

Manifold Clustering of Shapes

Dragomir Yankov
University of California
Riverside CA 92507, USA
dyankov@cs.ucr.edu

Eamonn Keogh
University of California
Riverside CA 92507, USA
eamonn@cs.ucr.edu

Abstract

Shape clustering can significantly facilitate the automatic labeling of objects present in image collections. For example, it could outline the existing groups of pathological cells in a bank of cyto-images; the groups of species on photographs collected from certain aerials; or the groups of objects observed on surveillance scenes from an office building.

Here we demonstrate that a nonlinear projection algorithm such as Isomap can attract together shapes of similar objects, suggesting the existence of isometry between the shape space and a low dimensional nonlinear embedding. Whenever there is a relatively small amount of noise in the data, the projection forms compact, convex clusters that can easily be learned by a subsequent partitioning scheme. We further propose a modification of the Isomap projection based on the concept of degree-bounded minimum spanning trees. The proposed approach is demonstrated to move apart bridged clusters and to alleviate the effect of noise in the data.

1. Introduction

The effectiveness of object recognition and content-based image retrieval systems is highly dependent on the accurate identification of shapes. Features such as color, texture, positioning etc., though important, are insufficient to convey the information that could be obtained through shape analysis [4, 15, 21, 26]. In this work we propose an algorithm for clustering of 2D shapes. The method is invariant to basic geometric transformations, e.g. scale, shift, and most importantly, rotation. It is robust to noise, sparsity in the data and outliers that may bridge clusters representing more similar classes.

The shape clustering problem is of practical importance in many areas where image or video data collections are used. Labeling objects in such collections usually requires manually examining huge volumes of data. Consider for

example the field of cytology or the task of video data analysis. For many medical projects large banks of microscope cell images need to be processed (Figure 1 top)¹. The ability to cluster different types of cells (normal cells or cells corresponding to pathologies and diseases) without human supervision could considerably facilitate the medical analysis. Botanists, on the other hand, are interested in detecting and documenting the genotypes populating certain aerials (Figure 1 bottom). In tasks such as automatic surveillance or content exploration, the detection of different groups of objects that appear in scene sequences is usually required. Again, for these applications, an unsupervised shape clustering approach would be extremely beneficial.

Constructing a robust clustering algorithm is not trivial, as it should consider certain specifics of the shape data and the intuitively expected outcome. One natural requirement

¹The malaria images are part of the Hoslink medical databank: <http://www.hoslink.com/>, and the diatoms images are part of the collection used in the ADIAC project [7]

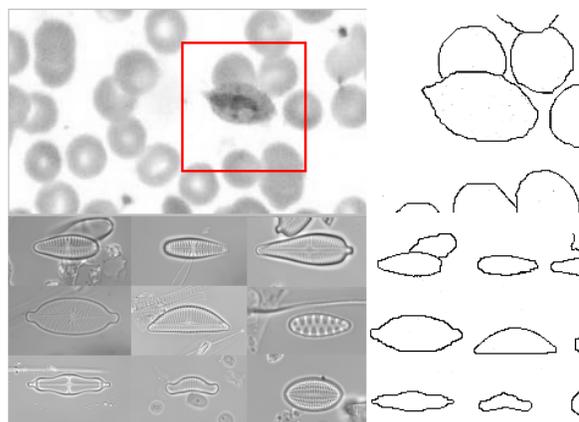


Figure 1: Cytology images. *Top:* *Plasmodium ovale* is one of the four malaria agents that can affect humans. The infected blood cells become larger with oval shape. *Bottom:* Diatoms are aquatic eukaryote plants, that appear in multiple shapes. Several types of diatoms can inhabit the same aerial

in shape recognition is to detect similarities invariant to basic geometric transformations. For example, in Figure 1 top, we would like to distinguish just two classes of cells - a normal and a pathological one, regardless of the many sizes and orientations that elements of each class could have. And while the scale and shift invariance are easily achievable with a suitable representation, the rotational invariance is much harder to deal with [14]. Important factors, that should be noted when dealing with rotational invariance, are how effective and efficient an algorithm is, as well as what level of control over the admissible rotations it provides. For example, in cytology analysis, we would like to consider all possible rotations when identifying the shape clusters, but in the case of handwritten character recognition we might need to confine the admissible rotations within the interval $[-15^\circ; 15^\circ]$. Otherwise we would detect as similar shapes that correspond to the digits “6” and “9” or the letters “b” and “q”.

Another challenge in the shape clustering task is introduced by the high dimensionality of the input space. Accurate shape representations generally require selecting a large number of features [12]. Additionally, there is significant amount of noise for many of the features, which is either related to the complexity of the studied shapes or is accumulated during certain preprocessing steps as image filtering and edge detection. Therefore, the resulting space is very high dimensional, with a lot of noise and possibly outliers. Clustering in such a space is practically meaningless, so a suitable dimensionality reduction should be applied.

A promising direction towards the outlined problems, relies in the fact that object data usually resides in some nonlinear embedding of the original space, that has a relatively low dimensionality [20, 24]. Nonlinear reduction techniques such as Isomap [24] or Locally Linear Embedding [20] are particularly suitable for projecting such data. Here we focus on the Isomap algorithm and demonstrate that it groups well shapes from equivalent classes, using a very low (two or three) dimensional representation. This suggests that shapes data are also isometric to some nonlinear embedding of the original space. Furthermore, as the classes tend to form compact, convex clusters, they are easy to learn with a subsequent partitioning algorithm, e.g the classical Expectation-Maximization (EM).

However, if different regions have different densities, or if there is considerable amount of noise, Isomap fails to reconstruct correctly the exact structure of the embedding. In sparse regions the embedding becomes disconnected, while in dense regions “short circuits” are formed between otherwise geodetically distant parts of the embedded surface [3]. As a result, some clusters representing elements from the same class are separated, while clusters representing different classes are often merged.

To project the shape data in cases of noise, bridged

or partially overlapping clusters, we introduce the idea of *degree-bounded Isomap*. The algorithm constructs a degree-bounded minimum spanning tree to approximate the underlying embedding. It is demonstrated to move further apart clusters corresponding to more similar classes and to decrease the effect of noise in the data.

The contributions of the paper can be summarized as:

1. The problem of clustering rotationally invariant shapes is studied and a robust approach for its solution is proposed.
2. An isometry between the shape space and a nonlinear low dimensional embedding is demonstrated, suggesting that nonlinear reduction algorithms should be preferred in learning from shape data for different tasks.
3. The question of Isomap’s topological stability is revised and a method is proposed that avoids the problem of having multiple disconnected components in the projection or forming short circuits between geodetically distant regions of the embedding.

The rest of this paper is organized as follows. In Section 2 we briefly review the shape recognition and manifold learning literature. Section 3 describes the selected representation and introduces the rotationally invariant metric, used for evaluating the shapes similarity. The proposed manifold clustering approach is described in Section 4. Section 5 provides an evaluation of the approach on several publicly available datasets. Section 6 concludes the discussion and outlines some directions for future research.

2. Related Work

A key factor in the efficiency of shape recognition systems is the selected shape representation. If the representation is not robust to noise, is ambiguous, or does not adapt to geometric transformations, then the clustering quality will be naturally poor. Here we briefly outline the possible shape representation techniques and point out some of their strengths and drawbacks. For more detailed information on the topic, we refer the reader to extensive surveys such as [8, 25, 30].

As outlined by Zhang et al. [30], the representation methods could roughly be divided into contour and region based. Region based methods extract features from the two dimensional image information, e.g. geometric moments, enclosed area or shape covering convex hulls. Some of the region based methods are computationally intensive and often require tracing the contour as well, so that better accuracy could be achieved. Others, such as the moment invariants, are not so robust to distortions and might be ambiguous if the shapes have more complex boundaries.

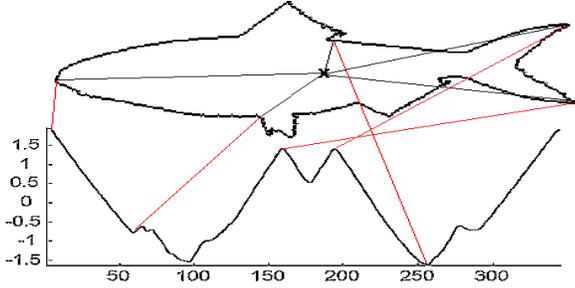


Figure 2: Global contour representation. The distance from the centroid of the shape to the contour points is measured and plotted as a time series.

Contour based representations construct a feature vector using only the points from the shape boundary. To obtain better efficiency, certain contour methods extract a very limited number of features that are either rotation invariant [17], or allow a corresponding alignment [5]. Both of the approaches, while suitable for particular settings, do not have good discriminative ability in the presence of noise and distortions [12, 30, 31]. For example, the alignment approach (also called *landmarking*) often uses the two principle axes of a shape to determine the features. This however is prone to ambiguities, as shapes from different classes may turn out to have similar axes [12].

In this work we adopt a *global* contour representation, in which the entire contour is converted to a 1D time series (see Figure 2, also Section 3). The representation is shift invariant, and by resampling all time series to the same length, or by using a warping metric to compare them, one could achieve invariance with respect to scale too. To obtain rotation invariance with this representation, all circular shifts for the time series need to be considered [1, 2], which renders a computationally intensive method. A potential way to deal with the problem is to consider the spectral information of the extracted time series by applying a Fourier transformation [10, 13, 27]. Charalampidis [10] and Klassen et al. [13] further utilize the transformation in partitioning and hierarchical shape clustering schemes. They demonstrate accuracy in performance, for cases when all rotations need to be considered. As we pointed out, however, we would like the approach to give us control over which rotations are admissible and which should be excluded.

Another drawback of a more complex representation, as the global contour one, is that many of the features might be irrelevant or noisy. To decrease the detrimental effect of such features, a suitable dimensionality reduction should be applied. Manifold approaches have been demonstrated to be particularly suitable for projecting image extracted data [6, 20, 22, 24]. In their clustering approach Srivastava et al. [23], also observe the manifold structure of the shape data. The authors implicitly assume a 2D structure for the

embedding and build a Markov model to partition the reconstructed 2D surface. Instead, we allow for a nonlinear reduction algorithm to automatically detect the best dimensionality for projecting the space. In particular, we focus on the Isomap [24] algorithm and demonstrate that clustering on the Isomap projection significantly outperforms clustering on the linearly projected data.

3. Measuring Shape Similarity

Formally, a shape representation technique transforms the shape space S into the vector space V through a particular mapping function ϕ :

$$\phi(\mathbf{s}_i) = \mathbf{v}_i \in V, \forall \mathbf{s}_i \in S$$

where ϕ constructs an ordered set of n descriptive features: $\mathbf{v}_i = (v_{i,1}, v_{i,2}, \dots, v_{i,n})$ [30]. The size n of the vectors depends on how many distinct features are necessary for the technique to describe the shapes in S . As we pointed out, many existing techniques target lower dimensionality in V in order to obtain better computational efficiency. The downside, however, is poor discriminative ability observed in multiple domains.

In the current work we adopt a global contour based representation, where every dimension $v_{i,j}$ corresponds to a point on the shape contour as illustrated in Figure 2. More precisely, here ϕ is the function that maps every contour point $s_{i,j}$ to the distance between this point and the shape’s mass center. This representation is known as *centroid-based* approach and has been introduced by Chang et al. [9]. The space V now consists of all time series extracted from shapes with the above mapping. The time series are further standardized to have mean zero and standard deviation one. The dimensionality of V is usually rather high, but we will demonstrate that a suitable nonlinear reduction in that space can preserve accurately the pairwise element similarities. It is interesting to note that, if several highly descriptive features do exist for a particular dataset, which is what landmarking relies upon, they will most likely be identified during the nonlinear reduction process automatically. If however such features are not present or are ambiguous, because of the shapes’ complexity or the presence of noise, the nonlinear reduction can still determine a suitable set of representative features on which to project V .

3.1 Rotationally Invariant Distance Measure

It is easy to see that due to the centroid-based representation, the selected mapping ϕ is shift invariant. Standardization alone, however, is insufficient to achieve scale invariance. Different object sizes or different image resolutions are likely to result in shape contours of variable

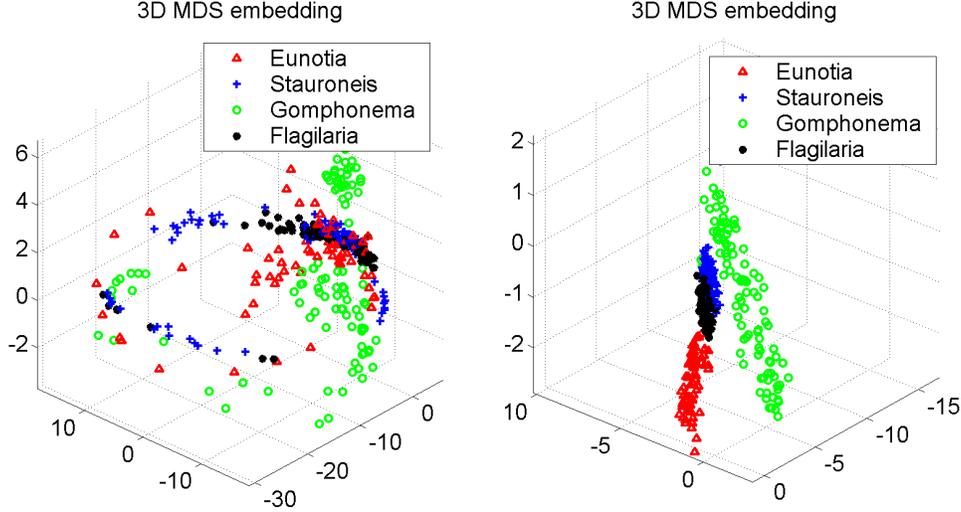


Figure 3: MDS projection of the diatoms dataset (Section 5.1). *Left:* the Euclidean distance fails to capture the similarities in the presence of rotations. *Right:* Using the rotationally invariant measure rd , elements of the same class are grouped together.

lengths too. Scale invariance could be achieved on the representation level by augmenting ϕ with a simple resampling step, during which all extracted time series are resized to the same length n . Another approach is to use a warping distance measure that will find the best alignment between the series and thus compensate for the different lengths. Using a warping distance, however, has been demonstrated to introduce little improvement over the basic resampling step followed by an Euclidean distance estimate [12].

A more difficult challenge is presented by rotation invariance. Even with the resampling step, ϕ is still unable to capture any possible similarities if rotations are present. As an example, consider Figure 3 left, which demonstrates the Multidimensional Scaling (MDS) projection for the diatoms dataset (see Section 5.1). Given a matrix of pairwise distances, MDS tries to compute the coordinates for the data that approximate best the information in the matrix. Once the coordinates are identified, the algorithm performs an eigen decomposition of the data, and projects it along the dimensions determined by the top few eigenvectors.

Here, the distances provided to MDS are the Euclidean distances between the resampled time series. The three dimensional projection was selected as the most accurate for the dataset. All elements have been uniformly randomly rotated, which leads to the spherical form of the projection (or circular if 2D projection was to be selected). The larger the angle of rotation, the further the examples from a single class are projected. At the same time, elements from different classes that should appear distant, are placed close to each other. As expected, the resulting clustering is essentially meaningless.

To approach the problem, it is important to notice that

all rotations of a shape s_i can be approximated by a suitably selected *circular shift* \mathbf{v}_i^r (also called *time series rotation*) of the original vector \mathbf{v}_i , where a circular shift is defined as:

$$\mathbf{v}_i^r = (v_{i,r+1}, v_{i,r+2}, \dots, v_n, v_{i,1}, \dots, v_{i,r}), r \in [0..n-1]$$

Detecting the clusters invariantly to shape rotations requires measuring the pairwise shape distances with respect to all possible rotations. In the vector space V this is equivalent to computing the minimum distance between all possible circular shifts of the two representative vectors:

$$rd(\mathbf{v}_i, \mathbf{v}_j) = \min_{0 \leq r \leq n-1} d(\mathbf{v}_i^r, \mathbf{v}_j) \quad (1)$$

In the following discussion the distance $d(\mathbf{v}_i, \mathbf{v}_j^r)$ is set to be the Euclidean distance between the corresponding vectors. By computing the minimum only over subintervals $[p, q] \subset [0, n-1]$, we can further restrict the admissible rotations. In this way, for example, we can avoid grouping together the shape representations of the handwritten digits “6” and “9”.

Using the newly introduced rotationally invariant distance, we apply again MDS to project the four class diatom dataset. The result is depicted in Figure 3 right. Unlike the non-rotated distance case, now the elements from the same class are grouped together. This demonstrates that a meaningful approach towards rotationally invariant shape clustering should consider the rd distance, rather than a simple application of the distance d . Yet, there is large overlap between some of the projected classes, which will deteriorate the accuracy of an arbitrary clustering scheme. In Section 4 we show that a nonlinear dimensionality reduction can mitigate this effect by separating better the projected clusters. But firstly, we look into a property of the distance measure

rd that will later be used in reconstructing the embedding, on which the non-linear projection operates.

3.2 Metric Properties of rd

An important property to be used in the proposed clustering approach is that, provided the distance d defines a metric over the vector space, the rotational distance rd defines a pseudo-metric² over V too.

The symmetry of rd follows directly from the fact that the distance d is symmetric. To show that rd satisfies the triangle inequality, without loss of generality, let us assume that the circular shift $r = p_0$ gives the minimum distance between vectors \mathbf{v}_i and \mathbf{v}_j , i.e. $rd(\mathbf{v}_i, \mathbf{v}_j) = d(\mathbf{v}_i^{p_0}, \mathbf{v}_j)$. Similarly, let $rd(\mathbf{v}_k, \mathbf{v}_j) = d(\mathbf{v}_k^{p_1}, \mathbf{v}_j)$, and $rd(\mathbf{v}_i, \mathbf{v}_k) = d(\mathbf{v}_i^{p_2}, \mathbf{v}_k)$. The following holds:

$$\begin{aligned} rd(\mathbf{v}_i, \mathbf{v}_j) + rd(\mathbf{v}_k, \mathbf{v}_j) &= d(\mathbf{v}_i^{p_0}, \mathbf{v}_j) + d(\mathbf{v}_k^{p_1}, \mathbf{v}_j) \\ &\geq d(\mathbf{v}_i^{p_0}, \mathbf{v}_k^{p_1}) \geq d(\mathbf{v}_i^{p_2}, \mathbf{v}_k) \\ &= rd(\mathbf{v}_i, \mathbf{v}_k) \end{aligned}$$

The first inequality above is true, because d is assumed to be metric, so it satisfies the triangle inequality. The second inequality follows from the fact that $d(\mathbf{v}_i^{p_2}, \mathbf{v}_k)$ is the distance between the optimal alignment of the two vectors \mathbf{v}_i and \mathbf{v}_k , while $(\mathbf{v}_i^{p_0}, \mathbf{v}_k^{p_1})$ also corresponds to some possible alignment of the same vectors.

4. Manifold Clustering of Shapes

We have already demonstrated that, when applied to the matrix of rotationally invariant distances, MDS provides a natural choice for a simple reduction of the time series space. Often, however, the data lies on a low dimensional nonlinear embedding (also called manifold), which linear projections cannot identify. The distances measured on the surface of the embedding are called *geodesic* distances. It may turn out that points that have large geodesic distance, and therefore should be treated as dissimilar, are very close in Euclidean sense. Linear projections operate in the Euclidean space and are inadequate to reconstruct the structure, implied by the geodesic distances. As a result, MDS might move apart otherwise similar (with respect to the manifold) elements or bring closer elements that come from different classes (again with respect to the manifold). This effect is the reason for the poor separability between the clusters demonstrated in Figure 3 right.

Vision data are often shown to reside on such nonlinear embedding [20, 24]. We demonstrate that shapes data also lie on an embedded space that could be reconstructed with

²Pseudo-metrics satisfy the triangle inequality and are symmetric. They do not guarantee, however, the positivity property, i.e. $d(v_i, v_j) = 0$ iff $v_i = v_j$

a suitable nonlinear dimensionality reduction technique. In particular, we study the performance of Isomap. After discussing briefly the specifics of the algorithm, we propose a modification for the cases when data are noisy, or when multiple bridging elements between different clusters deteriorate the stability of Isomap's projection.

4.1. Dimensionality Reduction With Isomap

To improve the chances for a subsequent clustering algorithm to detect any existing clusters, we need to preserve the compactness achieved by the MDS algorithm. For the elements of distinct clusters, however, the distances should be augmented and the clusters should be moved further apart. We obtain the effect by applying the Isomap projection algorithm.

For clarity of discussion a summary of the Isomap's steps, utilizing the rotationally invariant distance, is provided below:

1. Build the distance matrix $M_{m,m}$ for the data as follows: For all elements $v_i, i \in [1..m]$, if v_j is among the k -nearest neighbors to v_i , set $M(i, j) = rd(v_i, v_j)$. Otherwise set $M(i, j) = \text{inf}$.
2. In the graph defined by M , solve the all pairs shortest path problem (e.g. by applying Floyd-Warshall's algorithm). For all elements $v_i, i \in [1..m]$ set $M(i, j) = \text{shortest_path}(v_i, v_j)$
3. Run MDS on M obtained from the previous step.

The first step constructs a k -neighborhood graph as an approximation of the manifold surface and assigns small distances to pairs of elements that are very close on that surface. This is later preserved by the MDS reduction (step 3) in the projected space too. On the contrary, elements from different classes are less likely to be part of each others neighborhoods, and thus will be moved apart in the projection. The second step approximates the actual geodesic distances on the surface of the manifold with the shortest paths in the k -neighborhood graph. The 3D projection of the diatom dataset using Isomap is presented in Figure 4. A neighborhood of size $k = 16$, optimal for the projection (see Section 5.1), has been used. As seen from the figure, the clusters now are moved further apart, which supports the previous conjecture of an existing isometry between the shape space and a lower dimensional nonlinear embedding.

An important aspect to note is that the goodness of the geodesic distance approximation depends on the right choice of the neighborhood size k . Selecting k larger may

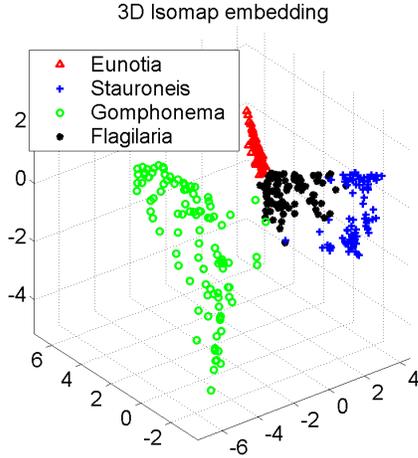


Figure 4: Isomap projection of the diatoms dataset. Clusters are better separated suggesting isometry between the shape space and a nonlinear embedding.

result in "short circuits" between distant elements, with respect to the manifold, similarly to the case when only Multidimensional Scaling is applied. In fact, in the asymptotic case when $k \rightarrow m$, Isomap is reduced simply to the MDS algorithm. On the other hand, selecting k too small may infer multiple disconnected components when building the neighborhood graph. In those cases MDS cannot reconstruct correctly the coordinates of the points. This results in a poor projection and thus in low clustering quality. And finally, depending on the sampling process, it may turn out that there is no one single k that is uniformly best across the whole dataset. For some samples a neighborhood of two elements may be most suitable, while for others, ten neighbors should be preferred. This dependence of the projection quality upon parameter k is referred to as *topological instability* of the Isomap algorithm. The impact, in the case of the shape clustering problem, can be observed in Figure 4, where the clusters of *Stauroneis* and *Flagilaria* diatoms are still not separated well, so that a clustering algorithm could discriminate them properly.

4.2 Stability Of The Isomap Projection

Balasubramanian et al. [3] argue that increasing the amount of noise in the data or having a comparatively sparse sample can cause multiple short circuits when Isomap tries to evaluate the correct geodesic distances. Softening the effect by selecting smaller neighborhood size k proves to be a poor solution, as in this case the constructed graph is split into multiple disconnected components. All distances between examples of two disconnected components are set by the algorithm to infinity and thus MDS cannot approximate correctly the coordinates for the elements. As a result, the

MDS projection deteriorates significantly.

The solution Tenenbaum suggests [3] is to optimize a tradeoff function between the percentage of elements omitted from the largest connected component and the variance in the distances, as computed on the manifold surface and in the Euclidean projection. Using large number of neighbors will decrease the percentage of omitted elements, but will also lead to improper evaluation of the right dimensionality. Decreasing k will lead to smaller variance, but will increase the percentage of not accounted elements. The globally optimal value of k , with respect to those two criteria, should be selected for the projection.

If, however, regions with different densities exist in the sample, the problem still persists. In denser regions the compromise globally optimal k might again lead to short circuits, while sparse regions will result in disconnected components. Wu et al. [28] suggest a different approach, in which the smallest distance edge between the disconnected components is identified and is added to the k -neighborhood graph. The authors demonstrate that the method is suitable for identifying multiple classes in data, where different classes reside in relatively distant regions on the manifold surface and even on different embeddings. The scheme is generalized by Yang [29], who argues that single edges between disconnected components do not reconstruct smoothly the surface of the manifold. He proposes building an l -connected graph in which for any possible split of the vertices into two groups there exist at least l edges connecting those groups.

Note that all of the above cases still lack flexibility in choosing the right neighborhood size k for individual graph nodes. Ideally we would like a method that defines stronger connectivity in dense regions of the data, but will loosen the requirement for the number of neighbors in sparser regions. Next we suggest one such approach based on degree- k -bounded minimum spanning trees.

4.3 Degree-bounded Isomap

The degree- k -bounded minimum spanning tree (k -MST) is an approximation of the MST of a connected graph, in which every vertex is allowed to have degree at most k [18]. The problem has emerged in the context of network modeling, where a network with minimum flow is needed but there is a limit imposed on the capacity of flow that can go through each node.

In the case of Isomap dimensionality reduction, we would like to approximate the k -neighborhood graph with a structure that will ensure connectivity between all vertices. For that purpose, a MST could be constructed. In a MST, however, the local information is not guaranteed to be preserved correctly. Many nodes can be of degree one, while few nodes (especially if residing in dense regions of the

data) may end up with some very high degree (e.g. forming stars). The k -MST avoids such undesired effects by restricting the degree of every vertex to be at most k . This also allows for the spanning tree to preserve better the locality around each node approximating the behavior of the k -neighborhood graph. In summary, the k -MST implicitly targets both of the problems outlined in the previous section, i.e. no disconnected components could be produced and there is no globally fixed neighborhood size k for all vertices.

Unfortunately, building the MST structure is a hard problem. In the case of $k = 2$, finding the k -MST is equivalent to the traveling salesman problem, which means that it is NP-complete. It has been demonstrated that constructing 3- and 4-MST is also NP-complete [18]. This may render the manifold representation with a k -MST impractical, yet we are going to approach the problem by making use of the metric properties derived earlier for the rotational distance rd .

Ravi et al. [19] prove that when the distance between the edges of a graph satisfies the triangle inequality, there exists a polynomial time algorithm for building an arbitrary k -MST with total cost at most twice the cost of the MST. We provide an outline of the algorithm below.

1. Build the MST for the data described by the distance matrix $M_{m,m}$ (e.g. we use Prim's algorithm). Select a root node \mathbf{r} for the tree.
2. Starting from \mathbf{r} do recursively for all non-leaf nodes \mathbf{v} : Assume that $(\mathbf{v}, \mathbf{v}_1), (\mathbf{v}, \mathbf{v}_2), \dots, (\mathbf{v}, \mathbf{v}_d)$, are the edges in increasing weight from \mathbf{v} to its children. If $degree(\mathbf{v}) > k$, replace the edges $(\mathbf{v}, \mathbf{v}_2), (\mathbf{v}, \mathbf{v}_3), \dots, (\mathbf{v}, \mathbf{v}_{d-k+2})$ with the edges $(\mathbf{v}_1, \mathbf{v}_2), (\mathbf{v}_1, \mathbf{v}_3), \dots, (\mathbf{v}_{d-k+1}, \mathbf{v}_{d-k+2})$

Step 2 above removes from \mathbf{v} as many edges to child nodes as necessary to keep its degree exactly k . The procedure is repeated recursively for all child nodes too, producing a degree- k -bounded tree. The fact that the cost of the edges is at most two times that of the MST follows from the ordering of the edges and the validity of the triangle inequality. For example, we have $rd(\mathbf{v}_1, \mathbf{v}_2) \leq rd(\mathbf{v}, \mathbf{v}_1) + rd(\mathbf{v}, \mathbf{v}_2) \leq 2rd(\mathbf{v}, \mathbf{v}_2)$, which implies that the cost of every added edge is at most twice the cost of the deleted one.

We will term the Isomap algorithm in which the k -neighborhood graph is replaced with a degree-bounded MST as *b-Isomap* (from bounded Isomap). The b-Isomap projection of the diatoms dataset is presented in Figure 5. In this example k has been set to 4.

The figure shows that the *Stauroneis* and *Flagilaria* classes are moved further apart as desired; the classes have less overlapping and just a few bridging elements between

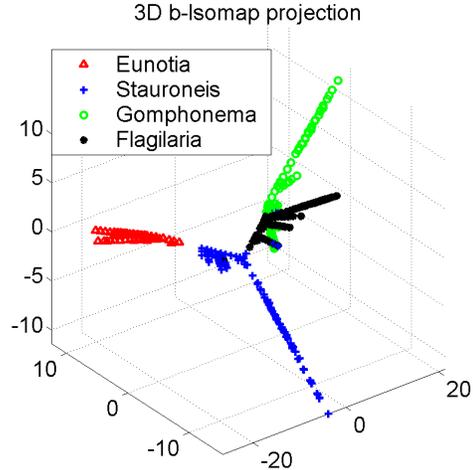


Figure 5: b-Isomap projection of the diatoms dataset. Sparser regions are loosely connected, which leads to better separability of bridged clusters such as the *Stauroneis* and *Flagilaria* ones

the clusters. The clusters are elongated, revealing that most of the elements from a certain class are represented by degree 2 nodes in the k -MST. One negative effect of the projection is that the clusters are not convex as in the case of the Isomap projection. Instead, there might be several elongated branches rooted as a subtree, representing elements from the same class. When multiple such branches exist, there is a higher chance that some of them will be assigned to different clusters degrading the quality of the clustering.

4.4. Shape Clustering Algorithm

We summarize the proposed clustering of rotationally invariant shapes in an end-to-end algorithm (see Algorithm 1). The algorithm builds on top of the introduced rotationally invariant distance metric rd , and uses a non-linear projection to discover the inherent dimensionality of the shape data at hand.

The clustering scheme can be used as both unsupervised or semi-supervised. In the presented evaluation we use a semi-supervised approach in which the cluster quality is checked upon the apriory known true labels of the elements. In an unsupervised procedure the mean square error with respect to the cluster centers could be tested instead.

As seen later from the experimental evaluation, which alternative to be used (Isomap or b-Isomap) depends on what prior information for the data we have. If there is no prior information, the Isomap approach should be preferred as the more consistent of the two (see Section 5). It should also be preferred when the existing classes of shapes are known to be relatively distinct and with small amount of noise. If the existing classes are believed to be comparatively similar (i.e. with large amount of overlap or bridging elements), or

Algorithm 1 Manifold Shape Clustering

procedure $[D_Labels] = \text{ShapeClustering}(D, C)$ **in:** D : dataset of converted to time series shapes;
 C : number of clusters**out:** D_Labels : cluster labels**Projection Step:**

- 1: $k = \text{Refine_k}(D, C)$; /*num.neighbors or degree*/
- 2: Alternative1: $D' = \text{Isomap}(D, k)$; /*projected data*/
- 3: Alternative2: $D' = \text{b-Isomap}(D, k)$; /*projected data*/

Clustering Step:

- 4: $IC = \text{Refine_Seeds}(D', C)$; /*initial seeds*/
 - 5: $D_Labels = \text{Cluster_EM}(D', IC, C)$;
-

there is large amount of noise in the data, then the b-Isomap projection should be applied. The projection parameter k , neighborhood size in the case of Isomap and maximum degree in the case of b-Isomap, can be selected using cross-validation (the approach used in our experiments) with subsamples of the data, or by applying the tradeoff optimization criterion discussed by Tenenbaum [3]. We decided to select a partitioning clustering algorithm, and EM in particular, as the clusters defined by Isomap and often by b-Isomap are convex and comparatively compact. The k-means algorithm in this setting is likely to fail due to the elongated structure of the clusters, while a k-medoid approach will have lower efficiency. The initial centers for the EM algorithm are selected as the best random seeds out of 10 runs again on subsamples of the data. An alternative approach is discussed by Fayyad et al. [11], which draws a set of very small subsamples and evaluates the centers that maximize the likelihood of the data based on those subsamples.

5. Experimental Evaluation

We test the performance of the two manifold approaches and the MDS projection on three publicly available datasets - diatoms, marine creatures and arrowheads. The datasets are selected to have different characteristics in terms of noise, sparsity in the data and similarity between the available classes. The actual labels of the samples are known and are used in measuring the accuracy of clustering.

The following evaluation procedure has been applied for all methods. A 10 times random sampling is used with 80% random subsamples from the original dataset. For each subsample, after the data is projected with the corresponding method, an EM clustering is performed. As EM relies on the correct initial center selection, it is repeated 10 times, each time with randomly selected centers. The accuracy from the best of the 10 clusterings is reported as accuracy of the method for this subsample.

5.1. Diatoms Dataset

Diatoms are eukaryote plants that live in aquatic environment. The dataset we use is collected as part of the ADIAC project [7]³. It contains approximately 360 images of diatoms from four classes - *Eunotia*, *Stauroneis*, *Gomphonema*, *Flagilaria* (see Figure 6). All time series for the dataset are resampled to a length of 345 points.

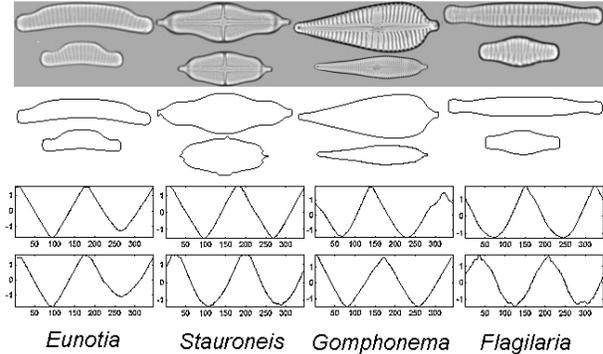


Figure 6: Diatoms dataset: original images - top, extracted shapes - middle, and time series representation - bottom. The four classes are relatively distinct with small similarities between some *Stauroneis* and *Flagilaria* diatoms.

To determine the number of dimensions that should be used in the projection, we measure the residual variance for any of the reduction methods as suggested by Tenenbaum et al. [24] (see Figure 7).

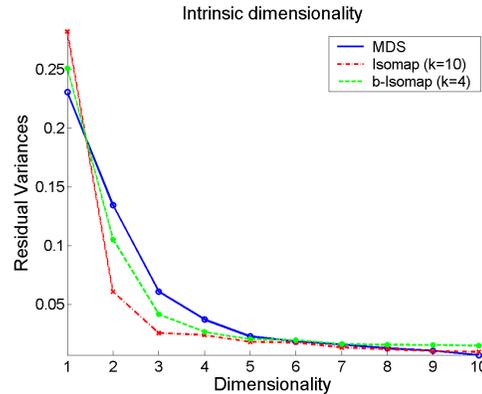


Figure 7: Detecting the intrinsic dimensionality of the data according to the three projections. The "elbow" of the curve points to the optimal number of dimensions to be used.

The "elbow" of the curve indicates the dimension beyond which adding new dimensions does not increase significantly the variance in the data, and thus no improvement

³The dataset can be downloaded from the DIADIST project web page: <http://rbg-web2.rbge.org.uk/DIADIST/>

in the projection can be expected. In the case of Isomap and b-Isomap, the variance also depends on the number of neighbors or the bounding degree parameter, still the structure of the curves remains similar for other values of the parameter too. The other datasets tested in the evaluation produced residual variance curves that differ in the speed with which they decay, but overall the best dimensions remain the same. Therefore, for all the datasets we tested the clustering accuracy, considering the 2D and 3D projections obtained by the three methods. The fact that two or three dimensions are descriptive for the data is not surprising, given the chosen representation. The time series usually have several extreme points, corresponding to those contour points that are closest/furthest from the shape centroid. It is the extreme points (global or in some case local extrema) that are usually detected as the most discriminative dimensions for the data.

Table 1: Clustering accuracy for the four class diatoms data.

Proj. Method	Dime-nsions	Parameter k	Average Accuracy (%)	Std (%)
MDS	3	N/A	62.3	5.2
Isomap	3	16	86.2	3.0
b-Isomap	3	4	83.0	3.6

Table 1 summarizes the details for the best accuracy obtained on the four class diatom dataset. Both nonlinear projections outperformed MDS with more than 20%. The best performance was obtained with the Isomap algorithm using three dimensional projection. The best number of neighbors for Isomap is relatively high (16), which implies that there is little noise and overlapping between the clusters (except for the *Stauroneis* and *Flagilaria* classes). The b-Isomap reduction performed slightly worse on average, and was also less consistent across the subsamples, which is represented by the larger variance in the accuracy (3.6%, column 5). An illustration of why b-Isomap’s projection was outperformed is presented for the two dimensional projection in Figure 8.

The figure compares the true labels (left graphs) with the labels as identified by the EM algorithm (right graphs). The ellipses drawn around each cluster have radii equal to twice the standard deviation along the corresponding dimension. Some of the *Stauroneis* and *Gomphonema* “branches” in the b-Isomap projection are incorrectly identified by EM to be part of the distribution for the *Flagilaria* class. The effect is not that strong for the Isomap projection because of the convex shape of the clusters.

We also compared the clustering accuracy between the two most overlapping classes, in which we additionally added to the time series Gaussian noise with mean zero and standard deviation 0.1.

The two dimensional projection in this case and the EM

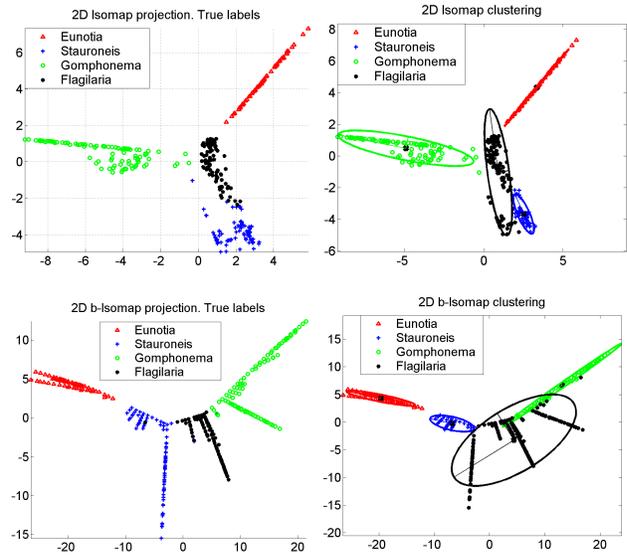


Figure 8: Clustering obtained from the 2D projections of Isomap (top) and b-Isomap (bottom). On the left are the true labels for the data, and on the right - the labels as computed by the EM algorithm. Note that the best projection is three dimensional, here two dimensions are shown for illustration only.

clustering are shown in Figure 9. The clusters produced by b-Isomap now have higher density, compared to the Isomap clusters, and are easier to detect with the EM algorithm. The sparsity in the Isomap clusters results from the multi-

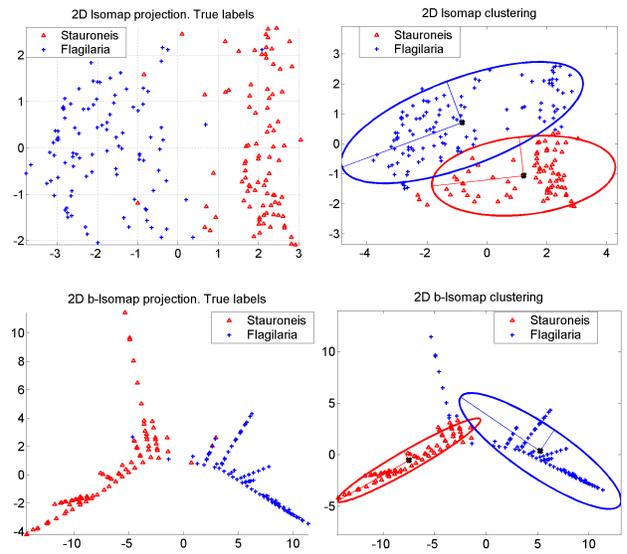


Figure 9: Clustering obtained from the 2D projections of Isomap (top) and b-Isomap (bottom) of the *Stauroneis* and *Flagilaria* classes only. On the left are the true labels for the data, and on the right - the labels as computed by the EM algorithm.

ple short circuits between the two similar classes. The clustering obtained with b-Isomap is almost perfect when three dimensional projection is used Table 2. Isomap performs better in three dimensions too (the best 2D accuracy for the algorithm is 89%), but still it is dominated by b-Isomap’s performance.

Table 2: Clustering accuracy for the two class diatom data.

Proj. Method	Dime-nsions	Parameter k	Average Accuracy (%)	Std (%)
MDS	3	N/A	90.2	1.3
Isomap	3	5	92.7	1.3
b-Isomap	3	3	98.3	0.9

The example demonstrates that significant improvement over Isomap can be achieved with the b-Isomap approach in the case of noise and when there is no strong distinction between the existing classes.

5.2. Marine Creatures Dataset

We used the prototype database of marine creatures discussed by Mokhtarian et al. [16]⁴. The images for four classes of different types of fish were selected, with each class containing 50 examples (Figure 10).

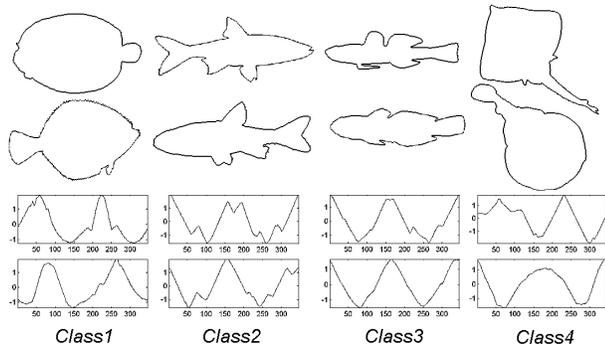


Figure 10: Marine creatures dataset: fish shapes - top, and their time series representation - bottom.

The time series extracted from the shape contours are again resampled to 345 time points (see Figure 10, bottom). For this dataset there is significant amount of within-class variability too. The contour of the shapes has more complex structure than that of the diatoms, which is reflected in the representation too. The time series contain more noise and there is no strong visual distinction between some elements from different classes. For example, *Class1* appears visually similar to *Class4*, while some elements of *Class2*

⁴The dataset can be downloaded from the SQUID project web page: <http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/>

are similar to elements of *Class3*. This similarity is a prerequisite for the formation of bridging elements between the projected clusters. In this sense, the dataset is similar to the two class diatom case. As expected, in this setting the b-Isomap projection is better than the one obtained with Isomap (Figure 11).

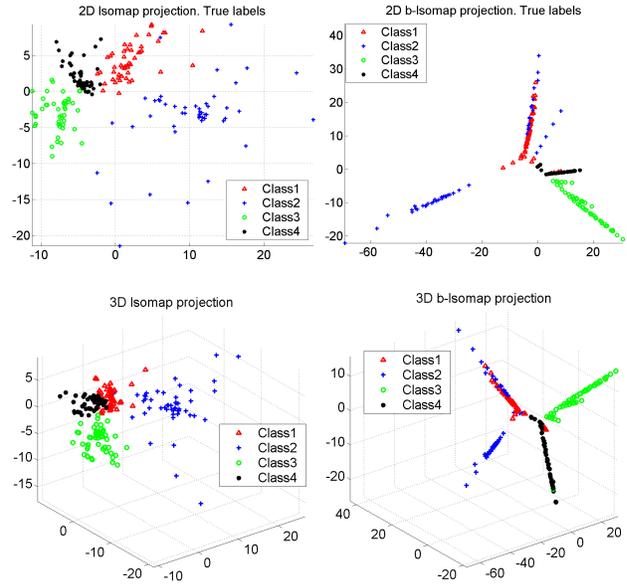


Figure 11: Marine creatures dataset: Isomap projection (left) compared to b-Isomap projection (right).

On average, clustering with b-Isomap is 2%-3% more accurate than clustering on the Isomap projection (Table 3). Again, the EM algorithm, applied on any of the nonlinear projections, significantly outperforms the clustering on the MDS projection. Yet, it is worth noting the larger variance of the nonlinear projections and especially of Isomap across the ten subsamples. This is partially due to the smaller number of examples (approximately 40 examples from each class are present in the subsamples), and partially to the larger amount of noise in the data.

Table 3: Clustering accuracy for the Marine creatures dataset.

Proj. Method	Dime-nsions	Parameter k	Average Accuracy (%)	Std (%)
MDS	2	N/A	61.0	3.0
Isomap	3	4	77.6	11.8
b-Isomap	3	4	80.0	7.8

5.3. Arrowheads Dataset

The dataset contains 600 images of randomly rotated arrowheads from a museum collection. The arrowheads are

representative of 6 distinct classes (Figure 12), with each class of 100 elements⁵. All time series have been resampled to 250 time points.

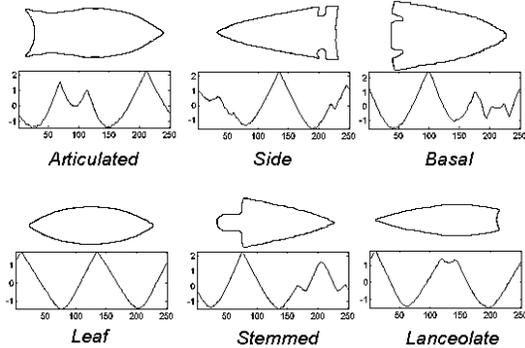


Figure 12: Arrowheads dataset: representative examples of the six classes and the corresponding time series representation.

The purpose of this evaluation was to test the behavior of the projections and the clustering algorithm when there is larger number of classes. Figure 13 demonstrates the 2D and 3D projections of Isomap and b-Isomap for the data.

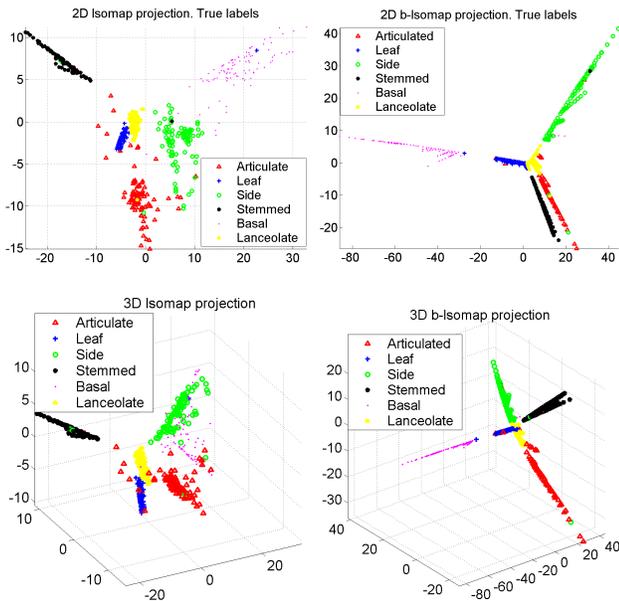


Figure 13: Arrowheads dataset: Isomap projection (left) compared to b-Isomap projection (right)

The performance of the three projections is summarized in Table 4. As the classes are distinct, and there is enough data from each class in the subsamples, Isomap reconstructs well the embedded structure and projects the classes in well defined sufficiently distant clusters. The two dimensional

⁵The dataset can be obtained upon request to dyankov@cs.ucr.edu

b-Isomap projection with bounding parameter $k = 6$ performed similarly well (85.1% accuracy). This is a result of the convexity of the clusters for this dataset. For most classes the degree-bounded spanning tree forms single long branches, which allows for all examples to be subsequently identified as coming from the same cluster. The b-Isomap clustering was also more consistent for the dataset, with twice smaller deviation as compared to Isomap. Both approaches again significantly outperformed the linear Multi-dimensional Scaling.

Table 4: Clustering accuracy for the arrowheads dataset.

Proj. Method	Dime-nsions	Parameter k	Average Accuracy (%)	Std (%)
MDS	3	N/A	75.6	5.7
Isomap	3	14	85.2	6.2
b-Isomap	2	6	85.1	3.1

6. Conclusions

We presented a method for clustering shape data invariantly of basic geometric transformations as shifting, scaling and most importantly rotation. The work demonstrated that an Isomap projection built on top of a rotationally invariant distance metric can detect correctly the intrinsic nonlinear embedding in which the shape examples reside. We have further introduced a modification of the Isomap algorithm, based on the concept of degree-bounded minimum spanning trees, that decreases the effect of bridging elements and noise in the data.

Our current efforts are targeted towards a hybrid solution that automatically combines the better features of both Isomap and b-Isomap. As we envision the approach, it should reconstruct the embedding by adaptively adjusting to the local densities in the data and at the same time preserve the compactness and convexity of the existing clusters.

References

- [1] T. Adamek and N. O’Connor. Efficient contour-based shape representation and matching. *Multimedia information retrieval*, pages 138–143, 2003.
- [2] E. Attalla and P. Siy. Robust shape similarity retrieval based on contour segmentation polygonal multiresolution and elastic matching. *Pattern Recognition*, 38(12):2229–2241, 2005.
- [3] M. Balasubramanian and E. Schwartz. The Isomap algorithm and topological stability. *Science*, 295, 2002.
- [4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

- [5] B. Bhanu and X. Zhou. Face recognition from face profile using dynamic time warping. pages IV: 499–502, 2004.
- [6] M. Breitenbach and G. Z. Grudic. Clustering through ranking on manifolds. In *Proceedings of the 22nd international conference on Machine learning (ICML)*, pages 73–80, 2005.
- [7] H. Buf, M. Bayer, S. Droop, R. Head, S. Juggins, S. Fischer, M. Bunke, J. Roerdink, J. Pech-Pacheco, G. Christobal, H. Shahbazkia, and A. Ciobanu. Diatom identification: A double challenge called ADIAC. In *Proceedings ICIAP*.
- [8] A. Cardone, S. Gupta, and M. Karnik. A survey of shape similarity assessment algorithms for product design and manufacturing applications. *J. Comput. Inf. Sci. Eng.*, 3(2):109–118, 2003.
- [9] C. Chang, S. Hwang, and D. Buehrer. A shape recognition scheme based on relative distances of feature points from the centroid. *Pattern Recognition*, 24(11):1053–1063, 1991.
- [10] D. Charalampidis. A modified k-means algorithm for circular invariant clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1856–1865, 2005.
- [11] U. Fayyad, C. Reina, and P. Bradley. Initialization of iterative refinement clustering algorithms. In *Knowledge Discovery and Data Mining*, pages 194–198, 1998.
- [12] E. Keogh, L. Wei, X. Xi, S. Lee, and M. Vlachos. LB_Keogh supports exact indexing of shapes under rotation invariance with arbitrary representations and distance measures. *VLDB*, 2006.
- [13] E. Klassen, A. Srivastava, M. Mio, and S. Joshi. Analysis of planar shapes using geodesic paths on shape spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(3):372–383, 2004.
- [14] D. Li and S. Simske. Shape retrieval based on distance ratio distribution. *HP Tech Report. HPL-2002-251*, 2002.
- [15] N. Logothetis and D. Sheinberg. Visual object recognition. *Annu. Rev. Neurosci.*, 19:577–621, 1996.
- [16] F. Mokhtarian, S. Abbasi, and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. In *Proc. International Workshop on Image Databases and Multimedia Search*, pages 35–42, 1996.
- [17] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transactions on Graphics*, 21(4):807–832, 2002.
- [18] C. Papadimitriou and V. Vazirani. On two geometric problems related to the travelling salesman problem. In *J. Algorithms*, pages 231–246, 1984.
- [19] R. Ravi, M. Marathe, S. Ravi, D. Rosenkrantz, and H. H. III. Many birds with one stone: Multi-objective approximation algorithms. In *ACM Symposium on Theory of Computing*, pages 438–447, 1993.
- [20] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.
- [21] Y. Rui, T. Huang, and S. Chang. Image retrieval: current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(4):39–62, 1999.
- [22] R. Souvenir and R. Pless. Manifold clustering. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV)*, pages 648 – 653, 2005.
- [23] A. Srivastava, S. Joshi, W. Mio, and X. Liu. Statistical shape analysis: Clustering, learning, and testing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4):590–602, 2005.
- [24] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323, 2000.
- [25] R. Veltkamp and L. Latecki. Properties and performance of shape similarity measures. *IFCS Conference: Data Science and Classification*, 2006.
- [26] R. Veltkamp and M. Tanase. Content-based image retrieval systems: a survey. *Technical Report*, 2001.
- [27] M. Vlachos, Z. Vagenas, P. Yu, and V. Athitsos. Rotation invariant indexing of shapes and line drawings. In *Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM)*, pages 131–138, 2005.
- [28] Y. Wu and K. Chan. An extended Isomap algorithm for learning multi-class manifold. In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*, volume 6, pages 3429–3433, 2004.
- [29] L. Yang. Building k-connected neighborhood graphs for isometric data embedding. *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 28:827–831, 2006.
- [30] D. Zhang and G. Lu. Review of shape representation and description techniques. *Pattern Recognition*, 37(1):1–19, 2004.
- [31] J. Zunic, P. Rosin, and L. Kopanja. Shape orientability. In *ACCV(2)*, pages 11–20, 2006.