

Structural Indexing of Line Pictures with Feature Generation Models

Hirobumi Nishida

Software Research Center, Ricoh Co., Ltd.
1-1-17 Koishikawa, Bunkyo-ku, Tokyo 112-0002, JAPAN
E-mail: hn@src.ricoh.co.jp

Abstract. Structural indexing is a potential approach to efficient classification and retrieval of image patterns with respect to a very large number of models. Essential problems caused by mapping image features to discrete indices are that the indexing is sensitive to noise, scales of observation, and local shape deformations, and that *a priori* knowledge or feature distributions of corrupted instances are not available for each class when a large number of training data are not presented. To cope with these problems, shape feature generation techniques are incorporated into structural indexing. The feature transformation rules obtained by an analysis of some particular types of shape deformations are exploited to generate features that can be extracted from deformed patterns. The generated features are used in model database organization and classification. Experimental trials with a large number of sample data show that the shape feature generation significantly improves the classification accuracy and efficiency.

1. Introduction

The needs for shape classification with a large number of models have become practical as pattern retrieval from image database has come up as an important application of image media technologies. In most of model-based methods for image pattern classification and recognition, independent data structures are considered for each model, but the efficiency of classification is drastically degraded when we deal with a large number of models, for instance, several hundreds or thousands. *Structural indexing* [1—4] is a potential approach to resolving this difficulty. This technique is based on the idea of distributing features associated with model identifiers over a large data structure prepared for a model set, along with classification by voting for models with reference to the extracted features. An example of such data structures is a large table where a shape feature has a correspondence to a table address and the table item stores a list of model identifiers with the particular shape feature corresponding to the table address. For complex shapes, the correct model is not necessarily selected as the top choice obtained by voting, and some other sophisticated algorithms need to be applied to the reduced sets of model candidates. Therefore, required properties for structural indexing are that the correct model is guaranteed to be included in the model candidate set composed of a few to 10 percent of the whole model set, and that the time complexity can be ignored compared with sophisticated algorithms. If these

requirements are satisfied, classification can be sped up by several tens times without degrading classification accuracy. Furthermore, these two properties are also essential in achieving adequate performance and efficiency for the application to retrieval from large image pattern databases.

We often encounter a situation where *only one sample pattern* is available for *each class* in image pattern database applications. Instance patterns are subject to noises and local shape deformations, and therefore, we need an efficient and effective way for searching a model pattern for the inputted pattern without *a priori* knowledge and parameter distributions of corrupted instances. In particular, structural features are, in general, are sensitive to noises and local shape deformations, and their transformations are intractable and often catastrophic in that the original, global structures are not preserved. We now need to address the following essential problems of the structural indexing approach: (1) How can the transformations or distributions of features be modeled when only one sample pattern is available for each class? (2) How can the robustness of structural indexing be improved against noises and local shape deformations? These problems need to be addressed adequately so that the structural indexing approach can be useful and effective in practice.

To address these problems, Stein and Medioni [2] extract structural features from several versions of polygonal approximations of the shape contour with a variety of error tolerances for approximations. Furthermore, Del Bimbo and Pala [4] proposed a method integrating the scale-space method into the structural indexing approach, constructing hierarchical data structures taking noises and local deformations into consideration based on structural feature descriptions from each scale. However, there are some technical questions about the computational complexity and stability of this approach, because the curves are smoothed with a variety of scales and correspondences of inflection points need to be found among difference scales. Therefore, there have been few effective, efficient approaches to the structural indexing that adequately address the above-mentioned problems.

In this paper, an efficient, effective algorithm along with data structures is presented for indexing and classification of closed contours based on structural features in terms of convex/concave parts and quantized directional features along contours. In particular, shape feature generation techniques are incorporated into structural indexing to improve the accuracy and robustness of shape classification against noises and local shape transformations. An analysis of the feature transformations is carried out for some particular types of shape deformations, leading to feature generation rules composed of a small number of distinct cases. The rules are exploited to generate features that can be extracted from deformed patterns caused by noises 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 contours. The effectiveness of the proposed method is demonstrated by experimental trials with a large number of sample data. Contours contain all information about objects and patterns, and therefore, there should be a number of applications ranging widely: the recognition of isolated objects and symbol/character patterns, the retrieval from very large databases of image patterns, and the selection of models and image transforma-

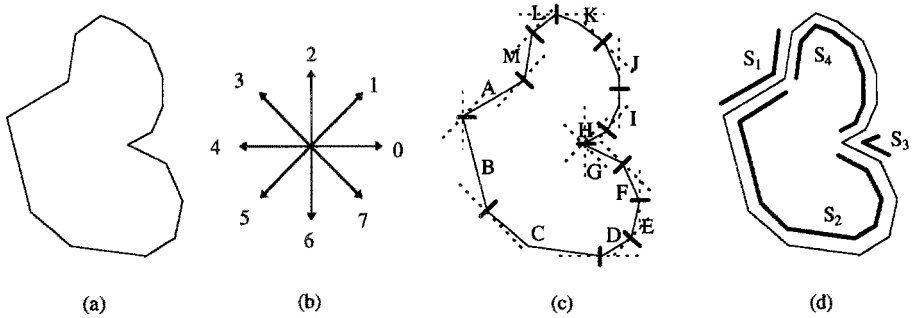


Fig. 1. (a) a closed contour with a polygonal approximation, (b) quantized-directional codes when $N=4$, (c) sub-segments when $N=4$, (d) segments when $N=4$.

tion parameters for model-based image coding for bi-level images along with very large databases of image pattern models.

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, the structural indexing is described along with the transformation rules of structural features to generate features which can be extracted from deformed patterns caused by noises and local shape deformations. In Section 4, the proposed method is validated by systematically designed experiments with a large number of testing data. Section 5 concludes this paper.

2. Structural Representation of Closed Contours

The structural representation of closed contours [5] is outlined in this section, based on quasi-convex/concave structures along contours incorporating $2N$ quantized-directional features (N is a natural number). As shown in Fig. 1a, the closed contour is first approximated by a polygon. 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. For adjacent sub-segments a and b , we write a concatenation of sub-segments a and b as $a \xrightarrow{j,k} b$. The arrow " \rightarrow " means that the sub-segment a is concatenated to b so that we turn counterclockwise when traversing them from a to b , and " j, k " denotes the direction codes of convexity formed by a and b .

Example 1: Fig. 1c illustrates the decomposition of a contour shown in Fig. 1a into sub-segments when $N=4$. We obtain the following concatenations for these sub-segments.

$$L \xrightarrow{3,3} M, K \xrightarrow{2,2} L, J \xrightarrow{1,1} K, I \xrightarrow{0,0} J, H \xrightarrow{7,7} I, H \xrightarrow{3,5} G, \\ F \xrightarrow{1,1} G, E \xrightarrow{0,0} F, D \xrightarrow{7,7} E, C \xrightarrow{6,6} D, B \xrightarrow{5,5} C, A \xrightarrow{3,4} B, A \xrightarrow{7,7} M$$

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

$$a_0 \xrightarrow{j(1,0),j(1,1)} a_1 \xrightarrow{j(2,0),j(2,1)} \dots \xrightarrow{j(n,0),j(n,1)} a_n. \quad (1)$$

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 that is not the joint with a_1 , and the ending point is as the end point of a_n that is not the joint with a_{n-1} . When the part of the contour corresponding to the segment is traversed from its starting point to its ending point, one turns counterclockwise around any joints of sub-segments.

A segment is characterized by a pair of integers $\langle r, d \rangle$, *characteristic numbers*, representing the degree of rotation and the direction of the segment:

$$r = \sum_{i=1}^n \left\{ (j(i,1) - j(i,0)) \% (2N) \right\} + \sum_{i=1}^{n-1} \left\{ (j(i+1,0) - j(i,1)) \% (2N) \right\} + 2 \quad (2)$$

$$d = j(1,0)$$

where $a \% b$ ($b > 0, 0 \leq a \% b < b$) is a remainder when a is divided by b . Based on the coordinate system defined by the bounding box of the contour (the upright rectangle just enclosing the shape) such that its center is located at $(0.5, 0.5)$ and the length of its longer side is 1, each segment is associated with eight parameters describing its size and position: location of its starting point (x_S, y_S) , location of its ending point (x_E, y_E) , location of the center (x_C, y_C) and size (W, H) of its bounding box. Furthermore, in the structural indexing and voting processes, these eight parameters are quantized into L intervals, treated as integers 0 through $L-1$. Therefore, features of a segment are described by a tuple of ten integers:

$$(r, d, \lfloor L \cdot x_S \rfloor, \lfloor L \cdot y_S \rfloor, \lfloor L \cdot x_E \rfloor, \lfloor L \cdot y_E \rfloor, \lfloor L \cdot x_C \rfloor, \lfloor L \cdot y_C \rfloor, \lfloor L \cdot W \rfloor, \lfloor L \cdot H \rfloor) \quad (3)$$

Adjacent segments are connected by sharing the first sub-segments or last ones of the corresponding sequences. The connection sharing the first sub-segments is called *h-connection*, and the one sharing the last sub-segments is *t-connection*, denoted by $s \overset{h}{\leftarrow} t$ and $s \overset{t}{\leftarrow} t$, respectively, for two adjacent segments s and t .

Example 2: The following segments, as shown in Fig. 1d, are generated from the 13 sub-segments shown in Fig. 1c:

$$S_1: A \xrightarrow{7,7} M, \quad S_2: A \xrightarrow{3,4} B \xrightarrow{5,5} C \xrightarrow{6,6} D \xrightarrow{7,7} E \xrightarrow{0,0} F \xrightarrow{1,1} G,$$

$$S_3: H \xrightarrow{3,5} G, \quad S_4: H \xrightarrow{7,7} I \xrightarrow{0,0} J \xrightarrow{1,1} K \xrightarrow{2,2} L \xrightarrow{3,3} M.$$

Their characteristic numbers are given by $\langle 2, 7 \rangle$, $\langle 8, 3 \rangle$, $\langle 4, 3 \rangle$, and $\langle 6, 7 \rangle$, respectively,

and the connections for the four segments are denoted as $S_1 \overset{h}{\leftarrow} S_2 \overset{t}{\leftarrow} S_3 \overset{h}{\leftarrow} S_4 \overset{t}{\leftarrow} S_1$.

In the sequel, we assume that segments are indexed sequentially so that the interior of the pattern or object lies on the left side.

Feature	Model Index List
...	...
(3, 4, 2, 3, 1, 0, 3, 2, 1, 1, 1)	3, 78, 346, 897
(3, 4, 2, 3, 1, 0, 3, 2, 1, 1, 2)	89, 298, 485, 837, 917
...	...
(5, 6, 2, 3, 1, 0, 3, 2, 1, 1, 1)	19, 289, 283, 584, 739, 937, 997
...	...

Fig. 2. Model database organization by structural indexing. Each table item stores a model identifier list with the segment feature corresponding to the table index.

3. Structural Indexing with Feature Generation Models

Features of each segment extracted from the contour curve are described by a tuple of 10 integers. A large table, as illustrated in Fig. 2, is constructed for a model set by assigning a table address to a feature (a tuple of 10 integers) and storing a list of the model identifiers with the corresponding feature as a table item. Furthermore, classification of the inputted pattern is carried out by voting for each model on the lists stored as the table item at the table address corresponding to each segment feature.

However, the features are sensitive to noises 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 noises 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 contours.

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

- (1) Change of convex/concave structures caused by perturbations along normal directions on the contour and scales of observation, along with transformations of characteristic numbers (rotation degrees and directions).
- (2) Transformations of characteristic numbers (rotation degrees and directions) caused by small rotations.

We describe these two types of transformation in the rest of this section, along with the voting and indexing processes.

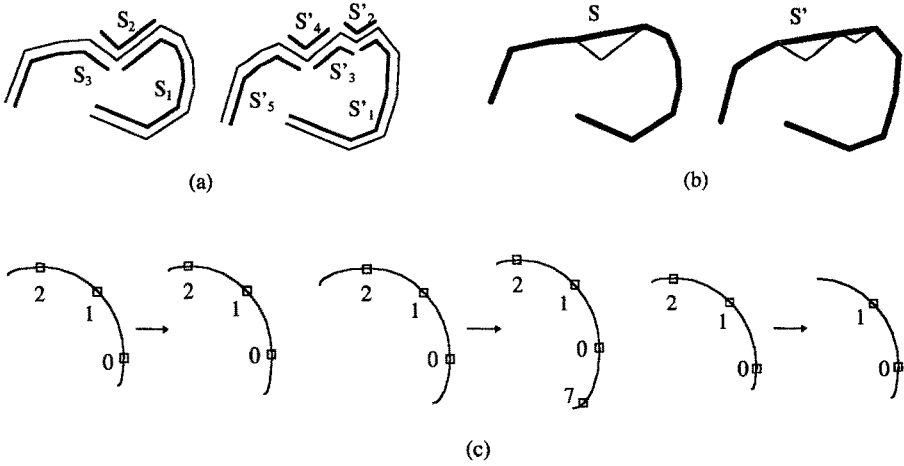


Fig. 3. (a) part of contours similar to one another in terms of global scales, (b) editing structural features by merging segment blocks, (c) transformations of characteristic numbers of segments by small rotations.

3.1. Transformations of Convex/Concave Structures

The convex/concave structures along the contour are changed by noises and local deformations, and also depend on scales of observations. For instance, two parts of contours 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 $S_1 \perp S_2 \overset{h}{\perp} S_3$ 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 $S'_1 \perp S'_2 \overset{h}{\perp} S'_3 \perp S'_4 \overset{h}{\perp} S'_5$ 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 contours 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 general, rules can be introduced for generating characteristic numbers from a segment block (a set of consecutive segments).

RULE 1: From a segment block, a characteristic number is generated according to the following rules:

- (1) From a segment block $\{S_i | i = 1, 2, \dots, n; S_1 \overset{h}{\perp} S_2 \perp \dots \perp S_n\}$, where n is odd, with characteristic numbers $\langle r_i, d_i \rangle$, a characteristic number $\left\langle \sum_{i=1}^n (-1)^{i+1} r_i, d_n \right\rangle$ is

generated if $r_{2i-1} - r_{2i} + r_{2i+1} \geq 2$ for $i = 1, 2, \dots, \lfloor n/2 \rfloor$ and $\sum_{i=1}^n (-1)^{i+1} r_i \geq 2$.

- (2) From a segment block $\{S_i | i = 1, 2, \dots, n; S_1 \perp S_2 \perp \dots \perp S_n\}$, where n is odd, with characteristic numbers $\langle r_i, d_i \rangle$, a characteristic number $\left\langle \sum_{i=1}^n (-1)^{i+1} r_i, d_1 \right\rangle$ is generated if $r_{2i-1} - r_{2i} + r_{2i+1} \geq 2$ for $i = 1, 2, \dots, \lfloor n/2 \rfloor$ and $\sum_{i=1}^n (-1)^{i+1} r_i \geq 2$.

These rules can be introduced from some mathematical properties mentioned in Nishida [5]. In the structural indexing and voting processes, for an integer M specifying the maximum number of segments to be merged, characteristic numbers are generated by applying RULE 1 to consecutive n segments ($n = 1, 3, \dots, M$).

3.2. Transformations of Characteristic Numbers by Small Rotations

The characteristic number $\langle r, d \rangle$ ($r \geq 2$) can be transformed by rotating the shape. Rules can be introduced for generating characteristic numbers by rotating the shape slightly (see Fig. 3c).

RULE 2: By applying a small rotation to the segment, the characteristic number $\langle r, d \rangle$ can be transformed into one of the following: (1) $\langle r, d \rangle$, (2) $\langle r+1, d-1 \rangle$, (3) $\langle r+1, d \rangle$, (4) $\langle r-1, d \rangle$ ($r \geq 3$), (5) $\langle r-1, d+1 \rangle$ ($r \geq 3$).

For instance, when $N = 4$ and $M = 3$, six characteristic numbers $\langle 2, 7 \rangle$, $\langle 8, 3 \rangle$, $\langle 10, 3 \rangle$, $\langle 4, 3 \rangle$, $\langle 6, 7 \rangle$, and $\langle 12, 7 \rangle$ are generated from RULE 1 from the four segments illustrated in Fig. 1d with characteristic numbers $\langle 2, 7 \rangle$, $\langle 8, 3 \rangle$, $\langle 4, 3 \rangle$, and $\langle 6, 7 \rangle$. Then, in total, 28 characteristic numbers $\langle 2, 7 \rangle$, $\langle 3, 6 \rangle$, $\langle 3, 7 \rangle$, $\langle 8, 3 \rangle$, $\langle 9, 2 \rangle$, $\langle 9, 3 \rangle$, $\langle 7, 3 \rangle$, $\langle 7, 4 \rangle$, $\langle 10, 3 \rangle$, $\langle 11, 2 \rangle$, $\langle 11, 3 \rangle$, $\langle 9, 3 \rangle$, $\langle 9, 4 \rangle$, $\langle 4, 3 \rangle$, $\langle 5, 2 \rangle$, $\langle 5, 3 \rangle$, $\langle 3, 3 \rangle$, $\langle 3, 4 \rangle$, $\langle 6, 7 \rangle$, $\langle 7, 6 \rangle$, $\langle 7, 7 \rangle$, $\langle 5, 7 \rangle$, $\langle 5, 8 \rangle$, $\langle 12, 7 \rangle$, $\langle 13, 6 \rangle$, $\langle 13, 7 \rangle$, $\langle 11, 7 \rangle$, $\langle 11, 8 \rangle$ are further generated by applying RULE 2 to these generated ones.

3.3. Indexing and Voting

In the model database organization step by structural indexing, features are generated from each model pattern by Rules 1 and 2, and the model identifier is appended to the model identifier list in the table item at the address computed from each generated feature. For each model i ($i = 1, 2, \dots, n$), let c_i be the number of features generated from segment features by Rules 1 and 2. For instance, $c_i = 28$ for the contour shown in Fig. 1a when $N = 4$ and $M = 3$. In the classification and retrieval by voting for models, from segment features extracted from the inputted curve, features are gener-

Table 1. Classification rates of deformed patterns by the proposed algorithm, with the level of noise and local shape deformations described by β , in comparison with an adaptation of Stein-Medioni method.

#models	β	Proposed Method				Stein-Medioni			
		Top 1%	Top 4%	Top 10%	Time (ms/sample)	Top 1%	Top 4%	Top 10%	Time (ms/sample)
100	0.0—0.5	96.4	99.7	99.9	17.2	84.1	95.8	97.6	61.7
	0.5—1.0	93.4	99.3	99.8		79.5	93.8	96.4	
	1.0—1.5	84.1	98.1	99.5		69.8	88.7	92.9	
	1.5—2.0	77.3	96.7	98.9		61.3	84.0	90.0	
500	0.0—0.5	98.8	99.9	100.0	23.0	90.2	95.6	97.5	63.3
	0.5—1.0	97.7	99.7	100.0		86.6	93.6	96.4	
	1.0—1.5	94.0	99.0	99.9		78.6	89.3	93.8	
	1.5—2.0	90.6	98.1	99.7		70.3	84.4	90.9	
1000	0.0—0.5	99.0	99.9	100.0	34.4	90.9	95.8	97.6	72.0
	0.5—1.0	98.0	99.7	100.0		87.4	93.9	96.6	
	1.0—1.5	94.7	99.1	99.9		79.6	89.7	94.0	
	1.5—2.0	91.5	98.3	99.7		71.5	84.9	91.1	

ated by Rule 1 and 2. Model identifier lists are retrieved from the tables by using the addresses computed from the generated features, and voting is carried out for each model on the lists. The inputted pattern is classified by selecting out some models according to the descendant order of v_i / c_i , where v_i is the number of votes for model i ($i = 1, 2, \dots, n$). For instance, for the inputted pattern presented on left of Fig. 4, forty-five top choices are shown from the model database composed of 1000 patterns.

4. Experiments

In this section, the proposed algorithm is evaluated statistically in terms of the robustness against noise and shape deformations, based on the systematically designed, controlled experiments with a large number of synthetic data. The experimental design is composed of the generation of model patterns and their deformations to be used as testing samples. A modification of the midpoint displacement algorithm [6] based on fractional Brownian motion is employed together with affine transformations for the model pattern generation. From each model pattern, a number of testing samples are generated by applying small rotations and random perturbations along the contours. One thousand models, as shown in Fig. 4, were created by the model generation algorithm.

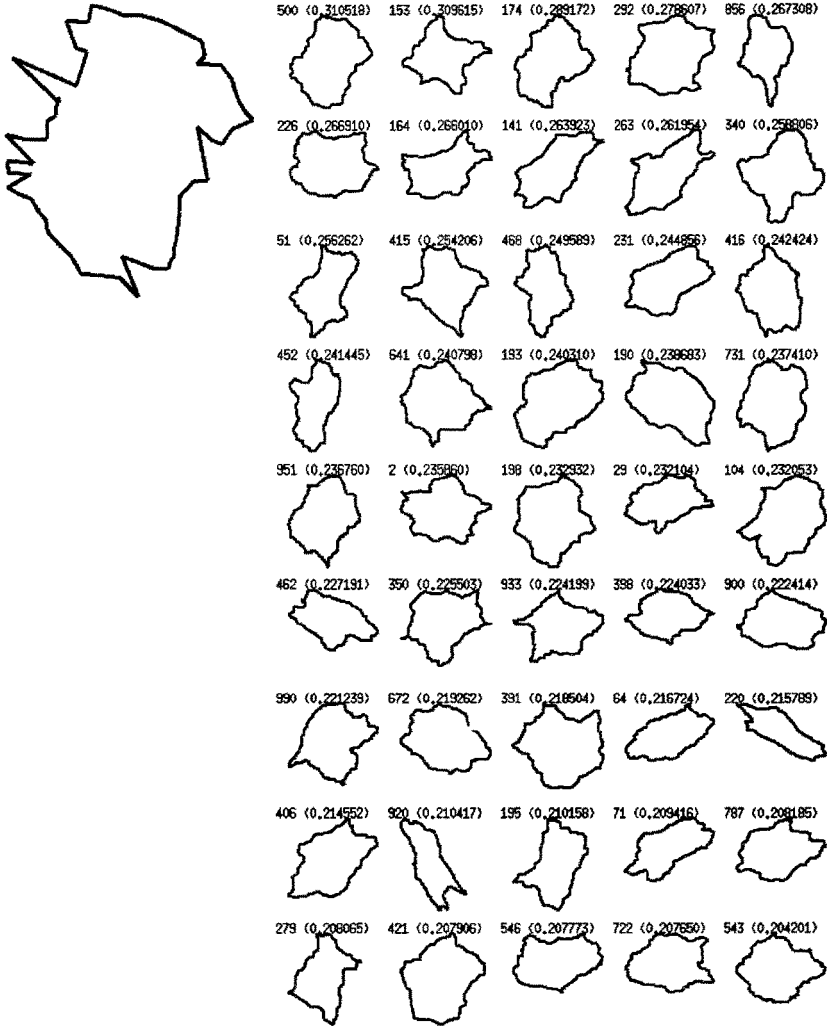


Fig. 4. An input pattern with 45 choices from the model database composed of 1000 patterns.

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, for comparison, we adapted Stein-Medioni method [2] to the model database organization and classification, extracting segment features from several versions of polygonal approximations of the shape contour with a variety of error tolerances for approximations. By changing the error tolerance for polygonal approximation of contours with Ramer's method from 1% to 10%, with a step of 1%, of the widest side of the bounding box of the contour, ten versions of polygonal approximations were created for each model pattern and inputted pattern.

Classification rates are presented for top 1% choices, top 4% choices, and top 10% choices, in Table 1. For instance, for 1000 models, correct models are included in top 40 choices with probability 98.3% for proposed algorithm, when $\beta \in [1.5, 2.0]$.

Clearly, significant improvements can be observed for the proposed method in terms of classification accuracy and processing time. Therefore, the effectiveness has been verified through the experiments for the shape feature generation models described in Section 3, along with the shape representation mentioned in Section 2.

Required properties for structural indexing are that the correct model is guaranteed to be included in the model candidate set composed of a few to 10 percent of the whole model set, and that the time complexity can be ignored compared with sophisticated algorithms. The result shows that these requirements are satisfied by the proposed method, and that classification can be sped up by several tens times without degrading classification accuracy.

5. Conclusions

We have presented an efficient, effective algorithm along with data structures for indexing of closed contours based on structural features in terms of convex/concave parts and quantized directional features along contours. In particular, we have addressed the problems of improving the accuracy and robustness of shape classification against noise and local shape transformations. To cope with these problems in structural indexing, shape feature generation techniques have been incorporated into structural indexing. The feature transformation rules obtained by an analysis of some particular types of shape deformations are exploited to generate features that can be extracted from deformed patterns. The generated features are used in model database organization and classification. Experimental trials have shown that the shape feature generation significantly improves the classification accuracy and efficiency.

References

1. A. Califano and R. Mohan, Multidimensional indexing for recognizing visual shapes, *IEEE Trans. Pattern Analysis and Machine Intelligence* **16**(6), 373—392 (1994).
2. F. Stein and G. Medioni, Structural indexing: efficient 2-D object recognition, *IEEE Trans. Pattern Analysis and Machine Intelligence* **14**(12), 1198—1204 (1992).
3. W.I. Grosky and R. Mehrotra, Index-based object recognition in pictorial data management, *Computer Vision, Graphics, and Image Processing* **52**, 416—436 (1990).
4. A. Del Bimbo and P. Pala, Image indexing using shape-based visual features, *Proceedings of 13th Int'l Conf. Pattern Recognition*, Vienna, Austria, vol. C, pp. 351—355 (1996).
5. H. Nishida, A structural model of curve deformation by discontinuous transformations, *Graphical Models and Image Processing* **58**(2), 164—179 (1995).
6. M. Miyojim and H.D. Cheng, Synthesized images for pattern recognition, *Pattern Recognition* **28**(4), 1995, 595—610 (1995).