Improving Shape Retrieval by Learning New Similarity Measures

Xingwei Yang[1], Xiang Bai[2,3], Longin Jan Latecki[1], Zhuowen Tu[3]

[1] Dept. of Computer and Information Sciences, Temple University, Philadelphia
{xingwei, latecki@temple.edu}
[2] Dept. of Electronics and Information Engineering, Huazhong University of Science and Technology, P. R. China
xiang.bai@gmail.com
[3] Lab of Neuro Imaging, University of California, Los Angeles
ztu@loni.ucla.edu

Abstract. Shape retrieval/matching is a very important topic in computer vision. The recent progress in this domain has been mostly driven by designing smart features 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 matching algorithms. It learns a better metric through graph transduction by propagating the model through existing shapes, in a way similar to computing geodesics in shape manifold. However, the proposed method does not require learning the shape manifold explicitly and it does not require knowing any class labels of existing shapes. 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%** on the MPEG-7 data set, which is the highest ever reported in the literature.

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 increasing the matching rate has been substantial in recent years. However, all of these approaches are focused on the nature of shape similarity. It seems to be an obvious statement that the more similar two shapes are, the smaller is their difference, which is measured by some distance function. Yet, this statement ignores the fact that some differences are relevant while other differences are irrelevant for shape similarity. It is not yet clear how the biological vision systems perform shape matching; it is clear that shape matching involves the high-level understanding of shapes. In particular, shapes in the same class can differ significantly because of distortion or non-rigid 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 all published shape distance measures [1–7]

are unable to address this issue. For example, based on the inner distance shape context (IDSC) [3], the shape in Fig. 1(a) is more similar to (b) than to (c), but it is obvious that shape (a) and (c) 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 irrelevant for this shape similarity judgment. On the other hand, much smaller shape details like the dog's ear and the shape of the head are of high relevance here. No matter how good a shape matching algorithm is, the problem of relevant and irrelevant 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 or a parameterized function.

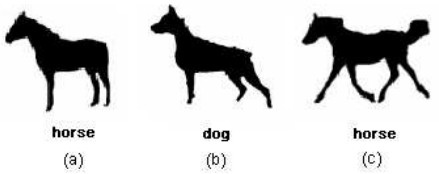


Fig. 1. Existing shape similarity methods incorrectly rank shape (b) as more similar to (a) than (c).

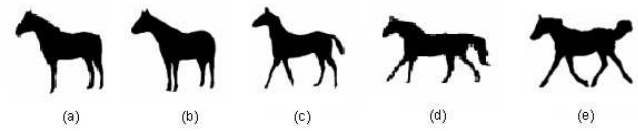


Fig. 2. A key idea of the proposed distance learning is to replace the original shape distance between (a) and (e) with a geodesic path in the manifold of know shapes, which is the path (a)-(e) in this figure.

In this paper, we propose to use 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 [8], which are time consuming to calculate. A better metric 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 [9] 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 datasets.

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 know 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 relevant and irrelevant 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. 1(a) as more similar to (c) than to (b). The reason is that the new distance function will replace the original distance (a) to (c) in Fig.1 with a distance induced by the shortest path between in (a) and (e) in Fig.2.

In more general terms, 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 shape A and shape 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 [10]. We propose a learning method to modify the original shape distance $d(A,C)$. 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 $d(A,C) > d(A,B_1) + \ldots + d(B_k,C)$, then our method learns a new $d'(A,C)$ such that $d'(A,C) \leq d(A,B_1) + \ldots + d(B_k,C)$. Since this path represents a minimal distortion morphing of shape A to shape C, we are able to ignore irrelevant shape differences, and consequently, we can focus on 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%** on part B of the MPEG-7 Core Experiment CE-Shape-1 data set [11], which is the highest ever bull's eye score reported in the literature. As the input to our method we used the IDSC, which has the retrieval rate of 85.40% 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 irrelevant.

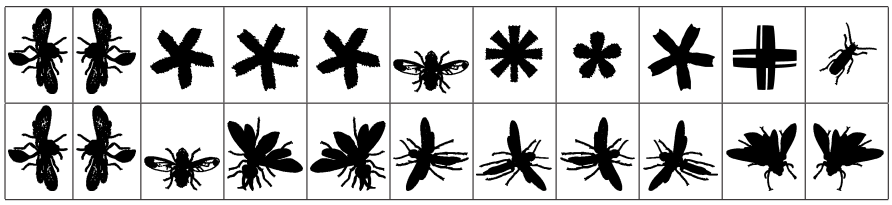


Fig. 3. The first column shows the query shape. The remaining 10 columns show the most similar shapes retrieved from the MPEG-7 data set. The first row shows the results of IDSC [3]. The second row shows the results of the proposed learned distance.

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 learning shape distances. 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 gives the experimental results to show the advantage of the proposed approach. Conclusion and discussion are given in Section 7.

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 [12], generative model [13], Transductive Support Vector Machine (TSVM) [14]. Recently there have been some promising graph based transductive learning approaches proposed, such as label propagation [9], Gaussian fields and harmonic functions (GFHF) [15], local and global consistency (LGC) [16], and the Linear Neighborhood Propagation (LNP) [17]. Zhou et al. [18] modified the LGC for the information retrieval. The semi-supervised learning problem is related to manifold learning approaches, e.g., [19].

The proposed method is inspired by the label propagation. The reason we choose the framework of label propagation is it allows the clamping of labels. 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[9]. 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). This allows us to clearly demonstrate the retrieval rate improvements obtained by the proposed method. Belongie et al. [1] introduced a novel local representation of shapes called shape context. Ling and Jacobs [3] modified the shape context by considering the geodesic distance of contour instead of the Euclidean distance, which improved the classification of articulated shapes. Latecki and Lakaemper [4] used visual parts for shape matching. In order to avoid problems associated with purely global or local methods, Felzenszwalb and Schwartz [5] also described a hierarchical matching method. Other hierarchical methods include the hierarchical graphical models in [20] and hierarchical procrustes matching [6].

There is a significant body of work on distance learning [21]. Xing et al. [22] propose estimating the matrix W of a Mahalanobis distance by solving a convex optimization problem. Bar-Hillel et al. [23] also use a weight matrix W to

estimate the distance by relevant component analysis (RCA). Athitsos et al. [24] proposed a method called BoostMap to estimate a distance that approximates a certain distance. Hertz's work [25] uses AdaBoost to estimate a distance function in a product space, whereas the weak classifier minimizes an error in the original feature space. All these methods' focus is a selection of suitable distance from a given set of distance measures. Our method aims at improving the retrieval 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 image, document, key word, 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 integer) to each pair of objects.

We assume that x_1 is a query object(e.g., a query shape), $\{x_2, \ldots, x_n\}$ 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 the 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 so that for many pairs of objects it returns wrong results, although it may return correct scores for most 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 define a sequence of labeling functions $f_t : X \rightarrow [0, 1]$ with $f_0(x_1) = 1$ and $f_0(x_i) = 0$ for $i = 2, \ldots, n$. We use the following recursive update of function f_t:

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

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

$$f_{t+1}(x_1) = 1. \tag{2}$$

We have only one class that contains only one labeled element being the query x_1. We define a sequence of new learned similarity functions restricted to x_1 as

$$sim_t(x_1, x_i) = f_t(x_i). \tag{3}$$

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

We iterate steps (1) and (2) until the step $t = T$ for which the change is below a small threshold. We then rank the similarity to the query x_1 with sim_T.

Our experimental results in Section 6 demonstrate that the replacement of the original similarity measure sim with sim_T results in a significant increase in the retrieval rate.

The steps (1) and (2) 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 $\lim_{t\to\infty} f_t(x_i) = 1$ 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 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.

4 Relation to Label Propagation

Label propagation is formulated as a form of propagation on a graph, where node's label propagates to neighboring nodes according to their proximity. In our approach we only have one labeled node, which is the query shape. 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 irrelevant shape differences. Therefore, when learning is complete, it is able to focus on 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[9]. A graph is created where the nodes are all the data points, the edge between nodes i, j represents their similarity $w_{i,j}$. Larger edge weights allow labels to travel through more easily. We define a $n \times n$ probabilistic transition matrix P as a row-wise normalized matrix w.

$$P_{ij} = \frac{w_{ij}}{\sum_{k=1}^{n} w_{ik}} \tag{4}$$

where P_{ij} is the probability of transit from node i to node j. Also define a $l \times C$ 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 a $n \times C$ 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 $f(x_i) = y_i$ for $i = 1, \ldots, l$ and $f(x_j)$ arbitrarily (e.g., 0) for $x_j \in X_u$
2. Repeat until convergence: Set $f(x_i) = \frac{\sum_{j=1}^{n} w_{ij} f(x_j)}{\sum_{j=1}^{n} w_{ij}}$, $\forall x_i \in X_u$ and set $f(x_i) = y_i$ for $i = 1, \ldots, l$ (the labeled objects should be fixed).

In step 1, all nodes propagate their labels to their neighbors for one step. Step 2 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 sub-matrices

$$P = \begin{bmatrix} P_{LL} & P_{LU} \\ P_{UL} & P_{UU} \end{bmatrix} \tag{5}$$

As proven in [9] the label propagation converges, and the solution can be computed in closed form using matrix algebra:

$$f_U = (I - P_{UU})^{-1} P_{UL} Y_L \tag{6}$$

However, as the label propagation requires all classes 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 data 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 [26], local liner approximation [17], and adaptive kernel size selection [27].

However, in the case of shape similarity retrieval, a distance function is usually defined, e.g., [1, 3–5]. 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:

$$w_{ij} = \exp(-\frac{D_{ij}^2}{\sigma_{ij}^2}) \tag{7}$$

Previous research has shown that the propagation results highly depend on the kernel size σ_{ij} selection [17]. In [15], 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 use an adaptive kernel size based on the mean distance to K-nearest neighborhoods:

$$\sigma_{ij} = C \cdot \text{mean}(\{knnd(x_i), knnd(x_j)\}) \tag{8}$$

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 C is an extra parameter. Both K and C are determined empirically.

6 Experimental Results

In this section, we show that the proposed approach can significantly improve retrieval rates of existing shape similarity methods.

6.1 Improving Inner Distance Shape Context

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% on the MPEG-7 data set. The proposed distance learning method is able to improve the IDSC retrieval rate to **91.00%**. For reference, Table 1 lists some of the reported results on the MPEG-7 data set. The MPEG-7 data set consists of 1400 silhouette images grouped into 70 classes. Each class has 20 different shapes. The retrieval rate is measured by the so-called bull's 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 bull's eye retrieval rate is the ratio of the total number of shapes from the same class to the highest possible number (which is 20×1400). Thus, the best possible rate is 100%.

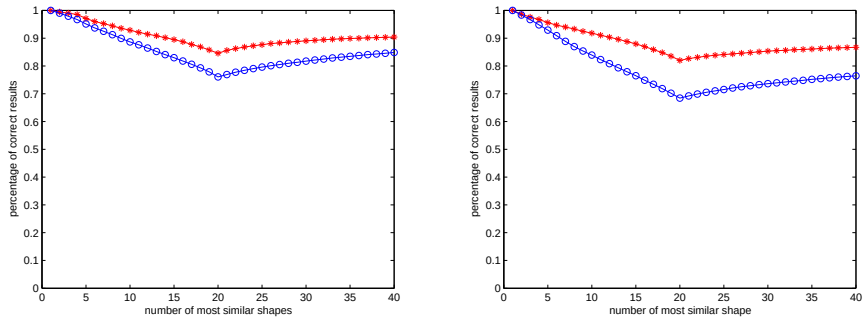
In order to visualize the gain in retrieval rates by our method as compared to IDSC, we plot the percentage of correct results among the first k most similar shapes in Fig. 4(a), i.e., we plot the percentage of the shapes from the same class among the first k-nearest neighbors for $k = 1, \ldots, 40$. 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 bull's 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: $C = 0.25$ and the neighborhood size is $K = 10$. As 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 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 1400×1400 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 can improve the retrieval rate but at the cost of computational efficiency.

Table 1. Retrieval rates (bull's eye) of different methods on the MPEG-7 data set.

Alg.	CSS [28]	Vis. Parts [4]	SC +TPS [1]	IDSC +DP [3]	Hierarchical Procrustes [6]	Shape Tree [5]	IDSC+DP + our method
Score	75.44%	76.45%	76.51%	85.40%	86.35%	87.70%	**91.00%**



(a) (b)

Fig. 4. (a) A comparison of retrieval rates between IDSC [3] (blue circles) and the proposed method (red stars). (b) A comparison of retrieval rates between visual parts in [4] (blue circles) and the proposed method (red stars).

In addition to the statistics presented in Fig. 4, Fig. 5 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 stem is relevant, although the tentacles of the octopuses 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 results 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 ten IDSC retrieval results of lizard are wrong. since IDSC cannot ignore the irrelevant 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 irrelevant differences between

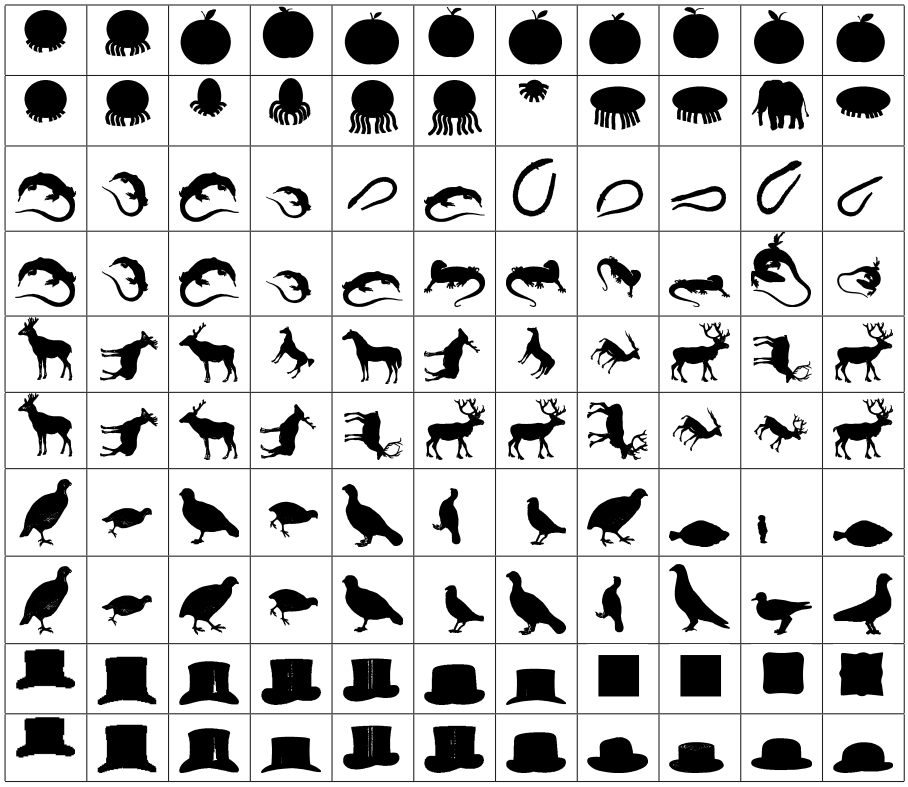
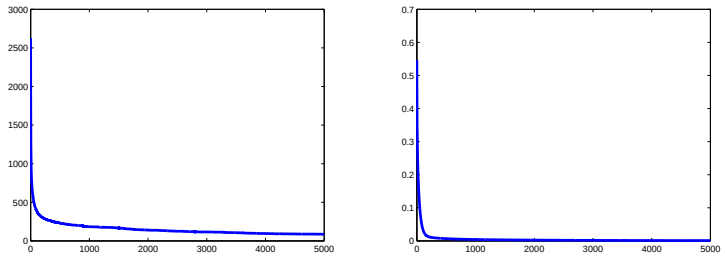


Fig. 5. The first column shows the query shape. The remaining 10 columns show the most similar shapes retrieved by IDSC (odd row numbers) and by our method (even row numbers).

lizards and the relevant differences between lizards and sea snakes. For the results of deer (fifth row), three of the top ten retrieval results of IDSC are horses. Compared to it, the proposed method (sixth row) eliminates all of the wrong results so that only deers are in the top ten results. It appears to us that our new method learned to ignore the irrelevant 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. 6(a), 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 violations [10]. Observe that the curve in

Fig. 6(a) 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 slow after about 1000 iterations, and at 5000 iterations they are nearly constant. Therefore, we selected $T = 5000$ as our stop condition. Since the situation is very similar in all our experiments, we always stop after $T = 5000$ iterations.



(a) (b)

Fig. 6. (a) The number of triangle inequality violations per iteration. (b) Plot of differences $||f_{t+1} - f_t||$ as a function of t.

Besides MPEG-7, We also present experimental results on the Kimia Data Set [29]. The database contains 99 shapes grouped into nine classes. As the database only contains 99 shapes, we calculate the affinity matrix based on all of the shape in the database. The parameters used to calculate the affinity matrix are: $C = 0.25$ 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 to 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.

Table 2. Retrieval results on Kimia Data Set [29]

Algorithm	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
SC [29]	97	91	88	85	84	77	75	66	56	37
Shock Edit [29]	99	99	99	98	98	97	96	95	93	82
IDSC+DP [3]	99	99	99	98	98	97	97	98	94	79
Shape Tree [5]	99	99	99	99	99	99	99	97	93	86
our method	99	99	99	99	99	99	99	99	97	99

6.2 Improving Visual Part Shape Matching

Besides the inner distance shape context [3], we also demonstrate that the proposed approach can improve the performance of visual parts shape similarity [4].

We select this method since it is based on very different approach 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 we set $C = 0.4$. The accuracy of this method has been increased from 76.45% to 86.69% on the MPEG-7 data set, which is more than 10%. This makes the improved visual part method one of the top scoring methods in Table 1. A detailed comparison of the retrieval accuracy is given in Fig. 4(b).

6.3 Improving Face Retrieval

We used a face data set from [30], 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 selecte 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. 7.

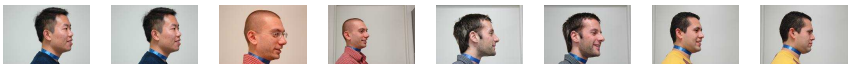
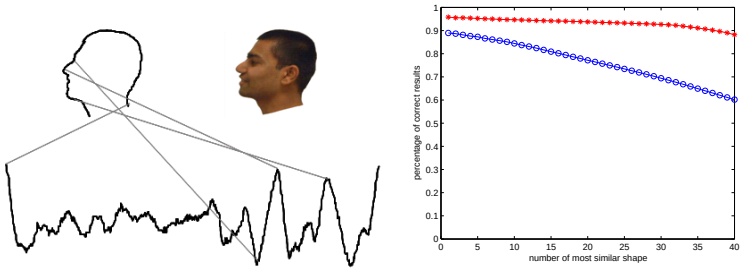


Fig. 7. A few sample image of the *Face (all)* data set.

The head profiles are converted to sequences of curvature values, and normalized to the length of 131 points, starting from the neck area. Fig. 8(a) illustrates how the profiles are transformed to sequences of curvature. The data set has two parts, training with 560 profiles and testing with 1690 profiles. The training set contains 40 profiles for each of the 14 classes. As reported on [30], we calculated the retrieval accuracy by matching the 1690 test shapes to the 560 training shapes. We used a dynamic time warping (DTW) algorithm with warping window [31] to generate the distance matrix, and obtained the 1NN retrieval accuracy of 88.9% By applying our distance learning method we increased the 1NN retrieval accuracy to 95.04%. The best reported result on [30] has the first nearest neighbor (1NN) retrieval accuracy of 80.8%. 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. 8(b). 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%, while the DTW rate dropped to 60.18%. Thus, the learned distance allowed us to increase the retrieval rate by nearly 30%. Similar to the above experiments, the parameters for the affinity matrix is $C = 0.4$ and $K = 5$.



(a) (b)

Fig. 8. (a) Conversion of the head profile to a curvature sequence. (b) Retrieval accuracy of DTW (blue circles) and the proposed method (red stars).

7 Conclusion and Discussion

In this work, we adapted a graph transductive learning framework to learn new distances with the application to shape retrieval. The key idea is to replace the distances in the original distance space with distances induces by geodesic paths in the shape manifold. The merits of the proposed technique have been validated by significant performance gains over the experimental results. However, like semi-supervised learning, if there are too many outlier shapes in the shape database, the proposed approach cannot 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 shape retrieval, which will also be part of our future work.

Acknowledgements

We would like to thank Eamonn Keogh for providing us the *Face (all)* dataset. This work was support in part by the NSF Grant No. IIS-0534929 and by the DOE Grant No. DE-FG52-06NA27508.

References

1. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Trans. PAMI **24** (2002) 705–522
2. Tu, Z., Yuille, A.L.: Shape matching and recognition - using generative models and informative features. In: ECCV. (2004) 195–209
3. Ling, H., Jacobs, D.: Shape classification using the inner-distance. IEEE Trans. PAMI **29** (2007) 286–299
4. Latecki, L.J., Lakämper, R.: Shape similarity measure based on correspondence of visual parts. IEEE Trans. PAMI **22(10)** (2000) 1185–1190
5. Felzenszwalb, P.F., Schwartz, J.: Hierarchical matching of deformable shapes. In: CVPR. (2007)
6. McNeill, G., Vijayakumar, S.: Hierarchical procrustes matching for shape retrieval. In: Proc. CVPR. (2006)
7. Bai, X., Latecki, L.J.: Path similarity skeleton graph matching. IEEE Trans. PAMI **30** (2008) 1282–1292

8. Srivastava, A., Joshi, S.H., Mio, W., Liu, X.: Statistic shape analysis: clustering, learning, and testing. IEEE Trans. PAMI **27** (2005) 590–602
9. Zhu, X.: Semi-supervised learning with graphs. In: Doctoral Dissertation. (2005) Carnegie Mellon University, CMU–LTI–05–192
10. Vleugels, J., Veltkamp, R.: Efficient image retrieval through vantage objects. Pattern Recognition **35 (1)** (2002) 69–80
11. Latecki, L.J., Lakämper, R., Eckhardt, U.: Shape descriptors for non-rigid shapes with a single closed contour. In: CVPR. (2000) 424–429
12. Brefeld, U., Buscher, C., Scheffer, T.: Multiview dicriminative sequential learning. In: ECML. (2005)
13. Lawrence, N.D., Jordan, M.I.: Semi-supervised learning via gaussian processes. In: NIPS. (2004)
14. Joachims, T.: Transductive inference for text classification using support vector machines. In: ICML. (1999) 200–209
15. Zhu, X., Ghahramani, Z., Lafferty., J.: Semi-supervised learning using gaussian fields and harmonic functions. In: ICML. (2003)
16. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Scholkopf., B.: Learning with local and global consistency. In: NIPS. (2003)
17. Wang, F., Wang, J., Zhang, C., Shen., H.: Semi-supervised classification using linear neighborhood propagation. In: CVPR. (2006)
18. Zhou, D., Weston, J., A.Gretton, Q.Bousquet, B.Scholkopf.: Ranking on data manifolds. In: NIPS. (2003)
19. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290** (2000) 2323–2326
20. Fan, X., Qi, C., Liang, D., Huang, H.: Probabilistic contour extraction using hierarchical shape representation. In: Proc. ICCV. (2005) 302–308
21. Yu, J., Amores, J., Sebe, N., Radeva, P., Tian, Q.: Distance learning for similarity estimation. IEEE Trans. PAMI **30** (2008) 451–462
22. Xing, E., Ng, A., Jordanand, M., Russell, S.: Distance metric learning with application to clustering with side-information. In: NIPS. (2003) 505–512
23. Bar-Hillel, A., Hertz, T., Shental, N., Weinshall, D.: Learning distance functions using equivalence relations. In: ICML. (2003) 11–18
24. Athitsos, V., Alon, J., Sclaroff, S., Kollios, G.: Bootmap: A method for efficient approximate similarity rankings. In: CVPR. (2004)
25. Hertz, T., Bar-Hillel, A., Weinshall, D.: Learning distance functions for image retrieval. In: CVPR. (2004) 570–577
26. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: NIPS. (2004)
27. Hein, M., Maier, M.: Manifold denoising. In: NIPS. (2006)
28. Mokhtarian, F., Abbasi, F., Kittler, J.: 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 (1997) 51–58
29. Sebastian, T.B., Klein, P.N., Kimia, B.: Recognition of shapes by editing their shock graphs. IEEE Trans. PAMI **25** (2004) 116–125
30. Keogh, E.: UCR time series classification/clustering page. (In: http://www.cs.ucr.edu/~ eamonn/time_series_data/)
31. Ratanamahatana, C.A., Keogh, E.: Three myths about dynamic time warping. In: SDM. (2005) 506–510