

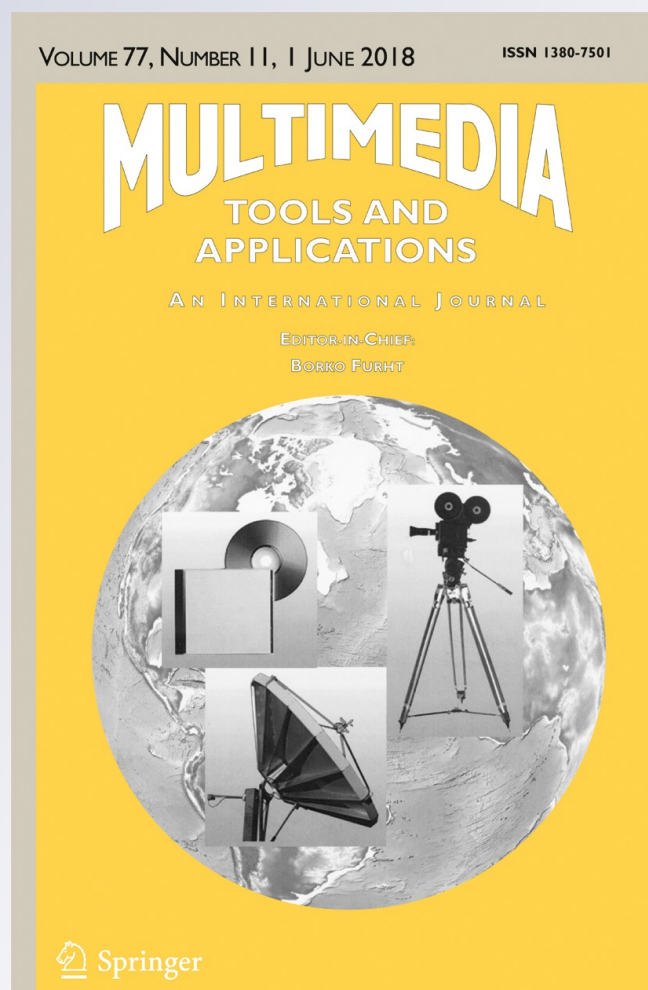
# *Shape retrieval through normalized B-splines curves*

**Nacéra Laiche & Slimane Larabi**

**Multimedia Tools and Applications**  
An International Journal

ISSN 1380-7501  
Volume 77  
Number 11

Multimed Tools Appl (2018)  
77:13891-13921  
DOI 10.1007/s11042-017-4998-x



**Your article is protected by copyright and all rights are held exclusively by Springer Science+Business Media, LLC. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your article, please use the accepted manuscript version for posting on your own website. You may further deposit the accepted manuscript version in any repository, provided it is only made publicly available 12 months after official publication or later and provided acknowledgement is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be accompanied by the following text: "The final publication is available at [link.springer.com](http://link.springer.com)".**



# Shape retrieval through normalized B-splines curves

Nacéra Laiche<sup>1</sup> · Slimane Larabi<sup>1</sup>

Received: 25 September 2016 / Revised: 2 May 2017 / Accepted: 3 July 2017 /

Published online: 15 August 2017

© Springer Science+Business Media, LLC 2017

**Abstract** This paper proposes a new technique for 2D shape modeling and retrieval based only on curves defined from shape boundary. Firstly, a shape representation system is build based on the decomposition of the outline into its constituent parts and their geometric description. The process decomposition is done using high curvature points located along the boundary. These obtained parts are then described by parametric curves using the B-spline approximation and normalized in order to eliminate scaling transformation. Finally, the resulting curves allow matching of shapes and retrieving that is robust to rotation, scale change and deformation. Experiments conducted on a variety of shape databases including Kimia-99, Kimia-216, MPEG-7 and our database created from a selection of ETH-80 shape database, illustrate the performance of the proposed approach when compared with existing algorithms in literature. Obtained results are presented and discussed.

**Keywords** Outline shape · Contour matching · Curvature points · B-spline approximation · Similarity measure · Shape retrieval

## 1 Introduction

One of the most important tasks in computer vision is measuring the similarity between two objects. Objects have various information such as color, shape and texture. Among these features, shape is the most widely used because it provides the main geometrical information that represents an object [17]. In the field of two-dimensional shape representation, there are two main categories: contour approaches and region approaches. Region-based methods rely on global features extracted from the shape such as geometric invariants [41], moment analysis

---

✉ Nacéra Laiche  
nlaiche@usthb.dz

Slimane Larabi  
slarabi@usthb.dz

<sup>1</sup> Computer Science Department, University of Science and Technology, Houari Boumediene (USTHB), Algiers, Algeria

[47]. Since region-based methods don't take into account details, they are robust to noise, little occlusion or deformation. Contour-based methods generally describe a shape by boundary information. Among them, we find Chain codes [30], curvature scale space [33]. Splines [10, 24] and polygons approximation [31] are also another type of boundary-shape methods which have been used for object representation. The B-splines curves possess very attractive properties such as smoothness, continuity and invariance to affine transformations. However there are few works have used the B-splines representation in a two-dimensional image analysis [34].

Therefore, this paper introduces a new boundary-based shape representation and description based on B-splines curves. The key idea is to exploit the geometric information embedded in shape boundary by analysing the shape boundary with the B-splines curve representation. For this, we propose first to extract the discriminative parts of the outline. Then the B-splines representation is applied to modelize each part by parametric curve. This approach is an extension to the approach previously presented [23] but we will present more robust techniques and more test results including a match example of measuring the similarity between shapes based on curve to curve matching. The proposed approach is invariant to geometric transformations and allows more robustness to occluded shapes and noise. Also we show that by using our proposed approach correct recognition is possible under partial occlusion and deformation.

The rest of this paper is organized as follows. The next Section presents a review of some related work on existing shape representation methods. Details on the proposed approach are described in Section 3. Experimental results are presented in Section 4. Section 5 concludes the paper.

## 2 Related work

Shape representation and matching is a current and difficult task in content-based image retrieval. A variety of algorithms have been proposed in the literature for 2D shape analysis and most of them can be classified into two categories: region-based methods and contour-based methods. Region-based methods extract features from the whole shape region without details as segments and dominant points. Therefore, they are easy to compute and robust to noise and shape distortions. Common region-based methods use moments to represent shapes such as Zernike moments [32] and Legendre moments [47] which have been demonstrated to achieve excellent performance. The major limitation is that the descriptors are sensitive to local changes and are not robust against occlusions. Region-based methods also include generic Fourier descriptor [50] and multi-scale Fourier-based description [12].

On the other hand, contour-based methods describe a shape by its outline information. These methods are more complicated, requiring sophisticated implementations but more suitable than global methods for recognizing partially visible objects. Different techniques are proposed such as spectral descriptors which include wavelet descriptors [13, 14, 22]. A curvature scale space (CSS) method is proposed in [33] which is based on finding the maxima of curvature zero-crossing points of boundaries curves at different scales to represent shapes. CSS has shown robustness under the similarity transformations. Scaling, orientation changes, translation and even noise can be easily handled by the representation and its associated matching algorithm but it is sensitive to occlusion. Multi-scale convexity concavity (MCC) representation [1] is another interesting multi-scale descriptor for shape representation and matching. The method is based on the relative displacement of a boundary point with respect

to its position in the precedent scale. Shape matching is achieved using dynamic programming. Authors in [2] proposed the use of the areas triangles formed by the boundary points at different scales. The representation is robust under geometric transformations and modest occlusions.

The medial axis transform or skeleton is a set of idealised lines which aims to give the part structure of the shape in a graph structure [37]. This description is used by Sebastian et al. for shape matching [42]. In their proposed approach a shock graph is created based on the medial axis segments and used for matching. Recently, a new similarity is learned through graph transduction by X. Bai et al. [6]. Wang et al. [45] proposed the use of height function (HF) for shape representation and matching. In their method, the boundary is given by a set of sample point. A height function is defined for each sample point using the distances of all the other points to its tangent line.

In [35], the shape context descriptor (SC) is developed to represent a shape by a set of reference points. A shape context at a reference point finds the correspondences between point sets. Corresponding points on two similar shapes have similar shape contexts. Ling et al. [29] extended the SC descriptor with the inner distance to propose a new descriptor called inner distance shape context (IDSC). The shape matching and comparison is achieved using dynamic programming. Later, Nanni et al. [36] exploit the local phase quantization descriptor to transform the descriptors obtained by the inner distance shape context, shape context and height functions into a matrix descriptor. The comparison between the extracted matrix descriptors is done with the Jeffrey distance. Based on SC and IDSC, Hu et al. [18] present a new descriptor called Coherent Distance Shape Contexts (CDSC) which can well deal with both hand shape and palm print recognitions. Recently, a distinct multiscale shape context (MSC) descriptor is considered by Li et al. [28] for deformable shape representation and retrieval. MSC is based on multiscale context, local features and soft assignment.

Based on both local and global shape features, a rich invariant shape descriptor is proposed in [48] for shape representation, matching and retrieval, the representation uses three types of invariants to extract shape characteristics from different aspects. The matching is achieved using the dynamic programming algorithm.

A major problem in object recognition is the variability in the appearance of an object due to different viewpoints or articulations of parts. So, organizing shape representations in terms of parts allows one to separate the representation of the shape of each individual part from the representation of the spatial relationships between the parts. This, in turn, leads to a more robust representation of shape. Latecki and Lakamper used the correspondence of visual parts for shape contours as a cognitively motivated shape similarity measure [25]. In their approach, a discrete curve evolution method is used as a filter for shape comparison that is a base for shape decomposition into visual parts. In [4], informative patches in images are derived from the training examples and are used as fragments. Then some features are extracted from each fragment and used to estimate the similarity between shapes.

Other techniques consist of approximate the shape contour by a polygon [7] and B-splines [10, 23]. A polygonal approximation of shape contours is considered for shape representation in [7]. In this approach, contours are divided into equal segments and all segments are used as local features for shape matching. Daliri and Torre [11] proposed a representation for shape-representation based on the extraction of the perceptually relevant fragments. According to this approach, each shape is transformed into a symbolic representation, using a predefined dictionary for contour fragments which is mapped to an invariant high-dimensional space used for recognition.

The B-splines curves are widely used for curve representation of the object boundaries [10, 16, 20, 38, 49] because they possess very attractive properties such as smoothness, continuity, affine invariance and local controllability. Although these methods are based on either the calculation of sum error of the corner points [38], the discrete re-sampled points on the B-spline curve [49], or a re-construction of B-splines with a fix number of control points [20]. These methods are noise sensitive and do not employ the property of continuous curve description of the B-splines.

In the above references, the B-splines are used to extract features from boundary or to curve representation. There are very few methods have been devoted to B-splines for curve matching and recognition [34].

In this paper, a method for shape representation and matching based on normalized B-splines curves and perceptual feature points located along the shape boundary is developed. In our algorithm for 2D object representation, we propose to combine the advantages of B-splines that are continuous curve representation and the geometric properties of a contour. The matching of two shapes consists to establish the optimal correspondences of normalized curves by using Chebyshev distance.

Our main contributions are following: Firstly, boundary shape of an object is decomposed into discriminative parts in order to understand and to give a rough description of the shapes. The decomposition process is based on perceptual feature points, which are estimated by curvature. Or when dealing with noisy shapes, the location of high curvature points can be difficult. So in order to locate significant points and to reduce false ones, we extend our work initially proposed in [23] by adding Gaussian filter that lead to better location then better results. Secondly, cubic B-splines curves are used to modelize the obtained parts and then normalized in order to eliminate scale change. Normalization is based on global features. Thirdly, an efficient matching technique that takes into account the orientation of curves is proposed which gives more accurate and offers more elasticity than that reported in our previous work [23]. Finally, several experiments are performed to evaluate the performance of the proposed approach for shape recognition using the different kind and commonly used databases.

### 3 Shape description

The main idea is to use polynomial pieces to describe the parts of outline shape. The advantages are obvious: this representation is rich, compact and local in the sense that a change of a control point in the original curve does not affect the representation entirely. The proposed algorithm for shape representation is done in three steps: first, the method preprocessing is used to smooth noisy points from the boundary shape, second, shape decomposition. Third, the obtained parts are modeled using the B-splines representation and their curvature points.

#### 3.1 Preprocessing step

##### – *Noise cancellation*

To improve the quality of the image for an efficient description and then best matching, a preprocessing method is applied which consists of noise cancellation. Various techniques have been proposed in the literature for noise cancellation like mean filter, Gaussian filter, Median filter, etc. In this paper, Gaussian filter is used in such a way that the representation be less sensitive to noise (see Fig.1).



**Fig. 1** Original shape and its smoothed shape

The Gaussian function in two dimensions is given bellow by the Eq. (1):

$$f(x,y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{1}$$

– **Shape orientation**

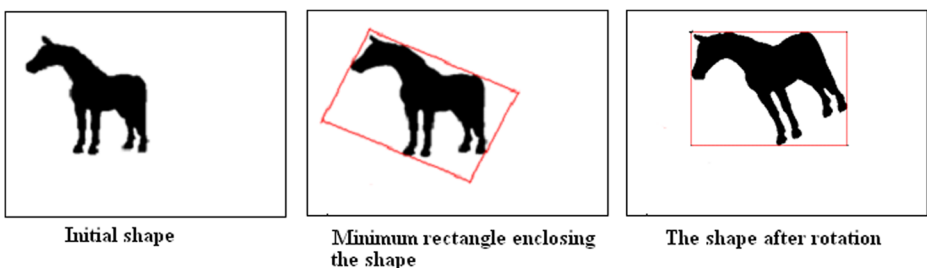
In order to ensure the invariance to shape rotation, the minimum rectangle enclosing the shape is used [15] as shown in Fig. 2. And then the description is computed relatively to this rectangle after its rotation so as the length coincides with rows and the width with columns. The rotation is done via the transformation:

$$x = x.\cos(\theta)-y.\sin(\theta), \quad y = x.\sin(\theta)+y.\cos(\theta) \tag{2}$$

where the shape rotation  $\theta$  is the rectangle's angle of inclination.

**3.2 Part decomposition step**

Outline decomposition into parts can be started by taking into account some features of boundaries which exert a crucial role in attracting the attention of an observer. Examples of such features are the high curvature points. These points give important clues for shape representation and analysis, because they can be used as local geometric measurement. The proposed algorithm for shape decomposition is based on the concave points. These points will be used as junction points and can be estimated by curvature. There are various techniques for estimating higher curvature points along the shape boundary. In the present work, the high curvature points are detected by using the Chetverikov's algorithm [9] which needs a pair of parameters:  $(d, \alpha)$ , where  $d$  represents the distance



**Fig. 2** The minimum rectangle MR encompassing shape

between points in the shape boundary and  $\alpha$  quantifies the curvature. In the following we will give the full algorithm that might decompose a shape into its different constituent parts:

- The first step consists of locating the high curvature points along the smoothed contour by using Chetverikov algorithm, Then we detect and localize the concave points.
- In the second step, we set a Euclidian distance  $\delta$  as a critical threshold in order to eliminate the redundant concave points. The points separated by  $\delta$  are deleted.
- Finally, the last step consists in selecting the most significant concave points. The curvature points obtained from the previous step are not all interesting points; we eliminate unwanted ones which correspond to those of low degree. Only curvature points with high degree of concavity are kept. There are several methods of approximating concavity degree. In our approach, the degree of each point is estimated as the ratio  $d/l$  where  $d$  is the distance from the concave point to associated chord of length  $l$  (see Fig. 3). Estimate the degree of concavity of each selected point.

Figure 4 shows some higher curvature points of some shapes.

After these different steps, we obtain the shape represented by a set of concave points. These points are used to split outline shape into a collection of curves  $C_i$  that can be combined to yield the shape as illustrated by Fig. 5.

### 3.3 Parts modeling

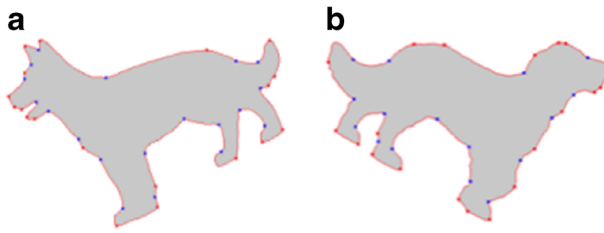
The description of the shape is the set of descriptions of its different parts by using the B-splines curves. B-splines representation is compact, robust to noise and has local controllability. The B-splines model is used to approximate each part  $C_i$  by a parameterized cubic B-spline curve without loss of information. This approximation is done such as the polynomial functions are associated to connected curves. The degree three is chosen instead of a higher degree because it has local differential geometry properties.

The B-spline approximation can be determined yielding some control points  $P_i = (x_i, y_i)$ ;  $i = 0$  to  $n$  extracted from the discrete curve. The parametric B-spline curve  $C(t)$  of order  $p$  is defined by:

$$C(t) = \sum_{i=0}^n N_{i,p}(t)P_i. \tag{3}$$

**Fig. 3** Concavity and convexity degree



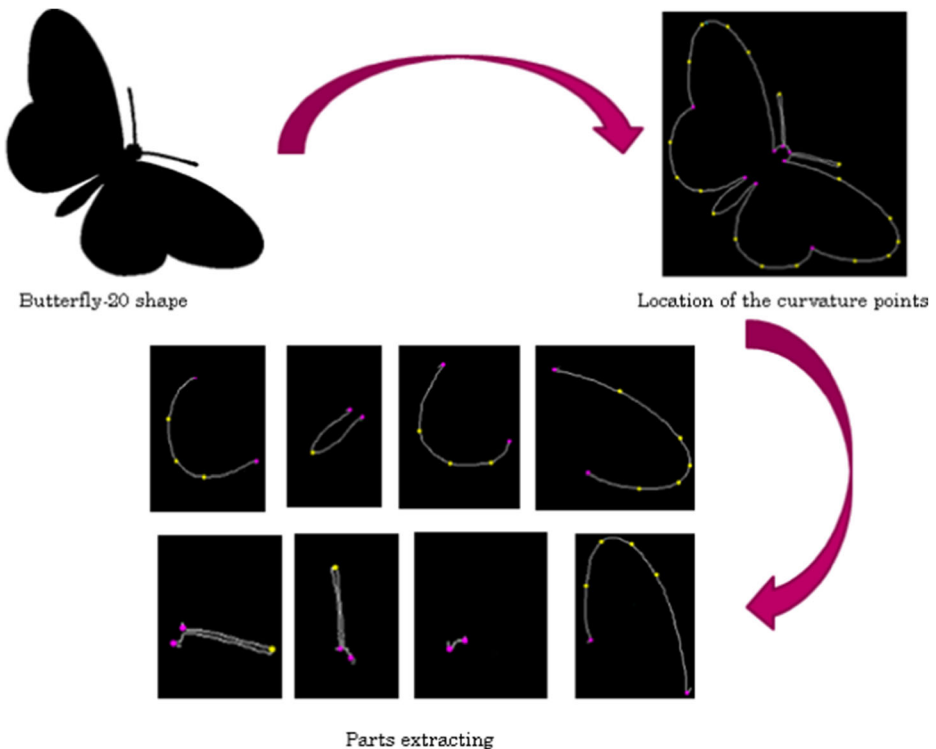


**Fig. 4** High curvature points extraction using Chetverikov algorithm (red points represent the convex points and the blue one represent the concave points) for (a)  $d = 9 \text{ pixels}$ ,  $\alpha = 156^\circ$ . (b) with  $d = 14 \text{ pixels}$ ,  $\alpha = 154^\circ$

Where  $N_{i,p}(t)$  are the splines basis functions of order  $p$  defined over a knot vector  $U = \{u_0, u_1, \dots, u_{n+p+1}\}$  [40]. The knot vector  $U$  consists of non decreasing real valued knots and mostly its first and last knot are repeated with multiplicity equal to order  $p$  as follows:

$$u_0 = \dots = u_p = 0, u_{n+1} = \dots = u_{n+p+1} = 1.$$

In practice, it is extremely difficult to specify an initial B-spline curve with a suitable number of control points distributed appropriately so as to yield a satisfactory approximating curve. Different solutions are proposed and the most of them use an iterative process for adjusting the number of control points and the knots in order to obtain a best approximation of the shape [44]. Based on square distance minimization, their approach makes an initial active B-spline curve which converges towards the target curve. Later, Yang et al. present an algorithm for adjusting the control points for B-spline approximation which is driven by the Pottmann et al.'s



**Fig. 5** Splitting of some outline shape using concave points in pink

optimization approach [46]. Recently, Chen et al. proposed a new approach for the B-spline curve approximation which is tangent to a target curve at a set of selected positions, called tangent points. These points are located using heuristic method [8]. In our case, the solution proposed for adjusting the control points of the B-spline curve is different from the conventional ones in curvature point's selection. The control points are located at high curvature points on the outline shape. More specifically, more of these points are selected as convex points.

In fact, when the control points are located as convexity points on the outline shape, it has a greatest impact on the description accuracy of the curve approximation (descriptor) and then on the matching process. The number of these points differs from a part to another. It depends of the geometry and the length of each curve and it is equal or higher than 4. The knot points which are defined as the junction between curve pieces are given by a linear combination of the located control points. In the experimental section, we will present some examples of the B-splines representation when the convex points are not considered.

Figure 6 illustrates high curvature points and resulting parts of the shape of a car from ETH-80 database [27], in addition a detailed description of the control points set  $N$  is given when the resulting parts are different.

The resulting B-splines functions are continuous in  $x$  and  $y$  and can accurately measure the similarity measure between two shapes.

### 3.4 Normalization

In order to make the proposed approach robust with respect to deformations and invariant to geometric transformations, we propose the following normalization methods.

– *Invariance to translation*

Invariance to translation is used as the first step of normalization. The obtained curves are translated so that the centroid of the shape is mapped to (0, 0) (see Fig. 7) by using the transformation below:

$$x' = x - x_g, \quad y' = y - y_g \tag{4}$$



Curves	Control points	N
$C_1$	(57.291,-191.849),(52.770,-198.405),(48.082,207.726),(44.669,-240.334), (65.053,-242.101),(81.860,248.691), (94.864,-245.691), (98.477,-252.896)	8
$C_3$	(346.840,260.031),(325.512,252.011),(373.404,267.020),(404.640,237,228), (368.231,-197.119),(330.888,-197.355),(313.013,-195.574),(313.044,- 186.829), (296.964,-188.505), (278.970,197.692)	10
$C_7$	(174.017,184.869), (151.531,-179.752), (127.970,-178.976), (105.019,177.859)	4

**Fig. 6** Illustration of control points. **a** Example shape. **b** Corresponding curves of outline shape and their control points. Bottom: Example of some curves with their N control points

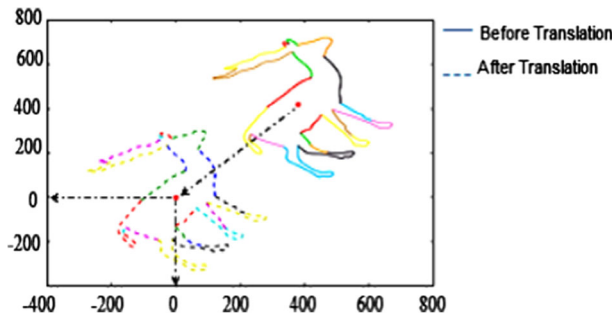


Fig. 7 Curves translation

Where  $(x', y')$  denotes the normalized curve point corresponding to the curve point  $(x, y)$  of shape and  $(x_g, y_g)$  are the spatial coordinates of the shape's centroid (Fig. 7). The shape's centroid  $C_g = (x_g, y_g)$  is given by:

$$x_g = M_{10}/M_{00}, \quad y_g = M_{01}/M_{00} \tag{5}$$

where  $M_{00}$ ,  $M_{10}$  and  $M_{01}$  are the first geometric moments of the shape defined as:

$$M_{pq} = \sum_x \sum_y x^p y^q f(x, y) \tag{6}$$

with  $f(x, y)$  is the mass distribution of the object,  $f(x, y) = 1$  or  $0$ .

– **Invariance to scale change**

The description of curves presented above changes when the outline shape is resized. To ensure the invariance to scale change without losing any information on the original curve, we carry out normalization by converting each B-spline curve as follow:

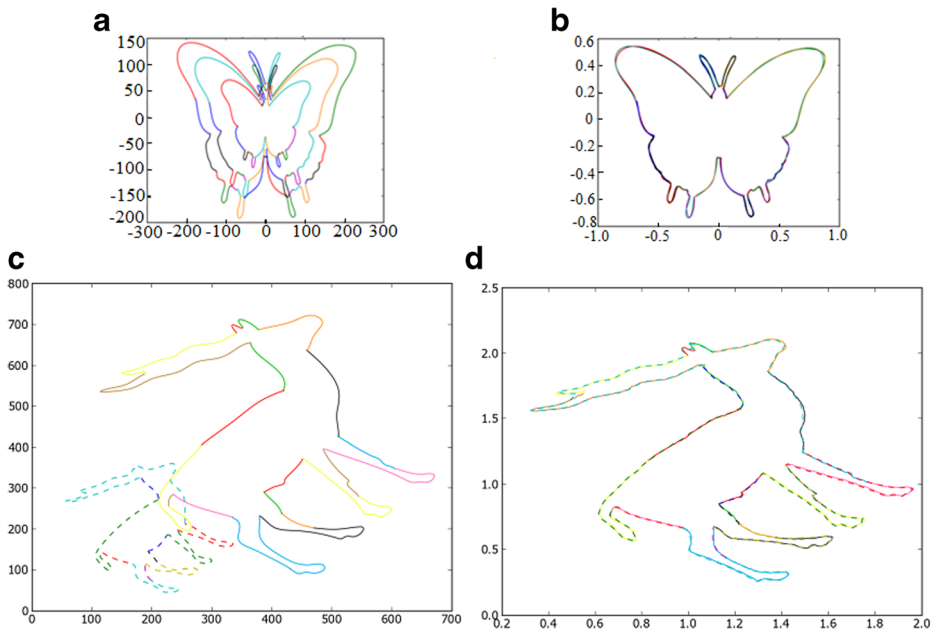
$$p_i(x_i, y_i) \rightarrow (x_i/D_{Max}, y_i/D_{Max}). \tag{7}$$

Where  $p_i(x_i, y_i)$  is a point of the B-spline curve and  $D_{Max} = \max \{d(C_g, P_i)\}$  represents the maximal distance from the centroid to the boundary curve (see Fig. 8).

This transformation allows us to bring back the different B-spline curves approximating the original boundary curve at different scales to the same neighborhood. Therefore the proposed transformation will give the same descriptor as illustrated by Fig. 9.

Fig. 8 Maximal distance





**Fig. 9** **a** Approximated boundary curve of butterfly-4 at 50%, 80% and 100% before normalization, **b** After normalization, **c** Boundary curve of deer-7 at 50% and 100% before scaling normalization, **d** After normalization

Figure 9 shows that the different B-splines curves at different scales are closely superposed.

## 4 Matching process

In this section, we propose an effective matching strategy to compute the similarity between two shapes based on their descriptors. For a query shape and an arbitrary model shape in the database, we can use our proposed shape representation approach, described in the previous section to extract their descriptor vector (the normalized B-splines curves). Then the similarity value between their descriptors is computed by the proposed matching process. Our matching process consists of two stages: global matching based on the first order of geometric moments and curve sequence matching. The details will be explained in the following subsections.

### 4.1 Global matching

Given an input (reference) shape with  $n$  curves, the search in a database of  $M$  shapes will require  $O(nmM)$  comparisons, where  $n$  is the average number of curves for each shape in the database. For speeding up the search while minimizing the risk of skipping good candidates, we narrow the search space. Global features are very important in this task since they are usually easy to extract and manipulate such as the normalized object centroid that will be used in the next. In this stage, we provide the global matching without defining the B-splines curves. We deal only about a global matching.

Shape centroid is a widely used feature to represent global information of the image. It has the properties of rotation, scale change and translation invariance. It can be computed as a mean value of coordinates of all the boundary points, so it allows us to analyze shape by using its outline only:

$$\left( \sum_{i=1}^N x_i/N, \sum_{i=1}^N y_i/N \right) \tag{8}$$

where  $N$  is the number of boundary points.

Shape centroid is often used for normalization. It is also used to construct shape descriptors see [21, 51]. In the present work, we consider the distance between the shapes' centroids to narrow the space research. For this a similarity measure  $D(C_g, C_g^m)$  between a pair of centroids  $C_g$  and  $C_g^m$  is defined. This measure is given as the normalized Euclidean distance in order to find correct responses even in deformations or occlusions. This normalization allows us to make our global matching insensible to shape's centroid positions when the shape is altered by some local deformations or articulations. Let  $C_g = (x_g, y_g)$ ,  $C_g^m = (x_g^m, y_g^m)$  be the centroids of query and shape model, the similarity measure  $D(C_g, C_g^m)$  is defined by the following equation:

$$D(C_g, C_g^m) = \left[ \left( \frac{x_g}{D_{Max}} - \frac{x_g^m}{D_{Max}^m} \right)^2 + \left( \frac{y_g}{D_{Max}} - \frac{y_g^m}{D_{Max}^m} \right)^2 \right]^{1/2} \tag{9}$$

where  $D_{Max}$  (resp.  $D_{Max}^m$ ) be the maximal distances from the centroid of query (resp. model) to the boundary curve.

Shapes for which  $D(C_g, C_g^m)$  is larger than a fixed threshold  $T_D$  are discarded, where  $T_D$  is fixed experimentally after various tests.

### 4.2 Curve sequence matching

Based on the results of the first global matching, this stage provides the local matching. As the basic idea of our proposed representation is that an object is identified by its various curves, the matching between a query and model shapes requires the similarity between of their all normalized curves.

– **Curve similarity measure**

To achieve curve matching, a similarity measure  $S(C_i, C_j)$  between two normalized curves  $C_i$  and  $C_j$  of two shapes  $Q$  and  $M$  respectively is defined using the Chebyshev distance. This similarity is given as the greatest of their differences along their points.

Let  $\{(x_1, y_1), \dots, (x_n, y_n)\} = C_i$  and  $\{(x'_1, y'_1), \dots, (x'_m, y'_m)\} = C_j$  be the sets of two curve points to match, the similarity measure is described as follows:

- We first check if the curves to match are oriented along the (oy) axis. If it is the case, the curve matching is done according to the (ox) axis: for each  $y_i$ , the distance  $|x_i - x'_i|$  is

computed, where  $(x_i, y_i) \in C_i$  and  $(x'_i, y'_i) \in C_j$ . Then the maximum value of this distance will be considered as the matching cost between the two curves.

- Otherwise, we apply the Chebyshev distance along the second coordinates which that consists to compute  $\max|y_i - y'_i|$ .

This similarity measure illustrates the absolute magnitude of the differences between coordinates of the two curves. So, a valid match between two normalized curves is considered if their matching cost is less than a given threshold  $T_C$ , otherwise they are different.

– **Shape similarity score**

Let  $S_1, \dots, S_p$  be the database of shapes and let  $Q$  be the query shape. Shape similarity between  $Q$  and an arbitrary shape  $S_i$  in the database is performed by comparing their corresponding descriptors. As the proposed shape descriptor is given as an ordered sequence of normalized curves, the descriptor vectors could be of different sizes. Therefore, the matching of two shape descriptors requires to define an effective matching strategy to compute the similarity between descriptor vectors such as for each curve of  $S_i$ , we establish the best correspondence with the curve of  $Q$ . On the other hand, this strategy must take into account curves' order of each shape. In order to make this possible without losing the notion of sequence and allowing elasticity in the shape matching, circular permutations are applied on the descriptor vector having the higher dimension. Then the matching between the two descriptors is achieved for each permutation. Finally, the similarity score of the two shapes  $Q$  and  $S_i$  is computed based on these similarity values.

Let  $Q$  and  $S$  be two shapes with  $n_q$  and  $n_m$  normalized B-splines curves respectively. We suppose that  $n_q \leq n_m$ , the shape similarity score can now be described while respecting the following steps:

- 1st Step: – The  $n_q$  curves of  $Q$  are compared with the  $n_q$  first curves of  $S$  according to their order; the obtained result is represented by the vector  $V_0$  whose constituents  $v_0^{i,i}$  represent the result of the comparison:

$$V_0 = (v_0^{1,1}, v_0^{2,2}, \dots, v_0^{k,k}, \dots, v_0^{n_q, n_q}); \quad 1 \leq k \leq n_q.$$

Where  $v_0^{i,i} = S(C_i, C_i^m)$ .

- Based on the curve similarity measure defined bellow, a noted vector  $W_0$  is associated for the vector  $V_0$ . The components of this vector are 0 or 1: 1 if  $v_0^{i,i}$  is a valid match and 0 otherwise.

- 2nd Step: The process of the first step is repeated with each permutation on the descriptor vector of  $S$  until matching all curves of  $Q$ . As we have  $(n_m - 1)$  permutations, we obtain  $(n_m - 1)$  vectors. The obtained results for the  $i^{\text{th}}$  permutation will be represented by the noted vector  $V_i$  given by the form:

$$V_i = \begin{cases} t(v_i^{1,i+1}, v_i^{2,i+2}, \dots, v_i^{n_q, i'}) & \text{if } : n_q < n_j. \\ t(v_i^{1,i+1}, v_i^{2,i+2}, \dots, v_i^{n_q, i'}) & \text{if } : n_q = n_j. \end{cases} \quad (10)$$

Such that the parameters  $r$  and  $s$  satisfy the following:

$$1 \leq r \leq n_j - 1 \text{ and } r \neq n_q, \quad 1 \leq s \leq n_q - 1 \text{ and } s \neq n_q.$$

The vector  $W_i$  associated to  $V_i$ , with components 0 or 1, is determined such as:

$$w_i = \begin{cases} 0 & \text{if } v_i^{k,l} > T_C \\ 1 & \text{if } v_i^{k,l} \leq T_C \end{cases} \quad (11)$$

where  $T_C$  is a threshold determined experimentally.

3rd Step: According to the vectors  $W_i$ , we can compute our shape similarity score between the query and a model shape. It is defined by the ratio:

$$Simscore(Q, S) = w/n_q \quad (12)$$

where  $w$  represents the maximal number of occurrence of the value 1 in the vectors  $W_i$ . It corresponds to the number of matched (similar) curves between  $Q$  and  $S$ ,  $n_q$  is the minimum number of curves of  $(Q, S)$ .

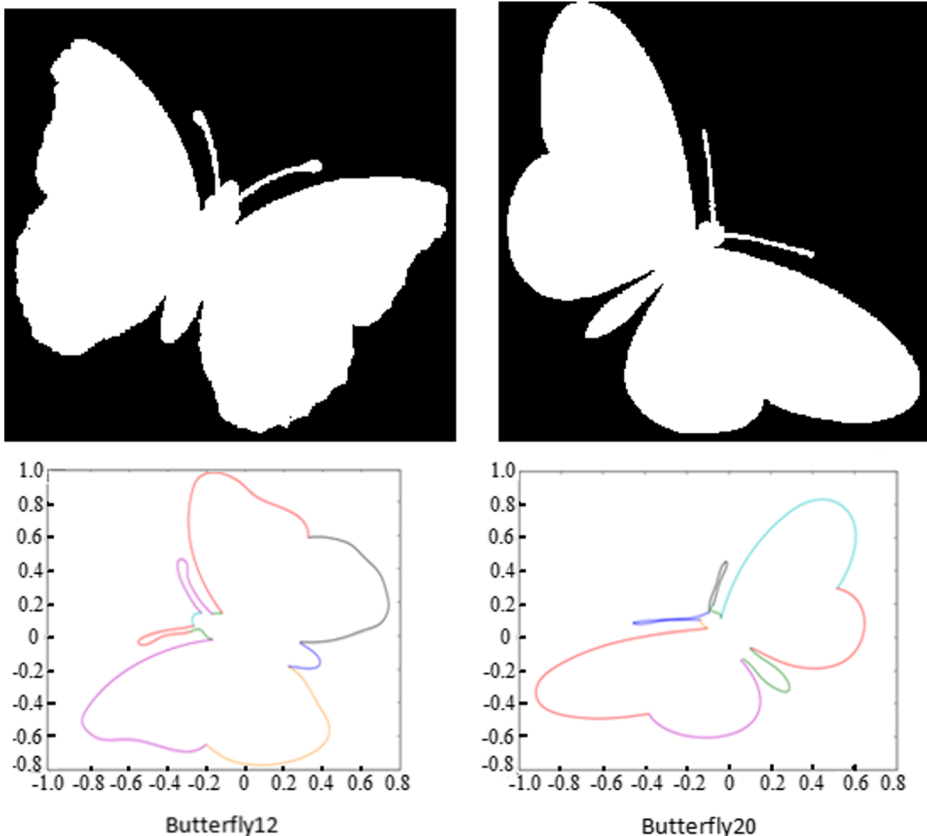


Fig. 10 Two butterfly shapes and their corresponding B\_splines curves at the bottom

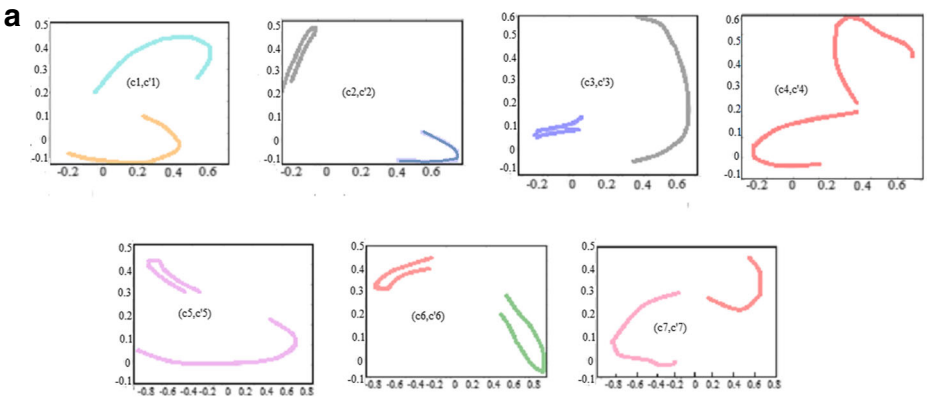
Therefore, to have the list of retrieval shapes, the similarity score between the query shape and each reference shape is computed and the matching results are ranked according to the order of their scores.

In Fig. 11, we illustrate some examples of curve similarity of two samples from the category butterfly: butterfly-12 and butterfly-20. Figure 10 shows these two examples of shapes and their corresponding B-splines representation.

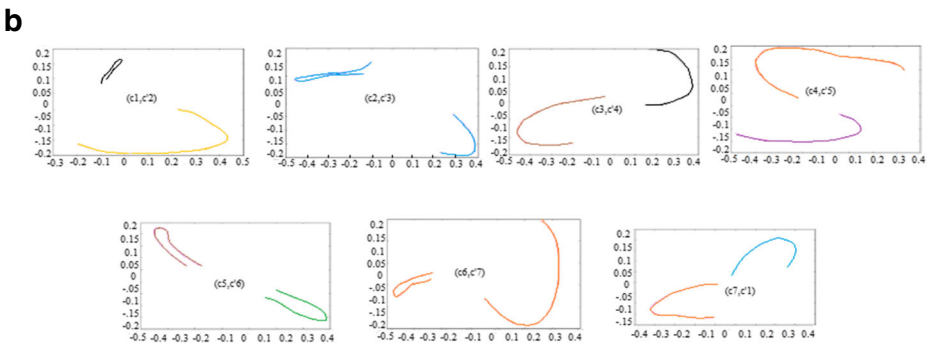
Figure 11 illustrates how the curve to curve matching is achieved at some selected permutations. Best matched pairs of curves are presented at the bottom of each case.

For example in the first case (see Fig. 11a), the curves of the two butterflies are compared in the same order. By applying the proposed curve matching algorithm, only the pair  $(C_1, C'_1)$  is considered as the best possible correspondence.

At the 5th permutation, the best pairs of matched curves, given by our system are  $(C_3, C'_7)$ ,  $(C_1, C'_5)$ ,  $(C_2, C'_6)$ ,  $(C_4, C'_1)$ ,  $(C_5, C'_2)$  and  $(C_7, C'_4)$ . We can see that in most cases, the curve matching results are achieved correctly.

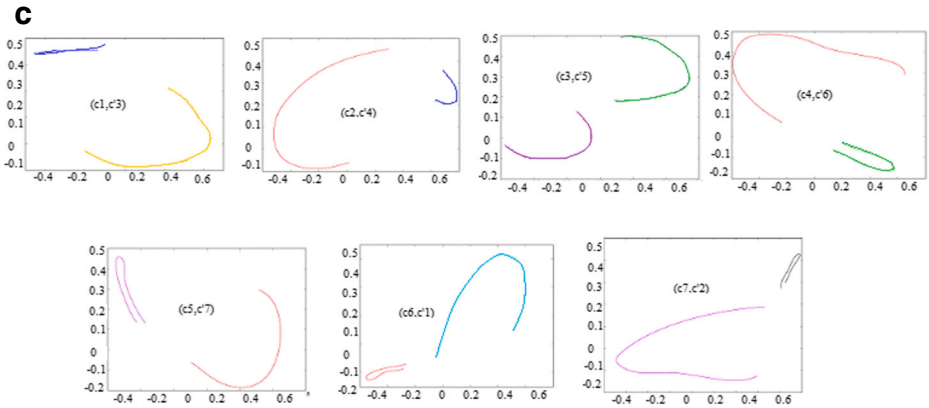


At zero permutation:  $(C_1, C'_1)$

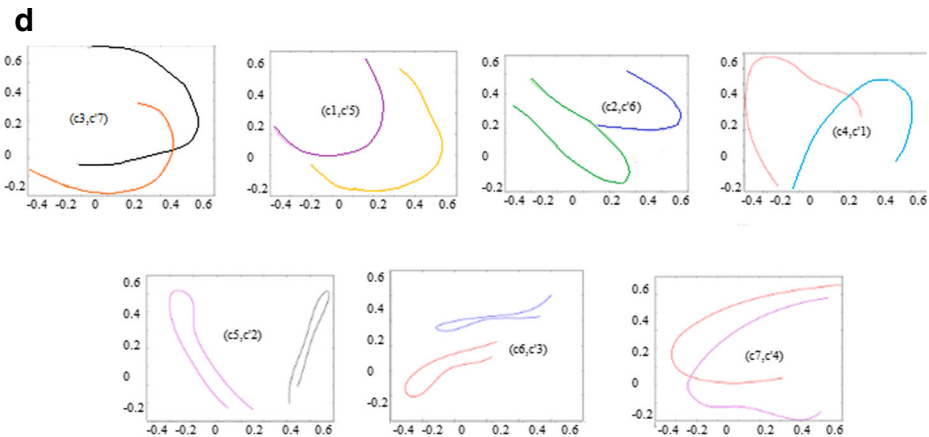


At the 1<sup>st</sup> permutation:  $(C_4, C'_5)$

**Fig. 11** Results of B-splines curves matching at some permutations



At the 2<sup>nd</sup> permutation: No matched curves found by our matching system.



At the 5<sup>th</sup> permutation:  $(C_3, C'_7), (C_1, C'_5), (C_2, C'_6), (C_4, C'_1), (C_5, C'_2)$  and  $(C_7, C'_4)$ .

**Fig. 11** (continued)

























## 5 Experimental evaluation

In this section, a series of some experiments are reported in order to evaluate the efficiency of the proposed approach for shape modeling and retrieval. All the experiments were conducted using benchmark shape databases: ETH-80, Kimia-99, Kimia-216 and set B of the MPEG-7.

### 5.1 ETH-80 database

We used the shapes database of Leibe and B.Schiele [27], where each object is represented by some views spaced evenly over the upper viewing hemisphere. These shapes are selected to have significant between class-differences. They are divided into 14 classes of objects consisting of horses, cows, apples, dogs, tomatoes, cups, pears, etc.. A sample of these shapes is illustrated by Table 1.

**Table 1** Some shapes of the database with rotation and scaling

Class 1						
Class 2						
Class 3						
Class 4						

Normalized curves for all models are constructed and stored in the model database in an off-line stage. Tests are conducted on objects of different shapes with different rotations and sizes. The proposed method is experimentally evaluated to assess its effectiveness, scalability and ability to handle occluded and partial queries. It is performed giving variety kinds of experiments:

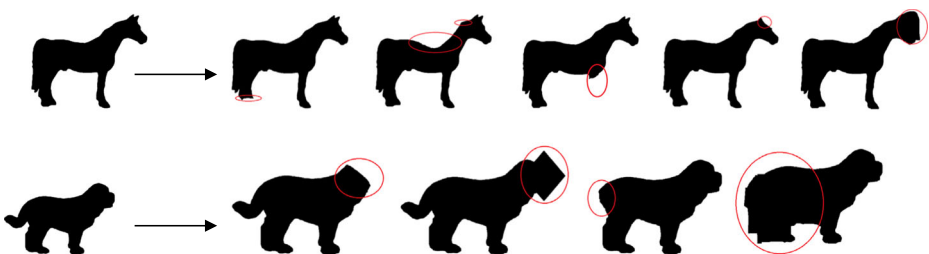
### 5.1.1 Deformation and shape's centroid position

For an experimental illustration that the centroid position and the proposed normalized measure cope well with shape deformations, we show some experimental results for selected examples. We consider the shapes horse, dog and cow which are geometrically distorted. Some of the sample shape distortions are illustrated by red circles as shown in Fig. 12.

In this study, our experimental procedure consists of creating different random deformations for each shape. Then computing the matching cost of the original and deformed shapes, using both the proposed measure defined in Eq.(9) and the Euclidian distance. The results of some examples of these experiments are presented at the bottom of each shape (see Fig. 13). Finally, we compute the average matching cost for each class.

A summary of the obtained average matching cost values over a selection of shapes of ETH-80 database, illustrated by Fig. 12, is given in Table 2. Observing the resulting matching cost values, we conclude that the proposed normalized measure based on the shape centroid deals well when deformations occur.

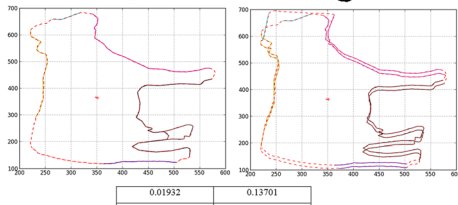
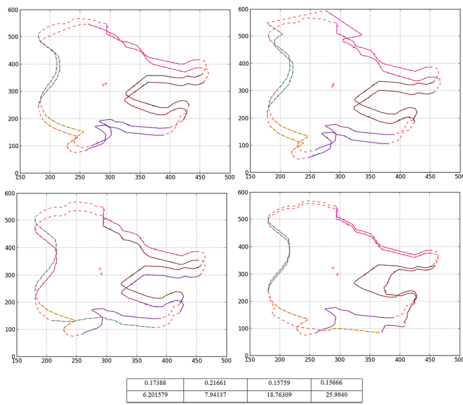
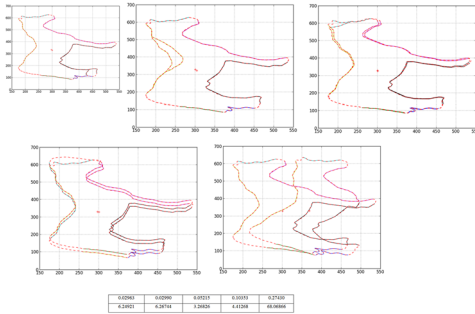
Equivalently, we examine the measure's performance in case of different shapes but of different shape classes. In order to achieve this experiment, we have selected some shapes as



The original shape

The deformed and the occluded shapes of the primary shapes

**Fig. 12** Samples of shapes from the ETH-80 database with their deformed and occluded shapes

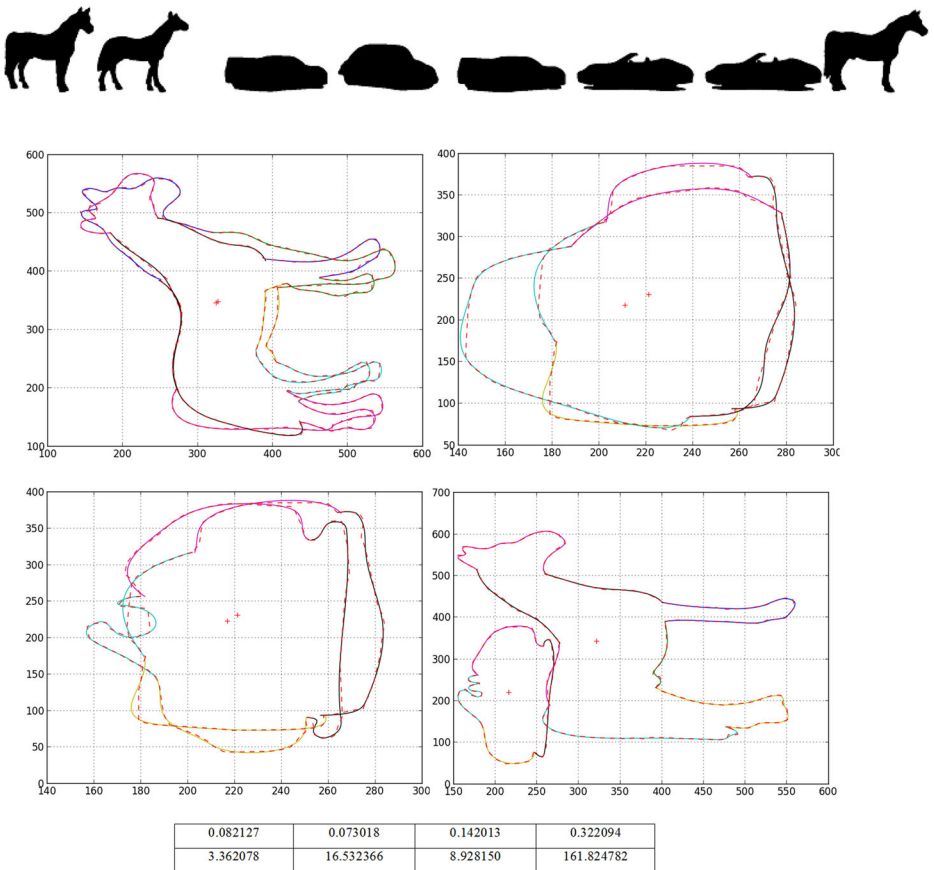


**Fig. 13** Examples of boundaries of different shapes of the same shape class. For each class, the first row represents the deformed shapes, the second one illustrates the superimposed B-spline curves of the deformed shapes with the original shape with their centroids and the results presented at the bottom of each class represent the matching cost values (the first row corresponds to the proposed normalized measure and the second one to the Euclidean distance)

**Table 2** Average matching cost values for different shapes of the same class

Methods	Shape class		
	Horse	Dog	Cow
Proposed measure	0.097902	0.176185	0.078165
Euclidean distance	17.65325	14.72500	3.59889

horse, dog, including their random deformations, cow and car. The matching cost between different pairs of shapes is estimated by the proposed measure and the Euclidean distance. Some matching examples are depicted in Fig. 14. The resulting matching values of these experiments show also that the proposed normalized distance based on shape's centroid seems to be quite efficient in the problem of shape similarity retrieval under deformations or occlusions. Another interesting observation made from the results shown in Fig. 14 is that, for a given pair of shapes of different types of objects but in the same class as for car and horse examples, the matching cost value remains very low compared to the rest examples as



**Fig. 14** Matching costs values for different pair of shapes of different shape classes using the proposed normalized measure and the Euclidean distance respectively

illustrated for example by the pair of shapes: (cow, dog) or (car, horse). This indicates that the measure proposed here can be used to classify shapes into different shape classes.

### 5.1.2 Matching results

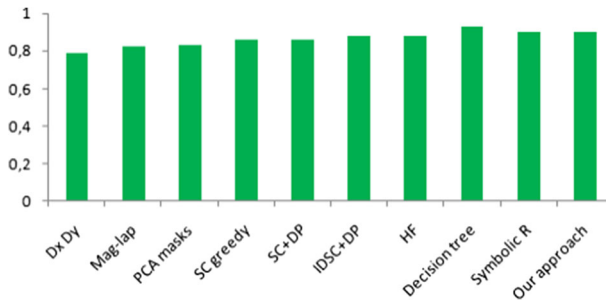
The first series of tests were conducted to evaluate the performance of our method to scale change and orientation. Some examples of shape retrieval and their matching scores are illustrated in Table 3. Results are depicted in every row in a decreasing order of similarity with the query. For each shape query, we can see that the method gives satisfying results as the selected shapes (best matches) belong to the same class.

**Table 3** Matching scores for some results

Query shape	First match	Second match	Third match
	88%	82%	76%
	100%	50%	33%
	100%	50%	50%
	100%	81%	62%
	100%	70%	70%
	90%	90%	63%
	83%	66%	25%
	40%	7%	5%
	5%	4%	40%
	25%	10%	6%
	None		

**Table 4** The three most similar shapes retrieved for occluded shapes and partial test queries

Query shape	Most similar	Second similar	Third similar
	50%	50%	40%
	16%	12%	11%
	12%	8%	7%
	48%	34%	28%
	17%	5%	5%
	32%	25%	20%
	40%	20%	20%
	21%	18%	15%
	26%	23%	20%
	75%	50%	50%
	None		
	23%	22%	20%
	40%	33%	33%
	40%	33%	33%

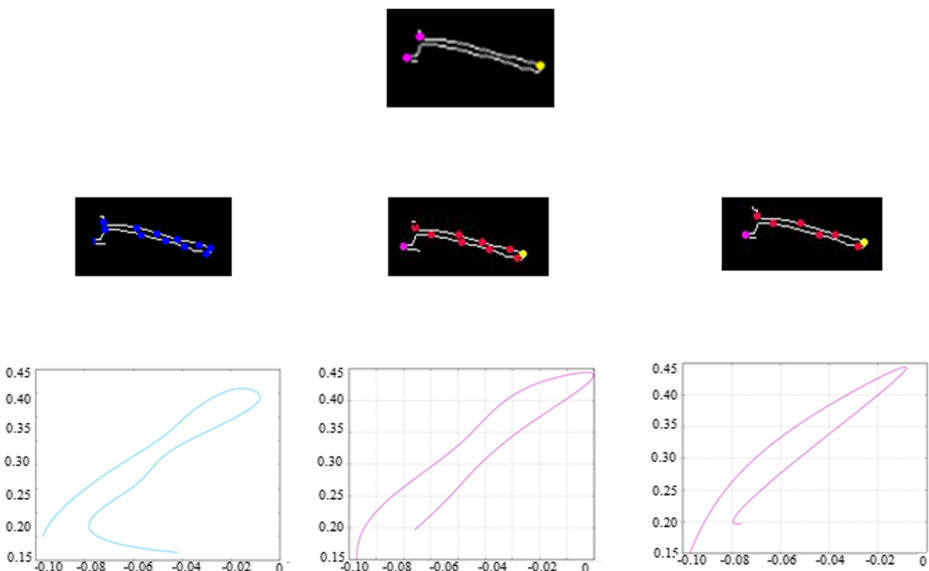


**Fig. 15** Retrieval rates on ETH-80 database

We also studied the robustness of our approach to find the most similar to query shape knowing that there isn't shapes of the same class in the database, as we can see by the last three examples of Table 3. For the human query, there is no similar shape found by our system, we obtain the query shape as a collection of its different curves. Towards these results, we can conclude that the proposed representation is more discriminating, also, a higher resistance.

In the second experiment, we evaluate the performance of the proposed approach to recognize occluded objects which is a difficult task in computer vision. Several shapes obtained as superimposing two shapes or more from the database are used as shape queries. Table 4 shows that for different degree of occlusion (50% for the three first queries and 90% for the fourth, fifth and sixth query) our proposed algorithm can classify correctly occluded shape. We considered that retrieved shape is correct if it is similar to at least one of the shapes composing the query.

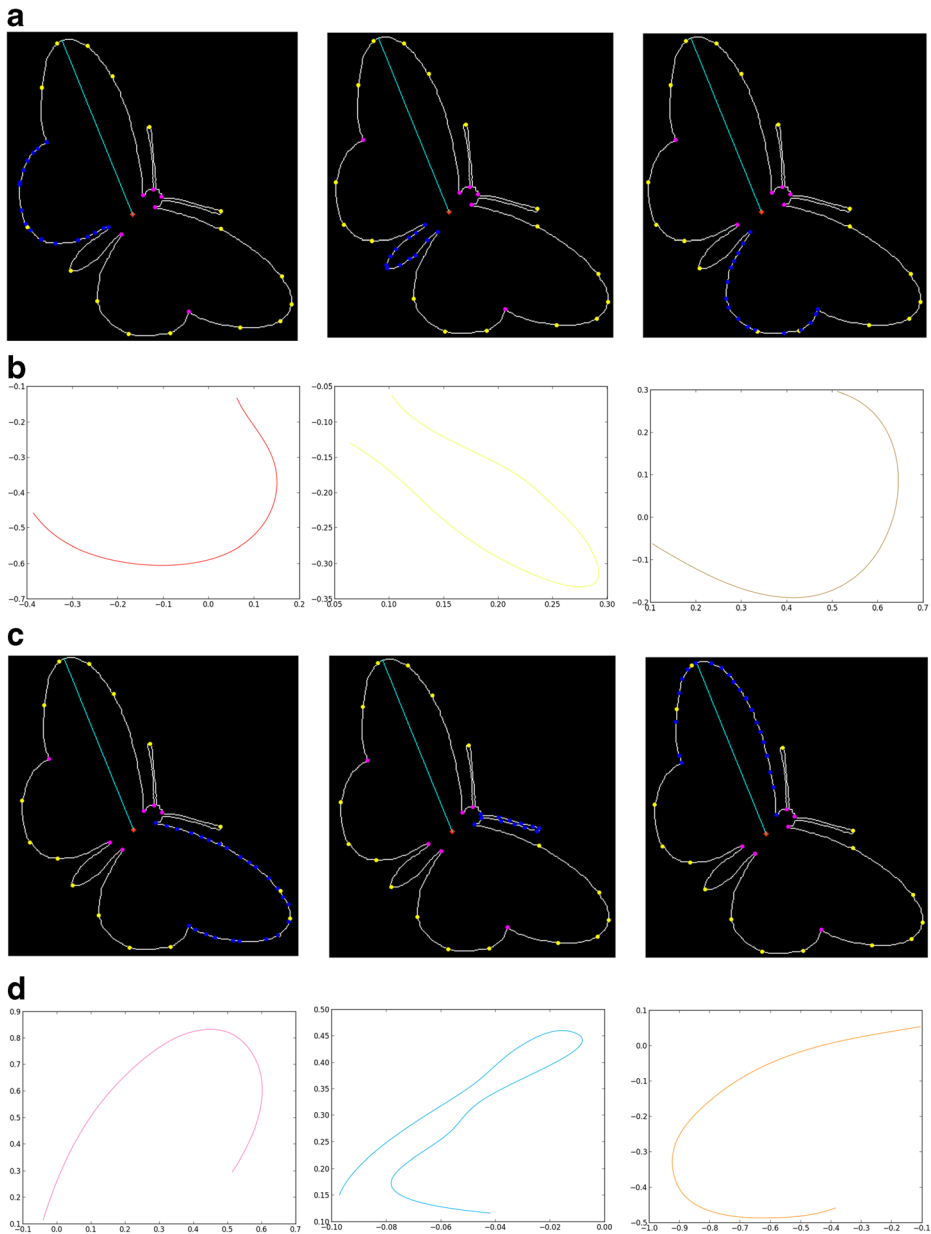
As our approach for shape matching and retrieval is based only on curves approximating the constituent parts of the shape. We propose to study if can we retrieve the shape using only a part of its outline. So to achieve this experiment, we performed some tests with incomplete shapes. Some examples are illustrated in Table 4 which shows the best retrieval results for the



**Fig. 16** Graphical representation of the control points (the middle row) and the associated B-splines curves at the bottom

partial queries with different degree of occlusion. From the obtained results, we can assert that our approach may recognize objects giving only a part of the outline shape as a query.

On the other hand, from the retrieval results presented in Table 4, one can see some recognition trials that are considered as failures. The irrelevant retrieved shapes are enclosed by rectangular boxes. The mismatches are mainly due to the property of the proposed



**Fig. 17** Representation of the B-splines curves with the proposed approach. **a** and **c** Represent the six first parts of the outline shape of the butterfly-20. **b** and **d** Represent their B-splines curves respectively

Query	Top ten results

Fig. 18 Retrieval results on five test queries shown in column 1

descriptor (normalized curves). Since the proposed approach is based on the different parts constituting the shape boundary and not the part relationships, it is possible for different shapes to share some similar parts. For example, the two last partial apples queries appear to be similar to tomato class shapes and cup. Observing the obtained results, this is not a negative aspect in favor of the proposed application, once our proposed descriptor deals only with the different parts constituting the boundary. However considering that the retrieval rates takes into account only the shapes belonging to the same class as the query shape, the performance is penalized.

### 5.1.3 Retrieval result

The ETH-80 database contains 80 shapes from eight classes with ten objects per class. To test the recognition rate, the leave-one-object-out method [45] is used. In comparison to previous results reported for the ETH-80 2D object database [27], our algorithm achieves 90,22% recognition rate for all shapes of our database. It is among the best proposed methods applied to this database and cited in [13, 45] as illustrated by Fig. 15.

## 5.2 MPEG-7 database

The well-known and widely used MPEG-7 CE-shape 1-Part B database [26] is used in our work to test the performance of the proposed approach. The total number of images in the database is 1400 consisting of 70 different classes of objects. Each class contains 20 shapes. In MPEG-7 database, there are many shapes where the inter-class object similarity is more than the intra-class object

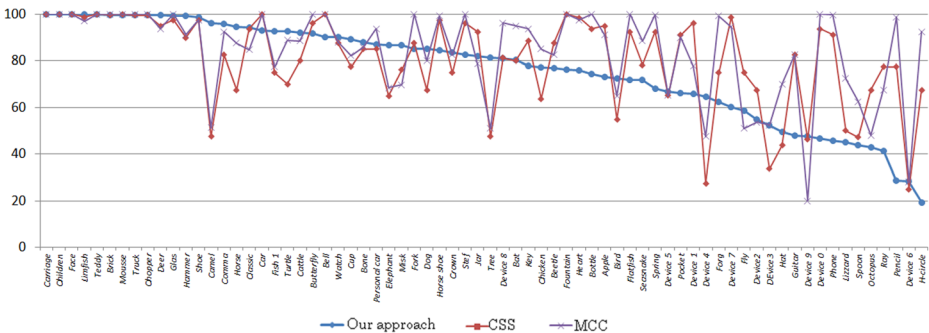




































Fig. 19 Retrieval rates for all classes from MPEG-7 database

Query	Methods											
	TSDIZ											
	BAS											
	Our approach											

**Fig. 20** Retrieval results for object bat-4. Comparative study for TSDIZ, BAS and the proposed approach

similarity. The retrieval rate is measured by the Bull’s-eye test [29, 45]: Each shape in the database is matched with all other shapes and the top forty best matched shapes according to the proposed matching process are retrieved. Among the forty best matched, the number of similar shapes which belong to the same class of the query are counted. Then the retrieval rate for the whole database is reported as a percentage of the maximal possible number of correct retrievals i.e. 28,000.

5.2.1 B-splines curves
































































– Control points

As presented in section 3.3, the choice of the control points has a closely impact on the accuracy of the curves. Figure 16 illustrates some examples of B-splines curves for the 6th part of the butterfly-20 shape. These curves are presented with control points located using the proposed representation and when are chosen randomly. The red labels associated to control points chosen randomly and the blue ones represent the selected control points given by our proposed representation.

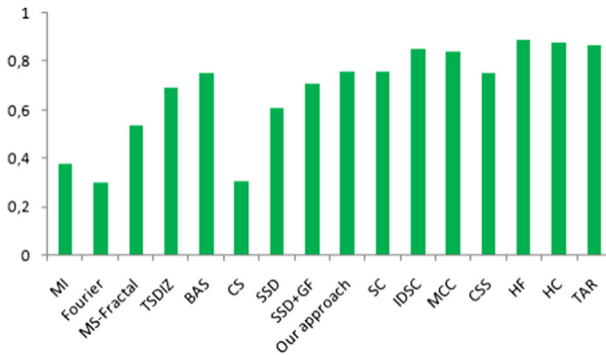
It can be seen from the Fig. 16 that the proposed technique for the selection of the control points gives a good accuracy of the B-spline curve.

– B-splines curves

Figure 17 shows some examples of B-splines representation for parts extracted from the butterfly-20 shape.

	SSD+GF											0
	Our method											0.4
	SSD+GF											0.2
	Our method											0.6
	SSD+GF											0.1
	Our method											0.9

**Fig. 21** Retrieval results on three queries (1st column) of two methods. Column 3–12 show the ten most similar shapes and column 13 gives the recognition rate over the top ten retrieved results



**Fig. 22** Comparison of the results of different approaches on the MPEG-7 (Bull's eye) test

### 5.2.2 Shape matching result

In this section, some examples of shape retrieval with the proposed approach are illustrated in Fig. 18. The queries are on the left column and in the second one, the best ten matches from the 1400 images of the database are listed. The obtained results show the invariance of affine transformations and deformations.

\Details of retrieval results for all classes of shapes from the MPEG-7 database are reported in Fig. 19. In the same figure we have reported the results obtained with both MCC [1] and CSS [33] approaches.

From the obtained results, we can observe that the most of classes can get very good retrieval results. The reason that the performs is low than 30% for some classes is that the description is based only on curves defined from shape, so some curves can be shared between shapes of different classes that are visually similar.










### 5.2.3 Comparative study

To further illustrate the performance of the proposed approach, a comparison with three algorithms: tensor scale descriptor with influence zones TSDIZ [3], beam angle statistic BAS [5] and the proposed method, is done. The shape bat-4 from MPEG-7 is selected as

**Table 5** Retrieval results on the Kimia-99 database for the proposed approach compared to other methods in the literature

Approaches	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	Total
Shape index [13]	43	51	58	52	52	49	51	47	45	44	492
ECCobj2D. h [19]	87	74	66	64	49	52	45	38	33	33	541
HC [13]	96	84	78	77	78	65	68	58	60	48	712
ECCobj2D [19]	94	85	81	73	81	73	64	59	56	35	701
Bernier and Landry [13]	97	94	92	85	74	73	65	54	43	36	713
Shape context [35]	97	91	88	85	84	77	75	66	56	37	756
Gen.model [13]	99	97	99	98	96	96	94	83	75	48	885
Proposed approach	98	99	98	97	97	95	94	92	88	66	924
Shock edit [13]	99	99	99	98	98	97	96	95	93	82	956
IDSC + DP [29]	99	99	99	98	98	97	97	98	94	79	958
IMC [48]	99	99	99	99	98	97	95	94	90	83	953
HF [45]	99	99	99	99	98	99	99	96	95	88	971

**Table 6** Retrieval rates per class for Kimia-99 database

Class									
Retrieval Rate	0.96	0.88	1	0.92	0.95	0.88	0.89	1	0.9

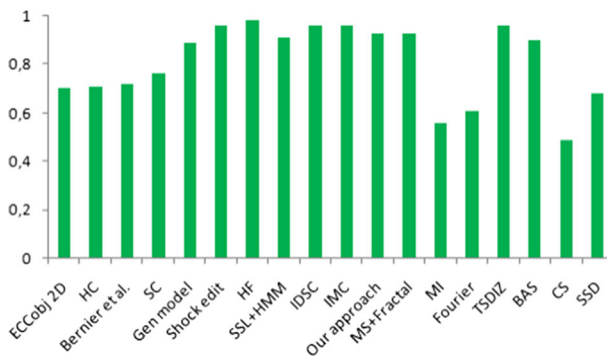
the query shape and the top ten retrieval results for the three algorithms are illustrated by the Fig. 20: first row, shape descriptor [3], followed by shape descriptor [5] and our approach. The shapes with a red line are the irrelevant retrieval results. It can see from the presented visual shape retrieval that there are only four similar shapes in the top ten retrieval results of BAS [5]. For TSDIZ [3] and our approach, the top ten results are exact.

To show the power of our boundary-based shape descriptor to preserve the geometric characteristics of shapes, we compare the retrieval results for three shapes selected from three different classes, based on SSD + GF descriptor [39] and our approach. We select the same queries as those cited in [39] and report the top ten retrieved shapes which are ranked by the proposed similarity measure. The obtained results are illustrated in Fig. 21.

The last column reports the rate of the retrieval result. From Fig. 21, we can see that the shape representation proposed here outperforms the method SSD + GF [39] and gives higher scores. The obtained results prove the efficacy and the competitiveness of our approach. This is due to the curve description based on the high curvature points which take into account the geometric description of the discriminate parts constituting the outline shape.

A summary of the obtained shape retrieval results on MPEG-7 database is reported in Fig. 22, using the bulls-eye score. In addition, a comparison with some known algorithms in the literature is made.

It shows from the Fig. 22 that the proposed approach performs competitively to SSD, TSDIZ, SSD + GF, SC, CSS and BAS algorithms in the literature and outperforms CS, MS-Fractal, MI, Fourier. The particularity of our work is that the geometric model proposed here gives a rough description about the shape as we can see by the examples of Fig. 21. It captures geometric information of the shape's boundary. Another observation is that the use of curves as match primitive preserves the geometry of the shape TSDIZ and allows robust retrieval against affine transformations and deformations as shown by the examples of Fig. 18.



**Fig. 23** Performance results on Kimia-99

**Table 7** Retrieval results on Kimia-216 database

Approaches	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	Total
Shape context [35]	214	209	205	197	191	178	161	144	131	101	78	1809
CPDH + EMD [48]	214	215	209	204	200	193	187	180	168	146	114	2030
IMC [48]	216	216	214	210	207	207	201	194	188	182	163	1991
Proposed approach	215	214	212	211	207	205	195	190	184	174	153	2160

### 5.3 Kimia-99 database

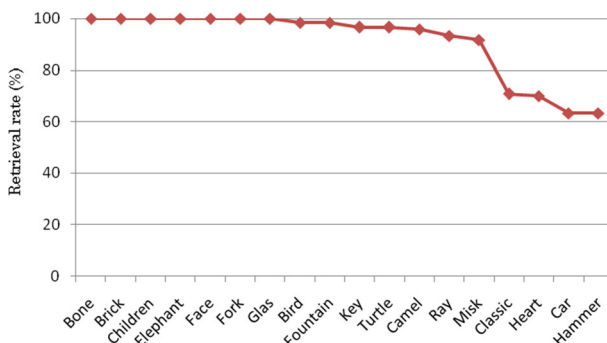
The proposed approach is also experimented with Kimia-99 database [42] which contains 99 shapes classified into 9 classes with 11 shapes in each class. Each shape of the whole database is selected as a query shape and then matched to the rest shapes. The ten most similar, excluding the query, are returned and only the correct results (results of the same class as a query) are counted. The details of this experiment are shown in Table 5 where the total number of top 1 to 10 closet matches are presented and compared to some relevant works in the literature. The reported results demonstrate that the proposed approach achieves satisfactory performance.

Table 5 shows also the ability of the proposed description to deal well with occluded shapes, articulated and shapes with deformations of parts, once most of the shapes of this database are under different levels of deformations, occlusion and articulation. Details of the obtained results for each classe of Kimia-99 database are presented in Table 6.

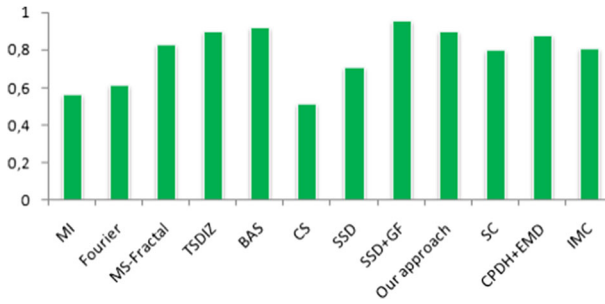
Figure 23 reports the retrieval performance of our approach and the corresponding retrievals for the above algorithms, including the methods IMC [48] and SSL + HMM [43]. As can be seen our approach has comparable results relatively to some well known algorithms. Fig. 23 shows also that the algorithms [13, 29, 48] present 0.03% better performance than our approach and HF [45] gives the best value.

### 5.4 Kimia-216 database

Kimia-216 database [42] is another widely used benchmark database for evaluating shape retrieval. It consists of 18 shape classes with 12 shapes in each class. In the experiment, each shape is considered as a query and the retrieval performance is evaluated using two measures of accuracy: the bull’s eyes score which counts the number of correct matches in the top 11 matches, excluding the query shape, and the precision at 10 measures. In Table 7, the number



**Fig. 24** Retrieval rate for all classes from Kimia-216



**Fig. 25** Retrieval rates on Kimia-216 database at 10 measures

of correct matches in each rank is summarized and compared to other methods. Note that the maximum number of correct retrievals in each case is 216. Figure 24 shows the average retrieval rates for each class. The overall retrieval rate for the proposed approach using the first measure is 0.89. Considering the precision at 10 measures, a retrieval rate of 0.90 is achieved. These results demonstrate the effectiveness of the proposed approach.

Figure 25 shows the reported results of our approach and many recent approaches on this database. Our approach performs comparably to the best known methods SSD + GF [39] and Bas [5].

## 6 Conclusion

In this paper, we have presented firstly a method for shape description based on cubic normalized curves. Outline shape is decomposed into a set of curves using points of high curvature. A cubic B-spline curve is then used for curve representation instead of a higher degree because it has a local control property and is less wiggly. The choice of the control points for curve approximation has been discussed and presented. Normalizing curve description using the shape's centroid guarantees the invariance of the description to scale change and invariance to rotation is ensured by the use of the minimum rectangle encompassing the shape.

Similarity measure between two shapes is proposed using Chebyshev distance. Experimental results on various databases of shapes show that the proposed descriptor is effective and has good retrieval accuracy. Using cubic curves as the description primitive has growing interest in preserving geometric properties of the shape. Consequently, the proposed approach has the ability to retrieve shapes even if we use them with occlusion or using only parts of them.

## References

1. Adamek T, O'Connor NE (2004) A multiscale representation method for nonrigid shapes with a single closed contour. *IEEE Trans Circuits Syst Video Technol* 14(5):742–754
2. Alajlan N (2010) Fast shape matching and retrieval based on approximate dynamic space warping. *Artificial Life and Robotics Journal* 15:309–315
3. Andal FA, Miranda PAV, Torres RD, Falco AX (2010) Shape feature extraction and description based on tensor scale. *Pattern Recogn* 43:26–36
4. Argawal S, Awan A, Roth D (2004) Learning to detect objects in images via a sparse part-base representation. *IEEE TPAMI* 26(11):1475–1490
5. Arica N, Vural FTY (2003) BAS: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recogn Lett* 24(9–10):1627–1639

6. Bai X, Yang X, Latecki LJ, Liu W, Tu Z (2010) Learning context-sensitive shape similarity by graph transduction. *IEEE Trans Pattern Anal Mach Intell*, PAMI 32(5):861–874
7. Carmona-poyato A, Madrid-Cuevas FJ, Medina-Camicer R, Munoz-Salinas R (2010) Polygonal approximation of digital planar curves through break point suppression. *Pattern Recogn* 43(1):209–216
8. Chen X, Ma W, Paul J (2010) Cubic B-spline curve approximation by curve unclamping. *Comput Aided Des* 42(6):523–534
9. Chetvericov D (2003) A Simple and efficient algorithm for detection of high curvature points in planar curves. In: 10th International Conference, CAIP, pp 1501–1504
10. Cohen FS, Huang Z, Yang Z (1995) Invariant matching and identification of curves using B- spline curve representation. *IEEE Trans Image Process* 4(1):1–10
11. Daliri MR, Torre V (2010) Classification of silhouettes using contour fragments. *Comput Vis Image Underst* 113(9):1017–1025
12. Direkoglu C, Nixon MS (2011) Shape classification via image-based multiscale description. *Pattern Recogn* 44(9):2134–2146
13. Ebrahim Y, Ahmed M, Abdelsalam W, Chau SC (2009) Shape representation and description using the Hilbert curve. *Pattern Recogn Lett* 30:348–358
14. El-Rube I, Kamel M, Ahmed M (2004) Wavelet-based affine invariant shape matching and classification. *ICIP Internat Conf on Image Processing*, Singapore
15. Graham RL (1972) An efficient algorithm for determining the convex hull of a finite planar set. *Inf Process Lett* 1:132–133
16. Gu Y, Tjahjadi T (2000) Coarse-to-fine planar object identification using invariant curve features and B-spline modeling. *Pattern Recogn* 33(9):1411–1422
17. Hayward WG (1998) Effects of outline shape in object recognition. *J Exp Psychol Hum Percept Perform* 24: 427–440
18. Hu RX, Jia W, Zhang D, Gui J, Song LT (2012) Hand shape recognition based on coherent distance shape contexts. *Pattern Recogn* 45:3348–3359
19. Ion A, Artner NM, Peyré G, Kropatsch WG, Cohen LD (2011) Matching 2D and 3D articulated shapes using the eccentricity transform. *Comput Vis Image Underst* 115:817–834
20. Juhasz I, Roth A (2013) A class of generalized B-spline curves. *Comput Aided Geom Des* 30:85–115
21. Klette R, Zunic J (2012) Distance between shape centroids versus shape diameter. *Comput Vis Image Underst* 116:690–697
22. Krishnamoorthy R, Sathiya Devi S (2013) Image retrieval using edge based shape similarity with multiresolution enhanced orthogonal polynomials model. *Digital Signal Process* 23:555–568
23. Laiche N, Larabi S (2011) Retrieval of 2D objects and shape matching using the B-spline representation. In: *IEEE International Conference on Signal and Image Processing Applications, ICSIPA*, pp 495–500
24. Laiche N, Larabi S (2014) 2D shape matching based on B-spline curves and dynamic programming. In: *VISAPP 2014, 9th International Conference on Computer Vision Theory and Applications*, pp 484–491
25. Latecki LJ, Lakamper R (2000) Shape similarity measure based on correspondence of visual parts. *IEEE TPAMI* 22(10):1182–1190
26. Latecki LJ, Lakamper R, Eckhardt U (2000) Shape descriptors for non-rigid shapes with a single closed contour. In: *Computer Vision and Pattern Recognition CVPR 2000*, pp 1424–1429
27. Leibe B, Scheille B (2003) Analyzing appearance and contour based methods for object categorization. *International Conference on Computer Vision and Pattern Recognition, Madison*
28. Li Z, Kuang Z, Liu Y, Wang J (2016) Multiscale shape context and re-ranking for deformable shape retrieval. *Comput Graph* 54:8–17
29. Ling H, Jacobs DW (2007) Shape classification using the inner-distance. *IEEE Trans Pattern Anal Mach Intell* 29(2):286–299
30. Liu YK, Wei W, Wang PJ, Zalik B (2008) Compressed vertex chain codes. *Pattern Recogn* 40(11):2908–2913
31. Lu Y, Lien J-M, Ghosh M, Amato NM (2012)  $\alpha$ -decomposition of polygons. *Comput Graph* 36:466–476
32. Ma Z, Zhang Z, Yang L (2011) Shape feature descriptor using modified Zernike moments. *Pattern Anal Applic* 14:9–22
33. Mokhtarian F, Abbasi S, Kittler J (1996) Efficient and robust retrieval by shape content through curvature scale space. In: *International Workshop on Image Databases and Multimedia Search*, pp 35–42
34. Mongkolnam P, Nukoolkit C, Dechsakulthorn T (2007) Represent image contents using curves and chain code. In: *IAPR Conference on Machine and Vision Applications, MVA*, pp 355–358
35. Mori G, Belongie S, Malik J (2004) Efficient shape matching using shape contexts. *IEEE Trans Pattern Anal Mach Intell* 27(11):1832–1837
36. Nanni L, Braham S, Lumini A (2012) Local phase quantization descriptor for improving shape retrieval/classification. *Pattern Recogn Lett* 33(16):2254–2260

37. Niblack PB, Gibbons CW, Capson DW (1992) Generating skeletons and centerlines from the distance transform. *CVGIP, Graphical Models and Image Processing* 54(5):420–437
38. Park H, Lee J-H (2007) B-spline curve fitting based on adaptive curve refinement using dominant points. *Comput Aided Des* 39:439–451
39. Pedrosa GV, Batista MA, Barcelos CAZ (2013) Image feature descriptor based on shape salience points. *Neurocomputing* 120:156–163
40. Piegl L, Tiller W (1997) *The NURBS book*. Springer, Berlin
41. Rivlin E, Weiss I (1995) Local invariants for recognition. *IEEE Trans Pattern Anal Mach Intell* 17(3):226–238
42. Sebastian TB, Klein PN, Kimia BB (2004) Recognition of shapes by editing their shock graphs. *IEEE Trans Pattern Anal Mach Intell* 26:550–571
43. Shi Y, Wang G, Wang R, Zhu A (2013) Contour descriptor based on space symmetry. *Optik* 124:6149–6153
44. Wang W, Pottmann H, Liu Y (2006) Y, fitting B-spline curves to point clouds by curvature-based squared distance minimization. *ACM Trans Graph* 25(2):214–238
45. Wang J, Bai X, You X, Liu W, Latecki LJ (2012) Shape matching and classification using height functions. *Pattern Recogn Lett* 33:134–143
46. Yang H, Wang W, Sun J (2004) Control point adjustment for B-spline curve approximation. *Comput Aided Des* 36:639–652
47. Yang GY, Shu HZ, Toumoulin C, Han GN, Luo LM (2006) Efficient Legendre moments computation for grey level images. *Pattern Recogn* 39(1):74–80
48. Yang J, Wang H, Yuan J, Li Y, Liu J (2016) Invariant multi-scale descriptor for shape representation, matching and retrieval. *Comput Vis Image Underst* 145:43–58
49. Zaletelj J, Pecci R, Spaan F, Hanjalic A, Lagendijk RL (1998) Rate distortion optimal contour using cubic B-spline. *European Signal Processing Conference* 9:1501–1504
50. Zhang DS, Lu G (2002) Generic Fourier descriptor for shape-based image retrieval. In: *ICME: IEEE. International Conference on Multimedia and Expo*, pp 425–428
51. Zunic J, Aktas MA, Ortiz CM, Galton A (2011) The distance between shape centroids is less than a quarter of the shape perimeter. *Pattern Recogn* 44:2161–2169



**Nacéra Laiche** received the DES. Degree in mathematics from University of Houari Boumediene: USTHB, Algiers, Algeria in 1993, the Magister degree in Algebra from the same University in 1997. She got her Ph.D in Computer Science in 2014. She is currently a researcher at the Computer Science Department of USTHB. Her research interests include shape representation, content-based image retrieval, and computer vision.



**Slimane Larabi** received the Ph.D. in Computer Science from the Polytechnic National Institute of Toulouse, France, in 1991. He is currently Professor at the Computer Science Institute of University of Sciences and Technologies, Algiers, Algeria. He is the head of Computer Vision group at the laboratory of artificial Intelligence of the same university, and an author of numerous publications for conferences, proceedings, and journals.