Calculus of non-rigid surfaces for geometry and texture manipulation

Alexander M. Bronstein, *Student Member, IEEE,* Michael M. Bronstein, *Student Member, IEEE,* and Ron Kimmel, *Senior Member, IEEE*

***Abstract*— We present a geometric framework for automatically finding intrinsic correspondence between three-dimensional non-rigid objects. We model object deformation as near-isometries and find the correspondence as the minimum-distortion mapping. A generalization of multidimensional scaling is used as the numerical core of our approach. As the result, we obtain the possibility to manipulate the extrinsic geometry and the texture of the objects as vectors in a linear space. We demonstrate our method on the problems of expression-invariant texture mapping onto an animated three-dimensional face, expression exaggeration, morphing between faces and virtual body painting.**

***Index Terms*— isometric embedding, minimum-distortion mapping, generalized multidimensional scaling, correspondence problem, texture mapping, face animation, expression exaggeration, morphing, virtual dressing, virtual body painting, calculus of surfaces.**

I. Introduction

NON-rigid three-dimensional objects arise in numerous computer graphics problems, including facial animation [23] and modelling [30], [26], [3], caricaturization, expression exaggeration [5] and transplantation from one face to another [18], [28], [25], cross-parametrization [35], [34], texture mapping [35], and morphing [1], [22]. The common denominator of all the above applications is the *correspondence problem*, that is, the need to identify the corresponding points in two different deformations of the object. Unlike synthetic object animation, where such a correspondence is usually known, in general, for example, when the objects are acquired by a range scanner, the correspondence is not readily available. Hence, in most cases, it must be established from the geometry of the objects.

Standard approaches for finding correspondence between two objects search for a common parametrization for the objects. In most cases, this procedure is not fully automatic and demands a user-assisted selection of a set of fiducial points [27], [22], [33]. In the problem of 3D facial animation, it is possible to construct a parametrization of faces that is common to all expressions [13], [3]. A hybrid method based on fitting 2D facial images to a deformable 3D model of the face was proposed in [30], [26].

Recently, methods based on isometric embeddings have been introduced in the computer vision community for deformation-invariant object recognition [14]. It was noted that in cases when the deformations approximately preserve the metric structure, the intrinsic geometry can be used as an invariant description of the object. Such a description is created by mapping the object into a low-dimensional Euclidean space (generally, referred to as the *embedding space*) such that the geodesic distances are replaced with Euclidean ones. This procedure is called *isometric* (or more correctly, *minimum-distortion*) *embedding* and is carried out using a multidimensional scaling (MDS) algorithm. The embedding, in a sense, allows to "undo" the deformation, providing a representation which is, up to the isometric group of the embedding space (in the case of Euclidean embedding, rotations, translations and reflections), is invariant to isometric deformations of the object. This method was employed to find a degree of similarity between deformable objects like different expressions of the human face [6].

Embedding the objects into a plane can be thought of as a method of finding a common parametrization [35]. Yet, the simple Euclidean embedding has several drawbacks. First, in most cases it introduces an inevitable distortion due to the fact that a non-flat shape cannot be isometrically embedded into a Euclidean space. Second, an alignment stage is needed in order to resolve the remaining degrees of freedom in the embedding space (Euclidean isometries), which, in turn, requires a dense sampling of the object (thousands of points) in order for the alignment to be accurate. Third, attempts to use Euclidean embeddings for texture mapping were practically limited to objects homeomorphic to a disc [35].

In [9], we proposed a generalization of MDS (hereinafter, GMDS for short) that allows to embed one object into another rather than using a common embedding space. Such an embedding establishes a correspondence between the two objects. Here, we adopt this approach, as it has several important advantages over the Euclidean embedding computed by the traditional MDS. GMDS can be be applied to objects with arbitrary topology, it does not require alignment, and, since the embedding space can be chosen to be an isometry of the object itself, the metric distortion introduced by embedding into a common space is avoided. GMDS can be naturally adapted to finding correspondence between partially missing objects. This allows us to gracefully deal with occlusions, often encountered in objects acquired using range scanners. Furthermore, the number of points required for accurately determining the correspondence can be small (tens or hundreds). This can be an important criterion in real-time applications, where computational restrictions force meshes with low-polygon count.

In this paper, we present an automatic correspondence pro-

Manuscript received February 15, 2006. Revised June 5, 2006.

A. M. Bronstein, M. M. Bronstein and R. Kimmel are with the Department of Computer Science, Technion – Israel Institute of Technology.

cedure, based on intrinsic geometric properties of the objects, based on the assumption that the objects are approximately isometric. The intrinsic correspondence gives us the possibility to manipulate the extrinsic geometry and the texture of the objects as vectors in a linear space. The numerical core is the GMDS algorithm, which is computationally efficient and produces results competitive with previously used methods. We start with formulating the correspondence problem between non-rigid objects and introducing the concept of minimum-distortion embedding in Section II. Section III describes the GMDS problem and a numerical algorithm for its solution. In Section IV, we address the problem of finding correspondence between partially missing or topologically different objects. In Section V, we present our problem from a broader perspective of creating a (locally) linear space, in which surfaces can be handled as vectors. Experimental results related to texture mapping on the human body, morphing, and animation of human faces are presented in Section VI. Section VII concludes the paper.

II. FINDING CORRESPONDENCE BETWEEN NON-RIGID OBJECTS

We model a non-rigid object as a compact, connected Riemannian two-dimensional manifold (surface) $\mathcal{S}$, with the geodesic distances $d_{\mathcal{S}} : \mathcal{S}\times\mathcal{S} \to \mathbb{R}$ induced by the Riemannian metric. From the point of view of metric geometry, the pair $(\mathcal{S}, d_{\mathcal{S}})$ is a *metric space*, and $d_{\mathcal{S}}$ describes the *intrinsic geometry* of the object. A surface $\mathcal{Q}$ obtained by means of a bijective map $\varphi : \mathcal{S} \to \mathcal{Q}$ is called a *deformation* of $\mathcal{S}$. If $d_{\mathcal{S}}(s, s') = d_{\mathcal{Q}}(\varphi(s), \varphi(s'))$ for all $s, s' \in \mathcal{S}$, we say that the map φ is an *isometry* and that $\mathcal{S}$ and $\mathcal{Q}$ are *isometric*. In practice, deformations preserve the distances only approximately, such that

$$|d_{\mathcal{S}}(s, s') - d_{\mathcal{Q}}(\varphi(s), \varphi(s'))| \leq \epsilon.$$

We call such deformations *ϵ-isometries* (or in general, *near-isometries*, without having ϵ specified).

The essence of the correspondence problem is finding the map φ, establishing the *correspondence* between the objects $\mathcal{S}$ and $\mathcal{Q}$, from their geometry. If we knew a common parameterization for $\mathcal{S}$ and $\mathcal{Q}$, say $\pi_{\mathcal{S}} : U \subset \mathbb{R}^2 \to \mathcal{S}$ and $\pi_{\mathcal{Q}} : U \subset \mathbb{R}^2 \to \mathcal{Q}$, we could compute the correspondence as $\varphi = \pi_{\mathcal{Q}} \circ \pi_{\mathcal{S}}^{-1}$. However, the mappings $\pi_{\mathcal{S}}$ and $\pi_{\mathcal{Q}}$ are unknown in practice. Correspondence algorithms based on common parametrization usually enforce U to be, for instance, the unit square. When only the geometry is available, constructing such a common parametrization in a consistent way is a challenging problem. Theoretically, the mappings $\pi_{\mathcal{S}}$ and $\pi_{\mathcal{Q}}$ can be estimated by finding correspondence between some fiducial points or features located on both objects [22]. Yet, the main limitation of feature-based approaches is the fact that they require a robust feature detector. In some cases, feature detection can be done automatically,[1] but usually it is user-assisted [27], [33].

Zigelman *et al.* [35] proposed using MDS to embed $\mathcal{S}$ and $\mathcal{Q}$ into the plane and thus recover the parametrizations $\pi_{\mathcal{S}} : \mathbb{R}^2 \to \mathcal{S}$ and $\pi_{\mathcal{Q}} : \mathbb{R}^2 \to \mathcal{S}$. This idea can be problematic for non-flat objects or objects with complicated topology. Moreover, since the embedding is performed into the whole $\mathbb{R}^2$, there is no guarantee that $\pi_{\mathcal{S}}$ and $\pi_{\mathcal{Q}}$ have the same domain.

A. Minimum-distortion embedding

In many applications, the deformations of an object can be described as near-isometries. For example, different postures of humans and animals are isometric deformations of their respective bodies. In [6], we showed empirically that the deformations of a human face due to natural expressions can be also approximated by isometries (an example of such deformations is shown in Figure 1). Relying on this knowledge, we can find φ as a map with the smallest distortion of the geodesic distances, e.g. measured as

$$\operatorname{dis}\varphi \equiv \sup_{s,s' \in \mathcal{S}} |d_{\mathcal{S}}(s, s') - d_{\mathcal{Q}}(\varphi(s), \varphi(s'))|.$$

If $\mathcal{S}$ and $\mathcal{Q}$ are ϵ-isometric, it is guaranteed that $\operatorname{dis}\varphi \leq \epsilon$. Our goal is to find φ with the minimal distortion $\operatorname{dis}\varphi$, which, according to our model, will give a good correspondence between $\mathcal{S}$ and $\mathcal{Q}$.

In practical applications, we work with discrete objects. The surface $\mathcal{S}$ is sampled at N points, $\{s_1, ..., s_N\}$, and represented as a triangular mesh. The geodesic distances between the samples are represented as an $N \times N$ matrix $\mathbf{\Delta}_{\mathcal{S}} = (d_{\mathcal{S}}(s_i, s_j))$, which is computed numerically using, for example, the *fast marching method* (FMM) [21]. Similarly, the surface $\mathcal{Q}$ is represented as $(\{q_1, ..., q_M\} \subset \mathcal{Q}, \mathbf{\Delta}_{\mathcal{Q}})$. In this discrete setting, we are looking for a map $\varphi : \{s_1, ..., s_N\} \to \mathcal{Q}$, such that $d_{\mathcal{S}}(s_i, s_j)$ is as close as possible to $d_{\mathcal{Q}}(\varphi(s_i), \varphi(s_j))$ for all $i, j = 1, ..., N$, that is,

$$\begin{aligned} \varphi &= \underset{\varphi}{\operatorname{argmin}} \max_{i,j=1,...,N} |d_{\mathcal{S}}(s_i, s_j) - d_{\mathcal{Q}}(\varphi(s_i), \varphi(s_j))| \\ &= \underset{\varphi}{\operatorname{argmin}} \operatorname{dis}\varphi. \end{aligned} \quad (1)$$

We refer to such φ as a *minimum-distortion embedding* of $\mathcal{S}$ into $\mathcal{Q}$; φ is a genuine isometry only if $\mathcal{S}$ and $\mathcal{Q}$ are isometric. Note that $(\mathcal{Q}, d_{\mathcal{Q}})$ is tacitly assumed to be a continuous surface here, as $\varphi(s_i)$ can be any point on $\mathcal{Q}$, not necessarily coinciding with $\{q_1, ..., q_M\}$. In practice, the values of $d_{\mathcal{Q}}$ must be approximated numerically from $(\{q_1, ..., q_M\} \subset \mathcal{Q}, \mathbf{\Delta}_{\mathcal{Q}})$. Generally, $\mathcal{S}$ can be a subset of $\mathcal{Q}$ (up to a nearly-isometric deformation); we address this case in Section IV.

[1] In the specific problem of finding correspondence between human faces, only a few points such as the eyes and the nose tip can be detected sufficiently accurately based on the surface geometry. This is due to the fact that the geometry of the facial surface contains mostly low-frequency information, while feature detection usually requires high-frequency information. Blanz *et al.* [3] establish dense correspondence using optical flow applied to the texture. However, such an approach is not applicable when the texture is not available.

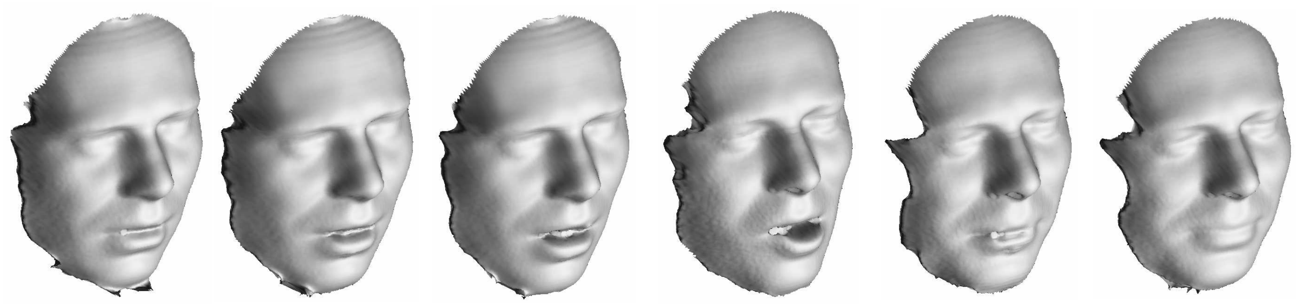

Fig. 1. Example of deformations of a non-rigid surfaces: a video sequence of one of the author's face, acquired using a real-time 3D scanner. Facial expressions can be modeled as near-isometries of the reference facial surface ("neutral expression").

III. Generalized multidimensional scaling

Problem (1) is apparently untractable, as it requires optimization over all the maps $\varphi : \{s_1, ..., s_N\} \rightarrow \mathcal{Q}$. Yet, denoting $q_i' = \varphi(s_i)\, , i = 1, ..., N$, we can reformulate (1) as an optimization over the image $\varphi(\{s_1, ..., s_N\})$, in an MDS-like spirit. For this end, we define the *generalized stress*

$$\sigma_p(q_1', ..., q_N') \;=\; \sum_{i>j} |d_{\mathcal{S}}(s_i, s_j) - d_{\mathcal{Q}}(q_i', q_j')|^p. \quad (2)$$

For $p = \infty$, we define

$$\begin{aligned} \sigma_\infty(q_1', ..., q_N') &= \max_{i,j=1,...,N} |d_{\mathcal{S}}(s_i, s_j) - d_{\mathcal{Q}}(q_i', q_j')| \\ &= \operatorname{dis}\varphi, \end{aligned} \quad (3)$$

The embedding φ is computed by minimization of the generalized stress,

$$\{q_1', ..., q_N'\} = \underset{q_1',...,q_N'}{\operatorname{argmin}}\; \sigma_p(q_1', ..., q_N'), \quad (4)$$

thus establishing a correspondence between the given N points $\{s_1, ..., s_N\} \subset \mathcal{S}$ and N points $\{q_1', ..., q_N'\}$ on $\mathcal{Q}$. Note that this approach is based only on the intrinsic geometry of the surfaces, and thus independent of the alignment of the surfaces in the Euclidean space. Unlike methods based on fiducial points, here we obtain a correspondence between a dense set of points, since N can be as large as necessary.

Problem (4) can be considered as a generalization of *multidimensional scaling* (MDS) [4] to arbitrary metric spaces. We call it the *generalized* MDS or GMDS for short [9]. Like in traditional MDS, problem (4) is a non-convex optimization problem, and therefore convex optimization algorithms may converge to a local minimum rather than to the global one [4]. Nevertheless, convex optimization is widely used in the MDS community if some precautions are taken in order to prevent convergence to local minima. In Section III-C, we show a multiscale optimization scheme that in practical applications shows good global convergence.

In the case of $p = \infty$, the GMDS can be reformulated as a constrained optimization problem

$$\min_{q_1',...,q_N',\tau} \tau \quad \text{s.t.} \quad |d_{\mathcal{S}}(s_i, s_j) - d_{\mathcal{Q}}(q_i', q_j')| \leq \tau;\; i > j, \quad (5)$$

with the use of an artificial variable τ. This problem is intimately related to the computation of the Gromov-Hausdorff distance between metric spaces [19], [24], [7]. In practice, small values of p (e.g. $p = 2$) are usually preferred.

Finally, note that since $\{q_1', ..., q_N'\}$ may be arbitrary points on the mesh $\mathcal{Q}$, we have to compute the distances $d_{\mathcal{Q}}$ between every pair of points on $\mathcal{Q}$. For this purpose, we use the *three-point geodesic distance approximation*, which is detailed in [7]. The idea of this numerical procedure is to produce a computationally efficient $\mathcal{C}^1$-approximation for $d_{\mathcal{Q}}$ and its derivatives, interpolating their values from the matrix $\boldsymbol{\Delta}_{\mathcal{Q}}$ of pairwise geodesic distances on $\mathcal{Q}$.

A. Iterative solution of the GMDS problem

Our goal is to bring the generalized stress (2) to a (possibly local) minimum over $\{q_1', ..., q_N'\}$, represented in some parametrization domain as vectors of coordinates $\{\mathbf{u}_1, ..., \mathbf{u}_N\}$. For example, if the surface $\mathcal{Q}$ admits some global parametrization, e.g. $[0, 1)^2 \rightarrow \mathcal{Q}$, every point on $\mathcal{Q}$ can be represented by $\mathbf{u} \in [0, 1)^2$. Global parametrization is often readily available for objects acquired using a range scanner. For objects with more complicated topology, global parametrization may be cumbersome to construct; in this case, we may represent a point on $\mathcal{Q}$ by the index t of the triangle enclosing it and a vector $\mathbf{u}$ of *barycentric coordinates* [16] in the local coordinate system of that triangle. For the sake of simplicity, in the following, we freely switch between q_i' and their local or global parametric representation, $(t_i, \mathbf{u}_i)$ or $\mathbf{u}_i$, respectively. We refer to the latter case as the *parametric GMDS*.

The minimization algorithm starts with an initial guess $\mathbf{u}_i^{(0)}$ of the points and proceeds by iteratively updating their locations, thus producing a decreasing sequence of stress values. Let $\{\mathbf{u}_1^{(k)}, ..., \mathbf{u}_N^{(k)}\}$ be the optimization variables at the kth iteration and let $\{\mathbf{d}_1^{(k)}, ..., \mathbf{d}_N^{(k)}\}$ be a set of directions such that displacement of $\mathbf{u}_i^{(k)}$ along them by some step size $\alpha^{(k)}$ decreases the value of the stress σ_p. The simplest way to select the directions is $\mathbf{d}_i = -\nabla_{\mathbf{u}_i}\sigma_p$, known as the *gradient descent* algorithm. More efficient ways to chose the step direction exist, including conjugate gradients and quasi-Newton algorithm [2].

The step size α has to be chosen in such a way that it guarantees a sufficient decrease of σ_p. When constant step is used, there is generally a tradeoff between too small steps, which result in slow convergence, and too large steps, which are liable to increase the value of σ_p. In order to provide a guaranteed decrease of σ_p, we adaptively select the step size at every iteration using the *Armijo rule*, which first sets $\alpha = \alpha_0$ and then successively reduces it by some factor $\beta \in (0,1)$ until

$$\sigma_p(\mathbf{u}_1,...,\mathbf{u}_{N_0}) - \sigma_p(\mathbf{u}_1+\alpha\mathbf{d}_1,...,\mathbf{u}_{N_0}+\alpha\mathbf{d}_{N_0}) \geq -\gamma\alpha\sum_i \mathbf{d}_i^{\mathrm{T}}\nabla_{\mathbf{u}_i}\sigma_p(\mathbf{u}_1,...,\mathbf{u}_{N_0}),$$

where $\gamma \in (0,1)$. An empirical choice we use is $\gamma = 0.3$, and $\beta = 0.5$. We start with a large initial value of α_0, gradually refining it at each iteration. A similar rule can be applied when the update is performed for a single point per iteration, yielding a block-coordinate descent algorithm.

When a global parametrization is used, we must restrict $\mathbf{u}_i + \alpha\mathbf{d}_i$ to remain inside the parametrization domain. This is done by applying a projection operator P_U on the point coordinated after each iteration, forcing them to the parametrization domain U. When the barycentric representation is used, it is impossible to simply add $\alpha\mathbf{d}_i$ to $\mathbf{u}_i$, since the latter might leave the triangle t_i, thus invalidating the barycentric representation. Instead, the displacement is performed by following a poly-linear path starting at $\mathbf{u}_i$, propagating along a straight line in the direction $\mathbf{d}_i$ until the first intersection with the triangle boundary, then proceeding along a line inside the triangle adjacent to the intersected edge, and so on until the total length of the path is α by a *path unfolding* algorithm [7].

B. Complexity

The complexity of the generalized stress and its gradient computation is $\mathcal{O}(N^2)$. In practice, N varies between tens to hundreds of points, therefore, GMDS is computationally efficient. In our implementation of the parametric version of GMDS, the computation of the stress σ_p and its gradient $\nabla\sigma_p$ in a problem with $N = 100$ points takes about $80\,msec$ on a mobile Intel Pentium IV 2 GHz CPU. The number of function and gradient evaluations required for the optimization is usually of the order of 100.

C. Multiresolution optimization

Multiresolution methods are widely employed to resolve the problem of local convergence in non-convex problems, such as one we have here. The key idea of a multiresolution optimization scheme is to work with a hierarchy of problems, starting from a coarse version of the problem containing a small number of variables (points). The coarse level solution is interpolated to the next resolution level, and is used as an initialization for the optimization at that level. The process is repeated until the finest level solution is obtained. Such a multiresolution scheme can be thought of as a smart way of initializing the optimization problem. Small local minima tend to disappear at coarse resolution levels, thus reducing the risk of local convergence which is more probable when working at a single resolution.

Formally, let us denote by $\mathcal{S}^1 \subset \mathcal{S}^2 \subset ... \subset \mathcal{S}^R = \mathcal{S}$ an R-level hierarchy of our data. We denote $|\mathcal{S}^r| = N^r$, where $N^R = N$. The points at the $(r+1)$-st resolution level are obtained by removing part of the points in the rth level. The corresponding distance matrices $\mathbf{\Delta}^1,...,\mathbf{\Delta}^R = \mathbf{\Delta}_{\mathcal{S}}$ are created as sub-matrices of $\mathbf{\Delta}_{\mathcal{S}}$. One possibility to construct such a hierarchy is the *farthest point sampling* (FPS) strategy [15]. As the coarsest resolution level $\mathcal{S}^1$, we select N^1 points. If some prior information about the object is available, it can be employed for the initialization of the coarsest level (in the human body, the initialization can be, for example, with points located at the hands and legs); otherwise, random initialization is used. Note that unlike in feature-based approaches, these points are used only as initialization and need not be located precisely. At the next resolution level, we add points in the following manner: s_{N^1+1} is selected as the most distant point from $\mathcal{S}^1$, and so on, $s_{N^r+k} = \mathrm{argmax}_{s\in\mathcal{S}}\, d_{\mathcal{S}}(s, \{s_1,...,s_{N^r+k-1}\})$. Taking the first N^r points from the sequence produced in this manner, we obtain $\mathcal{S}^r$.

Let us assume that at the rth resolution level, $\mathcal{S}^r = \{s_1,...,s_{N^r}\}$ is embedded into $\mathcal{Q}$ using the iterative minimization algorithm described above. As a result, the set of images $\varphi_t(\mathcal{S}^r) = \{s'_1,...,s'_{N^r}\}$ on the mesh $\mathcal{Q}$ is obtained. At the next resolution level, we have to embed a larger set $\mathcal{S}_0^{r+1}$ into $\mathcal{Q}$, solving the minimization problem for $\{s'_1,...,s'_{N^{r+1}}\}$. The initialization for the first N^r points is readily available from the solution at the previous level. The initial locations for the remaining points q'_i for $i = N^r+1,...,N^{r+1}$ have to be interpolated.

It is reasonable to initialize q'_i as a point on $\mathcal{Q}$ such that the geodesic distances from it to the points $q'_1,...,q'_{N^r}$ are as close as possible to the geodesic distances from s_i to $s_1,...,s_{N^r}$. Formally, q'_i can be expressed as

$$q'_i \;=\; \arg\min_q \sum_{j\in\mathcal{N}(s_i)} \left(d_{\mathcal{Q}}(q,q'_j) - d_{\mathcal{S}}(s_i,s_j)\right)^2, \quad (6)$$

where $\mathcal{N}(s_i)$ denotes the neighborhood of s_i on $\mathcal{S}$. Note that practically the minimum can be found by exhaustively searching over all samples or even a coarser subset of $\mathcal{Q}$. The complexity of such a search is $\mathcal{O}(N^r M)$, which is of the same order as the complexity of the iterative minimization process.

IV. Partial embedding

When working with objects acquired by means of a range scanner, due to occlusions, parts of the objects may be missing. In some cases, missing data can result in the objects having different topology. Think, for example, of a hole in one of the objects, which contradicts our fundamental assumption that the objects are near-isometric. Let us assume to be given $\mathcal{S}' \subset \mathcal{S}$, a *patch* of the surface $\mathcal{S}$. Our approach requires the embedding of $\mathcal{S}'$ into $\mathcal{Q}$. Yet, if we try to apply the GMDS straightforwardly, we may find significant distortions of the geodesic distances. This results from the fact that geodesic distances corresponding to geodesics that have passed through $\mathcal{S} \setminus \mathcal{S}'$ may change, while we have tacitly assumed that the metric on $\mathcal{S}'$ is the restricted metric $d_{\mathcal{S}}|_{\mathcal{S}'}$. We call geodesic distances that violate this assumption *inconsistent*.

In order to guarantee a correct embedding, inconsistent distances must be excluded. In the discrete setting, given $\mathcal{S}'$ sampled at $\{s_1,...,s_{N'}\}$, we denote by $P \subseteq \{1,...,N'\} \times$

$\{1, ..., N'\}$ the set of pairs of points between which the geodesic distances are consistent. Consequently, the minimum-distortion embedding can be defined as

$$\varphi = \underset{\varphi}{\operatorname{argmin}} \sum_{(i,j)\in P} |d_{\mathcal{S}}(s_i, s_j) - d_{\mathcal{Q}}(\varphi(s_i), \varphi(s_j))|^p.$$

This can be equivalently formulated as the minimization of the *weighted generalized stress*

$$\sigma_p(q'_1, ..., q'_N) = \left(\sum_{i>j} w_{ij} |d_{\mathcal{S}}(s_i, s_j) - d_{\mathcal{Q}}(q'_i, q'_j)|^p \right)^{1/p} \quad (7)$$

where $w_{ij} = 1$ if $(i, j) \in P$ and 0 otherwise.

If the surface $\mathcal{S}$ is available, we can define the inconsistent distances as those in which $d_{\mathcal{S}'}(s_i, s_j) \neq d_{\mathcal{S}}(s_i, s_j)\,; i, j = 1, ..., N'$. Otherwise, we must remove distances between pairs of points (i, j) close to the boundary $\partial\mathcal{S}'$, for which

$$d_{\mathcal{S}'}(s_i, \partial\mathcal{S}') + d_{\mathcal{S}'}(s_j, \partial\mathcal{S}') < d_{\mathcal{S}'}(s_i, s_j).$$

In practice, when the surfaces are given in a discrete representation, the above criteria are applied to finite sets of points, and the geodesic distances are computed numerically.

V. Calculus of non-rigid objects

In a broader perspective, we can think of non-rigid objects as of points in some infinite-dimensional space. Let $\mathcal{S}$ be an object, and $\mathbb{M}$ denote an abstract subspace of all the near-isometric deformations of $\mathcal{S}$. It is known empirically that the intrinsic dimensionality of $\mathbb{M}$ is usually low and it can be represented approximately as an abstract manifold [29]. Assume that we have a sequence of smooth deformations of $\mathcal{S}$, represented as a smooth trajectory $\mathcal{S}_t : [0, T] \to \mathbb{M}$. For example, it can be a 3D video sequence acquired by a range scanner, where t is thought of as the time. Let $\mathcal{S}_t$ and $\mathcal{S}_{t+dt}$ be two adjacent samples on the trajectory $\mathcal{S}_t$ (or in other words, two consecutive frames in the video sequence), and let $\mathbf{s}_t$ and $\mathbf{s}_{t+dt}$ be the corresponding extrinsic coordinates. If the step dt is sufficiently small, the difference between $\mathcal{S}_t$ and $\mathcal{S}_{t+dt}$ is also small. We can therefore linearize the manifold $\mathbb{M}$ around the point $\mathcal{S}_t$, approximating its generally non-Euclidean structure by a Euclidean one (Figure 2). The piece of the trajectory $\mathcal{S}_{\tau\in[t,t+dt]}$ is replaced by a linear displacement, which can be represented abstractly as $d\mathcal{S} = \mathcal{S}_{t+dt} - \mathcal{S}_t$ (though the subtraction between surfaces is not yet formally defined). Broadly speaking, our construction resembles the notion of tangent space in Riemannian geometry. In this way, we obtain an ability to work with surfaces as with vectors in a linear space, which provides us with a *calculus of non-rigid objects*: the ability to "add" and "subtract" two surfaces.

The knowledge of the correspondence $\varphi_t : \mathcal{S}_t \to \mathcal{S}_{t+dt}$ between $\mathcal{S}_t$ and $\mathcal{S}_{t+dt}$ is crucial in order to think of surfaces as of vectors and be able to apply arithmetic operations on their extrinsic coordinates. Thus, in terms of extrinsic coordinates, we can write $d\mathcal{S}$ as $d\mathbf{s} = \mathbf{s}_{t+dt} \circ \varphi_t - \mathbf{s}_t$. Consequently, once establishing the correspondence φ_t using the GMDS, we can approximate $\mathcal{S}_{t+\lambda dt}$ by the following convex combination

$$\mathbf{s}_{t+\lambda dt}(s) = (1 - \lambda)\mathbf{s}_t(s) + \lambda \mathbf{s}_{t+dt}(\varphi_t(s)), \quad (8)$$

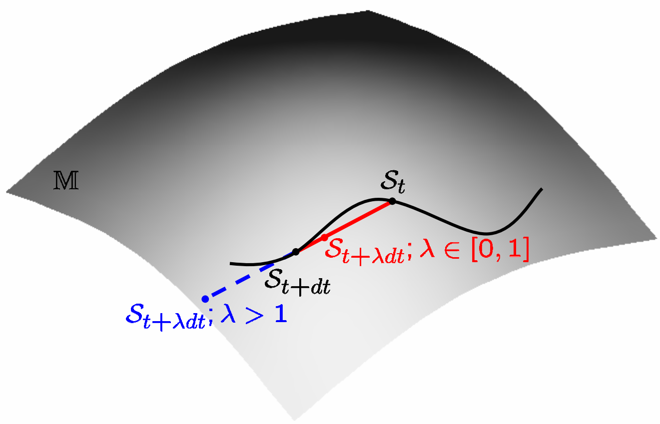

Fig. 2. Geometric illustration of interpolation and extrapolation of deformations of non-rigid objects.

for all $s \in \mathcal{S}_t$ and $\lambda \in [0, 1]$. If in addition the surfaces are endowed with the textures represented as vector fields $\alpha_t : \mathcal{S}_t \to \mathbb{R}^3$ and $\alpha_{t+dt} : \mathcal{S}_{t+dt} \to \mathbb{R}^3$, we can similarly construct the texture $\alpha_{t+\lambda dt}$ by blending between the corresponding pixels.

Varying the value of λ continuously from 0 to 1, we can create a linear interpolation between $\mathcal{S}_t$ and $\mathcal{S}_{t+dt}$ (see Figure 2 for a geometric illustration). Such an interpolation is useful, for example, as a method of *temporal super-resolution* of a 3D video. Since the video is given at a finite sampling rate, rarely exceeding 30 frames per second, we can produce the missing frames by linear interpolation between the given adjacent frames. If our objects are different faces $\mathcal{S}$ and $\mathcal{Q}$, such an interpolation will produce a morphing effect: a face which gradually turns from $\mathcal{S}$ into $\mathcal{Q}$. Note that the morphing is applied both to the extrinsic geometry of $\mathcal{S}$ and $\mathcal{Q}$ and their textures; the extrinsic geometries should be at least roughly aligned for a graceful morphing effect. This stage is trivial because we have the correspondence between the surfaces, therefore, a rigid transformation that will align them is found straightforwardly (in fact, it can be expressed analytically).

Allowing for $\lambda < 0$ or $\lambda > 1$, we can extrapolate the trajectory beyond $[t, t + dt]$. As a particular example in the facial animation problem, if $\mathcal{S}_t$ is a neutral posture of the face and $\mathcal{S}_{t+dt}$ is an expression, we can exaggerate this expression by taking $\lambda > 1$. This approach can be extended for facial features exaggeration or *caricaturization*. Suppose we are given an ensemble of representative faces $\mathcal{S}_1, ..., \mathcal{S}_N$. After finding correspondences between them, we can create the average face (*androgenus*) $\bar{\mathcal{S}}$. Given a new face $\mathcal{Q}$, we can compute the combination $\mathcal{Q}_\lambda = \lambda\mathcal{Q} - (1 - \lambda)\bar{\mathcal{S}}$. By taking $\lambda > 1$, we exaggerate the difference between $\mathcal{Q}$ and the average facial features $\bar{\mathcal{S}}$, thus emphasizing the non-average features of $\mathcal{Q}$ and thereby creating a *3D caricature*. The degree of caricaturization is controlled by the value of λ.

VI. Applications and results

A. Virtual body painting

A contemporary stream of art, known as *body painting*, presents the challenge of drawing clothes on the human body skin, in order to create an illusion of genuine clothes. When the person moves, the drawn picture deforms naturally with the skin, thus looking realistic practically in every pose of the body. In the computer graphics world, this “virtual dressing” effect can be achieved by texture mapping. As an illustration of a possible application, imagine that we would like a human actor to be used as a character in a computer game. The actor is scanned in several poses, then, an artist draws the texture that should be mapped on the character. In order to to avoid drawing a different texture for each pose, the texture from some reference pose Q must be transferred to the rest of the poses of the character. Assume that the texture α_Q is drawn on $\mathcal{Q}$. We wish to map it onto a deformed version of the object (a different pose of the character), $\mathcal{S}$. The new texture is given by $\alpha_{\mathcal{S}} = \alpha_{\mathcal{Q}} \circ \varphi$, where φ is the correspondence between $\mathcal{S}$ and $\mathcal{Q}$ computed using GMDS.

We demonstrate the GMDS approach in a virtual body painting experiment, using as a reference a public domain mesh of a head-less human body containing about 2600 vertices. Four poses were created by deforming the body in a CAD program. We painted two textures (Figure 3) that were mapped onto the reference surface. The GMDS algorithm was employed to embed 200 points on the reference object into its four different poses in order to establish the correspondence between the objects. Optimization was performed using the multi-resolution scheme with six resolution levels, created using the farthest point sampling. The correspondences obtained from the embedding are depicted in Figure 4. The texture mapping coordinates were transferred from the reference mesh to the four poses by inverse square distance-weighted interpolation. The final results are visualized in Figure 5. For better rendering, a head was manually added to each object. The mapping is good in general, despite some small yet noticeable artifacts, for example, in the fourth column of the second row in Figure 5.

B. Virtual makeup

In the motion pictures industry, one of the challenges is the creation of visually-realistic moving human faces. Current level of computer graphics allows to render a 3D animated head and embed it into the movie. Yet, such 3D animation is computationally-intensive and usually lacks the realism of genuine human face movements. On the other hand, the rapid development of 3D real-time video acquisition techniques [20] opens a new direction for creating a synthetic character, by scanning an actor and replacing his or her facial texture with a virtual one, automatically mapping a single image onto a 3D video sequence and creating a “virtual makeup” effect [8].

Thinking of the 3D video sequence frames as of deformable objects, and assuming the isometric model of facial expressions, the knowledge of the intrinsic correspondence between two facial surfaces allows expression-invariant texture mapping onto all the frames of the video sequence. We must note that expressions with open and closed mouth are topologically different, as opening the mouth creates a “hole” in the facial surface. This can be solved by imposing a topological constraint on the facial surface as described in

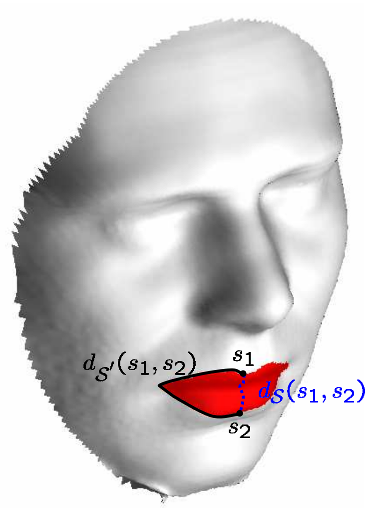

Fig. 6. The open mouth problem. Red: the cropped lips region $S \setminus \mathcal{S}'$. Blue dotted: a geodesic between the points s_1 and s_2 on $\mathcal{S}$. Black: the corresponding inconsistent geodesic on $\mathcal{S}'$.

Section IV, excluding the geodesics passing through the lips by setting the appropriate weights w_{ij} in problem (7) to zero (see Figure 6).

A scheme of the whole procedure is depicted in Figure 7. The reference surface $\mathcal{S}$ (e.g., the first frame in the video sequence) is first cropped to remove the lips and leave only the facial contour. The remaining surface $\mathcal{S}'$ is subsampled using farthest point sampling, geodesic distance between the samples are computed using FMM. The distances crossing the cropped lips region are assigned zero weights. The points are then embedded into the target surface $\mathcal{Q}$ (one of the 3D video frames) using GMDS, which produces the correspondence φ. The texture $\alpha_{\mathcal{S}}$ is transferred from the reference surface $\mathcal{S}$ onto $\mathcal{Q}$ similarly to the virtual body painting problem.

In our experiment shown here, we mapped “virtual makeup” to a real 3D video sequence of a face, acquired by a structured light scanner at 640×480 spatial resolution, three frames per second (Figure 1). The lip contour in the reference frame was segmented manually. The cropped reference frame was sampled at 100 points; all the rest of the frames were sampled uniformly at about 3000 points. The surfaces were triangulated using Delaunay triangulation; then, the geodesic distances were computed using FMM [21]. The correspondence was found by embedding 100 points on $\mathcal{S}$ into $\mathcal{Q}$ using a multiresolution optimization scheme initialized with 8 points at the coarsest resolution. Figure 8 depicts a synthetic Shrek-like character, created from the video sequence by mapping a synthetic face texture image. The obtained faces look real and the texture alignment is good even in cases of strong facial expressions. Slight artifacts can be attributed to alignment imperfections of the reference texture image.

C. Synthesis and exaggeration of facial expressions

The same 3D face video data were used to demonstrate the idea of calculus of non-rigid objects, presented in Section V. The correspondences found in the previous experiment were

Fig. 3. Two textures used in the virtual body painting experiment.

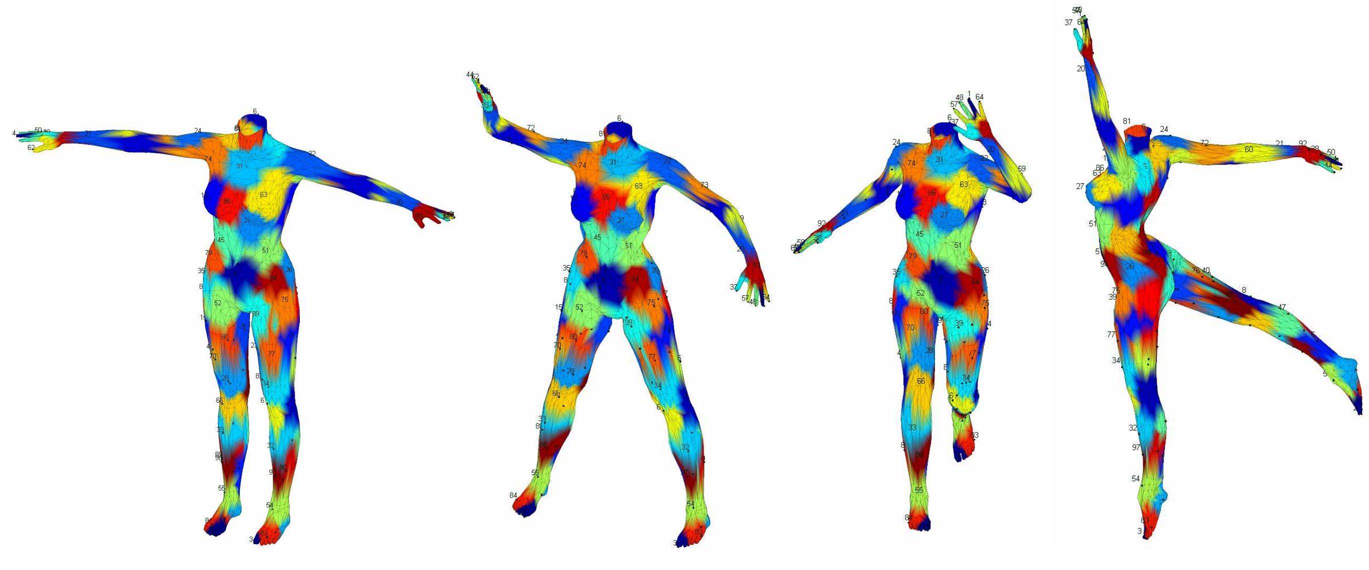

Fig. 4. Visualization of correspondence between poses of the human body, established using GMDS. Different colors depict corresponding patches built around 100 points on the objects used in the embedding.

used to transform the extrinsic geometry of the surfaces. Figure 9 shows interpolation between two frames computed according to formula (8). If, for example, we take the first frame to be a “neutral expression” and the second one to be a “sad” expression, varying λ continuously in the range $[0, 1]$ creates a natural transition between the “neutral” and the “sad” faces. Taking λ beyond 1 creates an exaggerated sad expression, depicted in Figure 10.

D. *Morphing between different faces*

So far, we have assumed that the deformations of our non-rigid object, e.g., the expressions of the face in the virtual

Fig. 5. Virtual body painting experiment. The texture is transferred from a reference pose of the human body (left column, outlined in gray) to its different poses. The correspondence between the objects is established by embedding 200 points on the reference object into its poses using the GMDS algorithm.

Fig. 8. Virtual makeup experiment. A few frames from the video sequence with a Shrek texture image mapped using the correspondence established by GMDS.

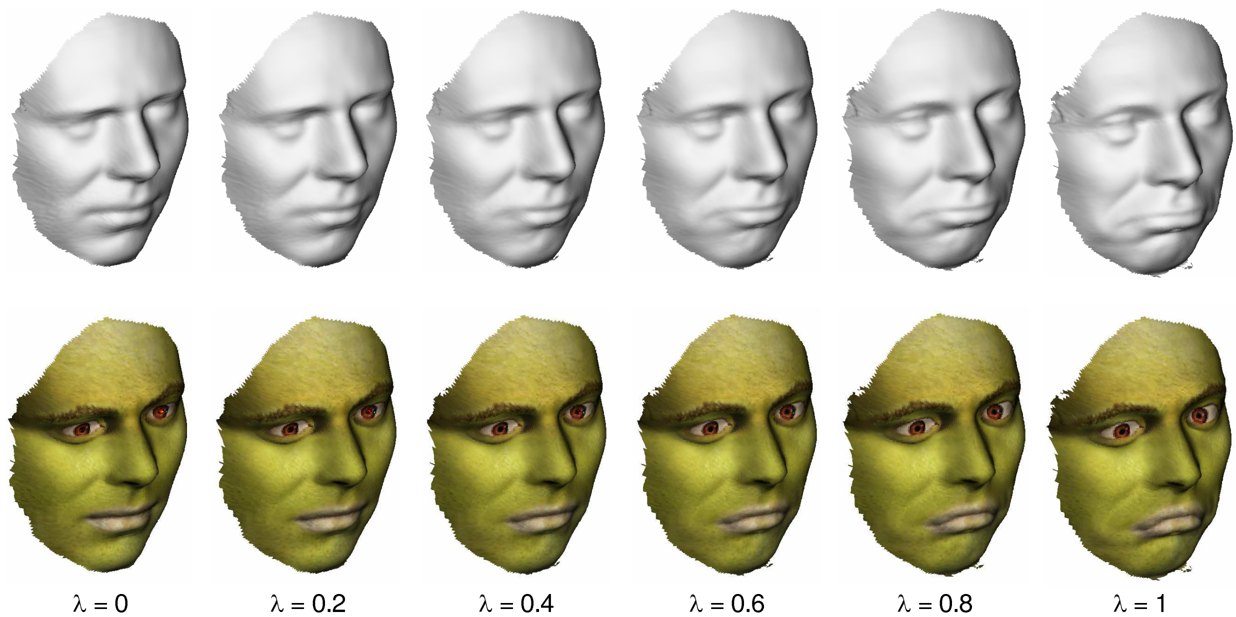

Fig. 9. Expression interpolation between two frames in the video sequence (in the first row the faces are shown without texture to emphasize the natural look of the synthetic expressions).

makeup experiment, can be described by near-isometries. Practice shows that the surfaces do not necessarily have to be isometric in order for the minimum-distortion mapping to be a good correspondence. For example, thinking of two faces as of flexible rubber masks, the correspondence problem is that of putting one mask onto the other, while trying to stretch it as little as possible. It is obvious that in most cases, the geometry features (nose, forehead, mouth, etc.) of the two masks will coincide, because in a broad sense, all human faces have similar geometry. Consequently, given two faces of different subjects, we can still use the same principle to find correspondence between them.

To exemplify this idea, we took a female and a male face from the Notre Dame database [12], [17] (denoted by $\mathcal{S}$ and $\mathcal{Q}$, with the textures $\alpha_{\mathcal{S}}$ and $\alpha_{\mathcal{Q}}$, respectively). Each face was subsampled at approximately 3000 points and triangulated. Fifty points were taken on $\mathcal{S}$ and embedded into $\mathcal{Q}$ using GMDS; the inverse of resulting correspondence was then used to map the texture $\alpha_{\mathcal{S}}$ from $\mathcal{S}$ to $\mathcal{Q}$, as $\alpha_{\mathcal{Q}} = \alpha_{\mathcal{S}} \circ \varphi^{-1}$. Figure 11 shows a synthetic face with male geometry and a female texture, obtained in this way. Figure 12 depicts a morphing effect between $\mathcal{S}$ and $\mathcal{Q}$, obtained by interpolating

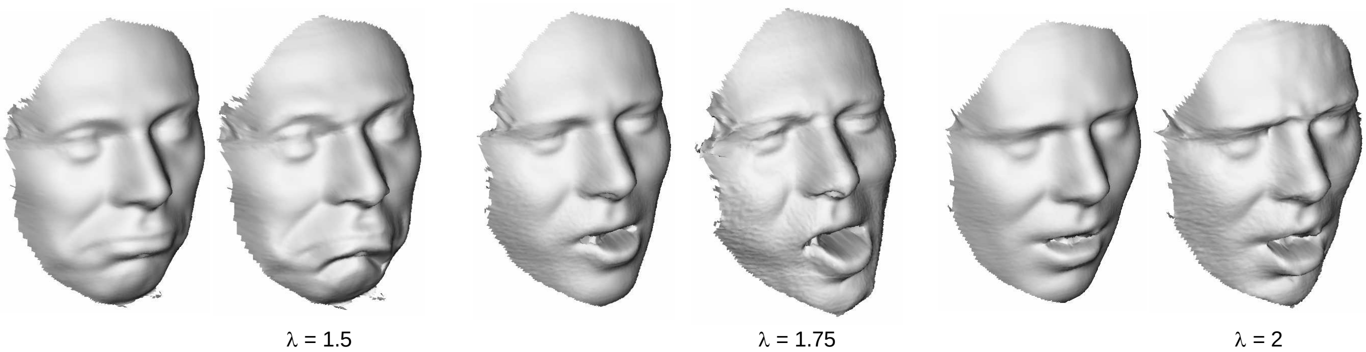

Fig. 10. Expression exaggeration. Shown left to right are three expressions and their exaggerated versions.

Fig. 11. Texture substitution. GMDS is used to find the minumum-distortion mapping between face $\mathcal{S}$ and $\mathcal{Q}$ (by embedding $\mathcal{S}$ into $\mathcal{Q}$). Using this mapping as a correspondence, the texture α is mapped onto $\mathcal{Q}$.

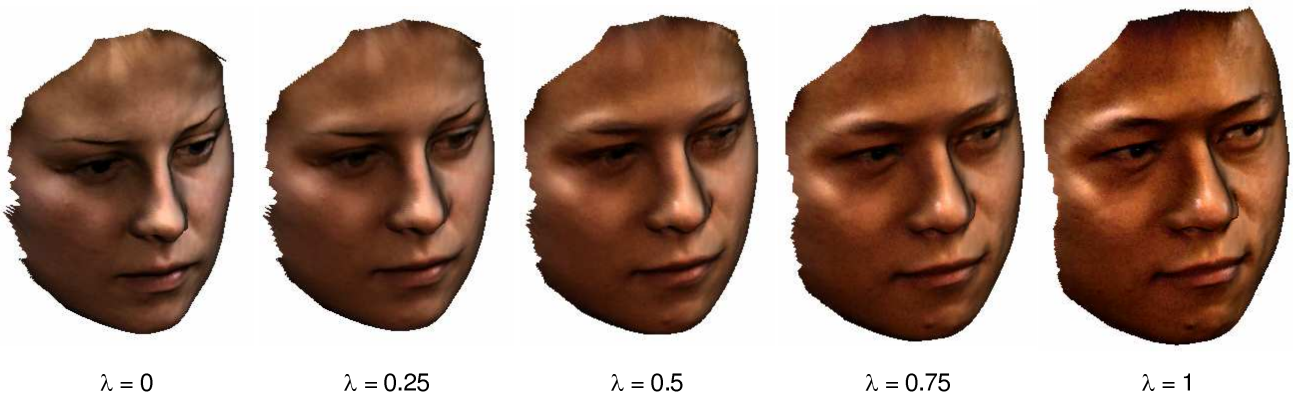

Fig. 12. Morphing. In the example from Figure 11, the correspondence is used to transform the texture and the extrinsic geometry of $\mathcal{S}$ into the corresponding texture and extrinsic geometry of $\mathcal{Q}$, creating a morphing effect.

the extrinsic geometry and the texture according to (8).

VII. Conclusion

We presented a procedure for establishing dense correspondence between non-rigid surfaces. Exploiting the empirical fact that facial expressions can be modeled as isometries, our approach is based on finding the minimum-distortion mapping between two surfaces. This is carried out by a procedure similar to multidimensional scaling. The algorithm is computationally efficient, though currently not real-time. Our preliminary results show that near-real-time performance can be achieved by exploiting multigrid optimization [10], [11] for the GMDS and implementation on graphics processors (GPU).

Being purely geometric, our approach is applicable when texture is not available. Since it is based on the intrinsic geometry (geodesic distances measured on the surfaces), the method is completely automatic and does not require any alignment based on the extrinsic geometry. Unlike feature-based methods, our approach does not require feature detection and tracking. We find correspondence between an arbitrarily dense set of points, as opposed to feature-based methods,

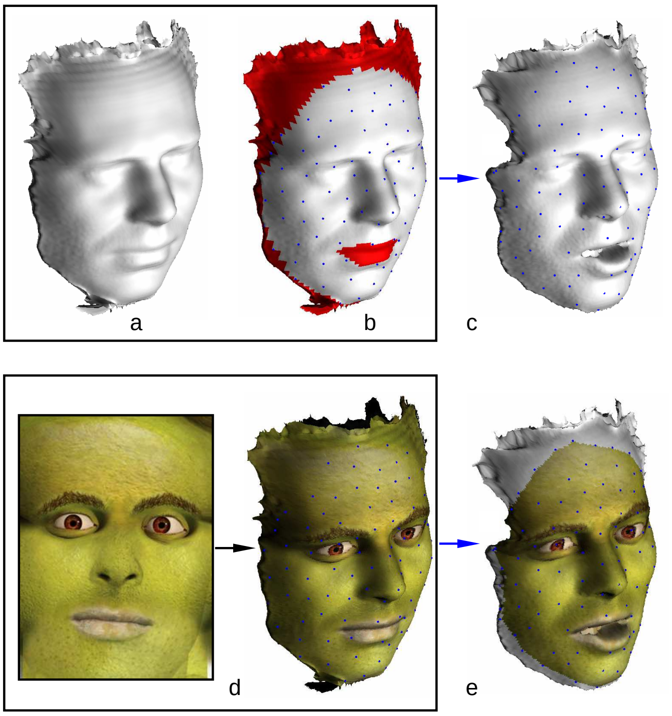

Fig. 7. Processing stages in the virtual makeup problem: (a) reference surface; (b) cropping and subsampling; (c) correspondence establishment using GMDS; (d) texture mapping onto the reference surface; (e) texture mapping onto the target surface.

which are usually limited to a small set of fiducial points that can be robustly detected and tracked. An additional advantage is that the minimum-distortion embedding approach uses a global criterion for finding the correspondence. This is especially important when working with noisy data. The fact that we use geodesic distances between all the points can be thought of as a means of regularization, which usually prevents outliers from compromising the correspondence quality. From this perspective, we can think of GMDS as a generalization of the elastic graph matching approaches [31], [32]. Finally, handling missing data is natural in our approach using the weighted generalized stress minimization.

The proposed method has a wide range of application in computer graphics and computer vision. We demonstrated only a few of them, including invariant texture mapping onto animated object, expression synthesis and exaggeration, texture substitution and morphing.

REFERENCES

[1] M. Alexa, "Merging polyhedral shapes with scattered features," *The Visual Computer*, vol. 16, pp. 26–37, 2000.
[2] D. Bertsekas, *Nonlinear programming*, 2nd ed. Atlanta Scientific, 1999.
[3] V. Blanz, C. Basso, T. Poggio, and T. Vetter, "Reanimating faces in images and video," *Computer Graphics Forum*, vol. 22, no. 3, pp. 641–650, 2003.
[4] I. Borg and P. Groenen, *Modern multidimensional scaling - theory and applications*. Springer-Verlag, Berlin Heidelberg New York, 1997.
[5] S. E. Brennan, "The caricature generator," *Leonardo*, vol. 18, pp. 170–178, 1985.
[6] A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Three-dimensional face recognition," *IJCV*, vol. 64, no. 1, pp. 5–30, August 2005.
[7] ——, "Efficient isometry-invariant surface matching," Dept. of Computer Science, Technion, Israel, Tech. Rep., 2006.
[8] ——, "Face2face: an isometric model for facial animation," in *Proc. AMDO*, 2006, to appear.
[9] ——, "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching," *Proc. National Academy of Sciences*, vol. 103, no. 5, pp. 1168–1172, January 2006.
[10] M. M. Bronstein, A. M. Bronstein, R. Kimmel, and I. Yavneh, "A multigrid approach for multidimensional scaling," in *Proc. Copper Mountain Conf. Multigrid Methods*, 2005.
[11] ——, "Multigrid multidimensional scaling," *Numerical Linear Algebra with Applications (NLAA)*, vol. 13, pp. 149–171, March-April 2006.
[12] K. Chang, K. W. Bowyer, and P. J. Flynn, "Face recognition using 2D and 3D facial data," in *ACM Workshop on Multimodal User Authentication*, 2003, pp. 25–32.
[13] G. J. Edwards, T. F. Cootes, and C. J. Taylor, "Face recognition using active appearance models," in *Proc. ECCV*, 1998.
[14] A. Elad and R. Kimmel, "On bending invariant signatures for surfaces," *IEEE Trans. PAMI*, vol. 25, no. 10, pp. 1285–1295, 2003.
[15] Y. Eldar, M. Lindenbaum, M. Porat, and Y. Y. Zeevi, "The farthest point strategy for progressive image sampling," *IEEE Trans. Image Processing*, vol. 6, no. 9, pp. 1305–1315, 1997.
[16] M. S. Floater and K. Hormann, *Advances on Multiresolution in Geometric Modelling*. Springer-Verlag, Heidelberg, 2004, ch. Surface Parameterization: a Tutorial and Survey, to appear.
[17] P. J. Flynn, K. W. Bowyer, and P. J. Phillips, "Assessment of time dependency in face recognition: an initial study," in *Audio- and Video-Based Biometric Person Authentication*, 2003, pp. 44–51.
[18] M. Gleicher, "Retargeting motion to new characters," in *Proc. ACM SIGGRAPH*, 1998, pp. 33–42.
[19] M. Gromov, *Structures métriques pour les variétés riemanniennes*, ser. Textes Mathématiques, 1981, no. 1.
[20] P. S. Huang, C. P. Zhang, and F. P. Chiang, "High speed 3-D shape measurement based on digital fringe projection," *Optical Engineering*, vol. 42, no. 1, pp. 163–168, 2003.
[21] R. Kimmel and J. A. Sethian, "Computing geodesic on manifolds," in *Proc. US National Academy of Science*, vol. 95, 1998, pp. 8431–8435.
[22] V. Kraevoy, A. Sheffer, and C. Gotsman, "Matchmaker: Constructing constrained texture maps," in *Proc. SIGGRAPH*, 2003.
[23] Y. Lee, D. Terzopoulos, and K. Waters, "Realistic modeling for facial animation," in *Proc. SIGGRAPH*, vol. 16, 1995, pp. 55–62.
[24] F. Mémoli and G. Sapiro, "A theoretical and computational framework for isometry invariant recognition of point cloud data," *Foundations of Computational Mathematics*, 2005, to appear.
[25] J. Noh and U. Neumann, "Expression cloning," in *Proc. ACM SIGGRAPH*, 2001, pp. 277–288.
[26] F. Pighin, R. Szeliski, and D. H. Salesin, "Modeling and animating realistic faces from images," *IJCV*, vol. 50, no. 2, pp. 143–169, November 2002.
[27] E. Praun, W. Sweldens, and P. Schröder, "Consistent mesh parameterizations," in *Proc. ACM SIGGRAPH*, 2001, pp. 179–184.
[28] R. W. Sumner and J. Popović, "Deformation transfer for triangle meshes," in *Proc. ACM SIGGRAPH*, 2004.
[29] J. B. Tennenbaum and V. de Silva abd J. C Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
[30] T. Vetter and V. Blanz, "A morphable model for the synthesis of 3D faces," in *Proc. SIGGRAPH*, 1999.
[31] L. Wiskott, *Labeled graphs and dynamic link matching for face recognition and scene analysis*, ser. Reihe Physik. Thun - Frankfurt am Main: Verlag Harri Deutsch, 1995, no. 53, phD thesis.
[32] L. Wiskott, J. Fellous, N. Kruger, and C. von der Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. PAMI*, vol. 19, no. 7, pp. 733–742, July 1997.
[33] J.-B. Yu and J.-H. Chuang, "Consistent mesh parameterizations and its application in mesh morphing."
[34] K. Zhou, J. Snyder, B. Guo, and H.-Y. Shum, "Iso-charts: Stretch-driven mesh parameterization using spectral analysis," in *Proc. ACM SGP*, 2004, pp. 45–54.
[35] G. Zigelman, R. Kimmel, and N. Kiryati, "Texture mapping using surface flattening via multi-dimensional scaling," *IEEE Trans. Visualization and computer graphics*, vol. 9, no. 2, pp. 198–207, 2002.