Learning Context-Sensitive Shape Similarity by Graph Transduction

Xiang Bai, *Student Member*, *IEEE*, Xingwei Yang, *Student Member*, *IEEE*, Longin Jan Latecki, *Senior Member*, *IEEE*, Wenyu Liu, *Member*, *IEEE*, and Zhuowen Tu, *Senior Member*, *IEEE*

Abstract—Shape similarity and shape retrieval are very important topics in computer vision. The recent progress in this domain has been mostly driven by designing smart shape descriptors for providing better similarity measure between pairs of shapes. In this paper, we provide a new perspective to this problem by considering the existing shapes as a group, and study their similarity measures to the query shape in a graph structure. Our method is general and can be built on top of any existing shape similarity measure. For a given similarity measure, a new similarity is learned through graph transduction. The new similarity is learned iteratively so that the neighbors of a given shape influence its final similarity to the query. The basic idea here is related to PageRank ranking, which forms a foundation of Google Web search. The presented experimental results demonstrate that the proposed approach yields significant improvements over the state-of-art shape matching algorithms. We obtained a retrieval rate of **91.61 percent** on the MPEG-7 data set, which is the highest ever reported in the literature. Moreover, the learned similarity by the proposed method also achieves promising improvements on both shape classification and shape clustering.

Index Terms—Shape similarity, shape retrieval, shape classification, shape clustering, graph transduction.

1 Introduction

Shape matching/retrieval is a very critical problem in computer vision. There are many different kinds of shape matching methods, and the progress in improving the matching rate has been substantial in recent years. However, nearly all of these approaches are focused on pairwise shape similarity measure. It seems to be an obvious statement that the more similar two shapes are, the smaller their difference is, which is measured by some distance function. Yet, this statement ignores the fact that some differences are more relevant while other differences are less relevant for shape similarity. It is not yet clear how biological vision systems perform shape matching; it is clear though that shape matching involves the high-level understanding of shapes. In particular, shapes in the same class can differ significantly because of in-class variation, distortion, or nonrigid transformation. In other words, even if two shapes belong to the same class, the distance between them may be very large if the distance measure cannot capture the intrinsic property of the shape. It appears to us that many published shape distance measures [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14] are unable to address this issue. For example, based on the inner distance shape context (IDSC) [3], the shape in Fig. 1a is more similar to Fig. 1b than to Fig. 1c, but it is obvious that shapes Figs. 1a and 1c belong to the same class. This incorrect result is due to the fact that the inner distance is unaware that the missing tail and one front leg are less relevant than much smaller shape details like the dog's ear and the shape of the head. No matter how good a shape matching algorithm is, the problem of more relevant and less relevant shape differences must be addressed if we want to obtain human-like performance. This requires having a model to capture the essence of a shape class instead of viewing each shape as a set of points, a parameterized function, or a manifold. In the proposed approach, each shape is considered in the context of other shapes in its class, and the class does not need to be known.

Given a database of shapes, a query shape, and a shape distance function, which does not need to be a metric, we learn a new distance function that is expressed by shortest paths on the manifold formed by the known shapes and the query shape. We can do this without explicitly learning this manifold. As we will demonstrate in our experimental results, the new learned distance function is able to incorporate the knowledge of intrinsic shape differences. It is learned in an unsupervised setting in the context of known shapes. For example, if the database of known shapes contains shapes (a)-(e) in Fig. 2, then the new learned distance function will rank correctly the shape in Fig. 1a as more similar to Fig. 1c than to Fig. 1b. The reason is that the new distance function will replace the original distance (a)-(c) in Fig. 1 with a distance induced by the shortest path between (a) and (e) in Fig. 2.

- *X. Bai and W. Liu are with the Department of Electronics and Information Engineering, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan, Hubei 430074, P.R. China. E-mail: xiang.bai@gmail.com, liuwy@hust.edu.cn.*
- *X. Yang and L.J. Latecki are with the Department of Computer and Information Sciences, Temple University, 1805 North Broad Street, Philadelphia, PA 19122. E-mail: {xingwei.yang, latecki}@temple.edu.*
- *Z. Tu is with the Lab of Neuro Imaging, University of California, Los Angeles, 635 Charles E. Young Drive South, Suite 225, Los Angeles, CA 90095. E-mail: zhuowen.tu@loni.ucla.edu.*

Manuscript received 10 Aug. 2008; revised 11 Jan. 2009; accepted 20 Mar. 2009; published online 9 Apr. 2009.
Recommended for acceptance by S. Belongie.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2008-08-0480.
Digital Object Identifier no. 10.1109/TPAMI.2009.85.

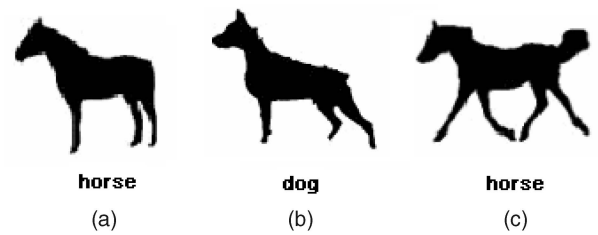

Fig. 1. Existing shape similarity methods incorrectly rank shape (b) as more similar to (a) than (c).

In the proposed approach, for a given similarity measure s_0, a new similarity s is learned through graph transduction. Intuitively, for a given query shape q, the similarity $s(q,p)$ will be high if neighbors of p are also similar to q. However, even if $s_0(q,p)$ is very high, but the neighbors of p are not similar to q, then $s(q,p)$ will be low. Thus, the new similarity s is context-sensitive, where a context of a given shape is defined by its neighbors, which are database shapes that are most similar to it. In this paper, we adopt a graph-based transductive learning algorithm to tackle this problem, and it has the following properties:

1. Instead of focusing on computing the distance (similarity) for a pair of shapes, we take advantage of the manifold formed by the existing shapes.
2. However, we do not explicitly learn the manifold nor compute the geodesics [15], which are time-consuming to calculate. A better similarity is learned by collectively propagating the similarity measures to the query shape and between the existing shapes through graph transduction.
3. Unlike the label propagation [16] approach, which is semi-supervised, we treat shape retrieval as an unsupervised problem and do not require knowing any shape labels.
4. We can build our algorithm on top of any existing shape matching algorithm and a significant gain in retrieval rates can be observed on well-known shape data sets.
5. The learned distance by our algorithm can also be used to improve the performance of the existing shape clustering methods.

Even if the difference between shape A and shape C is large, but there is a shape B which has small difference to both of them, we still claim that shapes A and C are similar to each other. This situation is possible for most shape distances since they do not obey the triangle inequality, i.e., it is not true that $d(A,C) \leq d(A,B) + d(B,C)$ for all shapes A, B, C [17]. If we have the situation that $d(A,C) > d(A,B) + d(B,C)$ for some shapes A, B, C, then the proposed method is able to learn a new distance $d'(A,C)$ such that $d'(A,C) \leq d(A,B) + d(B,C)$. Further, if there is a path in the distance space such that $d(A,C) > d(A,B_1) + \cdots + d(B_k,C)$, then our method learns a new $d'(A,C)$ such that $d'(A,C) \leq d(A,B_1) + \cdots + d(B_k,C)$. Since this path represents a minimal distortion morphing of shape A to shape C, we are able to ignore less relevant shape differences, and consequently, we can focus on more relevant shape differences with the new distance d'.

Our experimental results clearly demonstrate that the proposed method can improve the retrieval results of the existing shape matching methods. We obtained the retrieval rate of **91.61 percent** on part B of the MPEG-7 Core Experiment CE-Shape-1 data set [18], which is the highest ever bulls-eye score reported in the literature. We used the IDSC as our baseline algorithm, which has the retrieval rate of 85.40 percent on the MPEG-7 data set [3]. Fig. 3 illustrates the benefits of the proposed distance learning method. The first row shows the query shape followed by the first 10 shapes retrieved using IDSC only. Only two flies are retrieved among the first 10 shapes. The results of the learned distance for the same query are shown in the second row. All of the top 10 retrieval results are correct. The proposed method was able to learn that the shape differences in the number of fly legs and their shapes are not intrinsic to this shape class.

The remainder of this paper is organized as follows: In Section 2, we briefly review some well-known shape matching methods and the semi-supervised learning algorithms. Section 3 describes the proposed approach to

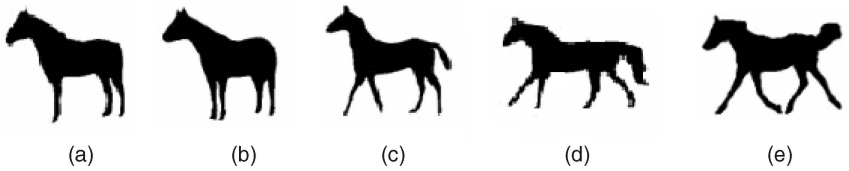

Fig. 2. A key idea of the proposed distance learning is to replace the original shape distance between (a) and (e) with a distance induced by geodesic paths in the manifold of known shapes. One such path is (a)-(e) in this figure.

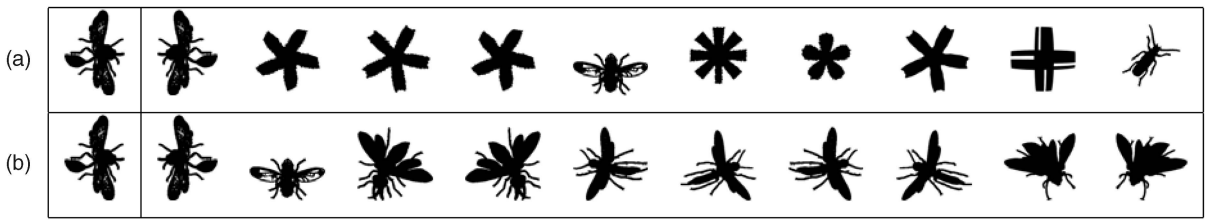

Fig. 3. The first column shows the query shape. The remaining 10 columns show the most similar shapes retrieved from the MPEG-7 data set. (a) Results of IDSC [3]. (b) Results of the proposed learned distance.

TABLE 1
Retrieval Rates (Bulls-Eye) of Different Methods on the MPEG-7 Data Set

Alg.	CSS [40]	Vis. Parts [4]	Shape Contexts [1]	Aligning Curves [41]	Distance Set [42]	Prob. Approach [43]	Chance Prob. [44]	Skeletal Context [45]	Gen. Model [2]	Optimized CSS [46]
Score	75.44%	76.45%	76.51%	78.16%	78.38%	79.19%	79.36%	79.92%	80.03%	81.12%
Alg.	Contour Seg. [47]	Multiscale Rep. [48]	Shape L'Âne Rouge [49]	Fixed Cor. [50]	Inner Distance [3]	Symbolic Rep. [30]	Hier. Procrustes [7]	Triangle Area [29]	Shape Tree [8]	IDSC [3] + our method
Score	84.33%	84.93%	85.25%	85.40%	85.40%	85.92%	86.35%	87.23%	87.70%	**91.61%**

learning shape distances, and relates it to PageRank. Section 4 relates the proposed approach to the class of machine learning approaches called label propagation. The problem of the construction of the affinity matrix is addressed in Section 5. Section 6.1 gives the experimental results on several famous shape data sets to show the advantage of the proposed approach. Conclusion and discussion are given in Section 7. A preliminary version of this paper appeared as [19]. In this paper, we introduce two new applications, shape clustering and retrieval of partially occluded shapes, and a systematic method for selecting optimal parameter setting in Section 6.1. We also relate the proposed approach to PageRank. Moreover, the experimental evaluation has been substantially extended.

2 Related Work

The semi-supervised learning problem has attracted an increasing amount of interest recently, and several novel approaches have been proposed. The existing approaches could be divided into several types, multiview learning [20], generative model [21], and Transductive Support Vector Machine (TSVM) [22]. Recently, there have been some promising graph-based transductive learning approaches proposed, such as label propagation [16], Gaussian fields and harmonic functions (GFHF) [23], local and global consistency (LGC) [24], and the Linear Neighborhood Propagation (LNP) [25]. Zhou et al. [26] modified the LGC for the information retrieval. The semi-supervised learning problem is related to manifold learning approaches, e.g., [27].

The proposed method is inspired by the label propagation method [16]. The reason we choose the framework of label propagation is that it allows clamping of labels. In other words, it fixes the label of labeled data points during the propagation process. Since the query shape is the only labeled shape in the retrieval process, the label propagation allows us to enforce its label during each iteration, which naturally fits in the framework of shape retrieval. Usually, GFHF is used instead of label propagation, as both methods can achieve the same results [16]. However, in the shape retrieval, we can use only the label propagation, the reason is explained in detail in Section 4.

Since a large number of shape similarity methods have been proposed in the literature, we focus our attention on methods that reported retrieval results on the MPEG-7 shape data set (part B of the MPEG-7 Core Experiment CE-Shape-1) [18]. This allows us to clearly demonstrate the retrieval rate improvements obtained by the proposed method. Belongie et al. [1] introduced a novel 2D histogram representation of shapes called Shape Contexts (SC). Ling and Jacobs [3] modified the Shape Context by considering the geodesic distance between contour points instead of the euclidean distance, which significantly improved the retrieval and classification of articulated shapes. Latecki and Lakämper [4] used visual parts represented by simplified polygons of contours for shape matching. Tu and Yuille [2] proposed the feature-driven generative models for probabilistic shape matching. In order to avoid problems associated with purely global or local methods, Felzenszwalb and Schwartz [8] described a dynamic and hierarchical curve matching method. Other hierarchical methods include the hierarchical graphical models in [28] and hierarchical procrustes matching [7]. Alajlan et al. proposed a mutiscale representation of triangle areas for shape matching, which also included partial and global shape information [29]. Daliri and Torre defined a symbolic descriptor based on Shape Contexts, and then used edit distance for final matching in order to overcome the difficulty caused by deformation and occlusions [30]. The methods above all focused on designing improved shape descriptors for single shapes and their comparison for pairs of shapes. Although the recent methods made some progress, the improvement is not obvious, as shown in Table 1 of Section 6.1. In this table, we summarize all the reported retrieval results on MPEG-7 database, and the retrieval rates of the recent publications are all around 85 percent. There are two main reasons that limit the progress in shape retrieval: 1) The case for large deformation and occlusions still cannot be handled well. 2) The existing algorithms cannot distinguish the more relevant and less relevant shape differences pointed out in Section 1.

There has been a significant body of work on distance learning [31]. Xing et al. [32] propose estimating the matrix W of a Mahalanobis distance by solving a convex optimization problem. Bar-Hillel et al. [33] also use a weight matrix W to estimate the distance by relevant component analysis (RCA). Athitsos et al. [34] proposed a method called BoostMap to estimate a distance that approximates a certain distance. Hertz's work [35] uses AdaBoost to estimate a distance function in a product space, whereas the weak classifier minimizes an error in the original feature space. All of these methods' focus is a selection of suitable distance from a

given set of distance measures. Our method aims at improving performance of a given distance measure.

3 Learning New Distance Measures

We first describe the classical setting of similarity retrieval. It applies to many retrieval scenarios like key word, document, image, and shape retrieval. Given is a set of objects $X = \{x_1, \ldots, x_n\}$ and a similarity function sim: $X \times X \rightarrow R^+$ that assigns a similarity value (a positive value) to each pair of objects.

We assume that x_1 is a query object (e.g., a query shape), and $\{x_2, \ldots, x_n\}$ is a set of known database objects (or a training set). Then, by sorting the values $sim(x_1, x_i)$ in decreasing order for $i = 2, \ldots, n$, we obtain a ranking of database objects according to their similarity to the query, i.e., the most similar database object has the highest value and is listed first. Sometimes, a distance measure is used in place of the similarity measure, in which case the ranking is obtained by sorting the database objects in increasing order, i.e., the object with the smallest value is listed first. Usually, the first $N \ll n$ objects are returned as the most similar to the query x_1.

As discussed above, the problem is that the similarity function sim is not perfect, and for many pairs of objects, it returns wrong results, although it may return correct scores for many pairs. We introduce now a method to learn a new similarity function sim_T that drastically improves the retrieval results of sim for the given query x_1.

Let $w_{i,j} = sim(x_i, x_j)$, for $i, j = 1, \ldots, n$, be a similarity matrix, which is also called an affinity matrix. We also define an $n \times n$ probabilistic transition matrix P as a rowwise normalized matrix w

$$P_{ij} = \frac{w_{ij}}{\sum_{k=1}^{n} w_{ik}}, \tag{1}$$

where P_{ij} is the probability of transit from node i to node j.

We seek a new similarity measure s. Since s only needs to be defined as similarity of other elements to query x_1, we denote $f(x_i) = s(x_1, x_i)$ for $i = 1, \ldots, n$. A key function is f and it satisfies

$$f(x_i) = \sum_{j=1}^{n} P_{ij}\ f(x_j). \tag{2}$$

Thus, the similarity of x_i to the query x_1, expressed as $f(x_i)$, is a weighted average over all other database objects, where the weights sum to one and are proportional to the similarity of the other database objects to x_i. In other words, we seek a function $f : X \rightarrow [0, 1]$ such that $f(x_i)$ is a weighted average of $f(x_j)$, where the weights are based on the original similarities $w_{i,j} = sim(x_i, x_j)$. Our intuition is that the new similarity $f(x_i) = s(x_1, x_i)$ will be large iff all points x_j that are very similar to x_i (large $sim(x_i, x_j)$) are also very similar to query x_1 (large $sim(x_1, x_j)$). Note that function f reaches equilibrium and an arbitrary function does not satisfy the equality.

The recursive expression (2) is closely related to PageRank. As stated in [36], a slightly simplified version of simple ranking R of a Web page u in PageRank is defined as

$$R(u) = \sum_{v \in B_u} \frac{c}{N_v} R(v), \tag{3}$$

where B_u is a set of pages that point to u, N_v is the number of links from page v, and c is a normalization factor.

Consequently, (2) differs from PageRank (3) by the normalization matrix, which is defined in (1) in our case, and is equal to $\frac{c}{N_v}$ for PageRank. The PageRank recursive equation takes a simple average over neighbors (a set of pages that point to a given Web page), while we take a weighted average over the original input similarities. Therefore, our equation admits recursive solution analog to the solution of the PageRank equation. Before we present it, we point out one more relation to recently proposed label propagation [16].

We obtain the solution to (2) by the following recursive procedure:

$$f_{t+1}(x_i) = \sum_{j=1}^{n} P_{ij}\ f_t(x_j) \tag{4}$$

for $i = 2, \ldots, n$, and we set

$$f_{t+1}(x_1) = 1. \tag{5}$$

We define a sequence of newly learned similarity functions restricted to x_1 as

$$sim_t(x_1, x_i) = f_t(x_i). \tag{6}$$

Thus, we interpret f_t as a set of normalized similarity values to the query x_1. Observe that $sim_1(x_1, x_i) = w_{1,i}$.

Steps (4) and (5) are used in label propagation, which is described in Section 4. However, our goal and our setting are different. Although label propagation is an instance of semi-supervised learning, we stress that we remain in the unsupervised learning setting. In particular, we deal with the case of only one known class, which is the class of the query object. This means, in particular, that label propagation has a trivial solution, in our case $\lim_{t \rightarrow \infty} f_t(x_i) = 1$ for all $i = 1, \ldots, n$, i.e., all objects will be assigned the class label of the query shape. Since our goal is ranking of the database objects according to their similarity to the query, we stop the computation after a suitable number of iterations $t = T$. As is the usual practice with iterative processes that are guaranteed to converge, the computation is halted if the difference $\|f_{t+1} - f_t\|$ becomes very slow, see Section 6.1 for details.

If the database of known objects is large, the computation with all n objects may become impractical. Therefore, in practice, we construct the matrix w using only the first $M < n$ most similar objects to the query x_1 sorted according to the original distance function sim. Our experimental results in Section 6.1 demonstrate that the replacement of the original similarity measure sim with sim_T results in a significant increase in the retrieval rate. The pseudocode of our algorithm is shown in Fig. 4.

4 Relation to Label Propagation

Label propagation belongs to a set of semi-supervised learning methods where it is usually assumed that class labels are known for a small set of data points. We have an extreme case of semi-supervised learning since we only

Input: The $n \times n$ row-wise normalized similarity matrix P with the query x_1, $f_1(x_1) = 1$, and $f_1(x_i) = 0$ for $i = 2, ..., n$.

while: $t < T$.

for $i = 2, ..., n$,

$f_{t+1}(x_i) = \sum_{j=1}^{n} P_{ij}\ f_t(x_j)$

end

$f_{t+1}(x_1) = 1$.

end

Output: The learned new similarity values to the query x_1: f_T.

Fig. 4. The pseudocode for the proposed algorithm.

assume that the class label of the query is known. Thus, we have only one class that contains only one labeled element being the query x_1. In our approach, we have a sequence of labeling functions $f_t : X \to [0, 1]$ with $f_0(x_1) = 1$ and $f_0(x_i) = 0$ for $i = 2, \ldots, n$, where $f_t(x_i)$ can be interpreted as probability that point x_i has the class label of the query x_1.

Label propagation is formulated as a form of propagation on a graph, where a node's label propagates to neighboring nodes according to their proximity. The key idea is that its label propagates "faster" along a geodesic path on the manifold spanned by the set of known shapes than by direct connections. While following a geodesic path, the obtained new similarity measure learns to ignore less relevant shape differences. Therefore, when learning is complete, it is able to focus on more relevant shape differences. We review now the key steps of label propagation and relate them to the proposed method introduced in Section 3.

Let $\{(x_1, y_1) \ldots (x_l, y_l)\}$ be the labeled data, $y \in \{1 \ldots C\}$, and $\{x_{l+1} \ldots x_{l+u}\}$ the unlabeled data, usually $l \ll u$. Let $n = l + u$. We will often use L and U to denote labeled and unlabeled data, respectively. The Label propagation supposes the number of classes C is known, and all classes are present in the labeled data [16]. A graph is created where the nodes are all the data points, and the edge between nodes i, j represents their similarity $w_{i,j}$. Larger edge weights allow labels to travel through more easily. Also, we define an $l \times C$ label matrix Y_L, whose ith row is an indicator vector for y_i, $i \in L : Y_{ic} = \delta(y_{i,c})$. The label propagation computes soft labels f for nodes, where f is an $n \times C$ matrix whose rows can be interpreted as the probability distributions over labels. The initialization of f is not important. The label propagation algorithm is as follows:

1. Initially, set $f_0(x_i) = y_i$ for $i = 1, \ldots, l$ and $f_0(x_j)$ arbitrarily (e.g., 0) for $x_j \in X_u$.
   Repeat until convergence:
2. Set $f_{t+1}(x_i) = \sum_{j=1}^{n} P_{ij}\ f_t(x_j)$, $\forall x_i \in X_u$.
3. Set $f_{t+1}(x_i) = y_i$ for $i = 1, \ldots, l$ (the labels of the labeled objects should be fixed).

In step 2, all nodes propagate their labels to their neighbors for one step. Step 3 is critical since it ensures persistent label sources from labeled data. Hence, instead of letting the initial labels fade way, we fix the labeled data. This constant push from labeled nodes helps to push the class boundaries through high-density regions so that they can settle in low-density gaps. If this structure of data fits the classification goal, then the algorithm can use unlabeled data to improve learning.

Let $f = \binom{f_L}{f_U}$. Since f_L is fixed to Y_L, we are solely interested in f_U. The matrix P is split into labeled and unlabeled submatrices

$$P = \begin{bmatrix} P_{LL} & P_{LU} \\ P_{UL} & P_{UU} \end{bmatrix}. \tag{7}$$

As proven in [16], the label propagation converges, and the solution can be computed in closed form using matrix algebra:

$$f_U = (I - P_{UU})^{-1} P_{UL} Y_L. \tag{8}$$

However, as the label propagation requires all classes to be present in the labeled data, it is not suitable for shape retrieval. As mentioned in Section 3, for shape retrieval, the query shape is considered as the only labeled datum and all other shapes are the unlabeled data. Moreover, the graph among all of the shapes is fully connected, which means the label could be propagated on the whole graph. If we iterate the label propagation infinite times, all of the data will have the same label, which is not our goal. Therefore, we stop the computation after a suitable number of iterations $t = T$.

5 The Affinity Matrix

In this section, we address the problem of the construction of the affinity matrix W. There are some methods that address this issue, such as local scaling [37], local liner approximation [25], and adaptive kernel size selection [38].

However, in the case of shape similarity retrieval, a distance function is usually defined, e.g., [1], [3], [4], [8]. Let $D = (D_{ij})$ be a distance matrix computed by some shape distance function. Our goal is to convert it to a similarity measure in order to construct an affinity matrix W. Usually, this can be done by using a Gaussian kernel:

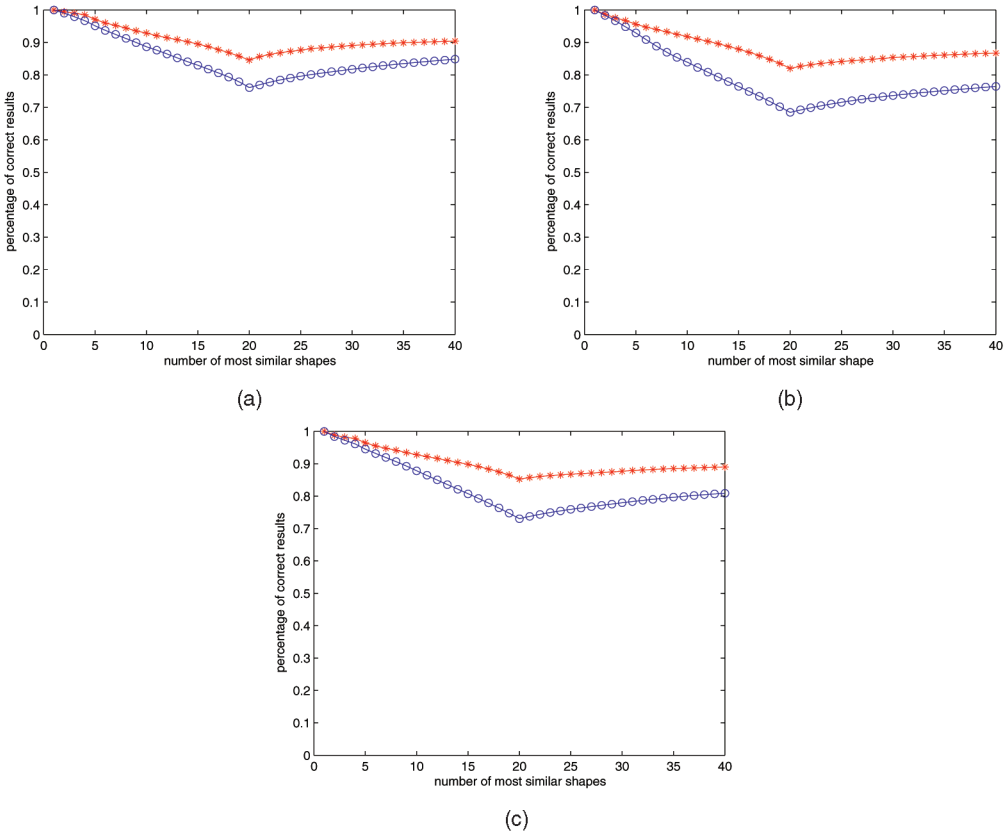

Fig. 5. (a) A comparison of retrieval rates between IDSC [3] (blue circles) and the result improved by the proposed method (red stars) for MPEG-7. (b) A comparison of retrieval rates between visual parts in [4] (blue circles) and the result improved by the proposed method (red stars) for MPEG-7. (c) A comparison of retrieval rates between Gen. Model [2] (blue circles) and the result improved by the proposed method (red circles) for MPEG-7.

$$w_{ij} = \exp\left(-\frac{D_{ij}^2}{\sigma_{ij}^2}\right). \tag{9}$$

Previous research has shown that the propagation results highly depend on the kernel size σ_{ij} selection [25]. In [23], a method to learn the proper σ_{ij} for the kernel is introduced, which has excellent performance. However, it is not learnable in the case of few labeled data. In shape retrieval, since only the query shape has the label, the learning of σ_{ij} is not applicable. In our experiment, we use an adaptive kernel size based on the mean distance to K-nearest neighborhoods [39]:

$$\sigma_{ij} = \alpha \cdot \text{mean}(\{knnd(x_i), knnd(x_j)\}), \tag{10}$$

where $\text{mean}(\{knnd(x_i), knnd(x_j)\})$ represents the mean distance of the K-nearest neighbor distance of the sample x_i, x_j and α is an extra parameter. Both K and α are determined empirically.

6 Experimental Results

In this section, we show that the proposed approach can significantly improve the performance of the existing shape retrieval, shape classification, and shape clustering methods.

6.1 Improving Shape Retrieval/Matching

6.1.1 Improving MPEG-7 Shape Retrieval

The IDSC [3] significantly improved the performance of shape context [1] by replacing the euclidean distance with shortest paths inside the shapes, and obtained the retrieval rate of 85.40 percent on the MPEG-7 data set. The proposed distance learning method is able to improve the IDSC retrieval rate to **91.61 percent**. For reference, Table 1 lists several reported results on the MPEG-7 data set. The MPEG-7 data set consists of 1,400 silhouette images grouped into 70 classes. Each class has 20 different shapes. The retrieval rate is measured by the so-called bulls-eye score. Every shape in the database is compared to all other shapes, and the number of shapes from the same class among the 40 most similar shapes is reported. The bulls-eye retrieval rate is the ratio of the total number of shapes from the same class to the highest possible number (which is $20 \times 1{,}400$). Thus, the best possible rate is 100 percent. From the retrieval rates collected in Table 1, we can clearly observe that our method made a significant progress on this database, and the second highest result is 87.70 percent obtained by Shape Tree [8].

In order to visualize the gain in retrieval rates by our method as compared to IDSC, we plot the percentage of correct results among the first k most similar shapes in Fig. 5a, i.e., we plot the percentage of the shapes from the same class among the first k-nearest neighbors for $k = 1, \ldots, 40$. Recall that each class has 20 shapes, which is why the curve increases for $k > 20$. We observe that the proposed method not only increases the bulls-eye score, but also the ranking of the shapes for all $k = 1, \ldots, 40$.

We use the following parameters to construct the affinity matrix: $\alpha = 0.25$ and the neighborhood size is $K = 14$. As

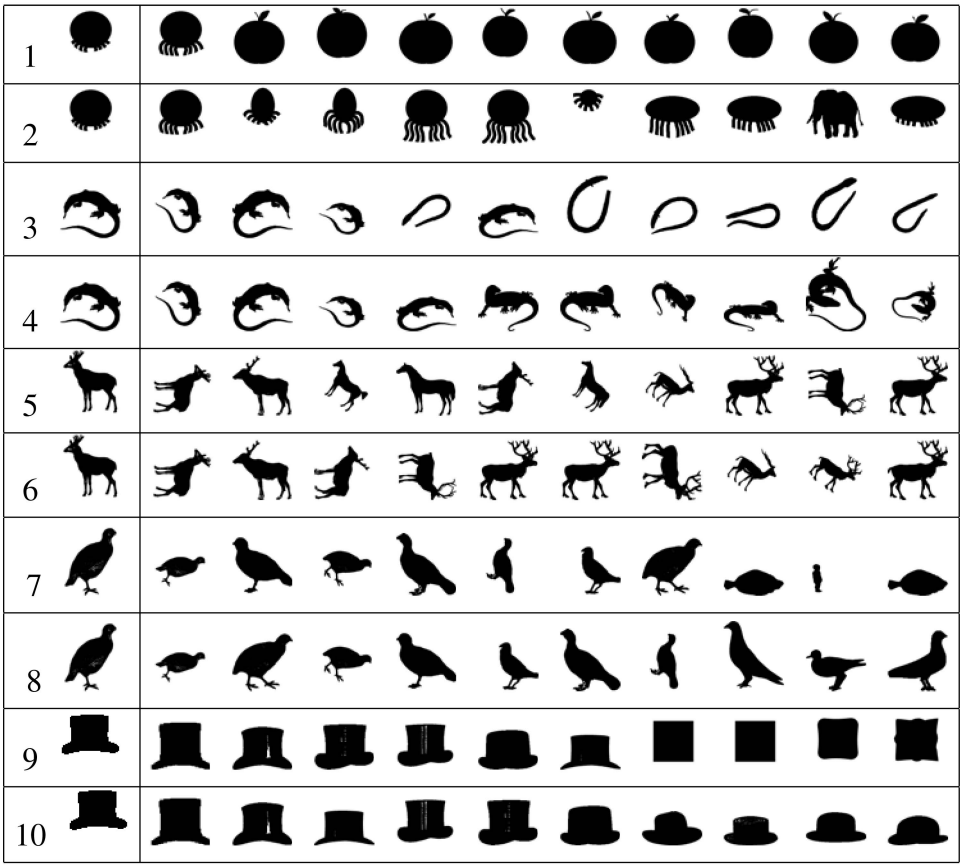

Fig. 6. The first column shows the query shape. The remaining 10 columns show the most similar shapes retrieved by IDSC (odd row numbers) and by our method (even row numbers).

stated in Section 3, in order to increase computational efficiency, it is possible to construct the affinity matrix for only part of the database of known shapes. Hence, for each query shape, we first retrieve 300 of the most similar shapes, and construct the affinity matrix W for only those shapes, i.e., W is of size 300×300 as opposed to a $1{,}400 \times 1{,}400$ matrix if we consider all MPEG-7 shapes. Then, we calculate the new similarity measure sim_T for only those 300 shapes. Here, we assume that all relevant shapes will be among the 300 most similar shapes. Thus, by using a larger affinity matrix, we could improve the retrieval rate but at the cost of computational efficiency. For each query, the average running time of our method on MEPG-7 is about 30 seconds in Matlab. For comparison, the running time of the original IDSC is about one minute for each query.

In addition to the statistics presented in Fig. 5, Fig. 6 illustrates also that the proposed approach improves the performance of IDSC. A very interesting case is shown in the first row, where, for IDSC only, one result is correct for the query octopus. It instead retrieves nine apples as the most similar shapes. Since the query shape of the octopus is occluded, IDSC ranks it as more similar to an apple than to the octopus. In addition, since IDSC is invariant to rotation, it confuses the tentacles with the apple stem. Even in the case of only one correct shape, the proposed method learns that the difference between the apple stems is very relevant, although the tentacles of the octopus exhibit a significant variation in shape. We restate that this is possible because the new learned distances are induced by geodesic paths in the shape manifold spanned by the known shapes. Consequently, the learned distances retrieve nine correct shapes. The only wrong result is the elephant, where the nose and legs are similar to the tentacles of the octopus.

As shown in the third row, six of the top 10 IDSC retrieval results of lizard are wrong since IDSC cannot discover the more relevant differences between lizards and sea snakes. All retrieval results are correct for the new learned distances since the proposed method is able to learn the less relevant differences between lizards and the more relevant differences between lizards and sea snakes. For the results of deer (fifth row), three of the top 10 retrieval results of IDSC are horses. Compared to it, the proposed method (sixth row) eliminates all of the wrong results so that only deer are in the top 10 results. It appears to us that our new method learned to ignore the less relevant small shape details of the antlers. Therefore, the presence of the antlers became a relevant shape feature here. The situation is similar for the bird and hat, with three and four wrong retrieval results, respectively, for IDSC, which are eliminated by the proposed method.

An additional explanation of the learning mechanism of the proposed method is provided by examining the count of the number of violations of the triangle inequality that involve the query shape and the database shapes. In Fig. 7a, the curve shows the number of triangle inequality violations after each iteration of our distance learning algorithm. The number of violations is reduced significantly after the first few hundred iterations. We cannot expect the number of violations to be reduced to zero since cognitively motivated shape similarity may sometimes require triangle inequality

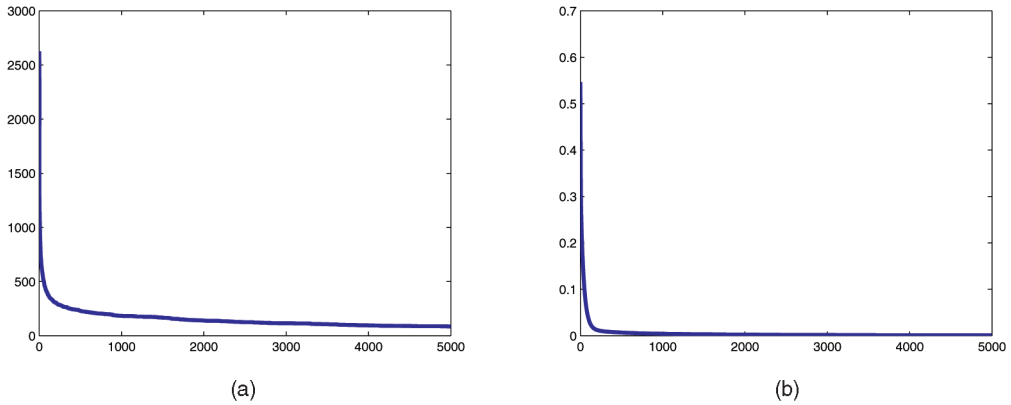

Fig. 7. (a) The number of triangle inequality violations per iteration. (b) Plot of differences $\|f_{t+1} - f_t\|$ as a function of t.

violations [17]. Observe that the curve in Fig. 7a correlates with the plot of differences $\|f_{t+1} - f_t\|$ as a function of t shown in (b). In particular, both curves decrease very slowly after about 1,000 iterations, and at 5,000 iterations, they are nearly constant. Therefore, we selected $T = 5{,}000$ as our stop condition. Since the situation is very similar in all of our experiments, we always stop after $T = 5{,}000$ iterations.

Besides the inner distance shape context [3], we also demonstrate that the proposed approach can improve the performance of visual parts shape similarity [4] and feature-driven generative model method [2]. We select these two methods since they are very different approaches than IDSC. In [4], in order to compute the similarity between shapes, first, the best possible correspondence of visual parts is established (without explicitly computing the visual parts). Then, the similarity between corresponding parts is calculated and aggregated. The settings and parameters of our experiment are the same as for IDSC as reported in the previous section except that we set $\alpha = 0.4$. The accuracy of this method has been increased from 76.45 percent to 86.69 percent on the MPEG-7 data set, which is more than 10 percent. This makes the improved visual part method one of the top scoring methods in Table 1. For feature-driven generative model method [2], the accuracy has been increased from 80.03 percent to 89.29 percent when we set $\alpha = 0.25$ and the other parameters are also the same as for IDSC. The detailed comparisons of the retrieval accuracy are given in Figs. 5b and 5c, respectively.

Besides the MPEG-7 data set, we also present experimental results on the Kimia's 99 data set [9]. The data set contains 99 shapes grouped into nine classes. In this data set, some images have protrusions or missing parts. Fig. 8 shows two sample shapes for each class of this data set. As the database only contains 99 shapes, we calculate the affinity matrix based on all of the shapes in the database. The parameters used to calculate the affinity matrix are: $\alpha = 0.25$ and the neighborhood size is $K = 4$. We changed the neighborhood size since the data set is much smaller than the MPEG-7 data set. The retrieval results are summarized as the number of shapes from the same class among the first top 1-10 shapes (the best possible result for each of them is 99). Table 2 lists the numbers of correct matches of several methods. Again, we observe that our approach could improve IDSC significantly, and it yields a nearly perfect retrieval rate, which is the best result in Table 2.

Fig. 8. Sample shapes from Kimia's 99 data set [9]. We show two shapes for each of the nine classes.

6.1.2 Improving Face Retrieval

We used a face data set from [51], where it is called *Face (all)*. It addresses a face recognition problem based on the shape of head profiles. It contains several head profiles extracted from side-view photos of 14 subjects. There exist large variations in the shape of the face profile of each subject, which is the main reason why we select this data set. Each subject is making different face expressions, e.g., talking, yawning, smiling, frowning, laughing, etc. When the pictures of subjects were taken, they were also encouraged to look a little to the left or right, randomly. At least two subjects had glasses that they put on for half of their samples. A few sample pictures are shown in Fig. 9.

The head profiles are converted to sequences of curvature values, and normalized to the length of 131 points, starting from the neck area. The data set has two parts, training with 560 profiles and testing with 1,690 profiles. The training set contains 40 profiles for each of the 14 classes. As reported in [51], we calculated the retrieval accuracy by matching the 1,690 test shapes to the 560 training shapes. We used a dynamic time-warping (DTW) algorithm with a warping window [52] to generate the distance matrix, and obtained the first nearest neighbor (1NN) retrieval accuracy of 88.9 percent By applying our distance learning method, we increased the 1NN retrieval accuracy to 95.04 percent. The best reported result in [51] has the first 1NN retrieval accuracy of 80.8 percent. The retrieval rate, which represents the percentage of the shapes from the same class (profiles of the same subject) among the first k-nearest neighbors, is shown in Fig. 10b.

The accuracy of the proposed approach is stable, although the accuracy of DTW decreases significantly when k increases. In particular, our retrieval rate for $k = 40$ remains high, 88.20 percent, while the DTW rate dropped to 60.18 percent. Thus, the learned distance allowed us to increase the retrieval rate by nearly 30 percent. Similar to

TABLE 2
Retrieval Results on Kimia's 99 Data Set [9]

Algorithm	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
SC [1]	97	91	88	85	84	77	75	66	56	37
Gen. Model [2]	99	97	99	98	96	96	94	83	75	48
Path Similarity [5]	99	99	99	99	96	97	95	93	89	73
Shock Edit [9]	99	99	99	98	98	97	96	95	93	82
IDSC [3]	99	99	99	98	98	97	97	98	94	79
Triangle Area [29]	99	99	99	98	98	97	98	95	93	80
Shape Tree [8]	99	99	99	99	99	99	99	97	93	86
Symbolic Rep. [30]	99	99	99	98	99	98	98	95	96	94
IDSC [3] **+ our method**	99	99	99	99	99	99	99	99	97	99

Fig. 9. A few sample images of the *Face (all)* data set.

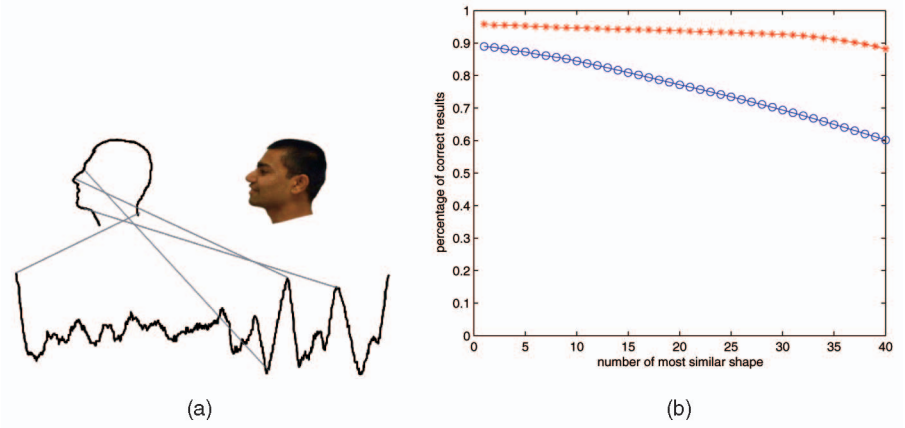

Fig. 10. (a) Conversion of the head profile to a curvature sequence. (b) Retrieval accuracy of DTW (blue circles) and the proposed method (red stars).

the above experiments, the parameters for the affinity matrix are $\alpha = 0.4$ and $K = 5$.

6.1.3 Improving Leaf Retrieval

The Swedish leaf data set comes from a leaf classification project at Linkoping University and Swedish Museum of Natural History [53]. Fig. 11 shows some representative examples. The data set contains isolated leaves from 15 different Swedish tree species, with 75 leaves per species. We followed the experimental setting for the Inner-Distance Shape Contexts used in [3], with 25 leaves of each species being used for training, and the other 50 leaves being used for testing. The 1NN accuracy reported in [3] is 94.13 percent, but the result we obtained with their software[1] is 91.2 percent. As shown in Fig. 12, the retrieval rate of the Swedish leaf is significantly improved by the proposed approach, especially, the 1NN recognition rate is increased from 91.2 to 93.8 percent. The parameters for the affinity matrix are $\alpha = 0.2$ and $K = 5$.

1. http://vision.ucla.edu/~hbling/code.

6.2 Improving 1NN Shape Classification

The k-nearest neighbor algorithm is among the simplest of all machine learning algorithms. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors. k is a positive integer, typically small. If $k = 1$, then the object is simply assigned to the class of its nearest neighbor. The proposed distance learning algorithm could improve the recognition rate of 1NN classification. The retrieval results of *Face (all)* and Swedish leaf databases have shown the

Fig. 11. Typical images from the Swedish leaf database [53], one image per species. Note that some species are quite similar, e.g., the first, third, and ninth species.

improvement. Besides, we divided the MPEG-7 data set into two sets: training set and testing set. For each class, 10 shapes are chosen as the training samples and the remaining 10 shapes are then used for testing. The results are shown in Table 3. We observe that the performance on these data sets has improved. The improvements on Swedish leaf and MPEG-7 are not so significant as on the Face data set, which might be related to the number of the training samples per class, which for the Swedish leaf and MPEG-7 data sets are much fewer than the Face data set. The parameters for all of the three data sets are the same as in the retrieval setting.

6.3 Improving Retrieval of Partially Occluded Shapes

It is well known that occlusion could potentially influence the performance of shape similarity approaches [54]. Since there is no standard test data set for occluded query shapes, we extended the MPEG-7 data set. In order to illustrate that the proposed approach has the potential to solve this problem, we selected several shapes and manually removed some of their parts. Then, the modified shapes are submitted as queries to the whole MPEG-7 data set for shape retrieval. The original distance matrix is obtained by IDSC. Fig. 13 shows the results of our experiments. The retrieval results in the odd rows are obtained by IDSC, and the results in the even rows are obtained by the proposed approach. It is clear that, although part occlusion influences the original IDSC a lot, our method can still improve the retrieval results. For example, we can interpret the results in the second row as 100 percent correct, while the original IDSC retrieved several incorrect shapes in the first row. We also observe that IDSC was unable to find the original fly from which the occluded fly query was made. Our method retrieved this fly as the first most similar shape to the query in the second row. The original retrieval IDSC results of the crown are even worse; only one result is correct, and most of the shapes belong to the class "fountain." Though the results of the proposed approach are not perfect, it still improves the performance a lot. Moreover, our query elephant is nearly half occluded, therefore, four of the top 10 results belong to the class "running person." The proposed approach could correctly retrieve all elephants. We also were able to obtain 100 percent correct retrieval for the occluded dog. The parameters for the part-occluded shapes are the same to them in the experiments on the whole MPEG-7 data set.

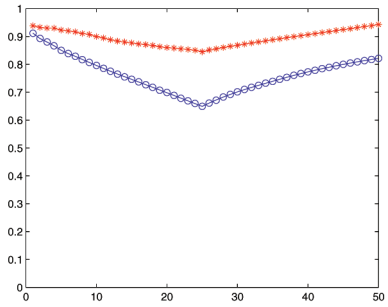

Fig. 12. Retrieval accuracy of IDSC (blue circles) and the proposed method (red stars).

6.4 Improving Shape Clustering

Besides the shape retrieval, the learned distance by the proposed approach can also be used for improving the performance of shape clustering. The difficulty of the shape clustering is also related to the shape similarity, which may have high variance of differences in the same class and, sometimes, small differences in different classes. Analogous to shape retrieval, the learned distance can improve the shape clustering results a lot.

In this paper, we choose Affinity Propagation [55] for shape clustering. Compared to other classic clustering algorithms, such as k-means, the main advantage of Affinity Propagation is that it does not require the prior knowledge of the number of clusters. As mentioned above, two shapes in the same class may be very different from each other and the distribution of differences is different for different classes. If the number of clusters is fixed before clustering, it may ruin the results because of the outliers. Therefore, Affinity Propagation is more suitable for the task of shape clustering as the outliers or unusual shapes which are totally different from other shapes in the same class will be automatically classified to separate clusters and will not affect other clusters. The details of Affinity Propagation are given in [55].

To evaluate the performance of the proposed approach on shape clustering, we applied the algorithm to three standard data sets: Kimia's 99 [9], shown in Fig. 8, and Kimia's 216 [9], which is a subset of the MPEG-7 data set. Fig. 14 shows two sample shapes for each class of Kimia's 216 shape data set. The third data set is the whole MPEG-7 data set. The score of the test is the ratio of the number of correct pairs of objects to the highest possible number of correct pairs and the best result would be 1. This score could represent the performance of shape clustering. If two shapes are clustered into one class and they have the same class label, it will be considered as a correct cluster result. Otherwise, if they do not have the same class label, the cluster result is wrong. Obviously, if two shapes are clustered into two different clusters, but they have the same true label, the proposed approach would not take it as a correct result. Finally, if the clustering algorithm could accurately cluster the MPEG-7 data set into 70 classes and each class contains the correct shapes, the score would be 1. Otherwise, it would be less than 1. The nearer to 1, the better the clustering algorithm. The IDSC [3] is used to obtain the input distance matrix for

TABLE 3
Results of 1NN Classification Improvement

	Original Distance	Learned Distance
Face (all)	88.9%	95.4%
Swedish leaf	91.2%	93.8%
MPEG-7 database	94.7%	95.7%

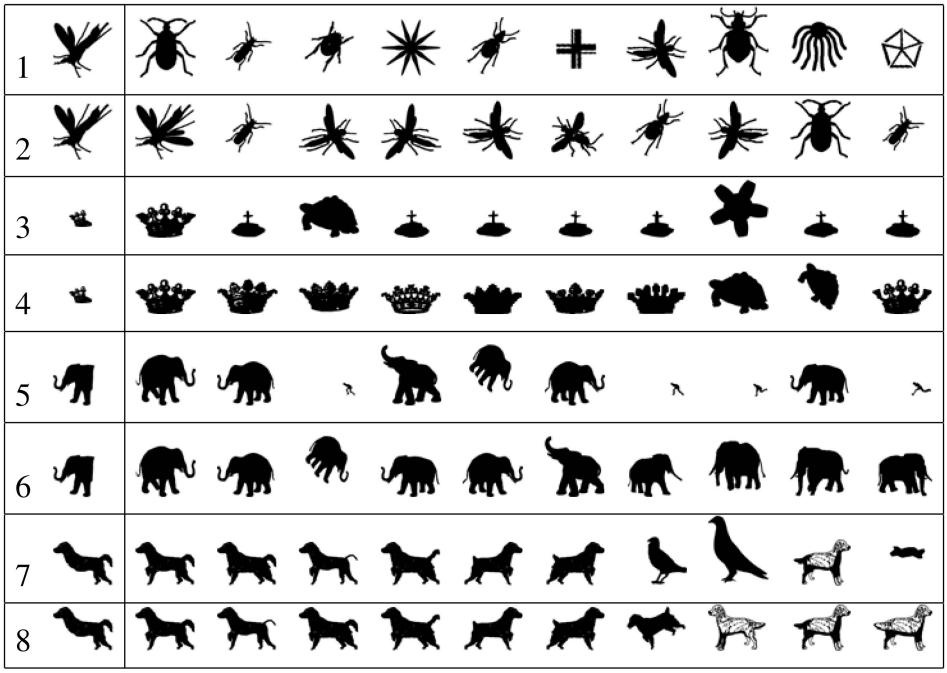

Fig. 13. The first column shows the query shape. The remaining 10 columns show the most similar shapes retrieved by IDSC (odd row numbers) and by our method (even row numbers).

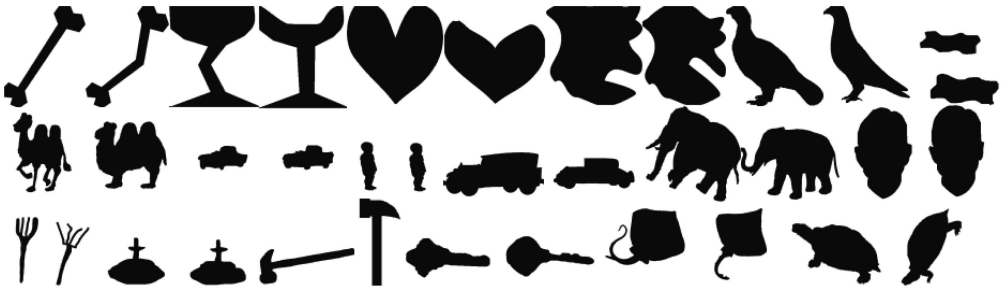

Fig. 14. Sample shapes from Kimia's 216 data set [9]. We show two shapes for each of the 18 classes.

each of three data sets. The shape clustering results based on the original distance by IDSC [3] and the learned distance by our algorithm are shown in Table 4 . Notice that the learned distance achieved a significant improvement on all data sets, and the numbers of the clusters are almost equal to the numbers of classes on Kimia's two data sets. We believe that some other methods, such as [15], can also be improved with our method. Here, we did not compare with the shape clustering method in [15] since they need to fix the number of cluster centers before clustering.

The number of iterations T is 1,000 for MPEG-7 data set and 300 for two Kimia's data sets. The parameters to calculate the affinity matrix for MPEG-7 are the same as for the retrieval. Besides, for Kimia's 99 shape database, the parameters are $K = 5$ and $\alpha = 0.33$, and for Kimia's 216 shape database, the parameters are $K = 7$ and $\alpha = 0.32$.

6.5 Choice of Parameters

There are three main free parameters for the proposed approach, α, K for affinity matrix, and the number of iterations T. In order to show the proposed approach is applicable in a reasonable range for parameters, we test the performance of the proposed approach on a range of parameter values. For T, as in Fig. 7b, it has been shown

TABLE 4
Clustering Results on the Kimia's 99 Data Set [9], Kimia's 216 Data Set [9], and MPEG-7 Data Set

	Kimia's 99 dataset		Kimia's 216 shape dataset		MPEG-7 dataset	
Number of Classes	9		18		70	
	Original Dist.	Learned Dist.	Original Dist.	Learned Dist.	Original Dist.	Learned Dist.
Number of Clusters	16	10	25	19	174	58
Accuracy	69%	95%	85%	97%	54%	86%

that, after several hundred iterations, the f is stable, which means that the approach is stable for T. Thus, we only consider the influence of the other two parameters. We randomly divide the whole MPEG-7 data set into two sets consisting of 700 shapes, in which each class contains 10 objects. One of them is chosen and bulls-eye score is calculated for each different pair of parameters. The new data set consists of 700 silhouette images grouped into 70 classes. Each class has 10 different shapes. For each query, the number of shapes from the same class among the 20 most similar shapes is reported. The bulls-eye retrieval rate is the ratio of the total number of shapes from the same class to the highest possible number (which is 10×700). The results are shown in Table 5.

In the above experiments, α ranges from 0.1 to 0.4 with an increase of 0.05 in each step and K ranges from 3 to 9 with an increase of 2 in each step. The best parameter is $\alpha = 0.25$ and $K = 7$. As the new data set is half of the MPEG-7, it is reasonable to double the K for the whole data set to $K = 14$ in the new data set. It is obvious that, in a proper range, the proposed approach is stable for the two parameters.

As manually choosing parameters is not proper for real application, we use a supervised learning framework to learn the parameters and obtain good results. We directly use the best learned parameters in the above experiments and then we perform the experiments on whole MPEG-7 data set based on these parameters.

7 Conclusion and Discussion

In this work, we adapted a graph transductive learning framework to learn new distances with the application to shape retrieval, shape classification, and shape clustering. The key idea is to replace the distances in the original distance space with distances induced by geodesic paths in the shape manifold. The merits of the proposed technique have been validated by significant performance gains in all presented experimental results. However, like semi-supervised learning, if there are too many outlier shapes in the shape database, the proposed approach may not be able to improve the results. Our future work will focus on addressing this problem. We also observe that our method is not limited to 2D shape similarity but can also be applied to 3D model retrieval, which will also be part of our future work.

TABLE 5
The Bulls-Eye Score for New Data Set
Based on Different Pairs of Parameters

	$K = 3$	$K = 5$	$K = 7$	$K = 9$
$\alpha = 0.1$	83.9%	88.11%	89.26%	89.84%
$\alpha = 0.15$	84.33%	88.67%	89.66%	90.31%
$\alpha = 0.2$	85.77%	90.29%	91.34%	91.84%
$\alpha = 0.25$	88.71%	92.17%	92.57%	92.56%
$\alpha = 0.3$	89.69%	91.16%	91.41%	91.16%
$\alpha = 0.35$	89.03%	90.39%	90.3%	90.2%
$\alpha = 0.4$	88.74%	89.99%	89.97%	89.84%

Acknowledgments

The authors would like to thank H. Ling for proving us his software for IDSC and the Swedish leaf database. The authors would also like to thank Eamonn Keogh for providing them the *Face (all)* data set. They also want to thank B.B. Kimia for proving his shape databases on the Internet. This work is supported in part by US National Science Foundation Grant No. IIS-0812118, US Department of Energy Grant No. DE-FG52-06NA27508, National Science Foundation of China Grant No. 60873127, and a grant from the PhD Programs Foundation of the Ministry of Education of China (No. 20070487028). This project is also in part supported by US National Institutes of Health Grant U54 RR021813 entitled Center for Computational Biology. This project is also supported in part by the US Office of Naval Research N000140910099 and National Science Foundation of China Grant 60903096. One of the authors (Xiang Bai) is supported by the MSRA Fellowship. The code and data used in this paper are available for free download at http://happyyxw.google pages.com/democodeeccv.

References

[1] S. Belongie, J. Malik, and J. Puzicha, "Shape Matching and Object Recognition Using Shape Contexts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 705-522, Apr. 2002.
[2] Z. Tu and A.L. Yuille, "Shape Matching and Recognition—Using Generative Models and Informative Features," *Proc. Eighth European Conf. Computer Vision*, pp. 195-209, 2004.
[3] H. Ling and D. Jacobs, "Shape Classification Using the Inner-Distance," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 286-299, Feb. 2007.
[4] L.J. Latecki and R. Lakämper, "Shape Similarity Measure Based on Correspondence of Visual Parts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1185-1190, Oct. 2000.
[5] X. Bai and L.J. Latecki, "Path Similarity Skeleton Graph Matching," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1282-1292, July 2008.

[6] B. Leibe and B. Schiele, "Analyzing Appearance and Contour Based Methods for Object Categorization," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2003.
[7] G. McNeill and S. Vijayakumar, "Hierarchical Procrustes Matching for Shape Retrieval," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2006.
[8] P.F. Felzenszwalb and J. Schwartz, "Hierarchical Matching of Deformable Shapes." *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2007.
[9] T.B. Sebastian, P.N. Klein, and B.B. Kimia, "Recognition of Shapes by Editing Their Shock Graphs," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 5, pp. 116-125, May 2004.
[10] K. Siddiqi, A. Shokoufandeh, S.J. Dickinson, and S.W. Zucker, "Shock Graphs and Shape Matching," *Int'l J. Computer Vision,* vol. 35, pp. 13-32, 1999.
[11] A. Shokoufandeh, D. Macrini, S. Dickinson, K. Siddiqi, and S.W. Zucker, "Indexing Hierarchical Structures Using Graph Spectra," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 7, pp. 1125-1140, July 2005.
[12] L. Gorelick, M. Galun, E. Sharon, R. Basri, and A. Brandt, "Shape Representation and Classification Using the Poisson Equation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 28, no. 12, pp. 1991-2005, Dec. 2006.
[13] I. Dryden, *Statistical Shape Analysis.* Wiley, 1998.
[14] F.L. Bookstein, "Principal Warps: Thin-Plate Splines and the Decomposition of Deformations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 11, no. 6, pp. 567-585, June 1989.
[15] A. Srivastava, S. H. Joshi, W. Mio, and X. Liu, "Statistic Shape Analysis: Clustering, Learning, and Testing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 4, pp. 590-602, Apr. 2005.
[16] X. Zhu, "Semi-Supervised Learning with Graphs," doctoral dissertation, Carnegie Mellon Univ., CMU-LTI-05-192, 2005.
[17] J. Vleugels and R. Veltkamp, "Efficient Image Retrieval through Vantage Objects," *Pattern Recognition,* vol. 35, no. 1, pp. 69-80, 2002.
[18] L.J. Latecki, R. Lakämper, and U. Eckhardt, "Shape Descriptors for Non-Rigid Shapes with a Single Closed Contour," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* pp. 424-429, 2000.
[19] X. Yang, X. Bai, L.J. Latecki, and Z. Tu, "Improving Shape Retrieval by Learning Graph Transduction," *Proc. European Conf. Computer Vision,* 2008.
[20] U. Brefeld, C. Buscher, and T. Scheffer, "Multiview Dicriminative Sequential Learning," *Proc. European Conf. Machine Learning,* 2005.
[21] N.D. Lawrence and M.I. Jordan, "Semi-Supervised Learning via Gaussian Processes," *Advances in Neural Information Processing Systems,* MIT Press, 2004.
[22] T. Joachims, "Transductive Inference for Text Classification Using Support Vector Machines," *Proc. Int'l Conf. Machine Learning,* pp. 200-209, 1999.
[23] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions," *Proc. Int'l Conf. Machine Learning,* 2003.
[24] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, and B. Scholkopf, "Learning with Local and Global Consistency," *Advances in Neural Information Processing Systems,* MIT Press, 2003.
[25] F. Wang, J. Wang, C. Zhang, and H. Shen, "Semi-Supervised Classification Using Linear Neighborhood Propagation," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2006.
[26] D. Zhou, J. Weston, A. Gretton, Q. Bousquet, and B. Scholkopf, "Ranking on Data Manifolds," *Advances in Neural Information Processing Systems,* MIT Press, 2003.
[27] S.T. Roweis and L.K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science,* vol. 290, pp. 2323-2326, 2000.
[28] X. Fan, C. Qi, D. Liang, and H. Huang, "Probabilistic Contour Extraction Using Hierarchical Shape Representation," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 302-308, 2005.
[29] N. Alajlan, M. Kamel, and G. Freeman, "Geometry-Based Image Retrieval in Binary Image Databases," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, no. 6, pp. 1003-1013, June 2008.
[30] M. Daliri and V. Torre, "Robust Symbolic Representation for Shape Recognition and Retrieval," *Pattern Recognition,* vol. 41, no. 5, pp. 1799-1815, 2008.
[31] J. Yu, J. Amores, N. Sebe, P. Radeva, and Q. Tian, "Distance Learning for Similarity Estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 30, pp. 451-462, Mar. 2008.
[32] E. Xing, A. Ng, M. Jordan, and S. Russell, "Distance Metric Learning with Application to Clustering with Side Information," *Advances in Neural Information Processing Systems,* pp. 505-512, MIT Press, 2003.
[33] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning Distance Functions Using Equivalence Relations," *Proc. Int'l Conf. Machine Learning,* pp. 11-18, 2003.
[34] V. Athitsos, J. Alon, S. Sclaroff, and G. Kollios, "Bootmap: A Method for Efficient Approximate Similarity Rankings," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2004.
[35] T. Hertz, A. Bar-Hillel, and D. Weinshall, "Learning Distance Functions for Image Retrieval," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* pp. 570-577, 2004.
[36] L. Page, S. Brin, R. Motwani, and T. Winograd, "The Pagerank Citation Ranking: Bringing Order to the Web," Stanford Digital Libraries Working Paper, 1998.
[37] L. Zelnik-Manor and P. Perona, "Self-Tuning Spectral Clustering," *Advances in Neural Information Processing Systems,* MIT Press, 2004.
[38] M. Hein and M. Maier, "Manifold Denoising," *Advances in Neural Information Processing Systems,* MIT Press, 2006.
[39] J. Wang, S.-F. Chang, X. Zhou, and T.C.S. Wong, "Active Microscopic Cellular Image Annotation by Superposable Graph Transduction with Imbalanced Labels," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2008.
[40] F. Mokhtarian, F. Abbasi, and J. Kittler, "Efficient and Robust Retrieval by Shape Content through Curvature Scale Space," *Image Databases and Multi-Media Search,* A.W.M Smeulders and R. Jain, eds., pp. 51-58, World Scientific, 1997.
[41] T. Sebastian, P. Klein, and B. Kimia, "On Aligning Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 1, pp. 116-125, Jan. 2003.
[42] C. Grigorescu and N. Petkov, "Distance Sets for Shape Filters and Shape Recognition," *IEEE Trans. Image Processing,* vol. 12, no. 7, pp. 729-739, Oct. 2003.
[43] G. McNeill and S. Vijayakumar, "2d Shape Classification and Retrieval," *Pro. Int'l Joint Conf. Artificial Intelligence,* 2005.
[44] B. Super, "Learning Chance Probability Functions for Shape Retrieval or Classification," *Proc. IEEE Workshop Learning at the IEEE CS Conf. Computer Vision and Pattern Recognition,* 2004.
[45] J. Xie, P. Heng, and M. Shah, "Shape Matching and Modeling Using Skeletal Context," *Pattern Recognition,* vol. 41, no. 5, pp. 1756-1767, 2008.
[46] F. Mokhtarian and M. Bober, *Curvature Scale Space Representation: Theory, Applications and MPEG-7 Standardization.* Kluwer Academic, 2003.
[47] E. Attalla and P. Siy, "Robust Shape Similarity Retrieval Based on Contour Segmentation Polygonal Multiresolution and Elastic Matching," *Pattern Recognition,* vol. 38, no. 12, pp. 2229-2241, 2005.
[48] T. Adamek and N. O'Connor, "A Multiscale Representation Method for Nonrigid Shapes with a Single Closed Contour," *IEEE Trans. Circuits and Systems for Video Technology,* vol. 14, no. 5, pp. 742-753, May 2004.
[49] A. Peter, A. Rangarajan, and J. Ho, "Shape L'âne Rouge: Sliding Wavelets for Indexing and Retrieval," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition,* 2008.
[50] B. Super, "Retrieval from Shape Databases Using Chance Probability Functions and Fixed Correspondence," *Int'l J. Pattern Recognition and Artificial Intelligence,* vol. 20, no. 8, pp. 1117-1137, 2006.
[51] E. Keogh, "UCR Time Series Classification/Clustering Page," http://www.cs.ucr.edu/~amonn/time_series_data/, 2009.
[52] C.A. Ratanamahatana and E. Keogh, "Three Myths About Dynamic Time Warping," *Proc. SIAM Int'l Conf. Data Mining,* pp. 506-510, 2005.
[53] O. Soderkvist, "Computer Vision Classification of Leaves from Swedish Trees," master's thesis, Linkoping Univ., 2001.
[54] A. Ghosh and N. Petkov, "Robustness of Shape Descriptors to Incomplete Contour Representations," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 11, pp. 1793-1804, Nov. 2005.
[55] B.J. Frey and D. Dueck, "Clustering by Passing Messages between Data Points," *Science,* vol. 315, pp. 972-976, 2007.

Xiang Bai received the BS and MS degrees in electronics and information engineering from Huazhong University of Science and Technology (HUST), Wuhan, China, in 2003 and 2005, respectively. He is currently working toward the PhD degree at HUST. From January 2006 to May 2007, he was with the Department of Computer Science and Information, Temple University. From October 2007 to October 2008, he was with the University of California, Los Angeles, as a joint PhD student. His research interests include computer graphics, computer vision, and pattern recognition. He is a student member of the IEEE.

Xingwei Yang received the BE degree in electronics and information engineering from Huazhong University of Science and Technology (HUST), Wuhan, China, in 2002. Currently, he is working toward the PhD degree in the Department of Computer and Information Science at Temple University. He is a student member of the IEEE.

Longin Jan Latecki received the PhD degree in computer science from Hamburg University, Germany, in 1992. He is an associate professor of computer science at Temple University, Philadelphia. His main research interests include shape representation and similarity, object detection and recognition in images, robot perception, data mining, and digital geometry. He has published more than 160 research papers and books. He is an editorial board member of *Pattern Recognition* and the *International Journal of Mathematical Imaging*. He received the Pattern Recognition Society Award together with Azriel Rosenfeld for the best article published in *Pattern Recognition* in 1998. He is the recipient of the 2000 Olympus Prize, the main annual award from the German Society for Pattern Recognition (DAGM). He is a senior member of the IEEE.

Wenyu Liu received the BS degree in computer science from Tsinghua University, Beijing, China, in 1986, and the MS and PhD degrees in electronics and information engineering from Huazhong University of Science and Technology (HUST), Wuhan, China, in 1991 and 2001, respectively. He is now a professor and associate dean of the Department of Electronics and Information Engineering, HUST. His current research interests include computer graphics, multimedia information processing, and computer vision. He is a member of the IEEE and the IEEE Systems, Man, and Cybernetics Society.

Zhuowen Tu received the ME degree from Tsinghua University and the PhD degree from Ohio State University. He is an assistant professor in the Laboratory of Neuroimaging (LONI), Department of Neurology, with a joint appointment in the Department of Computer Science, University of California, Los Angeles (UCLA). He is also affiliated with the UCLA Bioengineering Interdepartmental Program and the UCLA Bioinformatics Program. Before joining LONI, he was a member of the technical staff at Siemens Corporate Research and a postdoctoral fellow in the Department of Statistics at UCLA. He was a recipient of the David Marr Prize in 2003. He is a senior member of the IEEE.