

A Hierarchical Convex Polygonal Decomposition Framework for Automated Shape Retrieval

Sourav Saha, Jayanta Basak and Priya Ranjan Sinha Mahapatra

Abstract With the increasing number of images generated every day, textual annotation of images becomes impractical and inefficient. Thus, content-based image retrieval (CBIR) has received considerable interest in recent years. Keeping it as the primary motivational focus, we propose a method which exploits different degrees of convexity of an object's contour using a multi-level tree structured representation called Hierarchical Convex Polygonal Decomposition (HCPD) tree and the method also uses a special spiral-chain-code to encode the polygonal representation of decomposed shape at every node. The performance of the proposed scheme is reasonably good and comparable with existing state-of-the-art algorithms.

Keywords Convex polygon · Content based shape retrieval · Shape representation

1 Introduction

Due to the recent developments in digital imaging technologies, an increasing number of images are generated every day. In today's tech-savvy world, internet boom is continually driving interest in people to retrieve images of their interest from large data-pool. In order to achieve this task, images have to be represented by

S. Saha (✉) · J. Basak

Department of Computer Science and Engineering, Institute of Engineering and Management, Kolkata, India
e-mail: souravsaha1977@gmail.com

J. Basak

e-mail: lettertojayanta@gmail.com

P.R.S. Mahapatra

Department of Computer Science and Engineering, University of Kalyani, Kalyani, West Bengal, India
e-mail: priya_cksly@yahoo.co.in

© Springer India 2015

J.K. Mandal et al. (eds.), *Information Systems Design and Intelligent Applications*, Advances in Intelligent Systems and Computing 339, DOI 10.1007/978-81-322-2250-7_78

783

specific features. Early attempts tried to use textual annotation of images and then search images using their annotations. Clearly, this method is not practical for large databases. In addition, the textual annotation of image content by itself is a difficult and subjective process. Therefore, searching images using generic features has received considerable attention in recent years. Shape is considered the most promising for the identification of entities in an image. It can be argued that most real subjects are easily identified using only their silhouettes. A user survey in [1] indicated that 71 % of the users were interested in retrieval by shape. Shape is a concept which is widely understood yet difficult to define formally. The human perception of shapes is a high-level concept whereas mathematical definitions tend to describe shape with low-level features. However, for 2-D objects, Marshall [2] tried to define shape as a function of position and direction of simply connected curves within the two-dimensional field. Shape is important visual information that has received much attention from researchers in pattern recognition and computer vision in the past few decades. Most existing techniques for shape analysis and recognition are concerned with single-object shapes, i.e. the silhouette of an object. The main motivation of this research work is to focus on the geometric information, including shape and topology, for content-based image retrieval.

1.1 Related Work

In the past, contour and skeleton were usually used to analyze and represent the shape of objects. Contour-based is an important aspect of human visual perception. Polygonal approximation has been a very popular shape representation technique. It not only satisfactorily represents a shape, but also significantly reduces the amount of processing data for further applications. Therefore, many shape recognition (matching) methods through polygonal approximation [3] have been proposed. However, some conventional methods are somewhat sensitive to non-consistent results of polygonal approximation. Latrcki and Lakamper [4] proposed a convexity rule for shape decomposition based on discrete contour evolution. They concentrate some of decomposition of 2D objects into meaningful visual parts and proposed a contour evaluation method for identifying the visual part whether it is a significant convex part or not. The skeleton is another important method for object representation and recognition. Skeleton-based representations are the abstractions of objects, which contains both shape features and topological structures of original objects. Many researchers have made efforts to recognize the generic shape by matching skeleton structures represented by graphs or trees [5]. Unfortunately, these approaches have only demonstrated the applicability to objects with simple and distinctive shapes, and therefore, cannot be applied to more complex shapes like shapes in a MPEG-7 data set [6]. In this paper, we propose a method which exploits the different degrees of convex properties of contour curvature using multi-level tree structured representation called Hierarchical Convex Polygonal Decomposition (HCPD).

2 Proposed Scheme

The main objective of this research is to develop, implement and evaluate a prototype system for multi-level shape matching and retrieval. We model shape of image objects using a tree structured representation called Hierarchical Convex Polygonal Decomposition (HCPD). The hierarchy of the HCPD reflects the inclusion relationships between the objects' various curvatures along the boundary. To facilitate shape-based matching, a new spiral-chain code for each convex polygon is stored at the corresponding node in the HCPD. The similarity between two HCPD s is measured based on the maximum similarity at every level of the HCPD-tree, where a one-to-one correspondence is established between the nodes of the two trees. An effective string matching algorithm, called Fuzzy Levenshtein edit distance [7], is used to measure the similarity between the shape-attributed nodes' spiral-chain code representation of a polygon (Fig. 1).

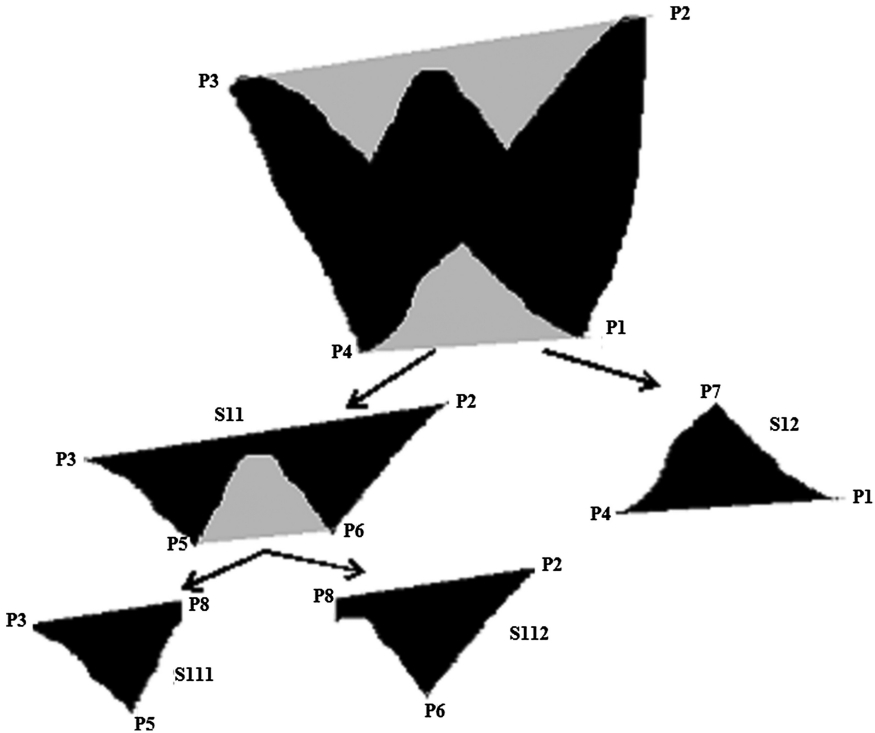


Fig. 1 Hierarchical convex polygonal decomposition

2.1 Hierarchical Convex Polygonal Decomposition

Given a binary image, our proposed method hierarchically decomposes the shape into several disconnected convex and non-convex sub-shapes by forming convex-polygon of the input shape. As shown in figure, shaded regions represent decomposed sub-shapes which are further decomposed at next lower levels. At first level, the convex polygon $\langle P1, P2, P3, P4 \rangle$ results in two sub-shapes, namely S11 and S12. At second level, S11 is further decomposed into S111 and S112 based on convex polygon $\langle P2, P3, P5, P6 \rangle$. As long as convex-polygonal boundary of a shape results in significant non-convex regions, the decomposition operation continues down the tree levels. Therefore, we have not decomposed S12, S111 and S112 further. Notably, we obtain convex-polygon for every decomposed shape generated as child node during hierarchical-tree structured decomposition process as shown in figure and encode each of them with a unique spiral chain directional encoding scheme as illustrated in subsequent section. The formal algorithm is presented below.

```

Algorithm: DecomposeShape(Shape S){
Input: Binary Object Shape S
Output: Hierarchical Tree of convex-polygons and their
corresponding spiral-chain-code
1 L <-FindBoundaryPoint(S);
2 FindConvexPolygon(L);
3 Determine & save Spiral-chain code of the polygon for
respective tree-node.
4 Determine non-convex regions;
5 for(each non-convex region:  $S_i$ ) do{
6     DecomposeShape( $S_i$ ) }
7 }

```

2.2 Boundary Point Tracing

Boundary point tracing is one of many preprocessing techniques performed on digital images in order to extract information about their general shape. In our proposed work, we followed Moore's neighborhood contour tracing strategy [8] to extract boundary points in a specific order (counter-clockwise) to decide which of them form convex-polygon of the object. The boundary point selection criteria examines eight-neighborhood of a point, P, namely locations P1, P2, P3, P4, P5, P6, P7 and P8 as shown in Fig. 2 in counter-clockwise direction. The general idea of Moore's neighborhood contour tracing technique is that every time the counter-clockwise scanning hits an object pixel-point, P, backtrack i.e. trace back to the

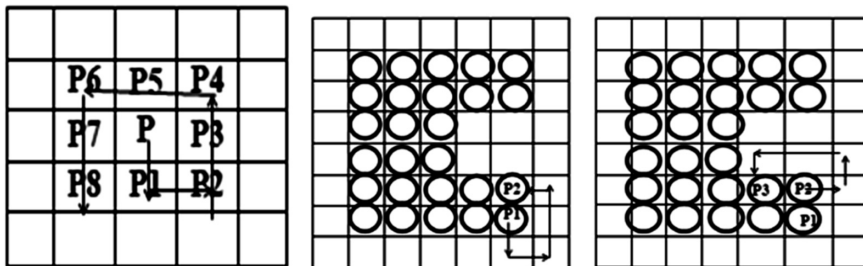


Fig. 2 Boundary point tracing

neighbor pixel from where the object pixel-point P’s location was entered and go around pixel P in anti-clockwise direction, visiting each pixel in its 8-neighborhood, until a new object pixel-point is encountered. The algorithm terminates when the start pixel is visited for a second time.

2.3 Convex Polygonal Approximation

Our convex polygonal covering of a shape is primarily inspired by Graham’s scan algorithm [9] for determining convex hull of an object. Given a digital object, the idea is to detect three successive boundary points in counter-clockwise direction to form an ordered triplet-points $\langle p_1, p_2, p_3 \rangle$ as a candidate and attempt to single out a point from the ordered triplet that needs either to be discarded or picked up for the convex-polygon. The point selection or elimination is based on whether these three points make a left turn or right at position p_2 . The equation as stated below helps us in deciding the turn-direction as it yields non-negative value for left turn but a right turn produces negative value. In case p_1, p_2, p_3 forms a left turn, we may consider boundary point p_1 as convex-polygonal point and the remaining points p_2, p_3 are set as first and second element for the next candidate triplet-points. On the other hand, a right turn implies that p_2 cannot be on convex-polygon of the object and in that case p_1, p_3 are set as first and second element for our next candidate triplet-points. Subsequently, another boundary point (P_4) in counter-clockwise direction is detected and added as the last point of our next candidate triplet-points. Once again we repeat the same procedure to find out the point from the ordered triplet that needs either to be discarded or picked up based on the above mentioned convexity analysis of point p_2 . Our approach deviates from Gram’s scan algorithm with regards to that fact that it considers only boundary points in counter-clockwise direction instead of every object points and discards inclusion of a point if the triangular area generated by the triplet $\langle p_1, p_2, p_3 \rangle$ falls below an empirically selected threshold value. Thus the modified algorithm basically results in an approximated convex-polygon covering the boundary of the input object (Fig. 3).

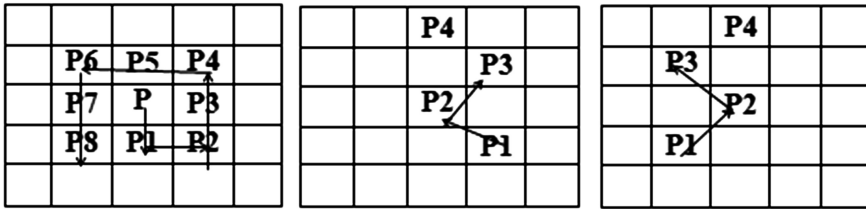


Fig. 3 Convex polygon vertex selection

$$f(p_1, p_2, p_3) = (p_2.x - p_1.x) * (p_3.y - p_1.y) - (p_3.x - p_1.x) * (p_2.y - p_1.y) \quad (1)$$

```

Algorithm: FindApproximateConvexPolygon {
Input: Boundary point-list: L
Output: Convex Polygonal Approximation

1 p1 <- first point in Boundary point-list: L;
2 p2 <- second point in Boundary point-list: L;
3 p3 <- third point in Boundary point-list: L;
4 while(p3 != first point of Boundary point-list: L){
5   if( p1, p2, p3 form a left turn){
6     if(the triangular area generated with p1, p2, p3
7       > AreaThreshold){
8       add p1 to convex polygon point list.}
9     p1<-p2;
10    p2<-p3;
11    p3 <- next point in Boundary point-list: L;}
12 else {
13   p2<-p3;
14   p3 <- next point in Boundary point-list: L;}
15 }}

```

2.4 Convex Polygon Spiral-Chain-Code

In our proposed scheme, we have developed a new spiral chain-code to encode a convex polygon. The idea behind the scheme is to arrange the sides in descending order of their length and consider the largest side as the first reference base. Subsequently, repeatedly we pick up the next available largest un-encoded side from the ordered list and the chain code for the newly chosen side is determined based on the direction of the vector drawn from the mid-point of the last chosen side to the

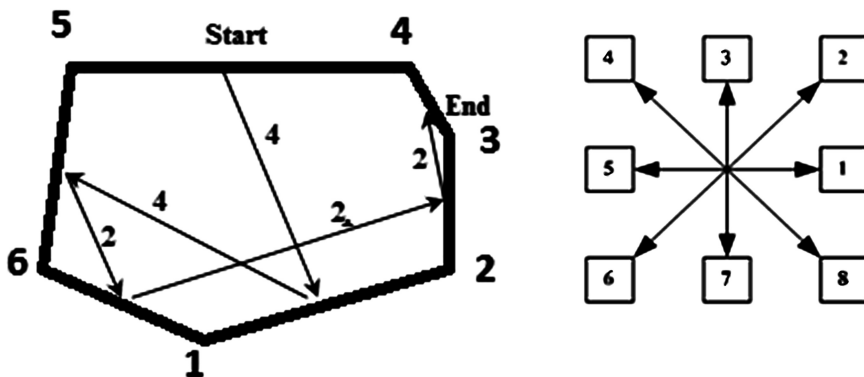


Fig. 4 Convex polygon spiral-chain-coding

mid-point of the newly chosen side. The chain code as shown in Fig. 4 which is used to encode the direction assumes last selected side as X-axis with its counter-clockwise direction (i.e. from node i to node $i + 1$) as positive direction. For example, the sides of the convex polygon shown in Fig. 4 are assumed to be ordered as $\{(4, 5), (1, 2), (5, 6), (6, 1), (2, 3), (3, 4)\}$ based on their length arranged from largest to shortest. The spiral-chain code for the polygon is $\{4, 4, 2, 2, 2\}$.

3 Experimental Results

To evaluate the performance of the proposed shape retrieval system, experiments have been conducted based on the MPEG-7 test database [6]. The dataset consists of 1,400 shapes grouped into 70 classes, each class containing 20 similar objects. Some of the shapes have experienced a number of transformations, such as scales, cuts and rotations and also the image resolution is not constant among them. Figure 5 presents a set of sample images from MPEG-7 test database.

3.1 Performance Evaluation Metric

Evaluation of retrieval performance is a crucial problem in content-based image retrieval, mainly due to the subjectivity of the human similarity judgment. The evaluation of a shape retrieval system depends on the application domain. However, many different methods for measuring the performance of a system have been created and used by researchers. Perhaps the most widely used measure, for retrieval effectiveness; in the literature is the ‘‘Bull’s eyes test’’ [6]. This frequently

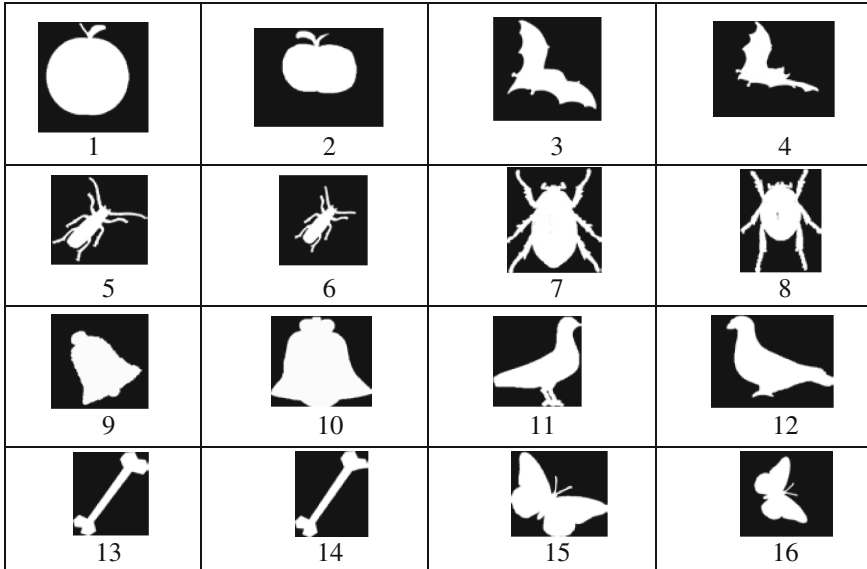


Fig. 5 Image sample data set (MPEG-7 database)

used test in shape retrieval enables the comparison of our approach against other performing shape retrieval techniques. Every shape in the dataset is compared to all other shapes, and the number of shapes retrieved from the same class among the top 40 retrieved similar shapes based on the applied algorithm is reported. Ideally the bull's eye retrieval rate for a query image is the ratio of the total number of retrieved shapes from the same class to the highest possible number which is 20 on MPEG-7. Thus the overall Bulls Eye Percentage (BEP) can be calculated taking average over individual BEP score for every query image from the data set.

3.2 Results

The following table presents the performance of our proposed algorithm for the sample data set shown in Fig. 2 as compared to popular Rammer's Polygonal Shape Chain-Code [3]. As described in previous section, every class of image data set contains 20 samples and relevant retrievals are the images belonging to the class to which the query image is ideally included. One of the interesting observations during experimentation is that as the number of sides in the polygonal shape-representation of an image increases, the performance of retrieval rate falls down. However, on the average the new proposed algorithm resulted in 83.5 % on Bull's eyes test, which seems reasonably good and comparable with existing 7 state-of-the-art algorithms (Table 1).

Table 1 Performance comparison table for the sample data set

Sample no.	Rammer's polygon-chain-code		Proposed algorithm	
	Relevant retrieval	BEP score (%)	Relevant retrieval	BEP score (%)
1	17	85	17	85
2	14	70	15	75
3	15	75	16	80
4	14	70	16	80
5	11	55	14	70
6	11	55	15	75
7	12	60	15	75
8	13	65	15	75
9	17	85	18	90
10	16	80	19	95
11	15	75	17	85
12	14	70	18	90
13	17	85	19	95
14	17	85	19	95
15	13	65	18	90
16	14	70	17	85

4 Conclusion

A novel framework is proposed for CBIR which exploits different degrees of convexity of an object's contour using a multi-level tree structured representation and the method also uses a special spiral-chain-code to encode the polygonal representation of decomposed shape at every node. The performance of the proposed scheme is reasonably good and comparable with existing state-of-the-art algorithms. However, we need to further investigate the performance issues based on complexity of contour curvature as well as various effective shape decomposition tree matching schemes.

References

1. Lambert, S., de Leau, E., Vuurpijl, L.: Using pen-based outlines for object-based annotation and image-based queries. In: 3rd International Conference on Visual Information and Information Systems, pp. 585–592. Amsterdam, The Netherlands (June 1999)
2. Marshall, S.: Review of shape coding techniques. *Image Vis. Comput.* **7**(4), 281–294 (1989)
3. Ramer, U.: An iterative procedure for the polygonal approximation of plane curves. *Comput. Graph. Image Process.* **1**(3), 244–256 (1972)
4. Latecki, L.J., Lakamper, R.: Convexity rule for shape decomposition based on discrete contour evolution. *Comput. Vis. Image Underst.* **73**(3), 441–454 (1999)

5. Ruberto, C.D.: Recognition of shape by attributed skeletal graphs. *Pattern Recogn.* **37**, 21–31 (2004)
6. <http://www.dabi.temple.edu/~shape/MPEG7/dataset.html>
7. Navarro, G.: A guided tour to approximate string matching. *ACM Comput. Surv.* **33**(1), 31–88 (2001)
8. http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/moore.html
9. Graham, R.L.: An efficient algorithm for determining the convex hull of a finite planar set. *Inf. Process. Lett.* **1**, 132–133 (1972)