Retrieval of Partially Visible Shapes through Structural Feature Indexing

Hirobumi Nishida

Ricoh Software Research Center, 1-1-17 Koishikawa, Bunkyo-ku, Tokyo 112-0002, Japan
hirobumi.nishida@nts.ricoh.co.jp

Abstract. Efficient and robust information retrieval from large image databases is an essential functionality for the reuse, manipulation, and editing of multimedia documents. Structural feature indexing is a potential approach to efficient shape retrieval from large image databases, but the indexing is sensitive to noise, scales of observation, and local shape deformations. It has now been confirmed that efficiency of classification and robustness against noise and local shape transformations can be improved through the feature indexing approach incorporating shape feature generation techniques. Based on this approach, an efficient, robust method is presented for retrieval of model shapes that have parts similar to the query shape presented to the image database. Effectiveness is confirmed through experimental trials with a large database of boundary contours, and is validated by systematically designed experiments with a large number of synthetic data.

1 Introduction

Information retrieval from large image databases is an essential functionality for the reuse, manipulation, and editing of multimedia documents. Images have some components in terms of representation, such as color, texture, and shape. Shape is an essential component, but shape analysis and representation are still difficult research subjects in spite of intensive research carried out for decades. In particular, shapes observed in natural scenes are often occluded, corrupted by noise, and partially visible. It is an important problem to retrieve efficiently model shapes that have parts similar to the query shape presented to the image database [1]. Shape retrieval from image databases has been studied recently for improving efficiency and robustness [2—4]. In particular, the problem is intractable when the shape is partially visible. Efficiency and robustness are important, but sometimes incompatible criteria for performance evaluation. The improvement of robustness implies that the scheme for classification and retrieval should tolerate certain types of variations and deformations for images. Obviously, it may lead to inefficiency if some brute-force methods are employed such as a generate-and-test strategy by generating various images with a number of difference parameters. A key to achieving both efficiency and robustness is through a compact and well-structured representation of images that tolerate variations and deformations. In particular, it has been confirmed that efficiency of classification and robustness against noise and local shape transformations can be improved at the same time by the feature indexing approach incorporating shape

F.J. Ferri et al. (Eds.): SSPR&SPR 2000, LNCS 1876, pp. 211-220, 2000.

feature generation techniques [5]. However, the application of this approach has been confined to sets of shapes represented as closed contours.

In this paper, based on the structural feature indexing with feature generation models, an efficient, robust method is presented for retrieval of model shapes that have parts similar to the query shape presented to the image database. This paper is organized as follows: In Section 2, a structural representation of curves by quasi-convex/concave features along with quantized-directional features [5] is outlined. In Section 3, based on the shape representation outlined in Section 2, we describe the shape signature, the model database organization through feature indexing, and the shape retrieval through voting. In Section 4, the transformation rules of shape signatures are introduced to generate features that can be extracted from deformed patterns caused by noise and local shape deformations. The proposed algorithm is summarized in Section 5. In Section 6, the proposed method is validated by systematically designed experiments with a large number of synthetic data. Section 7 concludes this paper.

2 Shape Representation

The structural representation of curves [5] is outlined in this section, based on quasi-convex/concave structures incorporating $2N$ quantized-directional features (N is a natural number). As shown in Fig. 1a, the curve is first approximated by a series of line segments. On a 2-D plane, we introduce N-axes together with $2N$ quantized-direction codes. For instance, when $N = 4$, eight quantized-directions are defined along with the four axes as shown in Fig. 1b. Based on these N-axes together with $2N$ quantized-direction codes, the analysis is carried out hierarchically.

A curve is decomposed into *sub-segments* at extremal points along each of the N-axes. Fig. 1c illustrates the decomposition of a contour shown in Fig. 1a into sub-segments when $N = 4$. For adjacent sub-segments *a* and *b*, suppose that we turn counterclockwise when traversing them from *a* to *b*, and the joint of *a* and *b* is an extremal point along the axes toward the directions $(j, j+1(\mathrm{mod}2N), \ldots, k)$. Then, we write the concatenation of these two sub-segments as $a \xrightarrow{j,k} b$. For instance, the

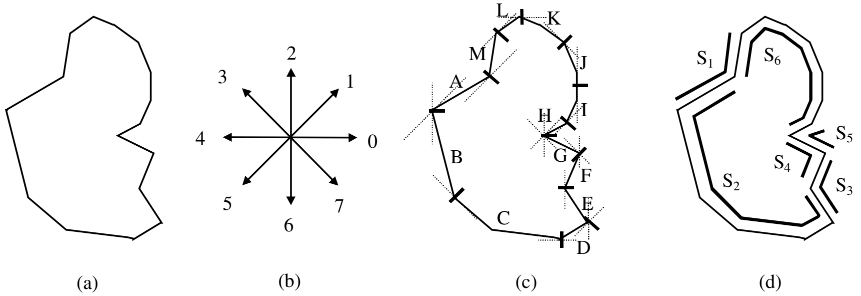

Fig. 1. (a) A closed contour with a polygonal approximation, (b) quantized-directional codes when $N = 4$, (c) sub-segments when $N = 4$, (d) segments when $N = 4$.

joint of sub-segments *H* and *G* in Fig. 1c is an extremal point along the three axes toward the directions 3, 4, and 5. Therefore, the concatenation of *H* and *G* is written as $H \xrightarrow{3,5} G$. In this way, we obtain the following concatenations for the sub-segments illustrated in Fig 1c.

$$L \xrightarrow{3,3} M,\quad K \xrightarrow{2,2} L,\quad J \xrightarrow{1,1} K,\quad I \xrightarrow{0,0} J,$$

$$H \xrightarrow{7,7} I,\quad H \xrightarrow{3,5} G,\quad F \xrightarrow{0,1} G,\quad F \xrightarrow{4,4} E,$$

$$D \xrightarrow{7,0} E,\quad C \xrightarrow{6,6} D,\quad B \xrightarrow{5,5} C,\quad A \xrightarrow{3,4} B,\quad A \xrightarrow{7,7} M$$

By linking local features around joints of adjacent sub-segments, some sequences of the following form can be constructed:

$$a_0 \xrightarrow{j(1,0),\,j(1,1)} a_1 \xrightarrow{j(2,0),\,j(2,1)} \cdots \xrightarrow{j(n,0),\,j(n,1)} a_n \qquad \textbf{(1)}$$

A part of the contour corresponding to a sequence of this form is called a *segment*. Furthermore, the starting point of the segment is defined as the end point of a_0, and the ending point is as the end point of a_n. When a segment is traversed from its starting point to its ending point, one turns counterclockwise around any joints of sub-segments. The following segments, as shown in Fig. 1d, are generated from the 13 sub-segments shown in Fig. 1c:

$$S_1 : A \xrightarrow{7,7} M,\quad S_2 : A \xrightarrow{3,4} B \xrightarrow{5,5} C \xrightarrow{6,6} D \xrightarrow{7,0} E,$$

$$S_3 : F \xrightarrow{4,4} E,\quad S_4 : F \xrightarrow{0,1} G,\quad S_5 : H \xrightarrow{3,5} G,$$

$$S_6 : H \xrightarrow{7,7} I \xrightarrow{0,0} J \xrightarrow{1,1} K \xrightarrow{2,2} L \xrightarrow{3,3} M.$$

A segment is characterized by a pair of integers $\langle r,d \rangle$, *characteristic numbers*, representing the angular span of the segment and the direction of the first sub-segment:

$$r = \sum_{i=1}^{n} \big(j(i,1) - j(i,0)\big)_{\mathrm{mod}\,2N} + \sum_{i=1}^{n-1} \big(j(i+1,0) - j(i,1)\big)_{\mathrm{mod}\,2N} + 2, \quad d = j(1,0) \qquad \textbf{(2)}$$

The characteristic numbers are given by $\langle 2,7 \rangle$, $\langle 7,3 \rangle$, $\langle 2,4 \rangle$, $\langle 3,0 \rangle$, $\langle 4,3 \rangle$, and $\langle 6,7 \rangle$, respectively, for the six segments shown in Fig. 1d.

Adjacent segments are connected by sharing the first sub-segments or last ones of the corresponding sequences. These two types of connection are denoted by $S \overset{h}{-} T$ and $S \overset{t}{-} T$, respectively, for two adjacent segments *S* and *T*. For instance, connections are denoted by $S_1 \overset{h}{-} S_2 \overset{t}{-} S_3 \overset{h}{-} S_4 \overset{t}{-} S_5 \overset{h}{-} S_6 \overset{t}{-} S_1$ for the six segments shown in Fig. 1d.

3 Shape Signature, Indexing, and Voting

Based on the shape representation outlined in Section 2, we describe the shape signature, the model database organization through feature indexing, and the shape retrieval through voting. For the model database organization, we assume that each model shape is presented to the system as line drawings or boundary contours of objects. In the shape retrieval, we assume that line drawings or parts of some model shape can be given as a query to the system.

3.1 Shape Signatures

In order to retrieve images for a query given as a partially visible shape, the shape signature is required to tolerate rotation, scaling, and translation. Therefore, features depending on orientation, size, and location cannot be employed as shape signatures. Based on the characteristic numbers and connections of segments extracted from model shapes or query shapes, the shape signature is constructed to satisfy this requirement. We assume that a series of n segments S_i $(i = 1,2,\ldots,n)$ have been extracted with characteristic numbers $\langle r_i, d_i \rangle$ and lengths l_i. The angular span r_i does not depend on orientation, size, or location. Furthermore, the lack of information due to dropping orientation, size, and location can be compensated by employing a triplet of the angular spans of two consecutive segments and their length ratio as the shape signature. From two consecutive segments S_i and S_{i+1} connected as $S_i \overset{c}{-} S_{i+1}$, $c \in \{h, t\}$, the shape signature is constructed as follows:

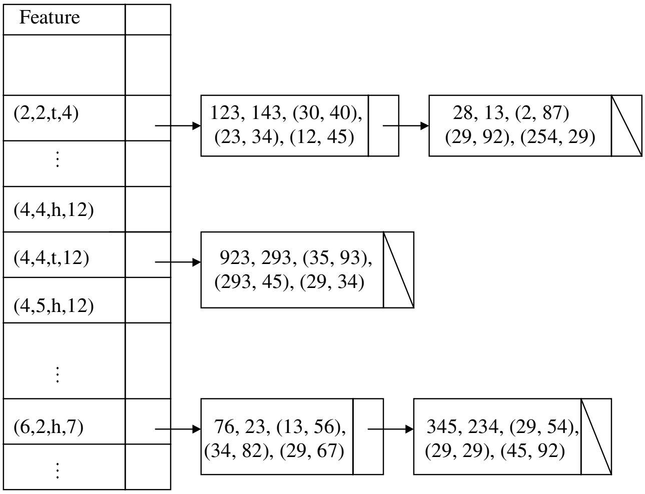

Fig. 2. Model database organization by structural indexing. Each table item stores a list whose element is composed of the model identifier, length, location of the center of gravity, and locations of the two end points of the curve segment.

$$\left(r_i, r_{i+1}, c, \left\lfloor Q \cdot \frac{l_{i+1}}{l_i + l_{i+1}} \right\rfloor\right),$$

where Q is the number of quantization levels for length-ratio parameters.

3.2 Indexing

From each model shape, shape signatures are extracted from all pairs of consecutive segments. A large table, as illustrated in Fig. 2, is constructed for a model set by assigning a table address to a shape signature and storing there a list whose item is composed of the following elements: the model identifier that has the corresponding shape signature, and shape parameters of the curve segment corresponding to the shape signature, namely length, location of the center of gravity, and locations of the two endpoints, computed on the model shape.

3.3 Voting

Classification of the query shape is carried out by voting for the transformation space associated with each model. For each model, voting boxes are prepared for the quantized transformation space $(\sigma, \theta, x_T, y_T)$, where σ is the scaling factor, θ is the rotation angle, and (x_T, y_T) is the translation vector. Shape signatures are extracted from the curve segment given as a query to the shape database. For each extracted shape signature, model identifiers and shape parameters are retrieved from the table by computing the table address. By comparing the shape parameters of the extracted shape signature with the registered parameters, the transformation parameters $(\sigma, \theta, x_T, y_T)$ can be computed for each model and the voting box corresponding to the transformation parameters associated with the model is incremented by one. In the implementation, transformation parameters σ and θ are computed from the line segment connecting the two endpoints, and (x_T, y_T) is computed from the location of the center of gravity.

4 Feature Generation Models

Shape signatures extracted from the curve are sensitive to noise and local shape deformations, and therefore, the correct model does not necessarily receive many votes as expected for the ideal case. Furthermore, when only one sample pattern is available for each class, techniques of statistical or inductive learning from training data cannot be employed for obtaining *a priori* knowledge and feature distributions of deformed patterns. To cope with these problems, we analyze the feature transformations caused by some particular types of shape deformations, constructing feature transformation rules. Based on the rules, we generate segment features that can be extracted from deformed patterns caused by noise and local shape deformations. In both processes of model database organization and classification, the generated features by the transformation rules are used for structural indexing and voting, as well as the features actually extracted from curves.

The following two types of feature transformations are considered in this work:

- Change of convex/concave structures caused by perturbations along normal directions on the curve and scales of observation, along with transformations of characteristic numbers (the angular span of the segment and the direction of the first sub-segment).
- Transformations of characteristic numbers caused by small rotations.

We describe these two types of transformation in the rest of this section.

4.1 Transformations of Convex/Concave Structures

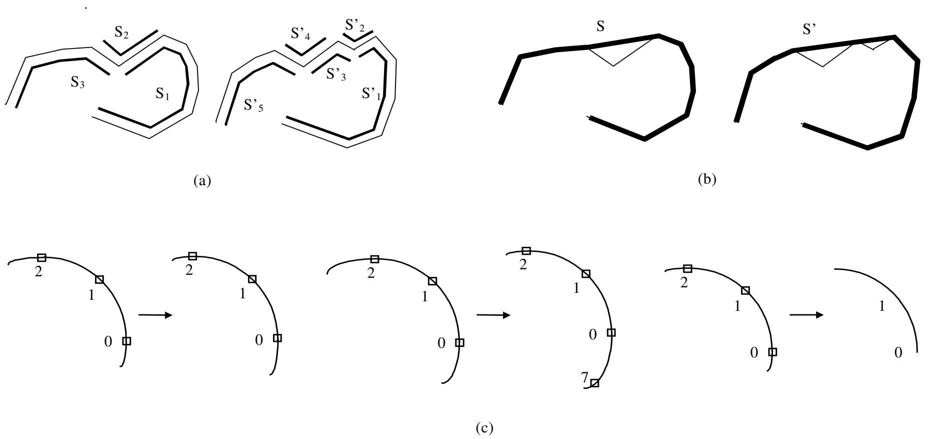

Fig. 3. (a) Part of curves similar to one another in terms of global scales, (b) editing structural features by merging segment blocks, (c) transformations of characteristic numbers of segments by small rotations.

The convex/concave structures along the curve are changed by noise and local deformations, and also depend on scales of observations. For instance, two parts of curves shown in Fig. 3a are similar to one another in terms of global scales, but their structural features are different. When $N = 4$, the curve shown on left is composed of three segments connected as $S_1 \overset{t}{-} S_2 \overset{h}{-} S_3$ with characteristic numbers $\langle 6,6 \rangle$, $\langle 2,6 \rangle$, and $\langle 3,2 \rangle$, whereas the one shown on right is composed of five segments connected as $S'_1 \overset{t}{-} S'_2 \overset{h}{-} S'_3 \overset{t}{-} S'_4 \overset{h}{-} S'_5$ with characteristic numbers $\langle 6,6 \rangle$, $\langle 2,6 \rangle$, $\langle 2,2 \rangle$, $\langle 2,6 \rangle$, and $\langle 3,2 \rangle$. To cope with such deformations, structural features on the two curves are edited so that their features can become similar to one another. For instance, the structural features illustrated in Fig. 3a can be edited by merging the two segment blocks $\{S_1, S_2, S_3\}$ and $\{S'_1, S'_2, S'_3, S'_4, S'_5\}$ to two segments S and S' as shown Fig. 3b. In the structural indexing and voting processes, for an integer M specifying the maximum number of segments to be merged, shape signatures are generated by applying RULE 1 described below to segment blocks.

Rule 1: Let $\langle r_i, d_i \rangle$ be the characteristic number of the segment s_i, and $\lambda(s_j s_{j+1} \cdots s_{j+k-1})$ be length of the curve composed of k consecutive segments $s_j s_{j+1} \cdots s_{j+k-1}$. From a segment block

$$\Sigma(s_j, m, n) = \left\{ s_{j+k} \,\middle|\, k = 0,1,\ldots,m,\ldots,m+n-1;\, s_j \overset{C_j}{-} s_{j+1} \cdots s_{j+m-1} \overset{*}{-} s_{j+m} \cdots s_{j+m+n-1} \right\}$$

where m and n are odd such that $1 \le m, n \le M$, a shape signature

$$\left(\sum_{k=0}^{m-1} (-1)^k r_{j+k}, \sum_{k=0}^{n-1} (-1)^k r_{j+m+k}, C_j, \left\lfloor Q \cdot \frac{\lambda(s_j s_{j+1} \cdots s_{j+m-1})}{\lambda(s_j s_{j+1} \cdots s_{j+m-1} s_{j+m} \cdots s_{j+m+n-1})} \right\rfloor \right),$$

is generated if $\sum_{k=0}^{m-1} (-1)^k r_{j+k} \ge 2$, $\sum_{k=0}^{n-1} (-1)^k r_{j+m+k} \ge 2$, $r_{j+2k-2} - r_{j+2k-1} + r_{j+2k} \ge 2$ for $k = 1,\ldots,(m-1)/2$, and $r_{j+m+2k-2} - r_{j+m+2k-1} + r_{j+m+2k} \ge 2$ for $k = 1,\ldots,(n-1)/2$.

For instance, when $N = 4$ and $M = 3$, from the six segments illustrated in Fig. 1d with characteristic numbers $\langle 2,7 \rangle$, $\langle 7,3 \rangle$, $\langle 2,4 \rangle$, $\langle 3,0 \rangle$, $\langle 4,3 \rangle$, and $\langle 6,7 \rangle$, the following shape signatures (length-ratio omitted) are generated by *Rule 1*: (2, 7, h), (2, 8, h), (7, 2, t), (7, 3, t), (8, 4, t), (2, 3, h), (2, 5, h), (3, 6, h), (3, 11, h), (3, 4, t), (5, 2, t), (4, 6, h), (4, 11, h), (6, 2, t), (11, 2, t), (11, 3, t). In total, at most $n \cdot \lceil M/2 \rceil$ shape signatures are generated from n segments.

4.2 Transformations of Characteristic Numbers by Small Rotations

The characteristic number $\langle r, d \rangle$ ($r \ge 2$) can be transformed by rotating the shape. Rules can be introduced for generating characteristic numbers by rotating the shape slightly (see Fig. 4c).

Rule 2: When the curve composed of the two consecutive segments S_1 and S_2 with characteristic numbers $\langle r_1, d_1 \rangle$ and $\langle r_2, d_2 \rangle$ is rotated by angle ψ $(-\pi/N \le \psi \le \pi/N)$, the angular spans r_1 and r_2 can be transformed into one of the 9 cases: (1) (r_1, r_2), (2) $(r_1, r_2 - 1)$, (3) $(r_1, r_2 + 1)$, (4) $(r_1 - 1, r_2)$, (5) $(r_1 - 1, r_2 - 1)$, (6) $(r_1 - 1, r_2 + 1)$, (7) $(r_1 + 1, r_2)$, (8) $(r_1 + 1, r_2 - 1)$, (9) $(r_1 + 1, r_2 + 1)$. Note that the cases (4—6) are applicable only if $r_1 \ge 3$, and that the cases (2), (5), and (8) are applicable only if $r_2 \ge 3$.

For instance, when $N = 4$ and $M = 3$, the 16 shape signatures have been generated by *Rule 1* from the six segments illustrated in Fig. 1d. Then, by applying *Rule 2* to these generated ones, 120 shape signatures, in total, are further generated.

5 Algorithm

In the model database organization step by structural indexing, shape signatures are generated from each model shape by *Rules 1* and *2*, and the model identifier with the shape parameters is appended to the list stored at the table address corresponding to each generated shape signature. For each model i ($i = 1,2,\ldots,n$), let c_i be the number of shape signatures generated by *Rules 1* and *2*. For instance, $c_i = 120$ for the contour shown in Fig. 1a when $N = 4$ and $M = 3$. In the classification and retrieval by voting for models and the transformation space, from shape signatures extracted from the query shape, shape signatures are generated by *Rules 1* and *2*. Model identifier lists are retrieved from the tables by using the addresses computed from the generated shape signatures, and the transformation parameters are computed for each model on the lists. The voting box is incremented by one for the model and the computed transformation parameters. Let v_i ($i = 1,2,\ldots,n$) be the maximum votes among the voting boxes associated with the model i. The query shape is classified by selecting out some models according to the descendant order of v_i / c_i. Examples of shape retrieval are given in Fig. 4, where query shapes are presented at top along with retrieved model shapes.

6 Experiments

In this section, the proposed algorithm is evaluated quantitatively in terms of the robustness against noise and shape deformations, based on the systematically designed, controlled experiments with a large number of synthetic data [5]. We examined the probability that the correct model is included in top *t%* choices for various values of the deformation parameter β [5] when curves composed of *r%* portions of a model shape are given as queries. For given values of *r* and β, a sub-contour of *r%* of length is randomly extracted from the model shape, and then, it is deformed by the deformation process as described in Nishida [5].

The main contribution of this work is to incorporate the shape feature generation into the structural indexing for coping with shape deformations and feature transformations. Therefore, the performance was compared with Stein-Medioni method [2] extracting features from several versions of piecewise linear approximations of the curve with a variety of error tolerances for approximations.

We carried out several experimental trials by changing the number of models from 200 to 500, examining the classification accuracy in terms of the deformed portions of model shapes given as queries to image databases. In the implementation of Stein-Medioni method, by changing the error tolerance with Ramer's method from 1% to 20%, with a step of 1%, of the widest side of the bounding box of the curve, twenty versions of approximations were created for each model shape and the query shape. Furthermore, the recommended values specified in [2] were used for some engineering parameters.

Table 1 presents the average classification rates for top 1%, 2%, 3%, 5%, and 10% choices when $\beta \in [0.0,0.5]$, $\beta \in [0.5,1.0]$, and $\beta \in [1.0,1.5]$. For instance, when a curve segment composed of 80% portions of a model shape subject to the deformation process with $\beta \in [0.5,1.0]$ is given as a query to shape databases of 500 models, the correct models are included in top 15 choices (3%) with probability 98.2% for proposed algorithm and with probability 83.9% for Stein-Medioni method. Clearly, significant improvements of robustness against noise and local shape deformations can be observed for the proposed algorithm in terms of classification accuracy without a significant degradation of efficiency. Through the experiments, the effectiveness has been verified through the experiments for the shape signature and the shape feature generation models.

7 Conclusion

Structural feature indexing is a potential approach to efficient shape retrieval from large image databases, but the indexing is sensitive to noise, scales of observation, and local shape deformations. It has now been confirmed that efficiency of classification and robustness against noise and local shape transformations can be improved at the same time by the feature indexing approach incorporating shape feature generation techniques [5]. In this paper, based on this approach, an efficient, robust method has been presented for retrieval of model shapes that have parts similar to the query shape presented to the image database. The effectiveness has been confirmed by experimental trials with a large database of boundary contours and has been validated by systematically designed experiments with a large number of synthetic data.

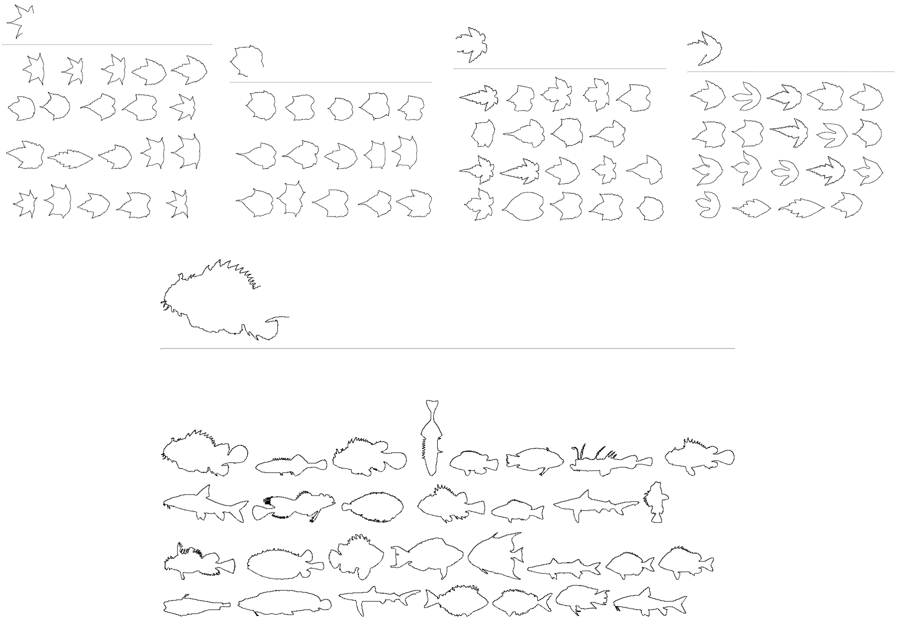

Fig. 4. Examples of shape retrieval.

References

1. R. Mehrotra and J.E. Gary, Similar-shape retrieval in shape data management, *Computer* **28**(9), 1995, 57—62.
2. F. Stein and G. Medioni, Structural indexing: efficient 2-D object recognition, *IEEE Trans. Pattern Analysis & Machine Intelligence* **14**(12), 1992, 1198—1204.
3. A. Del Bimbo and P. Pala, Image indexing using shape-based visual features, *Proc. 13th Int. Conf. Pattern Recognition*, *Vienna, August 1996*, vol. C, pp. 351—355.
4. F. Mokhtarian, S. Abbasi, and J. Kittler, Efficient and Robust Retrieval by Shape Content through Curvature Scale Space, *Proc. First International Workshop on Image Database and Multimedia Search, Amsterdam, August 1996*, pp. 35—42.
5. H. Nishida, Structural shape indexing with feature generation models, *Computer Vision and Image Understanding* **73**(1), 1999, 121—136.

Table 1. Average classification rates (%) of deformed patterns by the proposed algorithm in terms of the portion of model shapes (*r*%) presented as queries, in comparison with Stein-Medioni method.

β	Portion	Method	Classification rates (%) for top *t*% choices				
	(r)		*t* = 1	*t* = 2	*t* = 3	*t* = 5	*t* = 10
0.0— 0.5	100%	**Nishida**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
		Stein	92.8	96.8	98.3	98.8	99.7
	80%	**Nishida**	**99.6**	**99.7**	**99.7**	**99.8**	**99.9**
		Stein	86.3	92.3	94.8	97.3	98.9
	60%	**Nishida**	**92.4**	**95.0**	**95.9**	**96.7**	**98.0**
		Stein	61.6	73.1	79.0	84.7	91.6
0.5— 1.0	100%	**Nishida**	**99.8**	**99.8**	**99.8**	**99.9**	**99.9**
		Stein	80.6	87.4	89.9	92.8	96.5
	80%	**Nishida**	**96.0**	**97.8**	**98.2**	**99.1**	**99.5**
		Stein	69.5	79.8	83.9	88.1	93.1
	60%	**Nishida**	**75.1**	**82.7**	**85.7**	**89.0**	**93.4**
		Stein	39.5	51.2	58.3	66.8	78.1
1.0— 1.5	100%	**Nishida**	**91.1**	**94.3**	**96.2**	**97.3**	**98.2**
		Stein	55.1	65.8	71.0	76.2	84.2
	80%	**Nishida**	**68.4**	**75.9**	**79.8**	**84.9**	**90.1**
		Stein	40.7	52.0	57.7	66.0	76.5
	60%	**Nishida**	**36.6**	**46.0**	**51.7**	**59.0**	**68.5**
		Stein	17.5	27.3	33.3	41.3	55.5