ELSEVIER

Available online at www.sciencedirect.com
ScienceDirect

Pattern Recognition 41 (2008) 2189–2199

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/pr

# Detection and recognition of contour parts based on shape similarity

Xiang Bai[a], Xingwei Yang[b], Longin Jan Latecki[b,*]

[a]*HuaZhong University of Science and Technology, Wuhan, China*
[b]*Temple University, Philadelphia, USA*

Received 1 January 2007; received in revised form 12 November 2007; accepted 29 December 2007

## Abstract

Due to distortion, noise, segmentation errors, overlap, and occlusion of objects in digital images, it is usually impossible to extract complete object contours or to segment the whole objects. However, in many cases parts of contours can be correctly reconstructed either by performing edge grouping or as parts of boundaries of segmented regions. Therefore, recognition of objects based on their contour parts seems to be a promising as well as a necessary research direction.

The main contribution of this paper is a system for detection and recognition of contour parts in digital images. Both detection and recognition are based on shape similarity of contour parts. For each contour part produced by contour grouping, we use shape similarity to retrieve the most similar contour parts in a database of known contour segments. A shape-based classification of the retrieved contour parts performs then a simultaneous detection and recognition.

An important step in our approach is the construction of the database of known contour segments. First complete contours of known objects are decomposed into parts using discrete curve evolution. Then, their representation is constructed that is invariant to scaling, rotation, and translation.

*Keywords:* Shape similarity; Parts of visual form; Detection of contour parts

## 1. System overview

We begin with an overview of the system for contour-based object recognition. Based on psychophysical evidence [1], we can derive the following stages of contour-based object recognition (see Fig. 1):

(1) Edge detection ((a) → (b)).
(2) Contour grouping ((b) → (c)).
(3) Contour detection ((c) → (d)).

We do not discuss edge detection here, since it is an obvious step in image analysis that is also known to be performed by the human visual system. We view contour grouping as grouping of edge pixels to contour parts. It is a local process based on rule of good continuation (see Ref. [1]). As can be seen by comparing (b) and (c) in Fig. 1, contour grouping also means contour simplification and removal of small irrelevant features in our approach. We describe it in Section 5.

In the proposed system contour grouping is followed by contour detection, described in Section 6. For each contour part produced by contour grouping [32], the most similar contour parts in a database of known contour segments are retrieved using shape similarity. Therefore, contour detection also yields preliminary recognition results of contour parts, which can be viewed as recognition hypothesis. This fact is illustrated in Fig. 2. The second column shows most significant contour parts detected in images shown in first column. The contour part significance ranking is based on shape similarity to know contour parts, which are shown in columns 3–7. If we use first nearest neighbor classifier (1NN), then column 3, which shows the most similar database contour segments, illustrates the recognition results. Thus, contour part detection and recognition are based on shape similarity to know contour parts. Observe that the objects in all three query images are correctly recognized based

[*] Corresponding author.
*E-mail address:* latecki@temple.edu (L.J. Latecki).

0031-3203/$30.00 © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2007.12.016

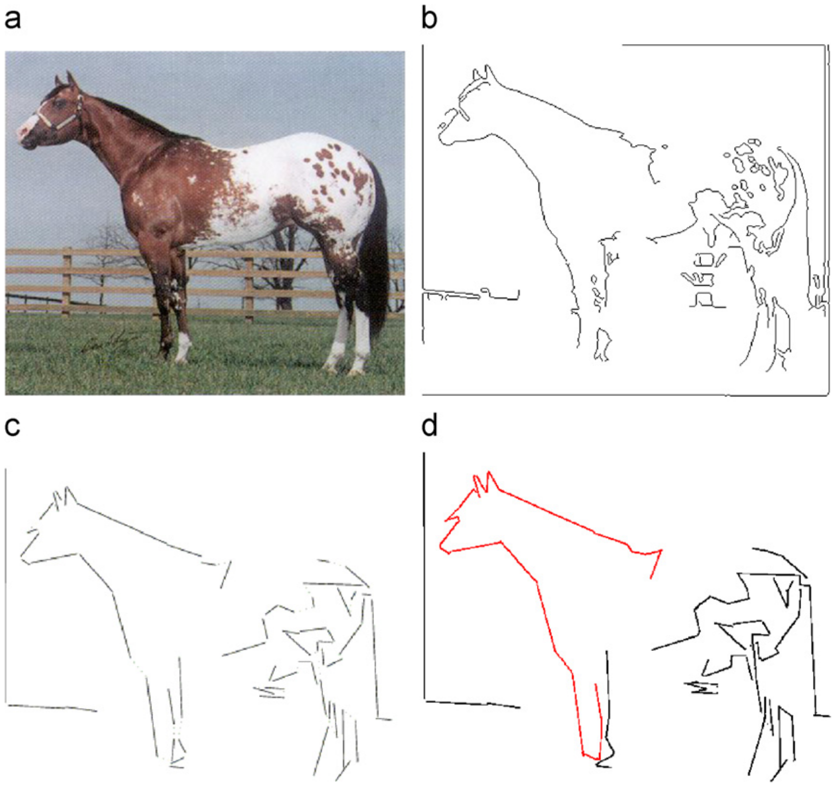

Fig. 1. (a) The original input image; (b) the edge map; (c) edge pixels group to line segments; (d) the most significant contour segment obtained by shape-based contour detection is marked in red.

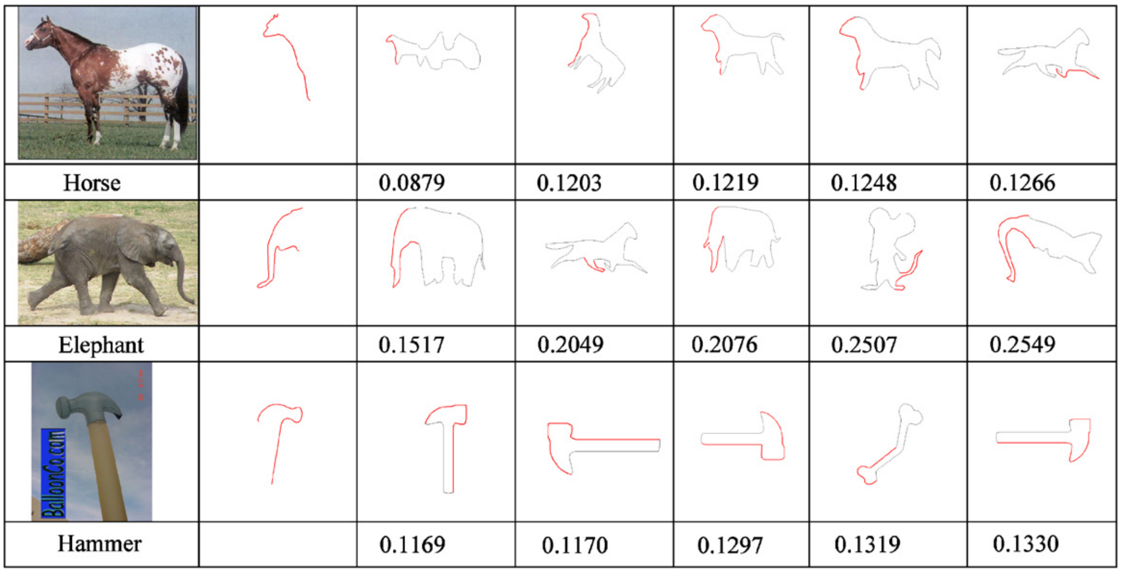

Fig. 2. Shape-based object recognition in images. After grouping of edge pixels (shown in Fig. 3) to contour parts, each contour part is compared to known contour segments using shape similarity. Column 2 shows contour parts extracted from images in column 1 that are most similar to know contour segments shown in Column 3. Columns 4–7 show further most similar known contour segments for the segments in column 2.

on the detected and recognized most significant contour parts shown in column 2. The edge maps of the three query images are shown in Fig. 3. The shape similarity measure we use is described in Section 4.

The performance of the proposed system is evaluated in Section 7. We used the MPEG-7 Shape 1 Part B dataset [2] for building the database of known contour parts (Section 3). The MPEG-7 dataset is composed of 70 object classes with

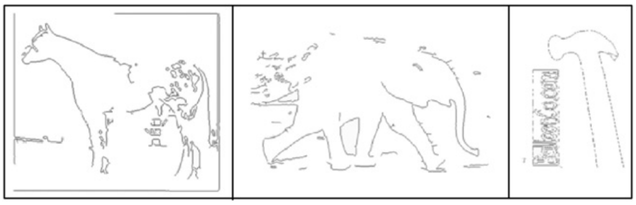

Fig. 3. Edges of color images in Fig. 2.

20 shapes in each class. We used shapes 1–10 in each class to construct the database. Based on the 700 complete contours, we generated nearly 58,000 contour segments. They represent the contour segments of objects known to our system. Thus, for each query contour part in the second column in Fig. 2, columns 3–7 show the most similar contour segments extracted from 58,000 contour segments. The objects in the three query images are different form the objects in the MPEG-7 Shape 1 Part B dataset, since the query images were not used in the construction of this database.

The main challenges of the proposed approach are contour grouping, detection of significant (grouped) contour parts, and similarity of contour parts. In this paper we address these challenges.

They belong to unsolved problems in computer vision, although they have received a substantial consideration in the literature. A brief overview of existing approaches is provided in Section 2.

## 2. Motivation and background

Given a relatively small object part, humans can recognize the object if the part is sufficiently unique. For example, it is obvious to us that Fig. 1 shows a horse. Moreover, the ease with which humans recognize articulated objects suggests that shape recognition in humans may solely be based on parts (See Fig. 4).

The cognitive importance of parts of visual form in human perception has been theoretically and experimentally verified, see e.g., Refs. [3,4,27–29]. However, identification of shapes given their parts is still an unsolved problem. Fig. 5 demonstrates the difficulties we may encounter in partial shape matching.

Given a significant part of visual form as a query, e.g., the fish tail shown in Fig. 5(a), our goal is to find similar shapes containing the query part; this means that we need to find the tail as a part of some other fish contours, e.g., it could be a part of the contours shown in Fig. 2(b) or (c). At least three serious problems arise:

(1) length problem,
(2) scale problem, and
(3) distortion problem.

Let us denote the query contour segment in Fig. 5(a) as $q$. Our goal is to find out that $q$ is similar to parts of contours in Fig. 5(b and c). However, the corresponding parts have different contour lengths and they are at different scales. Observe that problems (1) and (2) can be easily solved for complete contours by normalizing the contour lengths to be equal to one. The length normalization of contour segments does not solve the problems for the query part, since $q$ corresponds only to parts of the segments in Fig. 5(b and c). We would need first to find parts of these segments that correspond to $q$, and then normalize them. However, in order to do this, we need to know the proper length and scale of $q$ with respect to segments in Fig. 5(b and c), which leads to a chicken and egg problem. In addition, finding a part corresponding to $q$ in Fig. 5(c) is complicated by the spike on the fish tail. This illustrates the third problem of distortions (it affects whole contours as well). Distortions may arise from segmentation errors (segmentation artifacts) or from occlusions.

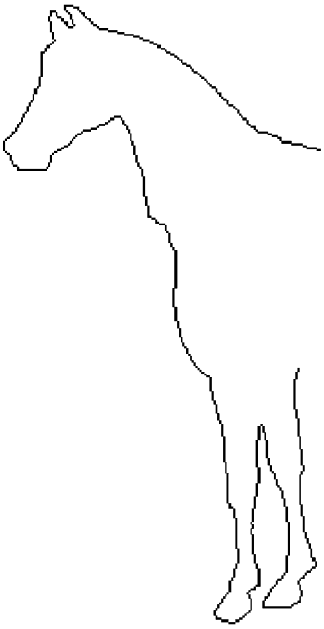

Fig. 4. Contour part.

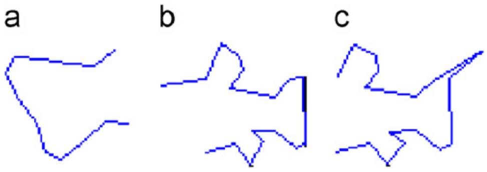

Fig. 5. (a) A fish tail contour; (b) a contour of a partially occluded fish; (c) a distortion led to a spike on the fish tail contour.

Problems 1, 2, and 3 make it very difficult to develop practically relevant similarity measure of contour parts. None of the current shape matching techniques provides solutions to these problems. Many approaches tolerate some minor occlusions or distortions, but most of the presented approaches require the whole contour or the whole object to be considered. This statement applies to all shape descriptors of whole contours, e.g., Refs. [5–8,33], in that they cannot be easily extended to handle similarity of contour parts. Only a small number of approaches address the issue of partial shape similarity [9–11,30,31]. The existing partial shape similarity measures (e.g., Ref. [9]) require that the query part is nearly identical to the corresponding part of the target contour. This is clearly an unrealistic assumption because of the noise distortions, and due to perspective projection changes. The approach in Ref. [9] is based on a matrix representation of pairwise distances between contour feature points. A matrix of a query contour part must match a submatrix of the target contour. Given the best match, an affine transformation is computed, and finally the Hausdorff distance of the transformed query part to the target contour is computed. Thus, this approach does not tolerate any distortions on the target part, and consequently, does not provide any solution to problem (3). Veltkamp and Tanase [10] proposed to use an extended dynamic programming approach directly on the turn angle function (also known as $\Psi$ function) representation of object contours. Their approach is not scale invariant, e.g., if part of the target contour similar to the query part is twice longer (due to a different scale) it is not recognized as similar. Observe that the $\Psi$ function itself is scale invariant (i.e., corresponding contour points have the same tangent directions), but its domain, which is the arc length contour parameterization, changes when the scale changes. Hence, it is a nontrivial problem to find the corresponding contour points. Consequently, this approach does not provide any solutions to problems (1) and (2). The approach in Ref. [10] also does not provide any solution to problem (3), since the distance of two $\Psi$ functions is the integral of their differences, which may be large in the presence of distortions on the target part, i.e., the matching of segments is not sufficiently elastic.

There exist a large number of approaches to contour grouping beginning with the work of Wertheimer [12] through the theory of visual interpolation in object perception of Kellman and Shipley [13] to the recent approaches that involve learning of contours from prior knowledge [14,15]. Stahl and Wang use geometric properties such as covexity [16] and symmetry [17] for contour grouping in a globally optimal fashion. Although closed contours are easier to group than open contour parts [18], this statement is only true if all edges that form a given close contour are present, which is seldom the case, e.g., due to occlusion or due to the performance of edge detectors. Therefore, we focus here on grouping open contour parts. We use a simple grouping approach based on a greedy strategy that includes good continuation and proximity. It is motivated by the approach in Ref. [13]. However, the main strength of our approach is evaluation of grouping hypotheses by using shape similarity to known contour parts. To our best knowledge none of the existing grouping approaches includes evaluation of grouping hypotheses.

## 3. Building a database of known contour parts

Our first goal is to build a representation of shapes that is suitable for shape-based recognition of contour parts. We assume that a set of complete contours of known shapes are given.

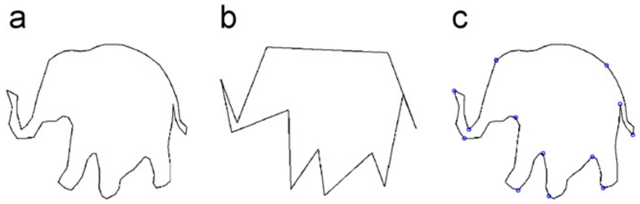

Fig. 6. (a) An elephant contour; (b) a DCE simplified polygon; (c) the common vertices of polygons in (a) and (b).

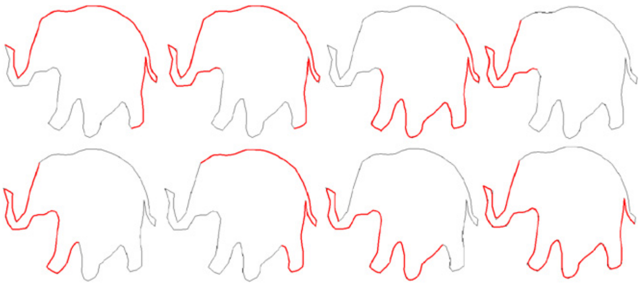

Fig. 7. Some contour parts between pairs of vertices in Fig. 6(c).

We face the following problems:

(1) How to extract meaningful contour parts?
(2) How to represent the contour parts so that they are invariant to scaling, translation, and rotation?

We provide answers to these questions in Sections 3.1 and 3.2, correspondingly.

### 3.1. Extraction of significant contour segments

It is shown in Ref. [19] that discrete curve evolution (DCE) yields a decomposition of complete contours that is in accord with human visual perception and is stable even in the presence of substantial contour deformations. DCE simplifies contour curves so that the main visual parts are preserved. For example, a DCE simplified polygon in Fig. 6(b) with only 13 vertices is still similar to the original contour in Fig. 6(a). In Fig. 6(c), the small circles illustrate the vertices of the simplified polygon on the original contour. The vertices of simplified polygon correspond to significant maxima and minima of the contour curvature, which are known to be important to decompose the contour into parts of visual form [3,4]. We use vertices of the simplified polygon to decompose the original complete contour into meaningful visual parts, which are defined as contour segments between each pair of the vertices. In Fig. 7, some visual part segments are shown in red. They are determined by vertices in Fig. 6(c).

DCE differs significantly form standard curve evolution approaches that are based on curve deformation guided by partial differential equations or their discrete analogs as presented in Bruckstein et al. [20]. DCE does not displace the contour points in the process of simplification. However, a key advantage of DCE in our context is the fact that it reduces the number of polygon vertices so that the remaining vertices represent significant contour points. The standard curve evolution approaches, including the approach in Ref. [20], keep the number of polygon vertices constant. They would therefore require an additional step after simplification to find significant contour points.

To summarize, for any contour polygon $w$, DCE yields a set of *critical points* $\Gamma(w)=\{u_i\}_{i=1}^n$, e.g., the small circles in Fig. 6(c). The contour segments of $w$ are defined as contour segments between all ordered pairs of points $(u_i, u_j)$ from $\Gamma(w)$ such that $u_i$ and $u_j$ are not adjacent. We will call them *visual segments*, since they are significant for shape recognition.

Now we briefly describe the process of DCE. Since any digital curve can be regarded as a polygon without loss of information (with possibly a large number of vertices), it is sufficient to study evolutions of polygonal shapes. Given the input boundary polygon $P$ with $n$ vertices, DCE produces a sequence of simpler polygons $P=P^n, P^{n-1}, \ldots, P^3$ such that $P^{n-(k+1)}$ is obtained by removing a single vertex $v$ from $P^{n-k}$ whose shape contribution measured by $K$ is the smallest. The order of removed vertices is determined by a relevance measure $K$ given by

$$K(s_1, s_2)=\frac{\beta(s_1, s_2)l(s_1)l(s_2)}{l(s_1)+l(s_2)}, \qquad (1)$$

where line segments $s_1$, $s_2$ are the polygon sides incident to a common vertex $v$, $\beta(s_1, s_2)$ is the turn angle at the common vertex $v$, $l$ is the length function. The main property of the relevance measure $K$ is that the higher value of $K(s_1, s_2)$, the larger is the contribution of the arc $s_1 \cup s_2$ to the polygon shape. The vertex removal idea of the DCE can be also expressed as segment replacement. In every evolution step, a pair of consecutive line segments $s_1$, $s_2$ is replaced by a single line segment joining the endpoints of $s_1 \cup s_2$. A stop condition that is adequate for shape similarity is given for DCE in Ref. [21]. It is based on the shape difference of the DCE simplified contour to the original input contour.

### 3.2. Normalization of visual segments

We use the method proposed in Ref. [22] to achieve invariance to planar similarity transformations (2-D translation, rotation, and uniform scaling) for the visual segments. Each visual segment $s$ is sampled with $n$ equidistant points $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. Then $s$ is transformed to a segment $s'$ in an invariant reference frame. The transformation is obtained by mapping $\mathbf{x}_1$ to $\mathbf{x}'_1=(0,0)$ and $\mathbf{x}_n$ to $\mathbf{x}'_n=(1,0)$, which allows us to compute the planar similarity transform mapping the remaining points to $\mathbf{x}'_2, \ldots, \mathbf{x}'_{n-1}$ in the normalize reference frame. In Fig. 8, the contour segment $ab$ is mapped to an invariant representation.

### 3.3. Database of known contour segments

Following the steps described in Sections 3.1 and 3.2, we constructed a database of nearly 58,000 known contour segments. They represent the contour segments of objects known to our system. We used the MPEG-7 Shape 1 Part B dataset for building this database [2]. The MPEG-7 dataset is composed of 70 object classes with 20 shapes in each class. We used shapes 1–10 in each class to construct the database of contour parts. We did not include in the database contour segments with too small $y$ variance in their invariant representation (Section 3.2), since such segments are nearly linear, and therefore bare no relevant shape information.

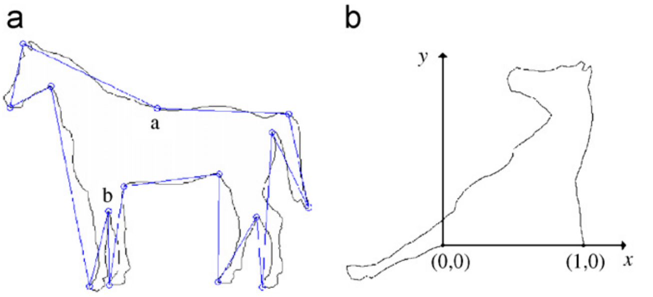

Fig. 8. (b) Shows an invariant segment representation of segment *ab* in (a).

Although the contour segments in the contracted database are normalized with respect to planar similarity transformations, they still provide a significant challenge to shape similarity measures. We may still face the problems described in Section 2 due to the instability of position of the critical points. For example, the position of the critical point $a$ on the complete contour in Fig. 8(a) is unstable. Even small noise or a small change in the perspective may displace point $a$ significantly. This implies that two similar contour segments may not necessarily match completely, but only parts of them may be similar. In other words, it may be impossible to correctly align two contour segments but only their parts.

The proposed approach is based on the assumption that the instability of the position of critical points is marginal. We assume that the displacement of most of the critical points is small. More precise, we assume that for each query contour part there exist a similar contour segment in our database. The large number of contour segments in our database clearly increases the probability that our assumption is correct. Moreover, critical points are known to be stable to view point changes [3,4]. Our experimental results presented in Section 6 verify the correctness of our assumption.

Due to our assumption, we need a shape similarity measure that can perform robustly in the presence of only small misalignment of parts of contour segments. Such a measure is described in Section 4.

## 4. Shape similarity of contour parts

The shape similarity measure used in our system is based on shape context [6]. We selected shape context, since it is robust to small misalignment of contour parts. Shape contexts are log-polar histograms of contour sample points relative to a given point on the shape [6,23]. A related representation was introduced in Ref. [8] as shape histograms. Every segment is represented by sample points: $P=\{p_1, \ldots, p_n\}$, $p_i \in R^2$. For a point $p_i$ on the contour, the shape context of $p_i$ is the histogram: $h_i(k)=\#\{q \neq p_i : (q-p_i) \in \text{bin}(k)\}$, where bins bin$(k)$ are defined using a uniform partition of directions in log-polar space and partition of distances from the point $p_i$, $h_i(k)$ is the number of contour points in the $k$th bin $bin(k)$ and $K$ is the total number of histogram bins. The bins are constructed by dividing the image plane into $K$ partitions (in a log-polar coordinate system) with $p$ as the origin. In this study, we use 5 intervals for the log distance $r$ and 12 intervals for the polar angle $\theta$, so $K = 60$, see Ref. [6] for details.

Consider a point $p_i$ on the first shape and a point $q_i$ on the second shape. Their similarity is measured as

$$C_{ij} = C(p_i, q_j) = \frac{1}{2}\sum_{k=1}^{K}\frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}. \tag{2}$$

To match and classify the query part accurately, we compute the shape contexts of all the points on the part. Since all the parts in our part segment database are normalized, we can skip the shape normalization in Ref. [6], and can directly compute the shape context distance:

$$D_{\mathrm{SC}}(Q, P) = \frac{1}{n}\sum_{p\in P}\arg\min_{q\in Q} C(p, q) + \frac{1}{n}\sum_{q\in Q}\arg\min_{p\in P} C(p, q), \tag{3}$$

where $p$ and $q$ denote the points on query part $Q$ and target part $P$ separately, and $n$ denotes the number of sample points on $Q$ and $P$.

Due to the size of the database of contour segments, for a given query contour part, we first use the shape contexts of center points to filter out clearly dissimilar segments. For each segment $s' = \{\mathbf{x}'_1, \ldots, \mathbf{x}'_n\}$ in an invariant frame, we compute its center of mass point

$$x'_m = \frac{1}{n}\sum_{i=1}^{n} x'_i,$$

and the mass shape context (MSC) for $s'$ as the histogram $h'_m : h'_m\ (k) = \#\{(x'_i - x'_m) \in \mathrm{bin}(k)\}$, where $i = 1, \ldots, n$. We use formula (2) to compute the matching cost of MSC of a query segment to all segments in the database. Then we use a threshold to construct a short list of candidate part segments. In our experiments described in Section 6, we used $T_1 = 20$ as a threshold to filter out the candidate part segments with MSC. These segments are then matched with formula (3) considering the shape contexts of all their points. The target part that has the minimal shape context distance to query part is considered as the most similar part, and the class that it belongs to is the classification result of the query part. Thus, we use 1-NN classification rule.

## 5. Extracting contour parts from images

We first group edge pixels to linear structures by applying an extended expectation maximization (EM) algorithm presented in Ref. [24]. This process is illustrated in Fig. 9. First a gray level edge map is computed, which is then approximated with a nonparametric probability density function (PDF). Then the extended EM algorithms fit line segments interpreted as a parametric PDF that minimizes Kullback–Leibler divergence (KLD) to the nonparametric PDF. This approach automatically determines the minimal number of model components, which are line segments in our applications. Four stages of line segment approximation computed this way are shown in Fig. 9.

The obtained line segments are then grouped to contour parts. The line segment grouping is based on two principles of perceptual grouping: proximity and good continuation [12]. We integrated these principles to a simple formula that measures the connectivity of two line segments as the weighted sum of the distance of their closest endpoints and their turn angle. We then iteratively connect line segments with the smallest connection value until the desired connection threshold is met. This is a simple grouping approach that is based only on a local heuristics. Definitely more sophisticated grouping methods are possible, e.g., a globally optimal usage of symmetry [17]. However, we stress that this simple rule was sufficient to group line segments to contour parts in our experiments. An example contour segment obtained by the line segment grouping is show in red color in Fig. 1(d). Clearly, this rule is not sufficient to group line segments to complete contours, since we only group adjacent line segments following the rule of good continuation. Here, we again benefit from the fact that we work with contour parts.

Our main contribution is evaluation of grouping hypotheses using shape similarity to known contour parts, which allows us to detect known contour parts as described in Section 6.

## 6. Contour detection

Let $I$ be a given digital image. We first extract all contour parts in $I$ using the contour grouping approach (Section 5). Then we compute their invariant representations (Section 3.2),

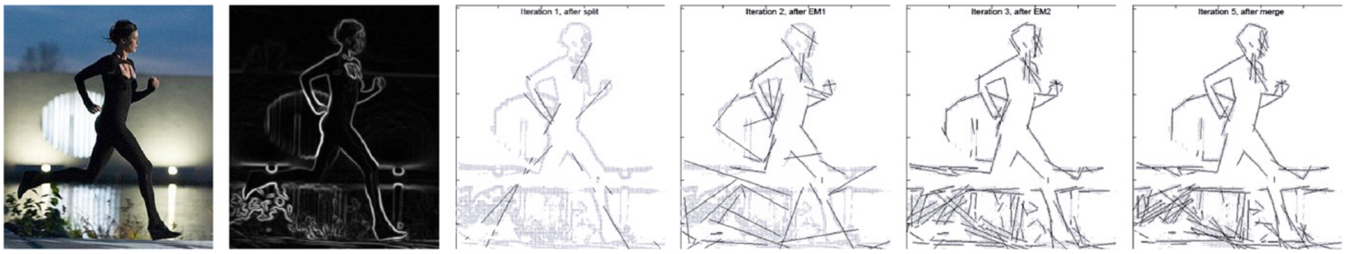

Fig. 9. We fit line segments to nonparametric pdf induced by a gray level edge map.

and eliminate the parts with too small $y$ variance (as we did for the database parts). Let $p_1, \ldots, p_k$ be the obtained contour parts in $I$. The parts $p_1, \ldots, p_k$ are compared to the contour segments in the database using the shape similarity $D_{SC}$ described in Section 4. Let $d(p_i)$ be the $D_{SC}$ distance to the database segment most similar to part $p_i$. We define now a significance relation of the grouped contour parts that is based on their similarity to known contour segments. Contour part $p$ is *more significant* than part $q$ if

$$d(p)\frac{\max(l(p), l(q))}{l(p)} < d(q)\frac{\max(l(p), l(q))}{l(q)}, \tag{4}$$

where $l(p)$ is the length of part $p$. Formula (4) panelizes shorter parts. For example, if part $q$ is two times shorter than part $p$, then the shape distance value $d(q)$is multiplied by 2. This relation is justified by the fact that the longer is the contour part obtained by contour grouping, the more likely it is part of a real contour in the image $I$. This is in accord with the theory of visual interpolation in object perception [13]. Finally, part $p$ is *most significant* if it is more significant than any other part in its image $I$.

Observe that the significance relation is based on 1-NN (first nearest neighbor) classifier. For each grouped contour part we only consider its dissimilarity value to the most similar segment in the database of known contour segments. Although 1-NN classifier is the simplest one [25], in many applications it outperforms more sophisticated classifiers. Xi et al. [26] note that "*1NN is an exceptionally competitive classifier, in spite of a massive research effort on time series/shape classification problems*". They compared 1-NN classifier to decision trees, Bayesian classifiers, neural networks, SVM, and rule-based methods and found that 1-NN has the best classification accuracy.

## 7. Performance evaluation

The goal of the proposed system is detection and recognition of contour parts extracted from images. We illustrated performance of our system on real images in Section 1. In order to perform more objective performance evaluation, we generated test images that contain one ground truth contour part and several distractor contour parts.

As the database of known contour segments, we used the database described in Section 3.3. We recall that it was constructed using contours of shapes with indices 1–10 of the 70 object classes in the MPEG-7 Shape 1 Part B dataset [2]. Each of the 70 object classes is composed of 20 shapes. Each of our test images contains one true contour part taken from shapes with indices 11–20 in one of the 70 classes plus 15 distractors, which are noisy contour parts. This way we assure that the query contour parts are known to our system.

Ten query test images are shown in the first column in Fig. 10. The second column in Fig. 10 shows the most significant contour parts detected in these images. The query part shown in column 2 was obtained by contour grouping in the query image shown in column 1. Since the query image is composed of several segments, the contour grouping did not extract the part identical to the part used to generate the query images. This is the reason why our query contour parts are not identical, the parts used to generate the images, and therefore, the query parts are not identical, to any parts in the database.

For each detected contour part in column two, its most similar database segment is shown in column three overlaid over its original contour. Columns 4 to 7 show further most similar segments. The shape dissimilarity values are shown under each database contour segment. Thus, we see for each query contour segment (column 2) the five most similar segments retrieved from 58,000 contour segments using shape similarity. The most significant contour part (column 2) is the one with the smallest shape dissimilarity value $d(p_i)$ scaled by the contour length as defined in Section 6.

All 10 most significant parts shown in column 2 in Fig. 10 are correctly classified. We classify them to the class of the most similar database segment (shown in column 3), i.e., we use 1-NN classifier.

Observe that the detection of most significant parts is closely link to their classification. In our framework, the detected contour part minimizes the dissimilarity to one of the contour segments in the database, and we assign it to the class of this database segment. We stress that the contour part detection may not necessary mean correct object classification but it is closely related. However, the class of the most similar database segment can be used as a classification hypothesis, which can then be verified, e.g., by fitting the whole original contour to the edge image. The verification is needed, because a given image may contain more than one signal contour and because it is possible that the detected contour is not correct in that its similarity to a database contour is accidental. This is illustrated in Fig. 10 by the fact that not all five most similar database segments (shown in columns 3 to 7) are from the class of the signal contour. Although some of the five most similar segments are in different classes, they are visually similar to the query parts. The only counterintuitive result is in third row and fourth column, which was the part of the bird contour is matched to a spiral. This result illustrates a drawback of the shape contexts. All corresponding points found by shape context have similar neighborhoods, but the global appearance of matched segment is different.

We evaluated the performance of proposed method using the 10 queries in Fig. 10. For each query part we counted the number of shapes from the same class among the first 10 most similar shapes. The overall retrieval rate is 52%. The rates for each query part are given in Table 1.

From the statistic in Table 1, we can observe that some query parts can get very good retrieval results but some cannot. There are two main reasons for this. The first one is if the part is too simple, it will be not unique and there will be a lot of parts accidentally similar to it. For example, the query part of the cattle is simple (20%). The second reason is if the part is too unique, the system cannot find the other parts except the part which is exactly the same with it. The query part of the elephant is too unique (10%).

Fig. 11 illustrates in the first column the second most significant part detected in the 10 query images. Columns 2 to

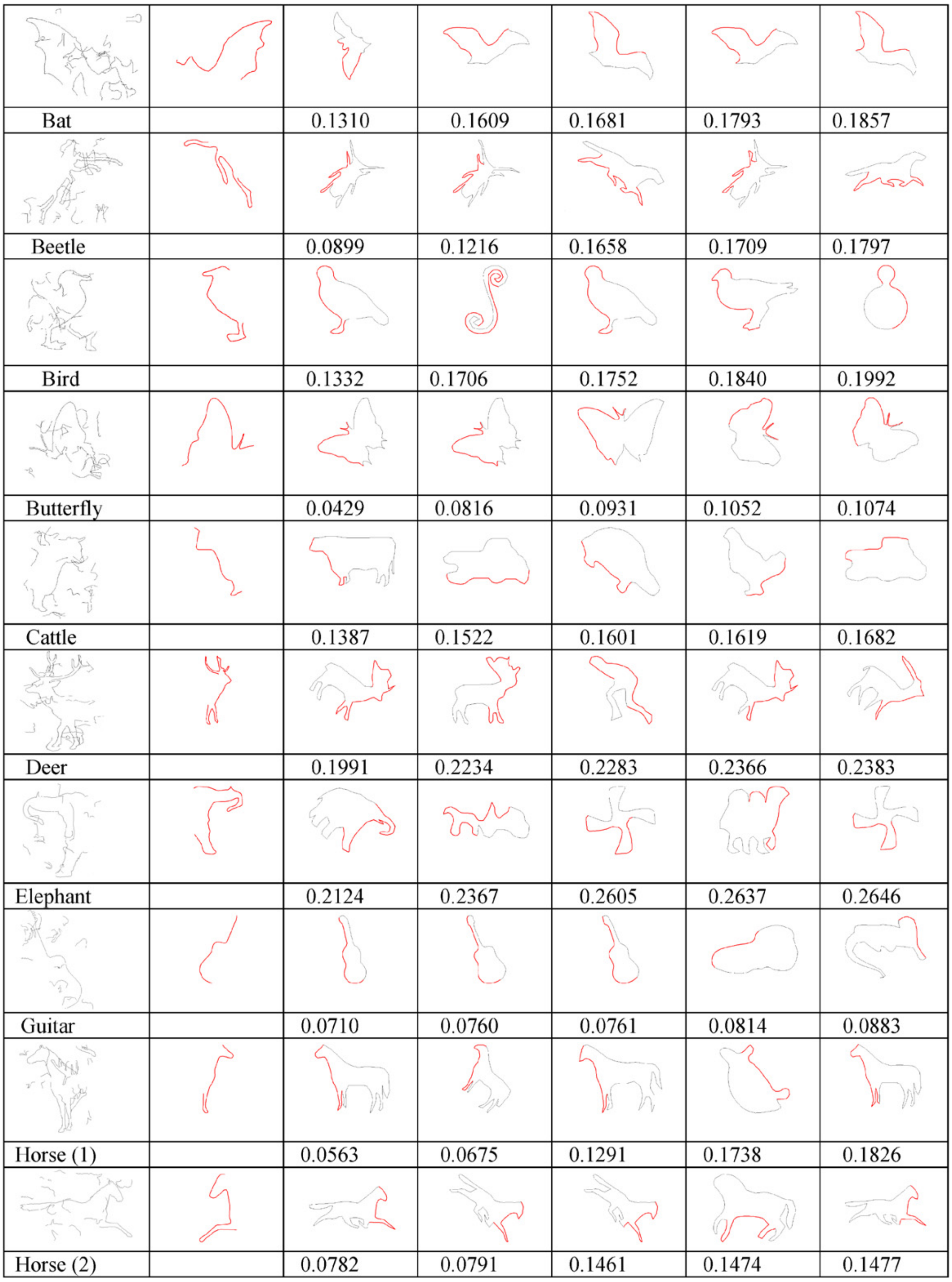

Fig. 10. Our retrieval results on 10 test images shown in column 1. Column 2 shows detected most significant contour parts. Columns 3–7 show most similar database segments to the detected contour parts.

6 show the most similar database segments to the second most significant parts. Observe that the retrieval results are significantly worse that in Fig. 10, which reflects the fact that the query parts in Fig. 11 are not true contour parts but distractors.

Some of our experimental results performed on real images are illustrated in Fig. 2 (Section 1). It is very interesting that the most similar database segment for the contour part of a horse in Fig. 2 belongs to the MPEG-7 class called carriage. This shows the limitations of keyword indexing of images, and

Table 1
The retrieval rates for 10 query shapes in Fig. 10

| Bat | Beetle | Bird | Butterfly | Cattle | Deer | Elephant | Guitar | Horse(1) | Horse(2) |
|---|---|---|---|---|---|---|---|---|---|
| 70% | 70% | 30% | 100% | 20% | 70% | 10% | 60% | 40% | 50% |

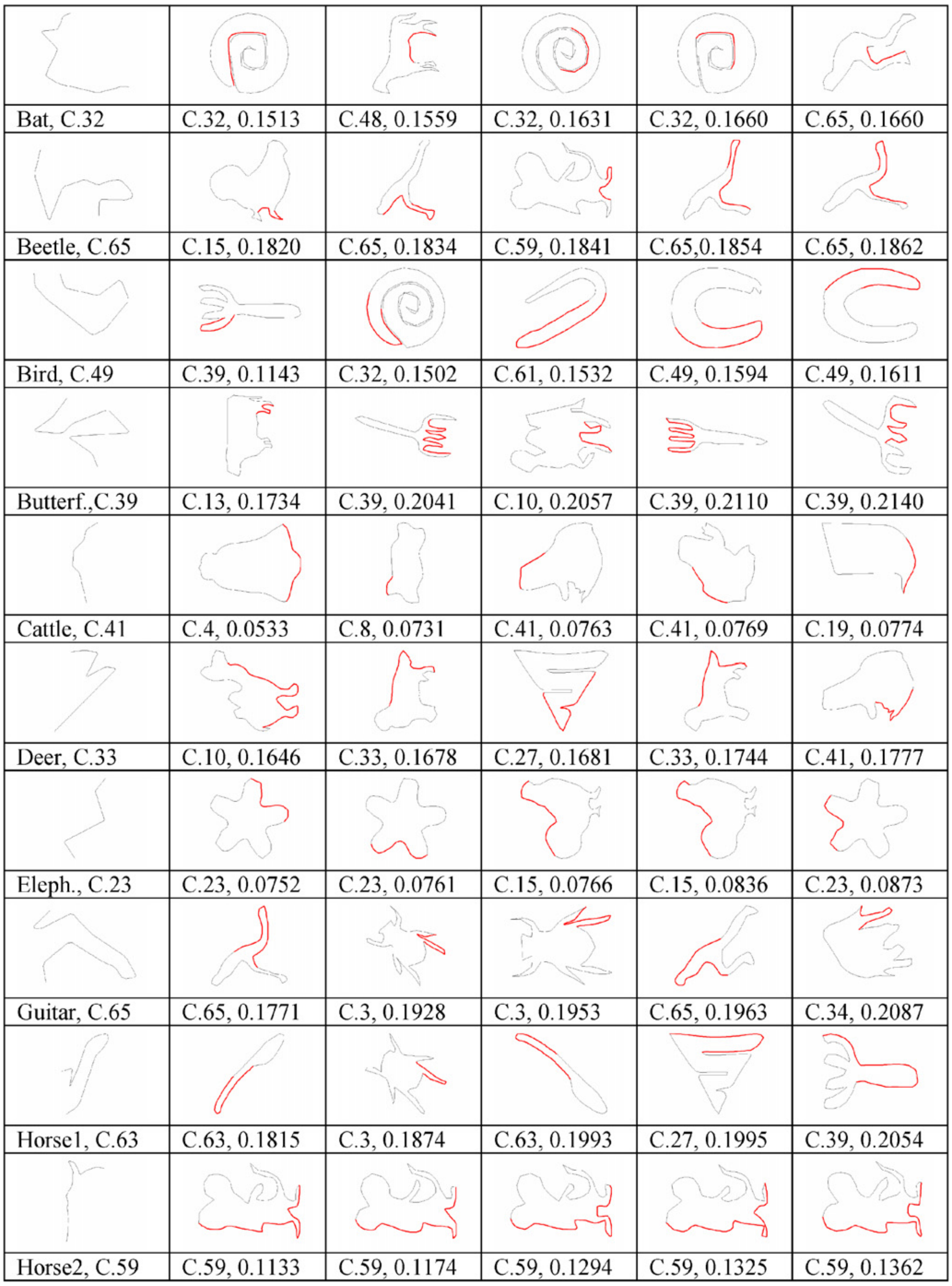

Fig. 11. The first column shows the second most significant contour parts extracted in the 10 test images. Columns 2–6 show the most similar database segments. The numbers following "C." indicate the MPEG-7 shape class.

it proves that our method can be useful for recognizing objects that due to an overlap were merged with other objects, i.e., the horse and carriage have one joint contour.

Some details in our experiment should be mentioned: our method is tested with $n = 100$ sample points on each contour segment. Usually the start point and end point of the part segments extracted from images may not be critical points, so we trimmed the segments to start and end with DCE critical points. Therefore, the detected contour parts shown in column 2 are subsegments of the signal contour segment. The average time for part retrieval is approximately 2 s, using Matlab 6.5 on a 1.5 GHz Xeon-based PC. The time required to set up the part segment database and MSC database is about 30 min.

## 8. Conclusions

The proposed approach applies a mixture of bottom-up and top-down processing to shape-based object recognition. After each bottom-up edge grouping step, top-down evaluation is applied to select the most promising grouping constellations. A promising grouping constellation is defined using cognitively motivated constraints. In accord with the cognitive simplicity principle known from Gestalt psychology [12], we propose to use partial shape similarity as a primary building block of such constraints. In accord with the newest results in human perception [1], grouping of edges to parts of object contours and recognition of the parts using shape similarity play a key role in object recognition. This means that object recognition is possible if only part of a contour is constructed, and the construction of the whole contour is not necessary for recognition. In particular, object recognition works in the presence of occlusion and segmentation errors.

Our experimental results demonstrate that the proposed system provides a good solution to the three problems described in Section 2. We conclude with the justification why this is the case.

Our method solves the length problem by keeping contour parts of different lengths in the database of known parts. We do not need to keep all possible parts of different lengths, since the shape context is tolerant for small length variations.

By normalizing the contour parts, following the method of Sun and Super [22], we have solved the problem of scale. The images in MPEG-7 dataset have a large variation of different sizes.

We do not provide a complete solution to the distortion problem, although our retrieval results are good in the presence of minor distortions. However, as shape context is sensitive to larger distortions, the proposed method cannot deal with significantly distorted shapes.

Another limitation of the proposed method is the usage of single contour parts as queries. Our future work will focus on using multiple contour parts as queries.

## Acknowledgments

This work was supported by NSF Grant no. IIS-0534929 and DOE Grant no. DE-FG52-06NA27508. We would like to thank Zygmunt Pizlo and Kevin Sanik (Purdue University) for providing us test images.

## References

[1] Z. Pizlo, Y. Li, G. Francis, A new look at binocular stereopsis, Vision Res. 45 (17) (2005) 2244–2255.

[2] L.J. Latecki, R. Lakaemper, U. Eckhardt, Shape descriptors for non-rigid shapes with a single closed contour, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2000.

[3] D.D. Hoffman, W.A. Richards, Parts of recognition, Cognition 18 (1984) 65–96.

[4] D.D. Hoffman, M. Singh, Salience of visual parts, Cognition 63 (1997) 29–78.

[5] L.J. Latecki, A. Gross, R. Melter (Eds.), Shape representation and dissimilarity for image databases, Pattern Recognition 35 (1) (2002) (special issue).

[6] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. PAMI 24 (2002) 509–522.

[7] C. Grigorescu, N. Petkov, Distance sets for shape filters and shape recognition, IEEE Trans. Image Process. 12 (9) (2003) 1274–1286.

[8] M. Ankerst, G. Kastenmüller, H.-P. Kriegel, T. Seidl, 3D shape histograms for similarity search and classification in spatial databases, Adv. Spat. Databases Int. Symp. (1999) 207–228.

[9] E. Saber, Y. Xu, A.M. Tekalp, Partial shape representation by sub-matrix matching for partial matching guided image labeling, Pattern Recognition 38 (2005) 1560–1573.

[10] R. Veltkamp, M. Tanase, Part-based shape retrieval, ACM Multimedia 12 (2005) 543–546.

[11] A. Pentland, Recognition by parts, in: Proceedings of International Conference on Computer Vision (ICCV), 1987, pp. 612–620.

[12] M. Wertheimer, Untersuchungen zur Lehre von der Gestalt II, Psycologische Forschung 4 (1923) 301–350.

[13] P.J. Kellman, T.F. Shipley, A theory of visual interpolation in object perception, Cognitive Psychol. 23 (1991) 141–221.

[14] Z. Tu, Probabilistic boosting-tree: learning discriminative models for classification, recognition, and clustering, in: 10th IEEE International Conference on Computer Vision (ICCV), 2005.

[15] J.H. Elder, A. Krupnik, L.A. Johnston, Contour Grouping with Prior Models, IEEE Trans. PAMI 25 (6) (2003) 661–674.

[16] J.S. Stahl, S. Wang, Convex grouping combining boundary and region information, in: IEEE International Conference on Computer Vision (ICCV), 2005, pp. 946–953.

[17] J.S. Stahl, S. Wang, Globally optimal grouping for symmetric boundaries, in: Proceedings of IEEE CVPR, New York, 2006.

[18] T. Tversky, W.S. Geisler, J.S. Perry, Contour grouping: closure effects are explained by good continuation and proximity, Vision Research 44 (2004) 2769–2777.

[19] L.J. Latecki, R. Lakaemper, Convexity rule for shape decomposition based on discrete contour evolution, Comput. Vision Image Understanding 73 (1999) 441–454.

[20] A.M. Bruckstein, G. Sapiro, D. Shaked, Evolutions of planar polygons, Int. J. Pattern Recognition Artif. Intell. 9 (1995) 991–1014.

[21] L.J. Latecki, R. Lakaemper, Shape similarity measure based on correspondence of visual parts, IEEE Trans. Pattern Anal. Mach. Intell. 22 (10) (2000) 1185–1190.

[22] K.B. Sun, B.J. Super, Classification of contour shapes using class segment sets, in: Proceedings of Computer Vision and Pattern Recognition (CVPR), 2005.

[23] G. Mori, S. Belongie, J. Malik, Efficient shape matching using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 27 (11) (2005) 1832–1837.

[24] L.J. Latecki, R. Lakaemper, Polygonal approximation of laser range data based on perceptual grouping and EM, in: IEEE International Conference on Robotics and Automation (ICRA), 2006.

[25] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.

[26] X. Xi, E. Keogh, C. Shelton, L. Wei, C.A. Ratanamahatana, Fast time series classification using numerosity reduction, ICML, 2006.

[27] I. Biederman, Human image understanding: recent research and a theory, CVGIP 32 (1985) 29–73.

[28] T. Binford, Visual perception by computer, in: IEEE Conference on Systems and Control, 1971.

[29] R. Brooks, Symbolic reasoning among 3D models and 2D images, Artif. Intell. 17 (1981) 285–348.

[30] L.J. Latecki, V. Megalooikonomou, Q. Wang, R. Lakaemper, C.A. Ratanamahatana, E. Keogh, Partial elastic matching of time series, in: Proceedings of the IEEE International Conference on Data Mining (ICDM), 2005, pp. 701–704.

[31] L.J. Latecki, R. Lakaemper, D. Wolter, Optimal partial shape similarity, Image Vision Comput. (IVC) 23 (2005) 227–236.

[32] L.J. Latecki, M. Sobel, R. Lakaemper, New EM derived from Kullback–Leibler divergence, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2006.

[33] G. McNeill, S. Vijayakumar, 2D shape classification and retrieval, in: International Joint Conference on Artificial Intelligence (IJCAI), 2005.

**About the Author**—XIANG BAI is a Ph.D. candidate at the Department of Electronics and Information Engineering of Huazhong University of Science and Technology (HUST). From January 2006 to May 2007, he worked in the Department of Computer Science and Information of Temple University supervised by Dr. L. J. Latecki. Currently, he has joined the Department of Neurology of the University of California, Los Angeles as a joint Ph.D. student. He has won the 2007 Microsoft Fellowship Award. His research interests include computer vision and machine learning.

**About the Author**—XINGWEI YANG received his B.E. degree in Electronics and Information Engineering from Huazhong University of Science and Technology (HUST), Wuhan, China, in 2002. Now, he is a Ph.D. candidate in Temple University.

**About the Author**—LONGIN JAN LATECKI is an associate professor in the Department of Computer and Information Sciences at Temple University in Philadelphia. He is the winner of the 25th Pattern Recognition Society Award together with Azriel Rosenfeld for the best paper published in the journal Pattern Recognition in 1998. He received the main annual award from the German Society for Pattern Recognition (DAGM), the 2000 Olympus Prize. He is a member of the Editor Board of Pattern Recognition and chairs the IS&T/SPIE annual conference series on Vision Geometry. He has published and edited over 150 research articles and books. His main research areas are shape representation and shape similarity, object recognition, robot mapping, video analysis, data mining, and digital geometry.