sum_{u \in \mathcal{H}_1} \omega(\delta_1(u), \delta_2(\phi(u))). \quad (2)$$

The isomorphism ϕ is a maximal similarity subtree isomorphism if there is no other subtree isomorphism $\phi' : \mathcal{H}'_1 \rightarrow \mathcal{H}'_2$ such that $\mathcal{H}_1 \subset \mathcal{H}'_1$ and $\Omega(\phi) < \Omega(\phi')$. It is called maximum similarity subtree isomorphism if $\Omega(\phi)$ is largest among all possible isomorphisms.

5.2 Solving the MSSI Problem

Here, an exact algorithm for solving the MSSI problem is presented. The algorithm recursively searches for all possible isomorphisms between two trees and selects the one yielding the maximum similarity. Unfortunately, such exhaustive search is computationally expensive; for two trees having n and m nodes, respectively, there are nm possible isomorphisms to be searched between subtrees rooted at each two

nodes in the two trees. However, due to the nonnegative property of the employed similarity function between the attributed nodes (explained in the next subsection), without loss of generality, the search space reduces to $n + m$ possible isomorphisms. The reason for this reduction is given as follows: Suppose an isomorphism $\tilde{\phi}(\tilde{u}, \tilde{v})$ is induced between two subtrees rooted at two arbitrary nonroot nodes $\tilde{u} \in \mathcal{T}_1$ and $\tilde{v} \in \mathcal{T}_2$, with parent nodes u and v, respectively. Since the node similarity function is nonnegative, adding u and v to the isomorphism $\tilde{\phi}$ increases its similarity weight or, at least, does not change it, i.e., $\Omega(\phi(u, v)) \geq \Omega(\tilde{\phi}(\tilde{u}, \tilde{v}))$. To achieve maximum similarity, an isomorphism has to include the parent nodes of the subtrees' root nodes until the root of either tree is included in the isomorphism. Therefore, the MSSI must include at least one of the root nodes of the two trees. In this case, the number of possible isomorphisms, which are candidates for MSSI, reduces to $n + m$.

Algorithm 1 is a pseudocode of our exact tree matching algorithm. It accepts two CTs as inputs and returns their MSSI along with its weight. The algorithm examines all subtree isomorphisms, between the nodes of the first CT and the root of the other CT, and vice versa, and returns the isomorphism yielding maximum similarity. The main function, *ExactMatch*, uses the function *TreeMatch* of Algorithm 2 to find the MSSI between two subtrees rooted at $u \in \mathcal{T}_1$ and $v \in \mathcal{T}_2$. *TreeMatch* initializes the isomorphism with $u \rightarrow v$ and its weight with $\omega(u, v)$. If both u and v are nonleaf nodes (both have at least a child node), then an optimal assignment procedure assigns the child node(s) of u to the child node(s) of v such that the sum of the pairwise similarities is maximum. This assignment problem is known as bipartite matching; for two sets of n_1 and n_2 elements and $n_1 \geq n_2$, there are $n_1!/(n_1 - n_2)!$ possible mappings between the elements of the two sets ($n!$ is the factorial of n). For this purpose, the Hungarian method is employed, which has a polynomial-time complexity of $O(n_1 n_2^2)$ [31], [33]. *TreeMatch* recursively matches the nodes of the subtrees rooted at u and v and returns their maximum similarity isomorphism. The recursive call of *TreeMatch* stops when all children of either u or v are leaf nodes. In this case, the pairwise similarities between the children of u and v are stored in a similarity matrix W. The rows and columns of W represent the children of u and v, respectively. Since the Hungarian procedure works on dissimilarity matrices, each element of W is subtracted from an upper bound value to obtain the dissimilarity version of W. This upper bound is chosen as the maximum value of W. Note that some children of u or v are left unmatched when $n_u \neq n_v$ (W is rectangular). Then, the Hungarian procedure returns the assignment that yields the minimum cost, i.e., the minimum sum of pairwise dissimilarities. To obtain the weight of the isomorphism, the cost is subtracted from the upper bound multiplied by the number of dissimilarities of the cost. This algorithm is similar to the algorithm in [43] except that the latter returns only the weight of the maximum isomorphism and does not return the isomorphism itself.

Algorithm 1 Exact tree matching (main algorithm):
$[\phi, \Omega] = ExactMatch(\mathcal{T}_1, \mathcal{T}_2)$
Notation:
$\mathcal{T}_1$, $\mathcal{T}_2$ are two CTs to be matched.
ϕ is a MSSI between the two trees.
Ω is the total weight of ϕ.
$\Omega \leftarrow 0$
for each node $u_i \in \mathcal{T}_1$ **do**
  $[map, sim] \leftarrow TreeMatch(u_i, root(\mathcal{T}_2))$
  if $sim > \Omega$ **then**
    $\Omega \leftarrow sim$
    $\phi \leftarrow map$
  end if
end for
for each node $v_j \in \mathcal{T}_2$ **do**
  $[map, sim] \leftarrow TreeMatch(root(\mathcal{T}_1), v_j)$
  if $sim > \Omega$ **then**
    $\Omega \leftarrow sim$
    $\phi \leftarrow map$
  end if
end for
return ϕ, Ω

Algorithm 2 Exact subtree matching:
$[map, sim] = TreeMatch(u, v)$
Notation:
n_u is the number of children of u.
n_v is the number of children of v.
$\delta(u)$ is the TAR signatures of node u.
ω is the similarity function given by (8).
W is the similarity matrix between the child nodes of u and v.
$hungarian(Z)$ is an optimal assignment function that assigns the rows to the columns of the cost matrix Z.
$sim \leftarrow \omega(u, v)$
$map \leftarrow \{(u, v)\}$
if $n_u \neq 0$ and $n_v \neq 0$ **then**
  for each child node c_i of u **do**
    for each child node c_j of v **do**
      $[M(i, j), W(i, j)] = TreeMatch(c_i, c_j)$
    end for
  end for
  $[assign, cost] = hungarian(max(W) * ones(n_u, n_v) - W)$
  $sim \leftarrow sim + max(W) * min(n_u, n_v) - cost$
  $map \leftarrow map \cup assign$
end if
return map, sim

5.3 Dynamic Space Warping

As explained above, the similarity function between two attributed nodes constitutes the kernel of our CT matching algorithm and is measured based on the TARs stored at the nodes (Section 4.1). To this end, we employ a very efficient dynamic programming method called dynamic space warping (DSW). Unlike the euclidean distance, DSW is based on nonrigid alignment between the points of the two boundaries.

Now, we present a brief description of the DSW algorithm in [2] and its adaptation to compute the similarity function between two CT nodes based on their TARs. At first, it is necessary to define the distance between two individual boundary points. Let $TAR_A(n, t_s)$ and $TAR_B(n, t_s)$ be the TARs for shapes A and B, respectively, where $n \in \langle 1, N \rangle$ is the index of the boundary points, and $t_s \in \langle 1, T_s \rangle$ is the triangle side length. Then, the distance between the two points $n \in A$ and $m \in B$ is defined as

$$D(n, m) = \frac{1}{T_s} \sum_{t_s=1}^{T_s} |TAR_A(n, t_s) - TAR_B(m, t_s)|. \quad (3)$$

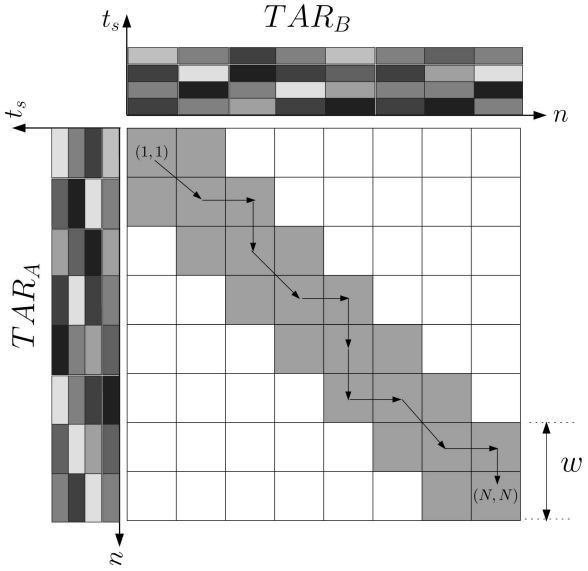


Fig. 4. Illustration of matching two TARs using the DSW algorithm. The rows and the columns of the DSW table represent the contour points of TAR_A and TAR_B, respectively. Note that $N = 8$ here for better visualization, and the intensity of each TAR point at specific t_s reflects its value.

Then, an $N \times N$ distance table, DT, is constructed to find the optimal correspondence between the points of the two boundaries. The columns of DT represent the points of one boundary, and the rows represent the points of the other. Initially, the elements of DT are set as

$$DT_{initial}(n,m) = \begin{cases} 0 & \max(1, n-w+1) \le m \\ & \le \min(N, n+w-1), \\ \infty & \text{otherwise}, \end{cases} \quad (4)$$

where $n, m \in \langle 1, N \rangle$, w is a predefined diagonal width for DT as illustrated in Fig. 4, and $max(a, b)$ and $min(a, b)$ are the maximum and minimum values of a and b, respectively. Only the elements of DT that fall within w are updated during the DSW search. This initialization of DT avoids computing the distances between all the points of two contours and restricts the distance computation to only those points that are more likely to correspond to each other. Therefore, the computational complexity is largely reduced, while more meaningful correspondences are obtained.

Starting at an arbitrary TAR point for both contours A and B, the distance table DT is searched through a predefined diagonal window of width w, left-to-right and up-to-bottom, starting from the upper left element, as shown in Fig. 4. The first row and first column elements are initialized as the distance between the corresponding points using (3). Then, the rest of the w-diagonal elements of DT are updated as

$$DT(n,m) = D(n,m) + \min \begin{cases} DT(n-1,m), \\ DT(n-1,m-1), \\ DT(n,m-1), \end{cases} \quad (5)$$

where $n, m \in \langle 1, N \rangle$.

The least cost path through the distance table is the value of element $DT(N, N)$, which corresponds to the best matching between the two TAR points, according to the selected starting points. However, it is clear that the established correspondence is sensitive to the starting point of each TAR. In order to achieve starting point (or rotation) invariance, it is sufficient to fix the starting point of one TAR and try all N starting points of the other TAR. Moreover, invariant to the mirror transformation can be obtained by flipping the points of one TAR and repeating the search for the N starting points again. The final least cost correspondence is taken as the minimum value of $DT(N, N)$ among all $2N$ runs of the DSW table search, denoted by DT_{min}.

To further increase the discrimination power of the DSW, *shape complexity* (SC) of each boundary is computed as the average, over all boundary points, of the absolute differences between the maximum and minimum TAR values at all scale levels (or triangle side lengths):

$$SC = \frac{1}{N} \sum_{n=1}^{N} \left| \max_{1 \le t_s \le T_s} \{TAR(n, t_s)\} - \min_{1 \le t_s \le T_s} \{TAR(n, t_s)\} \right|. \quad (6)$$

Then, the dissimilarity distance between two shapes, A and B, is given by

$$D_{dis}(A,B) = \frac{DT_{min}(A,B)}{1 + SC_A + SC_B}, \quad (7)$$

where $DT_{min}(A, B)$ is the minimum cost distance between shapes A and B, computed using the DSW table search, and SC_A and SC_B are the complexities of shapes A and B, respectively.

In this paper, the dissimilarity function of (7) must be normalized within a dynamic range, for instance, between 0 and 1 for two main reasons. First, this normalization allows computing the similarity directly from the dissimilarity distance (by subtracting the latter from its upper bound). The attributed tree matching algorithm described earlier requires a similarity function (ω), whereas (7) is a dissimilarity distance. Second, the distance normalization is essential to facilitate combining the contributions of different single-object shape similarities into one similarity function. Without such normalization, a single similarity value between two shapes in the two multiobject images may dominate the other similarity values just because its magnitude is large. In conclusion, normalizing the shape similarity function ensures equal emphasis of each individual similarity within the overall similarity function of (2). Here, the function of (7) has a lower bound of 0 (when $A = B$) since it represents a sum of absolute differences; therefore, it is sufficient to impose an upper bound on its value. For this purpose, (7) is divided by L_{AB}, the length of the least-cost path through the DSW table (see Fig. 4) to be upper bounded by 1 and then subtracted from 1 to give the shape similarity function ω:

$$\omega(A,B) = 1 - \frac{D_{dis}(A,B)}{2L_{AB}}. \quad (8)$$

6 Computational Complexity

The complexity of each of the representation and the matching stage is evaluated separately. It should be noted that the complexity of the matching stage is more critical since the representation of the database images can be computed prior to the time of the matching, whereas the matching usually takes place between the query image and most (if not all) of the database images at the time of retrieval.

The CT of a multiobject image of n shapes has n nodes; therefore, the order of the CT grows linearly with the

TABLE 1
Comparison of the Results of Different Methods on the MPEG-7 CE-Shape-1 Part B (Bulls-Eye) Test

Test	CSS [27]	MCC [1]	SC [5]	IDSC [23]	**TAR+DSW**
Part B	81.12%	84.93%	76.51%	85.40%	**87.13%**

number of objects and holes in the image. The TAR of each shape requires calculating the triangle area, according to (1), at each of the N points of the boundary. In addition, at each boundary point, the triangle area is calculated at different scales (or triangle side lengths). Typically, there are $\frac{N}{2}-1$ scales. Therefore, the computational complexity of the TAR is $O\big(N.(\frac{N}{2}-1)\big)$ or $O(N^2)$, and the overall complexity of the CT construction is $O(n.N^2)$.

The computational complexity of the tree matching algorithm depends on the number of CT levels and the branching factor (i.e., the number of children of a CT node). Let b be the maximum branching factor. For matching two CTs of n and m nodes, the main algorithm *ExactMatch* (of Algorithm 1) actually computes at most nm bipartite matchings: one for each possible pair of nodes from the two trees. The bipartite matching under nodes $u_i \in \mathcal{T}_1$ with p_i children and $v_j \in \mathcal{T}_2$ with q_j children, using the Hungarian procedure, has complexity $O(max(p_i, q_j)p_iq_j) \leq O(bp_iq_j)$. Summing over all possible matches, we have $\sum_{i=1}^{n}\sum_{j=1}^{m} bp_iq_j = bnm$. The node similarity is measured using the DSW algorithm of complexity given as follows: For the similarity function of (8), the length of the least-cost path L_{AB} is computed during the table search for computing D_{dis}; thus, the latter governs the complexity. Each of the SC terms SC_A and SC_B in the denominator of (7) requires $O(N)$ complexity, as given by (6). For the minimum cost distance term DT_{min} given by (5), the DSW table search is restricted within the diagonal w-width window; thus, the DSW table search complexity is $O(wN)$ (usually $w \ll N$). Since the DSW search is repeated for N starting points, the complexity becomes $O(wN^2)$; by considering the flipping operation, the total complexity of (8) is $O(2wN^2)$ or $O(N^2)$ (for $N = 128$, our experiments show that $w = 3$ is good enough, and larger w does not achieve better results). Therefore, the complexity of our exact CT matching algorithm is $O(bnmN^2)$.

The processing times for the representation and matching stages, obtained using Matlab (version 7.0) program running on Pentium IV 3.0 GHz PC, are given as follows: The average time for computing the TAR, per shape of the 1,400 shapes of the MPEG-7 CE-Shape-1 database [22], is 8 milliseconds. When each shape is used as a query, the average time per comparison for the DSW matching is 10 milliseconds. For the CT construction and matching, the average processing times per query of the multiobject medical image database [37] are 50 milliseconds and 0.4 second, respectively. It should be noted that the codes are not optimized, and better speeds can be obtained using other programming languages such as C and C++.

7 Experimental Results

In this section, we demonstrate the performance of our method using two experiments. In the first, the aim is to evaluate the effectiveness of our shape-matching method and to compare it with the state-of-the-art shape-matching methods on the standard MPEG-7 database. The second experiment demonstrates the retrieval performance on a database of 13,500 real and synthesized medical images.

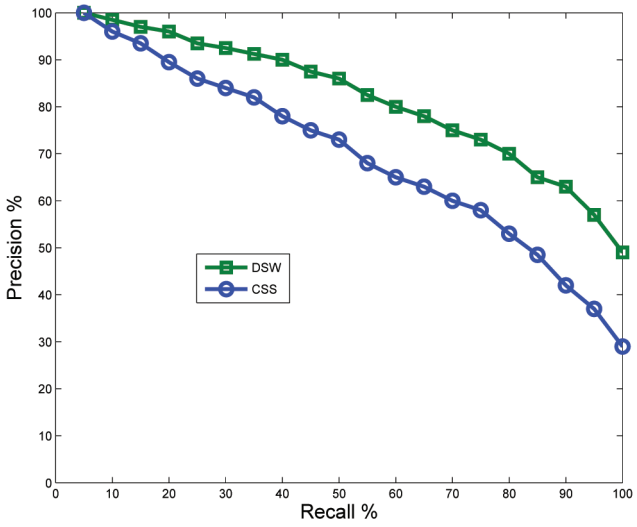


Fig. 5. Precision-recall curves of the DSW and the CSS methods.

7.1 Shape Retrieval Test

Here, the well-known MPEG-7 CE-Shape-1 database [22], which consists of 1,400 shapes semantically classified into 70 classes, is used. MPEG-7 part B test (also called *bulls-eye* test) is the most challenging shape retrieval test, where each shape is used as a query, and then, the number of correct matches is counted in the first 40 retrieved shapes. The overall retrieval rate is the average of all individual rates for the 1,400 shapes. The results of our TAR+DSW method, and many recent methods are shown in Table 1. Our method outperforms all other methods, as per this test. To get better insight about the retrieval performance of our method, Fig. 5 shows the precision-recall curves of the DSW and the CSS methods.[4] Clearly, the DSW outperforms the CSS, which has been selected by the MPEG-7 community [32] as the boundary-based shape descriptor, with a good margin at all recalls.

7.2 Medical Image Retrieval Test

Here, a database of 13,500 real and synthesized medical images is used [37]. The evaluation is based on human relevance judgements, as reported in [36]. Two images are considered similar if they contain similar objects in similar spatial relationships. Besides, the similarity is judged based on the query image, that is, a database image may contain extra objects without affecting the similarity but not the query. The following performance measures are used to evaluate the retrieval performance:

Precision is the ratio of the number of relevant retrieved images to the total number of retrieved images.

Recall is the ratio of the number of relevant retrieved images to the number of relevant images in the database. To compute recall, all database images relevant to the query have to be known. However, comparing every query with all 13,500 database images by a human referee is practically impossible. To overcome this problem, a sampling method,

4. The precision-recall results of the CSS method [27] in Fig. 5 were computed using our implementation.

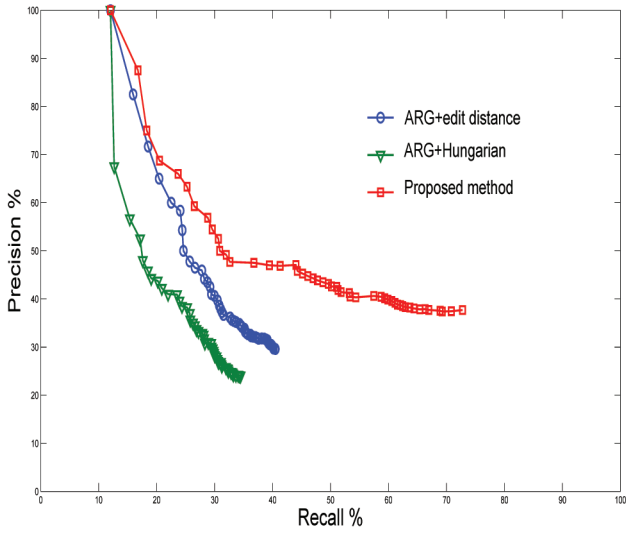


Fig. 6. Precision-recall curves of the proposed method and the ARG method with edit distance and Hungarian method [36].

known as pooling method in the text retrieval community [46], is employed, which results in labeling part of the database images, as explained in the rest of this section.

Ranking quality (R_q) measures the goodness of the retrieval ranking, that is, the method retrieves relevant images before the irrelevant images. The higher the value of R_q, the better the ranking quality of a method, and vice versa. Note that R_q is independent from the accuracy. R_q is computed as follows:

1. Each retrieved image is assigned a rank number according to its order and judged as relevant or irrelevant with respect to the query image.
2. Pairs of the retrieved images are taken such that each pair contains a relevant and irrelevant images, and the relevant one is first, and the irrelevant is second.
3. Let S^+ be the number of pairs with the relevant image's ranking better than the ranking of the irrelevant image, S^- is the number of pairs with the irrelevant image's ranking better, and $S = S^+ + S^-$. R_q is computed as

$$R_q = \begin{cases} \frac{1}{2}\left(1 + \frac{S^+ - S^-}{S}\right) & \text{if } S > 0, \\ 1 & \text{otherwise.} \end{cases} \quad (9)$$

Since the objects of the images in this database are manually segmented, overlapping between the objects can occur, which makes the inclusion relationships not clearly identified. Therefore, the CT construction and matching are modified accordingly. In this application, the CT edges are weighted to reflect the amount of inclusion of a child node in its parent node. To resolve the inclusion ambiguity caused by the overlapping, let object u with area A_u and object v with area A_v have an overlapping with area A_{uv} such that $A_u > A_v$. Then, node v is inserted in the CT as a child to the node u, and the weight of the edge $\vec{uv}$ equals A_{uv}/A_v. For the exact CT matching algorithm, since the incoming degree to any node in the CT equals one (except the root), only the node similarity function (ω) is modified to include the edge weights as follows: Let u_1, $v_1 \in \mathcal{T}_1$, u_2, $v_2 \in \mathcal{T}_2$, and ζ_1 and ζ_2 be the weights of the edges $u_1\vec{v}_1$ and $u_2\vec{v}_2$, respectively. Then, the similarity between nodes v_1 and v_2 is redefined as

TABLE 2
The Accuracy of Different Methods on the Medical Images Database Based on the Ground Truth Provided in [36]

Method	ARG+Hung.	ARG+ED	Proposed (exact)
Accuracy	24%	29.60%	40.91%

TABLE 3
The Ranking Quality of Different Methods on the Medical Images Database

Method	ARG+Hung.	ARG+ED	Proposed (exact)
R_q	0.0890	0.2390	0.5431

$$\omega(v_1, v_2) = 1 - (\alpha_1|\zeta_1 - \zeta_2| + \alpha_2|\hat{A}_{v_1} - \hat{A}_{v_2}| + \alpha_3 D_{dis}(v_1, v_2)/2L_{v_1v_2}), \quad (10)$$

where $\hat{A}_{v_1}$ is the normalized area of v_1 obtained by dividing A_{v_1} by the area of the largest object in $\mathcal{T}_1$ in order to achieve scale invariance (same for $\hat{A}_{v_2}$), D_{dis} is the dissimilarity function given by (7), $L_{v_1v_2}$ is the length of the DSW path, and α_i are positive weights such that $\sum_{i=1}^{3} \alpha_i = 1$ to maintain $0 \le \omega \le 1$. The values of these parameters are set according to the application under consideration. Due to the high subjectivity in evaluating the results of this test, obtaining optimum values of these parameters is an extremely difficult task. Nevertheless, our experiments showed that good results can be obtained over a wide range of the parameters' values. In our experiments, $\alpha_1 = 0.15$, $\alpha_2 = 0.35$, and $\alpha_3 = 0.5$ (these values are not optimized). Another possible strategy for the parameter calibration is to include the user in the loop using a relevance feedback scheme, which constitutes a main direction of our future research in this area.

The precision-recall pairs are computed as in [36]. In this retrieval test, 20 query images are presented, and for each query, the system retrieves the best 50 matches out of the 13,500 database images. In [36], the results of five methods and 19 of their variants are reported, which demanded 24,000 comparisons by human referees. These comparisons resulted in labeling 42 percent of the database images as relevant/irrelevant to any of the 20 queries. For consistency, we used the labeled database images as templates to classify the unlabeled retrievals for each query. Then, the ground truth is updated, and the precision-recall curves are computed as shown in Fig. 6. Each point in the figure is the average over the 20 query images; thus, each curve consists of 50 points. The figure shows the results of the proposed method and the ARG method [36] with two matching approaches: the graph edit distance and the Hungarian method. Clearly, the proposed method outperforms the ARG method, which outperformed many other methods [36].

Another test is performed based on the ground truth provided in [36] alone. For each query, the system retrieves the best 50 matches and the number of relevant images, n_r, and the number of irrelevant images, n_i, are counted. Note that both the relevant and irrelevant images are provided by the ground truth given in [36]; thus, our judgement in the evaluation is not considered in this test. Then, the accuracy for each query is computed as $n_r/(n_r + n_i)$. Table 2 shows the accuracy of our method is significantly higher than that of the ARG method.

Regarding the quality of retrieval quality (R_q), Table 3 shows that the proposed method also outperforms the ARG

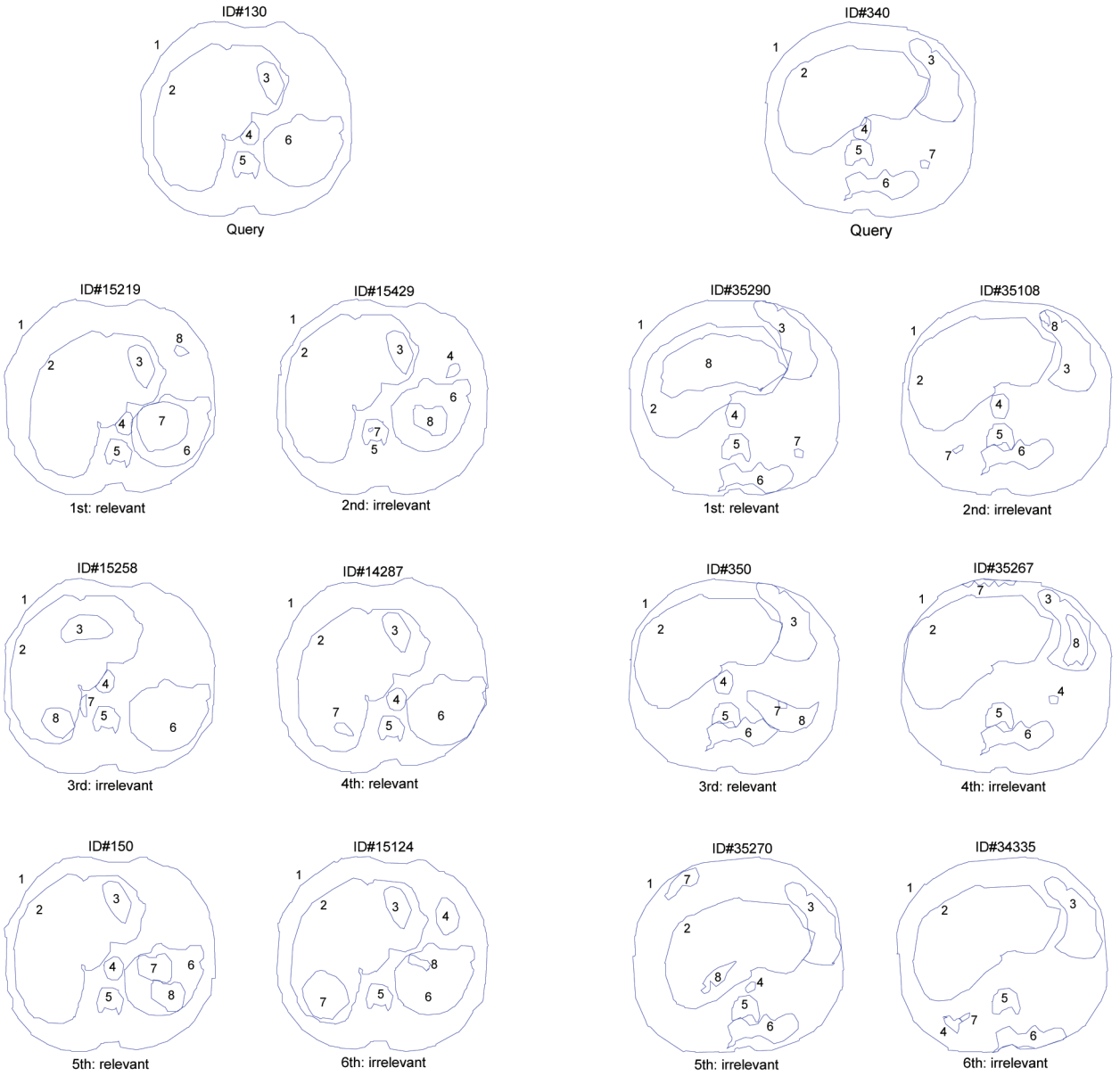


Fig. 7. A query image and the first six matches using the CT matching algorithm. The object labels denote the isomorphism returned by the algorithm.

Fig. 8. Another query image and the first six matches using the exact CT matching algorithm. The object labels denote the isomorphism returned by the algorithm.

method. Figs. 7 and 8 demonstrate two example queries and the first six matches retrieved by the CT matching algorithm. For clarity, the isomorphism returned by the algorithm is shown by objects with same labels in the query and each retrieved image.

It should be noted that conventional global shape descriptors such as the ART descriptor [20] are not suitable for this application due to the following reasons. First, the intraclass variability among the medical images is small. Second, the topological relationships (inclusion and overlapping) play an important role in defining the similarity as required by the domain experts; therefore, object-based methods are more appropriate. Third, the constraint on a database image to include similar objects with similar relationships as those in the query in order to be similar to the query, and not vice versa, makes it difficult for global descriptors to fulfill this requirement, unlike object-based methods. For example, for global descriptors, the third retrieval in Fig. 7 (judged as irrelevant by the domain experts) may look more similar to the query than the first, which is relevant. In fact, our experiments showed that the Zernike moments descriptor, which works in a similar fashion as the ART descriptor, have poor performance in this test.

8 Concluding Remarks

In this paper, the CT is introduced as a structured representation of multiobject images that encodes both shape and topology. Motivated by recent findings in psychology, the matching of two CTs is formulated as a maximum similarity subtree isomorphism (MSSI) problem. A matching algorithm is developed, which works in polynomial time and guarantees finding an exact solution to the MSSI problem. Experiments

with a database of 13,500 medical images, and a database of 1,400 shapes show the superiority of the proposed method over other methods in the literature.

The main strengths and limitations of the proposed approach for retrieving multiobject images are summarized in the following points:

Accuracy. The proposed method has the advantage of achieving higher retrieval accuracy than all other methods in the literature based on medical imaging and shape retrieval applications.

Flexibility. The decision on the similarity function between two multiobject images depends on the application according to the user requirements. The proposed method can handle various types of queries. The user can search a database for images containing a particular object or multiple objects with specific structural relationships; taking into consideration the shape of the objects, their topology, or both shape and topology at once. In the medical image retrieval considered in this paper, the user is interested in the objects of a database image similar to the objects in the query, and the remainder portion of the database image is insignificant [36]; thus, the similarity function is considered as the weight of the MSSI, Ω. However, this choice for the similarity function may not be suitable for other applications since the similarity increases proportionally with the size of the isomorphism. For the most general case, a normalized similarity function that accounts for the unmatched objects in both multiobject images is proposed. Once Ω is obtained, the final similarity function between two CTs is defined as

$$D_{CT} = \frac{\Omega}{2}\left(\frac{1}{|\mathcal{T}_1|} + \frac{1}{|\mathcal{T}_2|}\right). \quad (11)$$

Note that D_{CT} equals Ω normalized by the number of nodes in each CT. This normalization accounts for the unmatched parts in the two CTs.

Invariance. The rotation of (part of) the image objects can change the order of the sibling nodes in the corresponding CT; therefore, the employed tree matching method is invariant to the rotation angle of the image since it works for unordered trees. Regarding scale, the CT is clearly invariant to the scale of the image. In addition, the CT representation is invariant to the negative transformation of the binary image; changing the background/foreground from 0/1 to 1/0 only adds/deletes a child node to the root node.

Robustness. In general, the major limitation of the approach presented in this paper is related to the fact that it deals primarily with binary shapes. Segmentation is assumed already done, and the object boundary is clearly identified. In practice, noise and partial occlusion can drastically change the topology of the CT. One way to overcome this limitation is to use mathematical morphology in the segmentation stage, which is the basis for our future work in this area. For instance, if a dilation or erosion operation changes the topology of an image's CT, then both CTs are used for representing the image and several subtree isomorphisms are searched rather than just one.

REFERENCES

[1] T. Adamek and N.E. O'Connor, "A Multiscale Representation Method for Nonrigid Shapes with a Single Closed Contour," *IEEE Trans. Circuits and Systems for Video Technology,* vol. 14, no. 5, pp. 742-753, May 2004.

[2] N. Alajlan, I. Elrube, M.S. Kamel, and G. Freeman, "Shape Retrieval Using Triangle-Area Representation and Dynamic Space Warping," technical report, PAMI Group, Univ. of Waterloo, http://pami.uwaterloo.ca/nav.php?site=pub&action=list&researcher=naif, Aug. 2006.

[3] J. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. Shu, "Virage Image Search Engine: An Open Framework for Image Management," *Storage and Retrieval for Image and Video Databases,* pp. 76-87, Feb. 1996.

[4] O. El Badawy and M.S. Kamel, "Shape Representation Using Concavity Graphs," *Proc. 16th Int'l Conf. Pattern Recognition,* vol. 3, pp. 461-464, 2002.

[5] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 4, pp. 509-522, Apr. 2002.

[6] I. Biederman, "Recognition by Components: A Theory of Human Image Understanding," *Psychological Rev.,* vol. 94, no. 2, pp. 115-147, 1987.

[7] C.C. Chang and S.Y. Lee, "Retrieval of Similar Pictures on Pictorial Databases," *Pattern Recognition,* vol. 24, no. 7, pp. 675-681, 1991.

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

[9] J.P. Eakins, "Towards Intelligent Image Retrieval," *Pattern Recognition,* vol. 35, no. 1, pp. 3-14, 2002.

[10] J.P. Eakins, K.J. Riley, and J.D. Edwards, "Shape Feature Matching for Trademark Image Retrieval," *Proc. Int'l Conf. Image and Video Retrieval,* vol. 2728, pp. 28-37, Aug. 2003.

[11] J.P. Eakins, K. Shields, and J.M. Boardman, "Artisan: A Shape Retrieval System Based on Boundary Family Indexing," *Proc. Storage and Retrieval for Still Image and Video Databases,* vol. 2670, pp. 17-28, 1996.

[12] E.A. El-Kwae and M.R. Kabuka, "A Robust Framework for Content-Based Retrieval by Spatial Similarity in Image Databases," *ACM Trans. Information Systems,* vol. 17, no. 2, pp. 174-198, 1999.

[13] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz, "Efficient and Effective Querying by Image Content," *J. Intelligent Information Systems,* vol. 3, nos. 3/4, pp. 231-262, 1994.

[14] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by Image and Video Content: The Qbic System," *IEEE Computer,* vol. 28, no. 9, pp. 23-32, Sept. 1995.

[15] V. Gudivada and V. Raghavan, "Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity," *ACM Trans. Information Systems,* vol. 13, no. 2, pp. 115-144, 1995.

[16] F. Harary, *Graph Theory.* Addison-Wesley, 1969.

[17] M.K. Hu, "Visual Pattern Recognition by Moment Invariants," *IRE Trans. Information Theory,* vol. 8, pp. 179-197, 1962.

[18] A.K. Jain and A. Vailaya, "Shape-Based Retrieval: A Case Study with Trademark Image Databases," *Pattern Recognition,* vol. 31, no. 9, pp. 1369-1390, 1998.

[19] A. Khotanzad and Y.H. Hong, "Invariant Image Recognition by Zernike Moments," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, no. 5, pp. 489-497, May 1990.

[20] W.Y. Kim and Y.S. Kim, "A New Region-Based Shape Descriptor," Technical Report ISO/IEC MPEG99/M5472, Dec. 1999.

[21] Y.S. Kim and W.Y. Kim, "Content-Based Trademark Retrieval System Using a Visually Salient Feature," *Image and Vision Computing,* vol. 16, nos. 12-13, pp. 931-939, Aug. 1998.

[22] L.J. Latecki, R. Lakamper, and U. Eckhardt, "Shape Descriptors for Non-Rigid Shapes with a Single Closed Contour," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 424-429, 2000.

[23] H. Ling and D. Jacobs, "Using the Inner Distance for Classification of Articulated Shapes," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 719-726, June 2005.

[24] S. Loncaric, "A Survey of Shape Analysis Techniques," *Pattern Recognition,* vol. 31, no. 8, pp. 983-1001, 1998.

[25] D.G. Lowe, *Perceptual Organization and Visual Recognition.* Kluwer Academic, 1985.

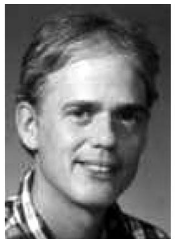
[26] A. Markman and D. Gentner, "Structure Mapping in the Comparison Process," *Am. J. Psychology,* vol. 113, no. 4, pp. 501-538, 2000.

[27] F. Mokhtarian and M. Bober, *Curvature Scale Space Representation: Theory, Applications, and MPEG-7 Standardization.* Kluwer Academic, 2003.

[28] F. Mokhtarian and A. Mackworth, "Scale-Based Description and Recognition of Planar Curves and Two-Dimensional Shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 8, no. 1, pp. 34-43, 1986.
[29] P. Monasse and F. Guichard, "Fast Computation of a Contrast-Invariant Image Representation," *IEEE Trans. Image Processing,* vol. 9, no. 5, pp. 860-872, 2000.
[30] P. Monasse and F. Guichard, "Scale-Space from a Level Lines Tree," *J. Visual Communication and Image Representation,* vol. 11, no. 2, pp. 224-236, 2000.
[31] J. Munkres, "Algorithms for Assignment and Transportation Problems," *J. Soc. Industrial and Applied Math.,* vol. 5, no. 1, pp. 32-38, 1957.
[32] *The MPEG Home Page,* http://www.chiariglione.org/mpeg, 2008.
[33] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity.* Prentice Hall, 1982.
[34] H.L. Peng and S.Y. Chen, "Trademark Shape Recognition Using Closed Contours," *Pattern Recognition Letters,* vol. 18, no. 8, pp. 791-803, Aug. 1997.
[35] A. Pentland, R. Picard, and S. Sclaroff, "Photobook: Content-Based Manipulation of Image Databases," *Int'l J. Computer Vision,* vol. 18, no. 3, pp. 233-254, 1996.
[36] E.G.M. Petrakis, "Design and Evaluation of Spatial Similarity Approaches for Image Retrieval," *Image and Vision Computing,* vol. 20, no. 1, pp. 59-76, Jan. 2002.
[37] E.G.M. Petrakis and C. Faloutsos, "Similarity Searching in Medical Image Databases," *IEEE Trans. Knowledge and Data Eng.,* vol. 9, no. 3, pp. 435-447, May/June 1997.
[38] E.G.M. Petrakis, C. Faloutsos, and K. Lin, "Imagemap: An Image Indexing Method Based on Spatial Similarity," *IEEE Trans. Knowledge and Data Eng.,* vol. 14, no. 5, pp. 979-987, Sept./Oct. 2002.
[39] P. Salembier and L. Garrido, "Binary Partition Tree as an Efficient Representation for Image Processing, Segmentation, and Information Retrieval," *IEEE Trans. Image Processing,* vol. 9, no. 4, pp. 561-576, 2000.
[40] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-Based Image Retrieval at the End of the Early Years," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 12, pp. 1349-1380, Dec. 2000.
[41] A. Soffer and H. Samet, "Using Negative Shape Features for Logo Similarity Matching," *Proc. 14th Int'l Conf. Pattern Recognition,* vol. 1, pp. 571-573, Aug. 1998.
[42] K. Thorisson, "Simulated Perceptual Grouping: An Application to Human Computer Interaction," *Proc. 16th Ann. Conf. Cognitive Science Soc.,* pp. 876-881, Aug. 1994.
[43] A. Torsello, D. Hidovic, and M. Pelillo, "Polynomial-Time Metrics for Attributed Trees," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 7, pp. 1087-1099, July 2005.
[44] R. Veltkamp and M. Hagedoorn, "State-of-the-Art in Shape Matching," *Principles of Visual Information Retrieval,* Springer, pp. 87-119, 2001.
[45] R. Veltkamp and M. Tanase, "Content-Based Image Retrieval Systems: A Survey," Technical Report UU-CS-2000-34, Inst. ICS, Utrecht Univ., 2000.
[46] *NIST Special Publication 500-242: The Seventh Text REtrieval Conf.,* E.M. Voorhees and D.K. Harmann, eds., Nat'l Inst. Standards and Technology, 1998.
[47] D. Zhang and G. Lu, "Review of Shape Representation and Description Techniques," *Pattern Recognition,* vol. 37, no. 1, pp. 1-19, 2004.

Naif Alajlan received the BSc (with first class honors) and MSc degrees in electrical engineering from King Saud University, Riyadh, Saudi Arabia, in 1998 and 2003, respectively, and the PhD degree from the University of Waterloo, Waterloo, Ontario, Canada, in 2006. He worked as a systems and control engineer in the Saudi Basic Industries (SABIC), Riyadh, from 1998 to 2000 and then joined the Electrical Engineering Department, King Saud University as a lecturer from 2000 to 2003. Since 2007, he has been an assistant professor at the Department of Electrical Engineering, King Saud University. His current research interests include shape retrieval, pattern recognition, and image processing.

Mohamed S. Kamel received the BSc degree (Hons) in electrical engineering from the University of Alexandria, Egypt, in 1970, the MSc degree in computation from McMaster University, Hamilton, Canada, in 1974, and the PhD degree in computer science from the University of Toronto, Canada, in 1981. Since 1985, he has been with the University of Waterloo, Department of Systems Design Engineering and, since 2004, in the Department of Electrical and Computer Engineering. He served as a consultant for General Motors, NCR, IBM, Northern Telecom, and Spar Aerospace. He is currently a Canada research chair professor in cooperative intelligent systems and the director of the Pattern Analysis and Machine Intelligence Laboratory. His research interests are in computational intelligence, pattern recognition, and distributed and multiagent systems. He is the author or coauthor of more than 200 papers in journals and conference proceedings, two patents, and numerous technical and industrial project reports. He is the editor in chief of the *International Journal of Robotics and Automation*, associate editor for the *IEEE Transactions on SMC Part A*, the *Intelligent Automation and Soft Computing*, *Pattern Recognition Letters*, and the *International Journal of Image and Graphics*. He also served as an associate editor of *Simulation* and the *Journal of the Society for Computer Simulation*. He is a member of the ACM, the AAAI, and the IEEE. He has been elevated to a fellow of the IEEE starting 2005. He is a member of the board of directors and cofounder of Virtek Vision Inc., Waterloo.



George H. Freeman received the BSc and PhD degrees in electrical engineering from the University of Waterloo in 1979 and 1984, respectively. He then joined the Continuous Speech Recognition Group, IBM T.J. Watson Research Center, Yorktown Heights, New York, where he worked on statistical language modeling and parallel implementations of speech recognition. Since September 1985, he has been with the Department of Electrical and Computer Engineering, University of Waterloo, where he is currently an associate professor. From 1993-1994, he spent a one-year sabbatical leave consulting for NCR Canada Ltd., Waterloo, Ontario, on the use of fractal and wavelet compression for scanned-document images. His current research interests are in signal processing, particularly lossless or near-lossless data compression and articulation-based hidden Markov modeling for speech processing.