Rock, Paper, and Scissors: extrinsic vs. intrinsic similarity of non-rigid shapes

Alexander M. Bronstein
Dept. of Computer Science
Technion, Haifa 32000, Israel
alexbron@ieee.org

Michael M. Bronstein
Dept. of Computer Science
Technion, Haifa 32000, Israel
bronstein@ieee.org

Ron Kimmel
Dept. of Computer Science
Technion, Haifa 32000, Israel
ron@cs.technion.ac.il

Abstract

This paper explores similarity criteria between non-rigid shapes. Broadly speaking, such criteria are divided into intrinsic and extrinsic, the first referring to the metric structure of the objects and the latter to the geometry of the shapes in the Euclidean space. Both criteria have their advantages and disadvantages; extrinsic similarity is sensitive to non-rigid deformations of the shapes, while intrinsic similarity is sensitive to topological noise. Here, we present an approach unifying both criteria in a single distance. Numerical results demonstrate the robustness of our approach in cases where using only extrinsic or intrinsic criteria fail.

1. Introduction

Most of us are familiar with the childhood Rock, Paper, Scissors game, in which the players bend their fingers in different ways to make the hand resemble one of the three objects (rock, represented by a closed fist; paper – by an open palm; or scissors – by the extended index and middle fingers). In the context of pattern recognition, this example demonstrates the difficulty of defining the similarity of non-rigid shapes: one may recognize the hand shapes as objects they intend to imitate, while others may say that all these "objects" are actually just deformations of the same hand. Using geometric terminology, the first similarity criterion is *extrinsic*, i.e., relates to the way the shape is embedded in the ambient space. The second criterion, on the other hand, is related to the *intrinsic* properties of the shape, described by its metric structure.

The problem of shape similarity is traditionally considered in computer vision, pattern recognition, and computational geometry literature, either implicitly or explicitly, from the extrinsic point of view (see, for example, [1, 11, 9]). A classical result for rigid object matching is the *iterative closest point* (ICP) method, introduced by Chen and Medioni [6] and Besl and McKay [2]. ICP methods try to find a Euclidean transformation between two shapes, minimizing an extrinsic distance between them, usually a variant of the Hausdorff distance. In some sense, one can think of ICP as finding the best possible rigid alignment between the shapes.

Intrinsic similarity was explored in the paper of Elad and Kimmel [7], who proposed a non-rigid shape recognition method based on Euclidean embeddings as a generalization of the previous work of Schwartz *et al.* [15]. The key idea is to map the metric structure of the shapes to a low-dimensional Euclidean space and compare the resulting images (called *canonical forms*) in this space. The canonical forms are computed using multidimensional scaling (MDS) [3]. One of the main disadvantages of this method is that the canonical forms can represent the intrinsic geometry of the shapes only approximately, as it is generally impossible to find an isometry between a non-flat surface and a flat Euclidean space. Mémoli and Sapiro [13] showed how the representation error can be theoretically avoided by using the Gromov-Hausdorff distance [8]. The Gromov-Hausdorff distance is invariant to isometries, which makes it a useful tool for intrinsic shape comparison. Mémoli and Sapiro showed an approximation of the Gromov-Hausdorff distance, related to the latter by a probabilistic bound. In a follow-up paper, Bronstein *et al.* [4] proposed a method for the computation of the Gromov-Hausdorff distance based on a numerical core similar to MDS, referred to as *generalized MDS* (GMDS).

The choice of whether to use intrinsic or extrinsic similarity depends significantly on the application. The drawback of extrinsic similarity is its sensitivity to non-rigid deformations. Using our example, a gesture of the hand can be extrinsically more similar to a rock or scissors rather than another hand. This make extrinsic criteria unsuitable for the analysis of non-rigid objects with a high degree of flexibility. The intrinsic criterion, on the other hand, is insensitive to deformations which can be approximated by isometries, since isometries preserve the metric structure of the shape. However, intrinsic similarity is sensitive to topological noise and can be problematic in cases where the objects are partially missing (for example, connecting the fingers of

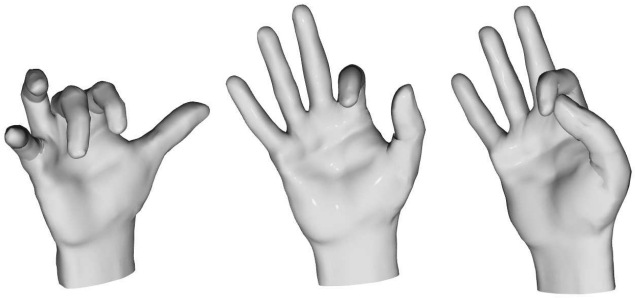


Figure 1. The difference between intrinsic and extrinsic similarity: right and middle shapes are extrinsically similar while being intrinsically dissimilar (the two shapes have different topology, since two fingers touch). Left and middle shapes are intrinsically similar (one can be obtained from the other by a near-isometric deformation), but extrinsically dissimilar.

the hand results in a significantly different intrinsic geometry, see Figure 1). Consequently, it appears that two semantically similar shapes can be substantially different both in their extrinsic and intrinsic geometries.

In this paper, we propose an approach for computing the similarity between non-rigid shapes trying to take the advantages of both intrinsic and extrinsic similarity criteria, while avoiding their shortcomings. The proposed similarity criterion is essentially a tradeoff between the extrinsic and the intrinsic criteria. As an illustration, one can think of fitting a rubber glove onto a hand. The extent to which the rubber surface is stretched represents the intrinsic geometry distortion. The fit quality, or in other words, how close the glove is to the hand surface, represents the extrinsic distance between the two objects.

The rest of this paper is organized as follows. In Section 2, we introduce the mathematical background and formulate the standard approaches to measuring intrinsic and extrinsic similarity. In Section 3, we present our approach for computing the joint intrinsic-extrinsic similarity. In Section 4, we show the numerical framework for computing this distance. Section 5 is dedicated to experimental results. Section 6 concludes the paper. Due to space limitations, derivations and technical details are omitted and will appear with additional experimental validation in the extended version of this paper.

2. Mathematical foundations

We model a non-rigid shape as a pair (X, d_X), where X is a two-dimensional smooth compact connected and complete Riemannian surface (possibly with boundary) embedded into $\mathbb{R}^3$, and $d_X : X \times X \to \mathbb{R}$ is the *geodesic metric* measuring the lengths of the shortest paths on the manifold, induced by the Euclidean length structure. For brevity of notation, we will write simply X, implying (X, d_X).

We broadly refer to properties described in terms of the metric d_X as to the *intrinsic geometry* of X, and to properties associated with the restriction of the Euclidean metric $d_{\mathbb{R}^3}|_X$ as the *extrinsic geometry*. From the intrinsic point of view, two objects X and Y are similar if the metrics d_X and d_Y are sufficiently close to each other. Formally, we say that X and Y are *ϵ-isometric* if there exists an *ϵ-surjective* map $\varphi : X \to Y$ (i.e., $d_Y(y, \varphi(X)) \leq \epsilon$ for all $y \in Y$), which has a *distortion*

$$\operatorname{dis} \varphi \quad = \quad \sup_{x,x' \in X} |d_X(x, x') - d_Y(\varphi(x), \varphi(x'))| = \epsilon.$$

Such a φ is called an *ϵ-isometry*. A 0-isometry is called simply an *isometry*, and two objects related by such a map are termed isometric. Isometric shapes are indistinguishable in terms of intrinsic geometry. An isometry from X to itself is called a *self-isometry*. The collection of all the self-isometries forms a group with the function composition operator and is referred to as the *isometry group*, denoted by $\operatorname{Iso}(X)$. Self-isometries express the intrinsic symmetries of an object.

Isometries are very different from ϵ-isometries. Particularly, an isometry is always bi-Lipschitz continuous [5], which is not necessarily true for an ϵ-isometry. If we relax the requirement of ϵ-surjectivity by demanding that φ has only $\operatorname{dis} \varphi \leq \epsilon$, we refer to such φ as an *ϵ-isometric embedding*.

2.1. Extrinsic similarity

The basic tool in extrinsic shape comparison is the *Hausdorff distance*, measuring the distance between two sets of points X and Y in $\mathbb{R}^3$,

$$d_{\mathrm{H}}^{\mathbb{R}^3}(X, Y) \quad = \quad \max\left\{ \sup_{x \in X} d_{\mathbb{R}^3}(x, Y), \sup_{y \in Y} d_{\mathbb{R}^3}(y, X) \right\},$$

where $d_{\mathbb{R}^3}(y, X) = \inf_{x \in X} \|y - x\|_2$ denotes the distance between the set X and the point y. A non-symmetric version of the Hausdorff distance,

$$d_{\mathrm{NH}}^{\mathbb{R}^3}(X, Y) \quad = \quad \sup_{x \in X} d_{\mathbb{R}^3}(x, Y), \tag{1}$$

is often preferred since it allows for partial comparison of shapes. The Hausdorff distance (both its symmetric and non-symmetric versions) regards the objects as sets of points in $\mathbb{R}^3$, is completely extrinsic, and consequently, is sensitive to non-rigid deformations of the shapes. Moreover, it also depends on rigid transformations of shapes; for example, translating one shape with respect to another changes the Hausdorff distance.

ICP-type methods try to get rid of this alignment ambiguity by minimizing the Hausdorff distance between X and

Y over all the isometries in $\mathbb{R}^3$ (i.e., rigid transformations, including rotations, translations and reflections),

$$d_{\mathrm{ICP}}(X,Y) \quad = \quad \inf_{i\in \mathrm{Iso}(\mathbb{R}^3)} d_{\mathrm{H}}^{\mathbb{R}^3}(i(X),Y).$$

Though resolving the rigid transformations ambiguity, ICP methods are still sensitive to non-rigid deformations. Numerically, d_{ICP} is computed using local optimization methods, liable to converge to a suboptimal solution (local minimum).

2.2. Intrinsic similarity

Canonical form-type methods compute the intrinsic similarity by posing the problem of intrinsic geometries comparison as the problem of extrinsic geometry comparison, which is relatively easy to compute. The approach consists of two stages. First, an extrinsic representation of the intrinsic geometry of the shapes (*near-isometric embedding*) in some common metric space $(\mathbb{Z}, d_{\mathbb{Z}})$ is constructed by finding two maps $\varphi : X \to \mathbb{Z}$ and $\psi : Y \to \mathbb{Z}$ with minimum distortions $\mathrm{dis}\ \varphi$ and $\mathrm{dis}\ \psi$. The embedding, in a sense, allows to "undo" all the isometric deformations of the objects (though, some degree of ambiguity stemming from isometries in $\mathbb{Z}$ still remains). Typically, $\mathbb{Z}$ is selected as the Euclidean space. The second stage is extrinsic comparison of the canonical forms $\varphi(X)$ and $\psi(Y)$, using, for example, the ICP distance $d_{\mathrm{ICP}}(\varphi(X), \psi(Y))$. Since achieving a true isometric embedding is usually impossible [12], the canonical forms are only an approximate representation of the intrinsic geometry of the shapes.

The problem of inaccuracy introduced by the embedding into $\mathbb{Z}$ can be resolved if we do not assume a given embedding space, but instead, include $\mathbb{Z}$ as a variable into the optimization. We can always find a sufficiently complicated metric space into which both X and Y can be embedded isometrically, and compare the images using the Hausdorff distance,

$$d_{\mathrm{GH}}(X,Y) \quad = \quad \inf_{\substack{\mathbb{Z} \\ \varphi:X\to\mathbb{Z} \\ \psi:Y\to\mathbb{Z}}} d_{\mathrm{H}}^{\mathbb{Z}}(\varphi(X),\psi(Y)),$$

(here φ and ψ are assumed to be isometric embeddings). Such an approach is referred to as *Gromov-Hausdorff distance* [8]. For compact surfaces, d_{GH} can be expressed in terms of the distortion obtained by embedding one surface into another,

$$d_{\mathrm{GH}}(X,Y) \quad = \quad \frac{1}{2} \inf_{\substack{\varphi:X\to Y \\ \psi:X\to Y}} \max\{\mathrm{dis}\,\varphi, \mathrm{dis}\,\psi, \mathrm{dis}\,(\varphi,\psi)\},$$

where

$$\mathrm{dis}\,(\varphi,\psi) \quad = \quad \sup_{x\in X, y\in Y} |d_X(x,\psi(y)) - d_Y(y,\varphi(x))|.$$

The computation of the distortions can be performed using GMDS, a procedure similar in its spirit to MDS, but not limited to spaces with analytically expressed geodesic distances. The Gromov-Hausdorff distance is a metric on the quotient space of non-rigid shapes under the isometry relation. Particularly, this implies that $d_{\mathrm{GH}}(X,Y) = 0$ if and only if X and Y are isometric. More generally, if $d_{\mathrm{GH}}(X,Y) \leq \epsilon$, then X and Y are 2ϵ-isometric and conversely, if X and Y are ϵ-isometric, then $d_{\mathrm{GH}}(X,Y) \leq 2\epsilon$ [5].

3. Joint similarity

Let us now return to our example of glove fitting. Assume that Y is the hand surface, and X is the glove we wish to fit. Our goal is to find such a deformation of X, denoted hereinafter by $Z = f(X)$, that is the most similar to Y in the extrinsic sense, yet, at the same time preserves its intrinsic geometry as much as possible (i.e., intrinsically similar to X). In a generic setting, we assume that Y (*probe*) is obtained from X (*model*) by means of some deformation (not necessarily isometric). We allow for the possibility that the topology of Y is different from that of X due to the presence of noise or as the result of the deformation (e.g, in the glove fitting example, the fingers of the hand can touch each other).

In order to quantify the intrinsic and extrinsic similarity, we define generic extrinsic and intrinsic distances d_{E} and d_{I}. We require that $d_{\mathrm{E}}(X,Y) = 0$ if X and Y are extrinsically similar (i.e., $X = Y$) and $d_{\mathrm{I}}(X,Y) = 0$ if X and Y are intrinsically similar (i.e., $d_X = d_Y$). Since the deformation f gives a one-to-one correspondence between X and Z, we can set $\varphi = f$ and $\psi = f^{-1}$ in the Gromov-Hausdorff distance, obtaining

$$\begin{aligned} d_{\mathrm{I}}(X,Z) \ &= \ \frac{1}{2}\mathrm{dis}\ \varphi \\ &= \frac{1}{2} \sup_{x,x'\in X} |d_X(x,x') - d_Z(f(x),f(x'))| \end{aligned} \quad (2)$$

as the intrinsic distance. Note that the correspondence between the surfaces is now fixed and does not participate anymore in the minimization. $d_{\mathrm{I}}(X,Z)$ defined this way measures the distortion in the intrinsic geometry of X introduced by the extrinsic deformation f. However, the geodesic distances in Z have to be re-computed every time the deformation changes.

Due to its sensitivty to noise, the L_∞ formulation of d_{I} can be less practical when working with real shapes. For that reason, we prefer to use its L_2 version,

$$\begin{aligned} &d_{\mathrm{I}}(x,Z) \ = \\ &\int_{X\times X} (d_X(x,x') - d_Z(f(x),f(x')))^2 da(x) da(x'), \end{aligned} \quad (3)$$

where da denotes the area element on X.

As the extrinsic distance, we use an L_2 version of the non-symmetric Hausdorff distance,

$$d_{\mathrm{E}}(Y,Z) \quad = \quad \int_Z \|z - y^*(z)\|_2 \, da(z), \qquad (4)$$

where $y^*(z) = \arg\min_{y \in Y} \|z - y\|_2$ is the closest point to z on Y (we write y^*, since in practice we work with compact objects, on which this minimum is always achieved). The set of closest points $y^*(Z)$ has to be recomputed every time the deformation changes. The lack of symmetry of $d_{\mathrm{E}}(Y,Z)$ allows for partial matching.

Since it is usually impossible to say which of the two criteria is more important, we judge the similarity as a tradeoff between them,

$$d_{\mathrm{J}}(X,Y) \quad = \quad \min_Z \, d_{\mathrm{I}}(X,Z) + \lambda \, d_{\mathrm{E}}(Z,Y), \qquad (5)$$

which we refer to as the *joint similarity* criterion. The parameter λ controls the relative significance of each criterion. This approach generalizes the similarity criteria based purely on extrinsic or intrinsic geometry. Setting $\lambda \ll 1$ penalizes the distortions in the intrinsic geometry, yielding a generalization of ICP, since we now allow for non-rigid isometries and not only for the rigid ones. On the other hand, setting $\lambda \gg 1$, the probe is forced to be attached to the model surface, which boils down to a problem similar to GMDS.

4. Numerical framework

For practical computations, we work with discretized shapes. The surface X is sampled at N points $\hat{X} = \{x_1, ..., x_N\} \subseteq X$, constituting an r-*covering* (i.e., $X = \bigcup_{n=1}^{N} B_X(x_n, r)$, where $B_X(x,r)$ is a metric ball of radius r centered at x). The extrinsic coordinates of $\hat{X}$ are represented as an $N \times 3$ matrix $\mathbf{X}$, whose ith row corresponds to a point $x_i \in \mathbb{R}^3$. The discrete shape is represented as a triangular mesh; each triangle is a triplet of indices of vertices belonging to it. Vertices connected by an edge (belonging to the same triangle) are said to be *adjacent*; we denote by E the set of all adjacent pairs of vertices in $\hat{X}$. The geodesic distances on $\hat{X}$ are approximated using the Dijkstra algorithm, forming an $N \times N$ matrix $\hat{\mathbf{D}}(\mathbf{X})$, whose elements are $\hat{d}_{ij}(\mathbf{X}) \approx d_X(x_i, x_j)$.

Assuming the deformed surface $\hat{Z} = f(\hat{X})$ maintains the connectivity of $\hat{X}$, we can formulate the following minimization problem with respect to the $N \times 3$ matrix $\mathbf{Z}$ of the extrinsic coordinates of $\hat{Z}$:

$$\begin{aligned} d_{\mathrm{J}}(\hat{X},\hat{Y}) \quad &= \quad \min_{\mathbf{Z}} \frac{1}{N^2} \sum_{i,j=1}^{N} (\hat{d}_{ij}(\mathbf{X}) - \hat{d}_{ij}(\mathbf{Z}))^2 \\ &+ \quad \frac{\lambda}{N} \sum_{i=1}^{N} d^2_{\mathbb{R}^3}(z_i, \hat{Y}) \end{aligned} \qquad (6)$$

where $d(z_i, \hat{Y})$ denotes the Euclidean distance from the point z_i to the discretized surface $\hat{Y}$. We denote the above cost function by $\sigma(\mathbf{Z})$. The first term in it is the discretization of d_{I}, whereas the second term is the discretization of d_{E}. In the following, we show how to compute these two terms and their derivatives with respect to $\mathbf{Z}$, required for the minimization of (6).

4.1. Intrinsic distance computation

The main challenge in optimization involving the computation of the intrinsic distance term d_{I} is the necessity to evaluate the geodesic distances on $\hat{Z} = f(\hat{X})$ and their derivatives with respect to the extrinsic geometry of $\hat{Z}$ changing at each iteration of the minimization algorithm. We first define the matrix $\mathbf{D}(\mathbf{Z})$ of *local distances*, whose elements are

$$d_{ij}(\mathbf{Z}) \quad = \quad \begin{cases} \|z_i - z_j\| & : \quad (i,j) \in E \\ 0 & : \quad (i,j) \notin E. \end{cases} \qquad (7)$$

Using the Dijkstra algorithm, we compute the set of the shortest paths between all pairs of points (i,j). For example, let $\mathcal{P}_{ij} = \{(i,i_1),(i_1,i_2),...,(i_{n-1},i_n),(i_n,j)\} \subset E$ be the shortest path between the points i and j. Its length is given by $L(\mathcal{P}_{ij}) = d_{i,i_1} + d_{i_1,i_2} + ... + d_{i_n,j}$, which is a linear combination of the elements of $\mathbf{D}(\mathbf{Z})$. We can therefore "complete" the missing entries in the matrix $\mathbf{D}(\mathbf{Z})$ by defining the matrix of approximate *global distances*

$$\hat{\mathbf{D}}(\mathbf{Z}) \quad = \quad \mathcal{I}(\mathbf{D}(\mathbf{Z}))\mathbf{D}(\mathbf{Z}), \qquad (8)$$

where $\mathcal{I}$ is a sparse fourth order tensor, with the elements $\mathcal{I}_{ijkl} = 1$ if the edge (k,l) is contained in the shortest path $\mathcal{P}_{ij}$, and 0 otherwise. Note that $\mathcal{I}$ depends on the connectivity E, which is assumed to be fixed, and the matrix of local distances $\mathbf{D}$, which, in turn, depends on $\mathbf{Z}$.

In order to compute the derivative of $\hat{\mathbf{D}}(\mathbf{Z})$ with respect to $\mathbf{Z}$, we assume that a small perturbation $d\mathbf{Z}$ of $\mathbf{Z}$ does not change the connectivity of the points on $\hat{Z}$, and as the results, the trajectory of the shortest paths between the points on $\hat{Z}$ traverses the same edges. Thus, we may write

$$\begin{aligned} \hat{\mathbf{D}}(\mathbf{Z} + d\mathbf{Z}) \quad &= \quad \mathcal{I}(\mathbf{D}(\mathbf{Z} + d\mathbf{Z}))\mathbf{D}(\mathbf{Z} + d\mathbf{Z}) \\ &= \quad \mathcal{I}(\mathbf{D}(\mathbf{Z}))\mathbf{D}(\mathbf{Z} + d\mathbf{Z}), \end{aligned}$$

and compute the derivative of $\hat{\mathbf{D}}(\mathbf{Z})$ as the derivative of the linear form $\mathcal{I}(\mathbf{D}(\mathbf{Z}))$. If $\mathcal{I}(\mathbf{D}(\mathbf{Z})) \neq \mathcal{I}(\mathbf{D}(\mathbf{Z} + d\mathbf{Z}))$, the assumption does not hold and the derivative of $\hat{\mathbf{D}}$ usually does not exist. Yet, the derivative of $\mathcal{I}(\mathbf{D}(\mathbf{Z}))$ belongs to the sub-gradient set of $\hat{\mathbf{D}}(\mathbf{Z})$ at the point $\mathbf{Z}$. This is sufficient for many minimization algorithms to work correctly [14].

The intrinsic distance can be written in terms of $\hat{\mathbf{D}}(\mathbf{Z})$ as

the Frobenius norm

$$d_{\mathrm{I}}(\mathbf{Z}) = \frac{1}{N^2}\|\hat{\mathbf{D}}(\mathbf{Z}) - \hat{\mathbf{D}}(\mathbf{X})\|_F^2 \quad (9)$$
$$= \frac{1}{N^2}\mathrm{tr}((\hat{\mathbf{D}}(\mathbf{Z}) - \hat{\mathbf{D}}(\mathbf{X}))^{\mathrm{T}}(\hat{\mathbf{D}}(\mathbf{Z}) - \hat{\mathbf{D}}(\mathbf{X}))).$$

Its derivative with respect to X is given by

$$\frac{\partial d_{\mathrm{I}}(\mathbf{Z})}{\partial \mathbf{Z}} = \frac{2}{N^2}(\hat{\mathbf{D}}(\mathbf{Z}) - \hat{\mathbf{D}}(\mathbf{X}))^{\mathrm{T}}\frac{\partial \hat{\mathbf{D}}(\mathbf{Z})}{\partial \mathbf{Z}}, \quad (10)$$

where

$$\frac{\partial \hat{d}_{ij}(\mathbf{Z})}{\partial \mathbf{Z}} = \sum_{k,l}\mathcal{I}_{ijkl}\frac{\partial d_{kl}(\mathbf{Z})}{\partial \mathbf{Z}}, \quad (11)$$

and the elements of $\frac{\partial d_{kl}(\mathbf{Z})}{\partial \mathbf{Z}}$ are given by

$$\frac{\partial d_{kl}}{\partial z_n^m} = \frac{1}{d_{kl}}\begin{cases} z_k^m - z_l^m & : \quad n = k \\ z_l^m - z_k^m & : \quad n = l \\ 0 & : \quad n \neq k, l \end{cases} \quad (12)$$

for $m = 1, 2, 3$.

4.2. Extrinsic distance computation

The computation of the extrinsic distance is similar to the one used in ICP algorithms, where the main difficulty arises from the need to re-compute the closest points each time the extrinsic geometry of $\hat{Z}$ changes. The extrinsic distance term can be written as

$$d_{\mathrm{E}}(\mathbf{Z}) = \frac{1}{N}\mathrm{tr}\left((\mathbf{Z} - \mathbf{Y}^*(\mathbf{Z}))(\mathbf{Z} - \mathbf{Y}^*(\mathbf{Z}))^{\mathrm{T}}\right) \quad (13)$$

where $\mathbf{Y}^*(\mathbf{Z})$ denotes the $N \times 3$ matrix, whose i-th row is the closest point on $\hat{Y}$ corresponding to x_i. The closest points are computed as a weighted average of the points on $\hat{Y}$ closest to x_i. The weights are selected in inverse proportion to the distance from x_i.

In ICP algorithms, it is common to assume $\mathbf{Y}^*(\mathbf{Z} + d\mathbf{Z}) \approx \mathbf{Y}^*(\mathbf{Z})$. By fixing $\mathbf{Y}^*$, $d_{\mathrm{E}}(\mathbf{Z})$ becomes a simple quadratic function, and its derivative can be written as

$$\frac{\partial d_{\mathrm{E}}(\mathbf{Z})}{\partial \mathbf{Z}} = \frac{2}{N}(\mathbf{Z} - \mathbf{Y}^*(\mathbf{Z}))^{\mathrm{T}}. \quad (14)$$

4.3. Iterative minimization algorithm

For $\mathbf{Z}$ in the neighborhood of some $\mathbf{X}$, the cost function that needs to be minimized can be approximated as

$$\sigma(\mathbf{Z}) \approx$$
$$\frac{1}{N^2}\mathrm{tr}\left(\hat{\mathbf{D}}(\mathbf{Z})^{\mathrm{T}}\hat{\mathbf{D}}(\mathbf{Z}) - 2\hat{\mathbf{D}}(\mathbf{X})^{\mathrm{T}}\hat{\mathbf{D}}(\mathbf{Z}) + \hat{\mathbf{D}}(\mathbf{X})^{\mathrm{T}}\hat{\mathbf{D}}(\mathbf{X})\right) +$$
$$\frac{\lambda}{N}\mathrm{tr}\left(\mathbf{Z}\mathbf{Z}^{\mathrm{T}} - 2\mathbf{Y}^*(\mathbf{X})\mathbf{Z}^{\mathrm{T}} + \mathbf{Y}^*(\mathbf{X})^{\mathrm{T}}\mathbf{Y}^*(\mathbf{X})\right), \quad (15)$$

where $\hat{\mathbf{D}}(\mathbf{Z}) \approx \mathcal{I}(\mathbf{D}(\mathbf{X}))\mathbf{D}(\mathbf{Z})$. Like in ICP, after finding a new $\mathbf{Z}$ which decreases $\sigma(\mathbf{Z})$, the closest points $\mathbf{Y}^*$ and the operator $\mathcal{I}$ are updated. The iterative minimization algorithm can be summarized as follows:

1. Compute the closest points $\mathbf{Y}^*(\mathbf{Z})$.
2. Compute the shortest paths between all pairs of points on $\hat{Z}$ and assemble $\mathcal{I}$.
3. Update $\mathbf{Z}$ by $\Delta\mathbf{Z}$ such that $\mathbf{Z} + \Delta\mathbf{Z}$ sufficiently decreasing the cost function (15).
4. If the change in $\mathbf{Z}$ is small, stop. Else, go to Step 1.

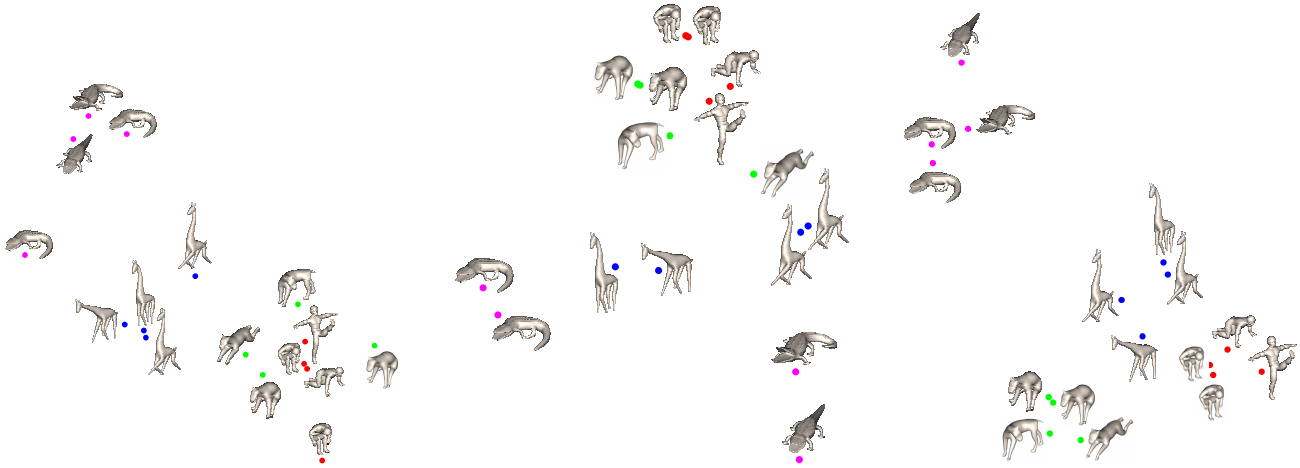
The update of $\mathbf{Z}$ in Step 3 can be safeguarded by evaluating the true cost function (with $\mathcal{I}(\mathbf{Z} + \Delta\mathbf{Z})$ and $\mathbf{Y}^*(\mathbf{Z} + \Delta\mathbf{Z})$ instead of $\mathcal{I}(\mathbf{Z})$ and $\mathbf{Y}^*(\mathbf{Z})$). In our implementation, no safeguard was used, and the minimization in Step 3 was done using conjugate gradients.

The initialization of the algorithm can be done in several ways, the simplest of which is $\mathbf{Z} = \mathbf{X}$. This choice works well when the extrinsic dissimilarity between $\hat{X}$ and $\hat{Y}$ is not too large; for large $d_{\mathrm{E}}(\hat{X}, \hat{Y})$, the algorithm will suffer from poor convergence similar to most ICP methods. Another choice is to initialize $\mathbf{Z}$ by the corresponding points on $\hat{Y}$ resulting from the solution of the GMDS problem. This choice is suitable for objects having sufficiently similar intrinsic geometries, making the intrinsic correspondence computed by GMDS meaningful.

5. Results

In order to demonstrate the proposed approach, we used a data set of four objects with four non-rigid deformations (Figure 2). While the first three deformations of each object were nearly isometric, the fourth one introduced topological changes modeled by welding points on the mesh. Three distances were computed: a non-symmetric L_2 approximation of the Gromov-Hausdorff distance, a non-symmetric L_2 approximation of the Hausdorff distance, and the joint distance d_{J}. The Gromov-Hausdorff distance was computed using GMDS with 100 points embedded into a surface represented with 1000 points. The Hausdorff distance was computed using ICP with meshes sampled at 1000 points. The joint distance was implemented in MATLAB and computed on meshes sampled at 100 points; its computation required approximately one minute per pair of shapes.

Figure 2 visualizes the computed similarities. It appears that while the intrinsic similarity is nearly invariant to isometric deformations, it is sensitive to topological changes, which introduce significant distortions in the intrinsic geometry. At the other end, the extrinsic geometry is insensitive to topological changes, yet sensitive to intrinsic deformation. The joint similarity criterion combines both advantages, and is insensitive to both isometries and topological changes.



Intrinsic similarity (d_{GH}). Extrinsic similarity (d_{ICP}). Joint similarity (d_{J}).

Figure 2. Visualization of different types of similarity (distances in the plane represent the degree of similarity).

6. Conclusions

We presented a new approach for the computation of non-rigid shape similarity as a tradeoff between extrinsic and intrinsic similarity criteria, which can be thought of as a hybrid of ICP and GMDS. Our approach can be illustratively presented as deforming one shape in order to make it the most similar to another from the extrinsic point of view, while trying to preserve as much as possible its intrinsic geometry. The joint intrinsic and extrinsic similarity appears to be advantageous over traditional purely extrinsic or intrinsic similarity criteria. While extrinsic similarity is sensitive to strong non-rigid deformations of the shapes and intrinsic similarity is sensitive to topology changes resulting from noise or non-rigid deformations, our joint similarity criterion allows to gracefully handle such problems. Experimental results demonstrate that it can be useful in situations where intrinsic and extrinsic similarities fail. In addition to shape analysis, our approach can be used for shape synthesis (e.g., morphing and animation problems), in a way similar to [10].

References

[1] R. Basri, L. Costa, D. Geiger, and D. Jacobs. Determining the similarity of deformable shapes. *Vision Research*, 38:2365–2385, 1998. 1

[2] P. J. Besl and N. D. McKay. A method for registration of 3D shapes. *IEEE Trans. PAMI*, 14:239–256, 1992. 1

[3] I. Borg and P. Groenen. *Modern multidimensional scaling - theory and applications*. Springer-Verlag, Berlin Heidelberg New York, 1997. 1

[4] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. National Academy of Sciences*, 103(5):1168–1172, January 2006. 1

[5] D. Burago, Y. Burago, and S. Ivanov. *A course in metric geometry*, volume 33 of *Graduate studies in mathematics*. American Mathematical Society, 2001. 2, 3

[6] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *Proc. IEEE Conference on Robotics and Automation*, 1991. 1

[7] A. Elad and R. Kimmel. On bending invariant signatures for surfaces. *IEEE Trans. PAMI*, 25(10):1285–1295, 2003. 1

[8] M. Gromov. *Structures métriques pour les variétés riemanniennes*. Number 1 in Textes Mathématiques. 1981. 1, 3

[9] D. Jacobs, D. Weinshall, and Y. Gdalyahu. Class representation and image retrieval with non-metric distances. *IEEE Trans. PAMI*, 22:583–600, 2000. 1

[10] M. Kilian, N. J. Mitra, and H. Pottmann. Geometric modeling in shape space. In *Proc. SIGGRAPH*, volume 26, 2007. 6

[11] L. J. Latecki and R. Lakamper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. PAMI*, 22(10):1185–1190, 2000. 1

[12] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some its algorithmic applications. *Combinatorica*, 15:333–344, 1995. 3

[13] F. Mémoli and G. Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Foundations of Computational Mathematics*, 5:313–346, 2005. 1

[14] B. T. Polyak. Minimization of non-smooth functionals. *USSR Computational Mathematics and Mathematical Physics*, 9:14–29, 1969. 4

[15] E. L. Schwartz, A. Shaw, and E. Wolfson. A numerical solution to the generalized mapmaker's problem: flattening nonconvex polyhedral surfaces. *IEEE Trans. PAMI*, 11:1005–1008, 1989. 1