1998 International Computer Symposium
Workshop on Image Processing and Character Recognition
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

# DEFORMED SHAPE RETRIEVAL BASED ON THE HIDDEN MARKOV MODEL

Fa-Shyang Chang and Shu-Yuan Chen@
Department of Computer Engineering and Science, Yuan-Ze University
135 Yuan-Tung Rd., Nei-Li, Chung-Li, Taoyuan, Taiwan, 320, R.O.C.

## ABSTRACT

A new deformed shape retrieval method based on the hidden Markov model (HMM) is proposed in this paper. Shape features as well as statistical and contextual information are incorporated into the HMM to derive probability values. Then, the probability values can be considered matching scores to retrieve similar shapes. The proposed method is translation, rotation and scale invariant. In addition, it is robust to various non-rigid deformations such as perspective, shear, occlusion and nonlinear distortions. The advantages are accomplished by the strategies we adopt. First, the proposed shape features are translation, rotation and scale invariant. Secondly, the flexibility of shape matching can be increased since the HMM is high tolerance to noise and distortion. Henceforth, non-rigid deformation can be coped with. Finally, although the HMM is computation intensive, only few high-level shape features are necessary for shape representation, thus, the computation efficiency is satisfactory. Our method has been applied on two databases: geometry and character. The experimental results prove the effectiveness, robustness and practicability of the proposed approach.

**Key words:** deformed shape retrieval, shape matching, hidden Markov model (HMM), non-rigid deformation, similarity measure, content-based retrieval.

## 1. INTRODUCTION

With the increasing popularity of the use of digital libraries, it becomes imperative to build an efficient content-based image retrieval system to browse through the entire library. A new shape-based image retrieval method is proposed in this paper. The proposed method is translation, rotation and scale invariant. In addition, it is robust to various non-rigid transformations such as perspective, shear, occlusion and nonlinear distortions. In other words, a shape being rotated, translated or scaled still looks alike itself, and even those are deformed non-rigidly such as perspective, shear, occlusion, and other nonlinear distortions. Examples of a shape and its deformed shapes are shown in Fig. 1.

In general, those approaches in the field of shape matching [1-14] can be broadly classified into two categories according to their objectives: recognition-oriented [1-7] and retrieval-oriented approaches [8-13].

The proposed approach is based on the polygonal approximation using the hidden Markov model. Shape features as well as statistical and contextual information are incorporated into the HMM to derive probability values. Then, the probability value can be considered matching scores to retrieve similar shapes. The advantages of our method are listed below.

(1)Most dynamic programming approaches used either global [5, 7, 8] or local features [4,5]. In general, global features convey overall property and are noise-insensitive but occlusion-sensitive, while local features convey local property and are noise-sensitive but occlusion-insensitive [4]. Our method adopts noise-insensitive global features, while deformation robustness including occlusion can be achieved by incorporating HMM into shape matching.

(2)The deviations resulting from polygonal approximation can also be handled by the HMM since the HMM is a flexible statistical model and tolerant to segmentation errors as proved in many applications [14, 16, 17]. The probability computed from the hidden Markov model can be regarded as the similarity measure [13,17], which in turn can be considered matching scores to retrieve similar shapes.

(a) (b) (c) (d) (e) (f) (g)

Fig. 1. Deformation of A. (a) Original character A; (b) rotated A; (c) perspective A; (d) sheared A; (e) pincushion A; (f) barrel A; (g) occlusion A.

@ To whom all correspondence should be sent.
Tel: 886-3-463-8800 Ext.357, Fax: 886-3-463-8850, E-mail: cschen@cs.yzu.edu.tw

♣ This work was supported partially by National Science Council, R.O.C. under grant NSC-88-2213-E-155-004.

(3) In general, a shape has more global features than local features. Since fewer numbers of global features are used, computation efficiency of our method can be satisfied, even when exhaustive searching by starting at each global feature is necessary for rotation invariant requirement.

The implementation of the proposed approach includes two steps. The first is the database creation step which is concerned with the construction of a HMM for each shape in the database. The second is the query comparison step, which deals with matching of the query shape with each HMM in the database.

The database creation step consists of two major phases: feature extraction and HMM construction. In the feature extraction phase, a contour is first extracted from each image. The shape features for each contour are then derived from corresponding breaking points, which are derived from critical points. Note that they are invariant to translation, rotation and size variations after suitable normalization is applied. Finally, we use shape features to construct HMMs for shape matching.

In the query comparison step, it also consists of two major phases: feature extraction as well as matching between query shape and HMMs in the database. The first phase is the same as that in the database creation step. To match query to HMMs in the database, matching steps must be performed. More specifically, the corresponding HMM can generate the probability values between the query and database shape. The probability values can provide useful information to derive similarity measure. In other words, the contours whose corresponding HMMs best match the query can be retrieved and ordered correctly.

In the remainder of this paper, basic concepts of HMM, feature extraction, HMM construction and the query comparison are described in Sections 2, 3, 4 and 5, respectively. Experimental results and conclusion are included in Sections 6 and 7.

## 2. BASIC CONCEPTS OF HMMS

Basic concepts of HMMs are briefly introduced in this section. A detailed description of HMMs can be found in the reference [15,16]. A left-to-right HMM, also called the Bakis model is used in this paper for simplicity. Let the observation sequence be denoted as $O = O_1, O_2, ..., O_T$, in which each observation $O_t$ characterizes the input signal at time t, and T is the number of observations in the sequence.

For an HMM, the notation $\lambda = (A, B, \Pi)$ is used to represent the model. The parameter set is described below.

1. N: the number of states in the model. We denote the set of states as $S = \{S_1, S_2, ..., S_N\}$, and the state at time t as $q_t$, where $1 \le t \le T$.
2. $V$: the observed value in each state. The range of $V$ may be infinite.
3. $A = \{a_{ij} | 1 \le i, j \le N\}$: the set of state-transition probability distributions, where $a_{ij} = P[q_{t+1} = S_j | q_t = S_i]$ is the probability of transition from state $S_i$ to state $S_j$ at time t. Obviously, each state transition coefficient is greater than or equal to zero, and the sum of each row in A will equal to one, i.e., $\sum_{j=1}^{N} a_{ij} = 1$.
4. $B = \{b_j(v) | 1 \le j \le N\}$: the set of observation probability distributions, in which $b_j(v) = P[O_t = v | q_t = S_j]$ is the probability of observing $v$ in state $S_j$ at time t.
5. $\Pi = \{\pi_i | 1 \le i \le N\}$: the set of initial state probability distributions, in which $\pi_i = P[q_1 = S_i]$ is the probability of the initial state $S_i$ at time 1.

In order to find the optimal state sequence, the Viterbi algorithm [15] is employed and stated as follows.

### Algorithm. Viterbi

Input. An HMM $\lambda = (A, B, \Pi)$ and an observation sequence $O = O_1, O_2, ..., O_T$.

Output. $p[O|\lambda]$, the probability of the observation sequence $O$ generated by the HMM $\lambda$.

Step1 $\delta_1(i) = \pi_i \cdot b_i(o_1)$ $\quad 1 \le i \le N$,

$\psi_1(i) = 0$.

Step2. $\delta_t(j) = \max_{1 \le i \le N}[\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(o_t)$ $\quad 2 \le t \le T$, $\quad 1 \le j \le N$,

$\psi_t(j) = \arg\max_{1 \le i \le N}[\delta_{t-1}(i) \cdot a_{ij}]$ $\quad 2 \le t \le T$, $\quad 1 \le j \le N$.

Step 3. $p[O|\lambda] = \max_{1 \le i \le N}[\delta_T(i)]$,

$\hat{q}_T = \arg\max_{1 \le i \le N}[\delta_T(i)]$.

Step 4. $\hat{q}_t = \psi_{t+1}(\hat{q}_{t+1})$

Step 5. Stop.

## 3. FEATURE EXTRACTION

The feature extraction consists of two major phases : polygonal approximation and feature derivation. In the former phase, contours are first extracted from images. Critical points and then breaking points are detected on the contours to represent the contour. In the latter phase, the shape features are derived on the basis of breaking points. In the remainder of this section, polygonal approximation and feature derivation are described in Sections 3.1 and 3.2, respectively.

### 3.1 Polygonal Approximation

Polygon approximation is essential to shape matching. Only if a good polygonal approximation is

is obtained , the subsequent matching work will be done well. Zhu et al. [18] and Gupta et al. [3] proposed novel methods to find out segmentation points. In the proposed method, a hybrid of the two approaches is proposed to obtain proper segmentation points (breaking points). The detection of critical points and then breaking points are briefly described below.

When a binary image is input, the corresponding edge image can be obtained by detecting all the black-to-white or white-to-black transitions. Actually, a pixel is an edge point if the pixel is different at least one of its 8-neighbors. When all the edges are detected, we can find out outer boundary contours by contour tracing.

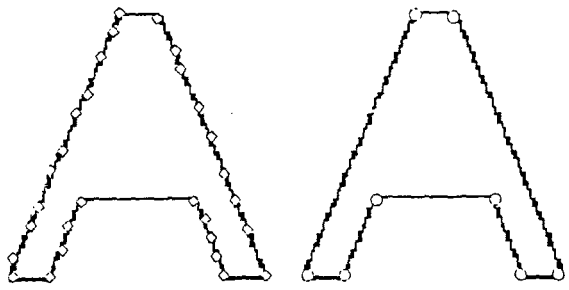
Then the modified Zhu-Chirlian algorithm [12] is employed to encode a contour as critical-point sequence. In general, the critical-points are the local maximum or minimum points of the boundary of the shape.

After each contour is first approximated by a sequence of critical points, they are then grouped together according to the colinear principle [3] to obtain global and stable representation of the contour. A set of critical points is said to be colinear if they lie on a straight line. The iterative algorithm proposed in [3] is adopted to locate pseudo breaking points from a sequence of critical points. Two constraints are then evaluated to screen out the actually breaking points. The two constraints are related with subtended angle as well as triangle region formed by consecutive pseudo breaking points as described.

(1)Let $Bp'_{i-1}$, $Bp'_i$ and $Bp'_{i+1}$ be three consecutive pseudo breaking points. If the subtended angle between $\overline{Bp'_{i-1}Bp'_i}$ and $\overline{Bp'_iBp'_{i+1}}$ is larger than 150°, the pseudo breaking point $Bp'_i$ is deleted.

(2) If the area of the triangle formed by three consecutive pseudo breaking points $\Delta Bp'_{i-1}Bp'_iBp'_{i+1}$ is smaller than one percentage of total area of the contour, the pseudo breaking point $Bp'_i$ is deleted.

In summary, a sequence of $CN$ critical points, denoted as ($Cp_1$, $Cp_2$, ..., $Cp_{CN}$), is obtained by the critical point-finding algorithm. Then, a sequence of $BN$ breaking points, denoted as ($Bp_1$, $Bp_2$, ..., $Bp_{BN}$), is obtained by the breaking-point-finding algorithm. An example of critical points and breaking points for character A is shown in Fig.2.



(a) (b)

Fig. 2. An example of critical points and breaking points for character A. (a) The critical points of character A; (b) the breaking points of character A.

## 3.2 Feature Derivation

In this paper, shape features are derived only from the breaking points since breaking points convey more high level and reliable information than critical points, Assume that the contour be represented as a sequence of $BN$ breaking points ($Bp_1$, $Bp_2$, ..., $Bp_{BN}$). Some notations used for the derivation of shape features are first defined followed by the feature derivation procedure. An illustration of shape features is shown in Fig. 3.

$$x_c = \frac{1}{N_b}\sum_{i\in O_B} x_i$$

$$y_c = \frac{1}{N_b}\sum_{i\in O_B} y_i$$

(1)Contour centroid: The contour centroid $P_c$ ($x_c$, $y_c$) is denoted as the centroid of the shape and can be computed as
where object $O_B$ denotes all the pixels belonging to enclosing region for the shape's contour and $N_B$ denotes the number of points inside object $O_B$.

(2)Contour area: The contour area $BA$ is denoted as the area of the enclosing region for the contour.

(3)Polar distance: The polar distance $\rho(Bp_i)$ is denoted as the distance between the point $Bp_i$ and the centroid $P_c$.

The shape features for the breaking point $Bp_i$ include the vertex angle, $B\phi'_i$, the chord subtended angle, $B\varphi'_i$, the chord distance, $B\rho'_i$, and the chord length, $Bl'_i$. All the features are normalized and described below.

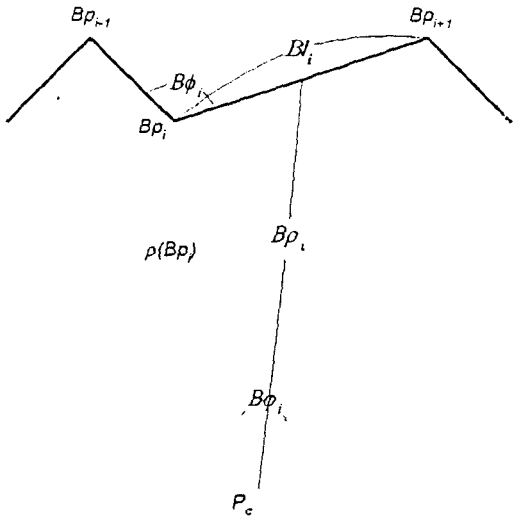



Fig. 3. Illustration of shape features.

(1)Vertex angle: The vertex angle $B\phi_i$ is demoted as the angle between the two line segments $Bp_{i-1}Bp_i$ and $\overline{Bp_i\ Bp_{i+1}}$, where $Bp_{i-1}$, $Bp_i$, and $Bp_{i+1}$ are three consecutive breaking points. Then the vertex angle can be normalized as $B\phi'_i$ using the following expression

$$B\phi_i' = \begin{cases} 1 - \dfrac{B\phi_i}{180} & \text{when} B\phi_i \geq 0 \\ \left|\dfrac{B\phi_i}{180}\right| - 1 & \text{when} B\phi_i < 0 \end{cases}$$

(2)Chord subtended angle: The chord angle $B\varphi_i$ is demoted as the angle subtended by the two line segments $\overline{Bp_iP_c}$ and $\overline{Bp_{i+1}P_c}$, where $Bp_i$ and $Bp_{i+1}$ are two consecutive breaking points and where $Bp_c$ is the centroid of the contour. Then the chord subtended angle can be normalized as $B\varphi'_i$ using the following expression

$$B\varphi_i' = \frac{B\varphi_i - B\varphi_{\min}}{B\varphi_{\max} - B\varphi_{\min}}$$

where $B\varphi_{max}$ and $B\varphi_{min}$ are the maximum and minimum values of the $B\varphi_i$ for all the values $i$ from $0$ to $BN$, i.e.

$B\varphi_{max} = \max(B\varphi_i) \forall i = 0,1,\ldots,BN-1$

$B\varphi_{min} = \min(B\varphi_i) \forall i = 0,1,\ldots,BN-1$.

(3)Chord distance: Let the point $Bpm_i$ be the middle point between the line segment $\overline{Bp_i\ Bp_{i+1}}$, where $Bp_i$ and $Bp_{i+1}$ are two consecutive breaking points. Then, the chord distance $B\rho_i$ is denoted as the polar distance $\rho(Bpm_i)$ for the point $Bpm_i$ and can be normalized as $Br'_i$ using the following expression

$$B\rho_i' = \frac{B\rho_i - B\rho_{\min}}{B\rho_{\max} - B\rho_{\min}}$$

where $B\rho_{max}$ and $B\rho_{min}$ are the maximum and minimum values of the $B\rho'_i$ for all the values $i$ from $0$ to $BN$.

(4)Chord length: The chord angle $Bl_i$ is demoted as the length of the chord segment $Bp_i\ Bp_{i+1}$, where $Bp_i$ and $Bp_{i+1}$ are two consecutive breaking points. Then, the chord length $Bl_i$ can be normalized as $Bl'_i$ using the following expression

$$Bl_i' = \frac{Bl_i - Bl_{\min}}{Bl_{\max} - Bl_{\min}}$$

where $Bl_{max}$ and $Bl_{min}$ are the maximum and minimum values of the $Bl_i$ for all the values $i$ from $0$ to $BN$.

Note that all the derived shape features are translation, rotation and scale invariant. The reasons are listed below. First, these features are either angle values or distance values which are translation and rotation invariant. Second, proper normalization as described above is performed to achieve scale invariance.

## 4. HMM CONSTRUCTION

After shape features for breaking points are extracted, two corresponding HMMs can be constructed. One is the ϕ-HMM with states related to vertex angle and the other is r-HMM with states related to chord subtended angle, chord distance and chord length. In general, the r-HMM inherit more spatial information than the ϕ-HMM. Thus, the matching in the r-HMM is more stringent than that in the ϕ-HMM. However, the ϕ-HMM has higher tolerance to deformation than the r-HMM.

The structures, the definitions of state transition and the definitions of observation probabilities for the two HMMs are described in Sections 4.1, 4.2 and 4.3, respectively.

### 4.1 Structure of HMMs

The structures of the ϕ-HMM and r-HMM are both of left-to-right HMMs associated with breaking points. Each vertex angle at breaking point represents a state in the ϕ-HMM, which models observation sequence of vertex-angles. Similarly, r-HMM models observation sequence of chord segments between two consecutive breaking points. When the input shape match to itself, the relationships between the observation sequence and the state sequence for ϕ-HMM are shown in Fig. 4. Note that the state $S_i$ with index $i$ is denoted as the index of the first end point $Bp_i$ of the chord segment $\overline{Bp_iBp_{i+1}}$.

### 4.2 Definition of State Transition Probability

The state transition probabilities for the ϕ-HMM and r-HMM are defined on the heuristic assumption that each state can forward jump at most four states. Moreover, the probabilities of the transition from each state to the allowed states are set equally for simplicity. On the basis of the concept, $a_{B\,jj}$ (from state $S_j$ to itself), and $a_{B\,jj+1}$ (from state $S_j$ to state $S_{j+1}$), $a_{Bjj+2}$ (from state $S_j$ to state $S_{j+2}$), $a_{Bjj+3}$ (from state $S_j$ to state $S_{j+3}$), and $a_{Bjj+4}$ (from state $S_j$ to state $S_{j+4}$) are the same. Hence, the state transition probabilities are defined by

$$a_{Bjj} = a_{Bjj+1} = a_{Bjj+2} = a_{Bjj+3} = a_{Bjj+4} = 1/5$$

### 4.3 Definition of the Observation Probability

In each state of the ϕ-HMM, a Gaussian distribution is used to estimate the probability that an input vertex angle is observed in the state. Actually, for each prototype contour in the database, the mean value of the Gaussian distribution of each state is the vertex angle of the corresponding breaking point; whereas, the standard deviation value of the Gaussian distribution is a prespecified value. In this study, the standard deviation is set as 35/180 empirically.

Thus, for a prototype contour with sequence of $BN$ breaking points, denoted as $(Bp_1^{(p)}, Bp_2^{(p)}, \ldots, Bp_{BN}^{(p)})$, the observation probability for an input breaking point $Bp_i^{(q)}$ with the vertex angle $B\phi'^{(q)}$ in the $i$th state of the ϕ-HMM is

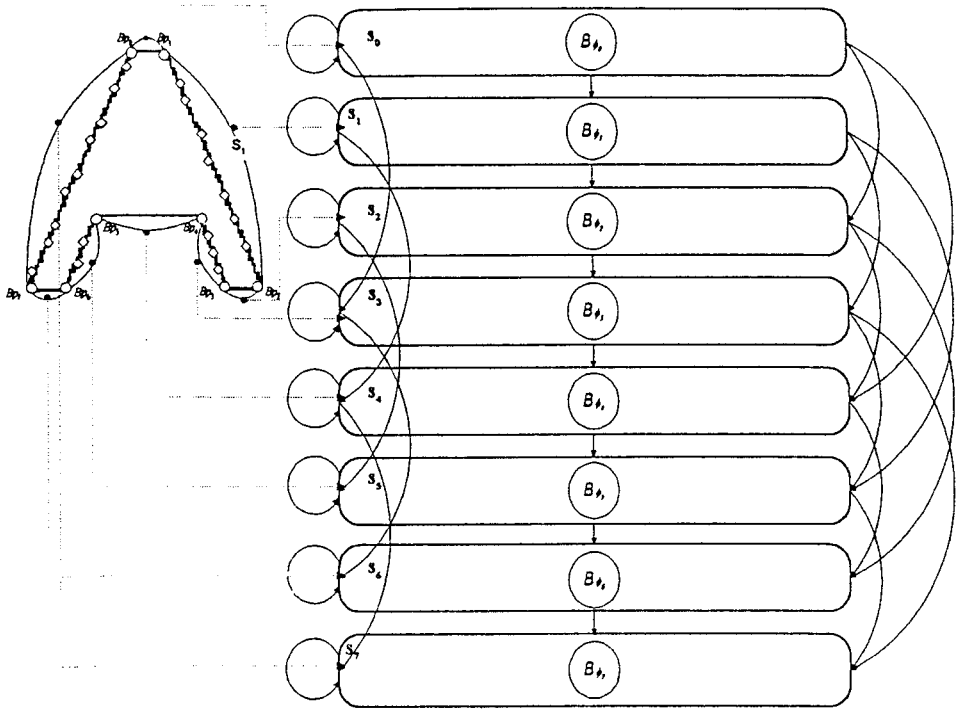


Fig. 4. Relationships between observation sequence of Bos and state seqence for Φ-HMM.

$$b_{B\phi}(B\phi^{(q)}, S^{(p)}) = \frac{\frac{1}{\sqrt{2\pi}\bullet 35/180}\exp\left\{-\frac{1}{2}\left[\frac{B_{\phi}^{(q)} - B_{\phi}^{(p)}}{35/180}\right]\right\}}{\sqrt{2\pi}\bullet 35/180}$$

where $B\phi_i^{(p)}$ is the vertex angle of the related breaking point $Bp_i^{(p)}$.

Similarly, in each state of the r-HMM, Gaussian distributions are used to estimate the probability that input shape features such as chord subtended angle, chord distance and chord length are observed in the state. In general, the features related to chord segment have more spatial information than those related to vertex angle since the spatial layout related to a contour can be implied by these features. For example, the contour formed by the character "I", "L" and "T" may have similar sequence of vertex angles, whereas they may have different sequence of the chord features. Thus, the matching in the r-HMM is more stringent than that in theϕ-HMM.

There are three features related to each chord segment, therefore, three Gaussian distributions, one for each feature, are needed in the r-HMM. The three observation probabilities can be defined in the similar for ϕ-HMM except that the standard deviation is set as 35/180 for angle feature and 1/8 for length feature

. Thus, for a prototype contour with sequence of $BN$ breaking points, denoted as $(Bp^{(p)}_1, Bp^{(p)}_2, \ldots, Bp^{(p)}_{BN})$, the observation probability, $b_{Bi}(B_p^{(q)}, S_i^{(p)})$ for an input breaking point $Bp^{(q)}$ with features $B\varphi^{(q)}$, $B\rho^{(q)}$, and $Bl^{(q)}$ in the $i$th state of the r-HMM is

$$b_{Bi}(Bp^{(q)}, Si^{(p)}) = b_{B\varphi}(Bp^{(q)}, Si^{(p)}) \bullet b_{B\rho}(Bp^{(q)}, Si^{(p)}) \bullet b_{Bl}(Bp^{(q)}, Si^{(p)}) \quad (2)$$

$$b_{B\varphi}(B_{\varphi}^{(q)}, S_i^{(p)}) = \frac{\frac{1}{\sqrt{2\pi}\bullet 35/180}\exp\left\{-\frac{1}{2}\left[\frac{B_{\varphi}^{(q)} - B_{\varphi}^{(p)}}{35/180}\right]\right\}}{\sqrt{2\pi}\bullet 35/180}$$

$$b_{B\rho}(B_{\rho}^{(q)}, S_i^{(p)}) = \frac{\frac{1}{\sqrt{2\pi}\bullet 1/8}\exp\left\{-\frac{1}{2}\left[\frac{B_{\rho}^{(q)} - B_{\rho}^{(r)}}{1/8}\right]\right\}}{\sqrt{2\pi}\bullet 1/8}$$

$$b_{Bl}(B_p^{(q)}, S_i^{(p)}) = \frac{\frac{1}{\sqrt{2\pi}\bullet 1/8}\exp\left\{-\frac{1}{2}\left[\frac{B_l^{(q)} - B_{li}^{(p)}}{1/8}\right]\right\}}{\sqrt{2\pi}\bullet 1/8}$$

where $B\varphi_i^{(p)}$, $B\rho_i^{(p)}$, and $Bl_i^{(p)}$ are the chord subtended angle, chord distance and chord length of the related breaking point $Bp_i^{(p)}$ in the prototype contour $P$.

## 5. QUERY COMPARISON

To match query contour to prototype HMMs in the database, the Viterbi algorithm must be employed to evaluate the matching probability values between the query and prototype contours. The probability values can be integrated to derive the matching scores to accomplish the goal of similar shape retrieval. In other words, the prototype contours whose HMMs best match the query contour can be retrieved. The derivation of similarity values is described below.

Let the $\phi$-HMM and r-HMM related to a prototype contour $P$ be $\lambda_{B\phi}^{(p)}$ and $\lambda_{Br}^{(p)}$. For a a query contour $Q$, the statistical probability values, $p(Q|\lambda_{B\phi}^{(p)})$ and $p(Q|\lambda_{Br}^{(p)})$,between $Q$ and $\lambda_{B\phi}^{(p)}$ and $\lambda_{Br}^{(p)}$, respectively, can be evaluated by the Viterbi algorithm. However, to employ the Viterb algorithm in our method, two important issues must be pointed out. First, under the assumption that the breaking point $Bp^{(q)}$ of the query contour $Q$ is observed at time t, the observation probability $b_j(o_t)$ in the Viterbi algorithm are just $b_{B\phi}(B_p^{(q)}, S_i^{(p)})$ and $b_{Br}(B_p^{(q)}, S_i^{(p)})$ for the $\phi$-HMM and r-HMM, as described by Eqs. (1) and (2), respectively. Second, to achieve rotation invariance, the initial probability for each state must be set the same value of 1. That means exhaustive search by starting the matching at each breaking is necessary. In general, the rotation invariance requirement can be overcome by rotating the query and the prototype contours to the same orientation. However, when the shape of the contour is deformed too seriously, the principal axis or the feature having the largest value may be changed and can not be the reference axis anymore. Thus, the exhaustive search is adopted by our method. Fortunately, such decision will not sacrifice the execution time seriously since the number of shape features is small as mentioned in Section 3.

After all, the overall similarity value, $S(Q,P)$ between the query contour $Q$ and the prototype $P$ can be defined as the sum of statistical probability values, $p(Q|\lambda_{B\phi}^{(p)})$ and $p(Q|\lambda_{Br}^{(p)})$. That is

$$S(Q,P) = p(Q|\lambda_{B\phi}^{(p)}) + p(Q|\lambda_{Br}^{(p)})$$

The similarity value can be used to rank the prototypes to retrieve similar shapes. In general, the prototypes having higher value is considered more similar to the query contour.

## 6. EXPERIMENTAL RESULTS

The proposed method is implemented under the Windows 95 operation system on a K6-233 PC with 128MB DRAM and Visual C++ 5.0 tool kits. The proposed method was tested on two databases: geometry and character as listed in [19]. To demonstrate the performance of the proposed approach, translation, rotation and scale invariance as well as perspective, pincushion/barrel, shear, occlusion and any other nonlinear distortions must be taken into account in the query-shape design.

A geometry database has six different geometry shapes, i.e., cross, triangle, star, rectangle, tree and ellipse. For each shape, its two scaled versions, two rotated versions, two perspective distortions, two shear distortions, two pincushion/barrel distortions, one occlusion distortion and three nonlinear distortions are used as query shapes to retrieve similar shapes in the database. The fourteen query shapes are produced by software package PhotoImpact. In a summary, 84 queries are performed for the geometry database.

A character database has twenty-six characters from "A" to "Z" of the same font Arial. Similarly, for each character, its two scaled versions, two rotated versions, two perspective distortions, two shear distortions, two pincushion/barrel distortions, one occlusion distortion and two nonlinear distortions are also used as query shapes to retrieve similar shapes in the database. The thirteen query shapes are also produced by software package PhotoImpact. In a summary, there are 338 queries for the character database.

We designed two independent experiments to query each of the two databases. However, the experiment schemes and the retrieval accuracy evaluation are the same for the two databases as described below. We pick each test shape to retrieve six similar shapes in the respective databases. On the average, the processing time for each test query is about 3 and 5 seconds for geometry and character databases, respectively.

Retrieval accuracy can be measured by the position (rank) of the required (original) image [10,11]. Ideally, the first retrieved image in the retrieval list should be the original image from which the query image is deformed. Thus, the ideal retrieval rank for each query must be 1. For each database, the average retrieval ranks are computed with respect to various shapes and various deformations, respectively. The average results for geometry database are listed in Tables 1 and 2, while for character database in Tables 3 and 4. For example, each entry in Table 1 is the average value of 14 retrieval ranks for 14 queries because each geometry shape in the geometry database has 14 various deformations, thus, there are 14 deformed queries to test whether the desired original shape can be retrieved. Some retrieval results are shown in Figs. 5 and 6.

Table. 1. Average retrieval rank for geometry database with respect to various geometry shapes.

| Cross | Triangle | Star | Rectangle | Tree | Ellipse | Average |
|---|---|---|---|---|---|---|
| 1.14 | 1.93 | 1.0 | 1.57 | 1.29 | 1.0 | 1.32 |

Table. 2. Average retrieval rank for geometry database with respect to various deformations.

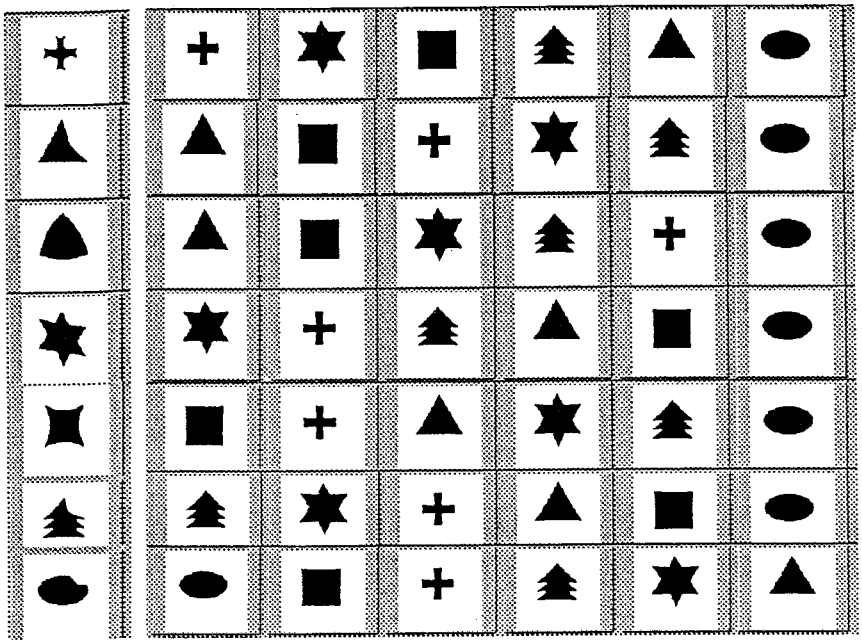
| Magnified | Minified | Right rotation | Left rotation | Perspective (top) | Perspective 2 (left) | Shear 1 (right) | Shear 2 (right-down) | Pincu-shion | Barrel | Occlusion | Nonlinear 1 | Non-linear 2 | Non-linear 3 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.17 | 1.00 | 1.33 | 1.00 | 1.5 | 1.00 | 1.17 | 1.17 | 1.83 | 1.33 | 1.67 | 1.17 | 1.83 | 1.33 | 1.32 |

Table. 3. Average retrieval rank for character database with respect to various characters.

| A | B | C | D | E | F | G | H | I | J | K | L | M | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.00 | 3.54 | 1.54 | 1.85 | 1.85 | 1.61 | 1.08 | 1.00 | 1.39 | 1.00 | 1.08 | 1.08 | 1.54 | |
| N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Average |

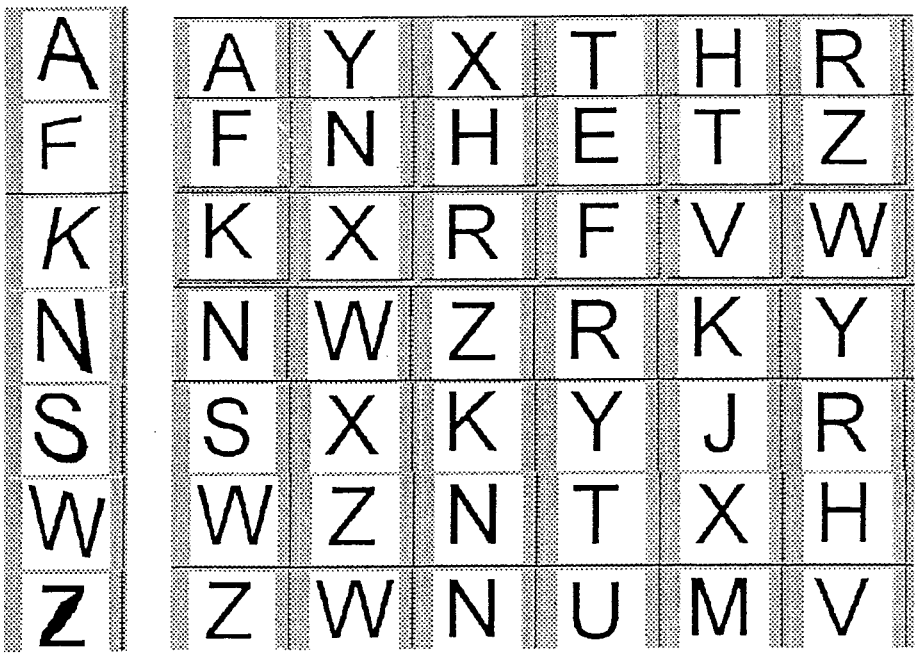
Table. 4. Average retrieval rank for character database with respect to various deformations.

| Magnified | Minified | Right rotation | Left rotation | Perspective 1 (top) | Perspective 2 (left) | Shear 1 (right) | Shear 2 (right-down) | Pincu-shion | Barrel | Occlusion | Nonlinear 1 | Nonlinear 2 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1.15 | 1.11 | 1.68 | 1.58 | 1.61 | 1.61 | 1.38 | 2.18 | 1.5 | 1.42 | 1.11 | 1.69 | 1.15 | 1.47 |



(a) (b)

Fig. 5. Some query results for geometry database. (a) Query shapes; (b) a list of six most similar shapes ordered by similarity values.



(a) (b)

Fig. 6. Some query results for character database. (a) Query shapes; (b) a list of six most similar shapes ordered by similarity values.

# 7. CONCLUSIONS

A new deformed retrieval method based on HMM is proposed in this paper. Since the shape features adopted in this study are invariant to translation, rotation and size variations, the proposed method can handle the invariant requirement. In addition, since HMM is high tolerance to noise as well as distortion and contextual information can be incorporated into HMM algorithm in a natural and elegant way, the flexibility for shape matching can be improved leading to deformation robustness results. Finally, although the HMM is computation intensive, only few high-level shape features are enough for similar retrieval, the computation efficiency will not be degraded seriously. Experimental results prove the effectiveness, robustness and practicability of our method.

Further research may be directed to the following topics. First, transition probability and parameter-learning strategy can be incorporated into HMM to improve the retrieval performance. Second, the correspondence information implied by the matching state sequence can be used to increase the retrieval accuracy and to solve correspondence problem. Third, the proposed method can be extended to natural images.

# REFERENCES

[1] L.K. Huang and M.J. Wang, "Efficient shape matching through model-base shape recognition," *Pattern Recognition,* Vol. 29, No. 2, pp. 207-215, 1996.

[2] X. Liu, S. Tan and S.H. Ong, "Fuzzy pyramid scheme for distorted object recognition", *Pattern Recognition,* Vol. 29, No. 10, pp. 1631-1646, 1996.

[3] L. Gupta and K. Malakapalli, "Robust partial shape classification using invariant breakpoints and dynamic alignment," *Pattern Recognition,* Vol. 23, No. 10, pp. 1103-1111, 1990.

[4] L. Gupta, T. Sortrakul and A. Charles, "Robust automatic target recognition using a localized boundary representation," *Pattern Recognition*, Vol. 28, No. 10,pp. 1587-1598, 1995.

[5] W.H. Tsai and S.S. Yu, "Attributed string matching with merging for shape recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 7, No. 4, pp. 453-462, 1985.

[6] D. Hong, T. Sarkodie-gyan, A.W. Campbell and Y. Yan, "A prototype indexing approach to 2-D object description and recognition," *Pattern Recognition*, Vol. 31, No. 6, pp. 699-725, 1998.

[7] S.W. Chen, S.T. Tung, C.Y. Fang, S. Cherng and A.K. Jain, "Extended attributed string matching for shape recognition," *Computer Vision and Image Understanding,* Vol. 70, No. 1, pp. 36-50, 1998.

[8] H.L. Peng and S.Y. Chen, "Trademark shape recognition using closed contours," *Pattern Recognition Letters*, Vol. 18, pp. 791-803, 1997.

[9] A. Pentland, R.W. Picard and S. Sclarogg, "Photobook: Content-based manipulation of image databases," *International Journal of Computer Vision,* Vol. 18, No.3, pp. 233-254, 1996.

[10] C. Faloutsos et al., "Efficient and effective querying by image content," *Journal of Intelligent Information Systems,* Vol. 3, pp. 231-262, 1994.

[11] A.K. Jain and A. Vailaya, "Image retrieval using color and shape, " *Pattern Recognition*, Vol. 29, No. 8, pp. 1233-1244, 1996.

[12] W.S. Lin abd S.Y. Chen, "Robust image retrieval using color and shape," *Proc. IPPR Conf. CVGIP*, pp.154-161, Taiwan, R.O.C., 1997.

[13] Y. He and A. Kundu, "2-D shape classification using hidden markov model," *IEEE Trans. Pattern Analysis and Machine Intelligence,* Vol. 13, No.11, pp. 1172-1184, 1991.

[14] A.D. Bimbo and P. Pala, "Visual image retrieval by elastic matching of user sketches," *IEEE Trans. Pattern Alaysis and Machine Intelligence,* Vol. 19, No. 2, pp.121-132, 1997.

[15] G.D. Forney, "The Viterbi algorithm," *Proceedings of the IEEE*, Vol. 61, No.3, pp. 268-278, 1973.

[16] L. R. Rabiner and B. H. Juang, *Fundamentals of speech recognition*. Prentice Hall Press, Englewood Cliffs, NJ(1993).

[17] H.C. Lin, L.L. Wang and S.N. Yang, "Color image retrieval based on hidden Markov models," *IEEE Trans. Image Processing,* Vol. 6, No. 2, pp. 332-339, 1997.

[18] P. Zhu and P.M. Chirlian, "On critical point detection of digital shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 8, pp. 737-748, 1995.

[19] F.S. Chang, Deformed Shape Retrieval Based on the Hidden Markov Model, *Master Thesis of the Institute of Electrical Engineering and Computer Engineering and Science*, Yuan Ze University, R.O.C., 1998.