T U M

I N S T I T U T F Ü R I N F O R M A T I K

Minimal Surfaces for 3D Shape Matching: A Linear Programming Solution

Thomas Windheuser, Ulrich Schlickewei and Daniel Cremers

TUM-I1101
Januar 11

T E C H N I S C H E U N I V E R S I T Ä T M Ü N C H E N

TUM-INFO-01-I1101-0/1.-FI

Druck: Institut für Informatik der
Technischen Universität München

Minimal Surfaces for 3D Shape Matching: A Linear Programming Solution

Thomas Windheuser, Ulrich Schlickewei and Daniel Cremers
Department of Computer Science, TU München

Abstract

We propose a novel method for computing a spatially dense matching between two $3D$ *shapes. It is based on finding a minimal surface in the high-dimensional space representing the product of two shape surfaces. Computationally, the proposed approach leads to a binary linear program whose relaxed version can be solved efficiently in a globally optimal manner. We consider two cost functions for matching. The first one aims at computing a matching which minimizes a thin-shell energy measuring the physical energy necessary to deform one shape into the other. The second one additionally imposes that corresponding elements on either surface should have a similar local feature descriptor. Experimental results demonstrate that the proposed LP relaxation allows to compute high-quality matchings which reliably put into correspondence articulated* $3D$ *shapes.*

1. Introduction

An increasing number of digitized three-dimensional objects has become available over the last years due to the technical progress in acquisition hardware like laser scanners or medical imaging devices. Such objects originate from a variety of different domains including biology, medicine, industrial design or computer animation. This rapid growth in stored data brings about the need for reliable algorithms to organize this data. One of the cornerstone problems in this context is the matching problem: Given two three-dimensional objects, find a meaningful correspondence between the object's surfaces. To date there is no efficient and optimal algorithm for this problem.

In this work, we propose a novel framework for finding an optimal matching between pairs of points on either surface. We formulate shape matching as a minimal surface problem which allows for a linear programming discretization. This model comes with a sound physical interpretation and allows to compute high-quality matching without need for initialization.

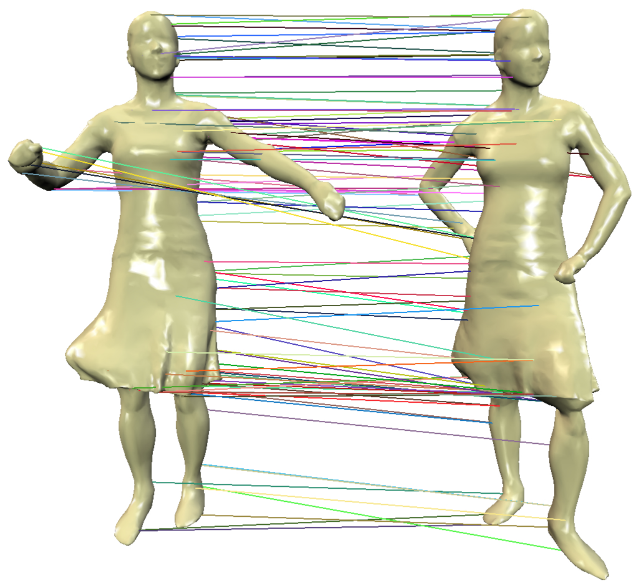


Figure 1. We propose to cast the matching of surfaces in $3D$ as a codimension-two minimal surface problem which aims at minimizing the distortion when transforming one shape into the other. We show that a consistent discretization of this minimal surface problem gives rise to an integer linear program. By means of LP relaxation we can compute near-optimal matchings such as the one shown above.

1.1. Related Work

Interestingly, in one less dimension the matching of *planar* shapes can be solved by means of dynamic programming in runtimes which are subcubic in the number of points on each shape [15]. Unfortunately the concepts of dynamic programming and Dijkstra's shortest path algorithm do not extend to higher-dimensions where the problem is no longer a shortest path problem. Therefore existing approaches for three-dimensional shape matching typically rely on *local* optmimization techniques.

The paradigm of the Gromov–Hausdorff framework, proposed by Mémoli and Sapiro in [13], is to find the correspondence which minimizes the geodesic distortion. Bronstein et al. [1] proposed an efficient method for computing such correspondences in a coarse-to-fine strategy much akin to optical flow algorithms. In [2] the same Gromov–Hausdorff was merged with the idea of diffusion distances.

Other approaches to shape matching employ techniques from conformal geometry [19, 22, 12] or Riemannian geometry [9]. The physically motivated energy model we use in this work is related to the works of Litke et al. [11] and of Rumpf and Wirth [20].

All the above-mentioned methods have in common that they use a local optimization technique to minimize a non-convex energy. As a consequence, the quality of solutions depends heavily on a good initialization and an appropriately designed coarse-to-fine strategy. In addition, solutions do not come with any optimality guarantees, which implies that in principle they can be arbitrarily bad.

Recently, two methods with a more *global* flavour have been proposed. On the one hand, Zeng and coworkers [23] formulate shape matching as a graph matching problem of third order and apply the QPBO algorithm [17]. Although the overall approach does not guarantee globally optimal solutions, it is able to detect when a proposed matching pair is globally optimal. Two major drawbacks of this approach are that firstly it suffers from a very high computational complexity, considering all triples of possible matchings. In practice it allows only the matching of a few feature points which is then postprocessed with a local method. Secondly, this approach lacks a continuous counterpart, as it merely matches discrete points rather than surface elements.

On the other hand, Lipman and Daubechies [10] recently proposed to compare surfaces of genus zero and open surfaces using optimal mass transport and conformal geometry. Computationally, this amounts to solving a linear programm in n^2 variables where n is the number of vertices used in the discretization of the surfaces. The problem with this approach is that no spatial regularity is imposed on the matchings.

1.2. Contribution

We propose a novel formulation for the shape matching problem based on finding an optimal surface of codimension 2 in the product of the two shape surfaces. This surface minimizes the physical deformation energy needed for deforming one shape into the other. We derive a consistent discretization of the continuous framework and show that the discrete minimal surface problem amounts to a linear program. Compared to existing approaches the proposed framework has the following advantages:

- The LP formulation is a global approach allowing to compute matchings which are independent of initialization with no postprocessing.
- The proposed method guarantees for spatially consistent matchings.
- We provide a discretization of the set of surface diffeomorphisms by means of linear constraints. This is quite remarkable because in previous formulations the diffeomorphism constraint is highly non-linear and computationally very difficult [21].
- The algorithmic formulation is independent of the particular choice of deformation energy and can be applied universally. As an example, we show that one can also incorporate local feature similarity in order to improve performance.
- Experiments demonstrate that reliable and dense matchings are obtained even for larger problem instances with no need for postprocessing.

2. From Shape Matching to Minimal Surfaces: The Continuous Setting

In this section we outline a shape matching model which is based on minimizing physically motivated energies. We then show how this problem can be translated into an equivalent problem of finding a minimal codimension-two surface in a four-dimensional space.

2.1. Shape Matching based on Minimizing Deformation Energies

Let $X, Y \subset \mathbb{R}^3$ be two differentiable, oriented, closed surfaces. We formulate the shape matching problem as an optimization problem over the set of orientation preserving diffeomorphisms between X and Y

$$\inf_{f \in \mathrm{Diff}^+(X,Y)} E(f) + E(f^{-1}) \tag{1}$$

where E is a suitable energy on the class of all diffeomorphisms between surfaces and $\mathrm{Diff}^+(X,Y)$ is the set of orientation preserving diffeomorphisms between X and Y. Note that we choose a symmetric problem formulation, penalizing at the same time deformation energy of X into Y and of Y into X. This is necessary because usually E takes different values on f and on f^{-1}.

The energy functional we use is borrowed from elasticity theory in physics [3]. Interpret the surfaces X and Y as "thin shells". Now we try to find the deformation of X into Y which requires the least stretching and bending energy. Such models usually consist of a membrane energy E_{mem} and a bending energy E_{bend} penalizing deformations in the first and in the second fundamental forms of the surfaces. In our work we use the following formulation:

$$\begin{aligned} E(f) = & \underbrace{\int_X (\mathrm{tr}_{g_X} \mathbf{E}) + \mu \, \mathrm{tr}_{g_X}(\mathbf{E}^2)}_{E_{\mathrm{mem}}} \\ & + \lambda \underbrace{\int_X (H_X(x) - H_Y(f(x))^2}_{E_{\mathrm{bend}}} \end{aligned} \tag{2}$$

where $\mathbf{E} = f^* g_Y - g_X$ is the difference between the metric tensors of X and Y, typically called the *Lagrange strain tensor*, H_X and H_Y denote the mean curvatures and μ and λ are parameters which determine the elasticity and the bending property of the material. This energy is a slightly simplified version of Koiter's thin shell energy [8].

2.2. Diffeomorphisms and their Graph Surfaces

Given an orientation preserving diffeomorphism $f : X \to Y$ we obtain a surface $\Gamma \subset X \times Y$ in the Euclidean product of X and Y by passing to the graph

$$\Gamma = \{(x, f(x)) \mid x \in X\} \subset X \times Y. \tag{3}$$

The surface Γ comes with two natural projections $\pi_X : \Gamma \to X, (x, f(x)) \mapsto x$ and $\pi_Y : \Gamma \to Y, (x, f(x)) \mapsto f(x)$. A diffeomorphism is completely characterized by its graph:

Proposition 1 (graph surfaces)**.** *Let Γ be the graph of a diffeomorphism $f : X \to Y$. Then*

(i) Γ is a differentiable, connected, closed surface in $X \times Y$

(ii) The projections π_X and π_Y are both diffeomorphisms.

(iii) The two orientations which Γ naturally inherits from X and Y coincide.

Viceversa, any surface $\Gamma \subset X \times Y$ which satisfies (i),(ii) and (iii) is the graph of a diffeomorphism between X and Y. We call such surfaces **graph surfaces**

The energy $E(f)$ can be expressed as

$$E(f) = \tilde{E}(\Gamma) \tag{4}$$

where $\tilde{E}(\Gamma) = E(\pi_Y \circ (\pi_X)^{-1}) + E(\pi_Y \circ (\pi_Y)^{-1})$.

The outcome of the above discussion is that the optimization problem (1) can be phrased as an optimization problem over the set of surfaces in $X \times Y$ which then reads

$$\begin{array}{ll} \inf & \tilde{E}(\Gamma) \\ \text{subject to} & \Gamma \subset X \times Y \text{ is a graph surface} \end{array} \tag{5}$$

We remark that the idea of casting optimal diffeomorphism problems as minimal surface problems has been applied previously in the theory of nonlinear elasticity [7].

3. The Discrete Setting

In this section we develop a discrete counterpart of the notion of graph surfaces in $X \times Y$ introduced in Section 2.2. We start in 3.1 with the definition of discrete surface patches in $X \times Y$. These patches will be the building blocks for discrete graph surfaces intruduced in 3.2. Finally in 3.3 we give a discrete version of the energy.

3.1. Discrete Surface Patches

Let $X = (V_X, E_X, F_X)$ be a triangulated oriented surface mesh, consisting of a set of vertices V_X, of directed edges E_X and of oriented triangles F_X.

A priori, edges on X do not have a direction. Here we have fixed for each edge on X an arbitrary direction. Thus, whenever two vertices a_1 and a_2 of X are connected by an edge, either $\left(\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right) \in E_X$ or $\left(\begin{smallmatrix} a_2 \\ a_1 \end{smallmatrix}\right) = -\left(\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right) \in E_X$. We extend the set of edges by *degenerate edges*

$$\overline{E}_X = E_X \cup \{\left(\begin{smallmatrix} a \\ a \end{smallmatrix}\right) \mid a \in V_X\}. \tag{6}$$

By assumption, the triangles of X have an orientation. If the vertices a_1, a_2, a_3 build an oriented triangles on X, then $\left(\begin{smallmatrix} a_1 \\ a_2 \\ a_3 \end{smallmatrix}\right) = \left(\begin{smallmatrix} a_2 \\ a_3 \\ a_1 \end{smallmatrix}\right) = \left(\begin{smallmatrix} a_3 \\ a_1 \\ a_2 \end{smallmatrix}\right) \in F_X$. As for the edges, we extend the set of triangles by *degenerate triangles*

$$\overline{F}_X = F_X \cup \left\{ \left(\begin{smallmatrix} a_1 \\ a_2 \\ a_2 \end{smallmatrix}\right) \mid a_1, a_2 \in V_X,\ \pm\left(\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right) \in \overline{E}_X \right\}. \tag{7}$$

Notice that degenerate triangles can consist of only one or of two vertices.

Next, we introduce product surface triangles for two triangular meshes X and Y. Define the product of X and Y by the set of vertices $V = V_X \times V_Y$, the set of edges $E = \overline{E}_X \times \overline{E}_Y$ and the set of product triangles

$$F := \left\{ \begin{pmatrix} a_1, b_1 \\ a_2, b_2 \\ a_3, b_3 \end{pmatrix} \;\middle|\; \begin{array}{r} f_1 = \left(\begin{smallmatrix} a_1 \\ a_2 \\ a_3 \end{smallmatrix}\right) \in \overline{F}_X, \\ f_2 = \left(\begin{smallmatrix} b_1 \\ b_2 \\ b_3 \end{smallmatrix}\right) \in \overline{F}_Y, \\ f_1 \text{ or } f_2 \text{ non-degenerate} \end{array} \right\} \tag{8}$$

The product triangles in F are the basic pieces which are later glued to discrete graph surfaces. For shape matching, a product triangle

$$\begin{pmatrix} a_1, b_1 \\ a_2, b_2 \\ a_3, b_3 \end{pmatrix} \in F \tag{9}$$

is interpreted as setting vertex $a_i \in V_X$ in correpondence with vertex $b_i \in V_Y$.

3.2. Discrete Surfaces

As we have seen in Section 2 a diffeomorphism can be represented as a surface $\Gamma \subset X \times Y$ satisfying conditions (i), (ii) and (iii). In this section we derive discrete versions of these properties.

Definition 2. *A* **discrete surface in** $X \times Y$ *is a subset* $\Gamma \subset F$*. The set of all discrete surfaces is denoted by* $\mathrm{surf}(X \times Y)$.

Discrete version of (i): Recall that the boundary operator for triangle meshes [5] maps triangles to their oriented boundary. We extend this definition to the product graph G.

As for the sets E_X and E_Y we choose arbitrary orientations for each product edge $e \in E$. By means of these orientations we define for any edge $\binom{v_1}{v_2}$ connecting two vertices $v_1, v_2 \in V$ a vector in $O\binom{v_1}{v_2} \in \{-1, 0, 1\}^{|E|}$ whose e-th entry is given by

$$O\binom{v_1}{v_2}_e = \begin{cases} 1 & \text{if } e = \binom{v_1}{v_2} \\ -1 & \text{if } e = \binom{v_2}{v_1} \\ 0 & \text{else} \end{cases} \tag{10}$$

We remark that by definition of E_X and E_Y only one of the cases appears, so that exactly one entry of $O\binom{v_1}{v_2}$ is non-zero.

Definition 3. *The* boundary operator $\partial : F \to \{-1, 0, 1\}^{|E|}$ *is defined by*

$$\partial \begin{pmatrix} a_1, b_1 \\ a_2, b_2 \\ a_3, b_3 \end{pmatrix} := O\begin{pmatrix} a_1, a_2 \\ b_1, b_2 \end{pmatrix} + O\begin{pmatrix} a_2, a_3 \\ b_2, b_3 \end{pmatrix} + O\begin{pmatrix} a_3, a_1 \\ b_3, b_1 \end{pmatrix}, \tag{11}$$

where the $a_i \in V_X$ and $b_i \in V_Y$ form triangles on X resp. on Y.

The boundary operator is linearly extended to a map

$$\text{surf}(X \times Y) \to \mathbb{Z}^{|E|}. \tag{12}$$

A discrete surface in $X \times Y$ Γ is **closed** *if $\partial\Gamma = 0$.*

The natural discrete version of (i) is a closed connected surface in $X \times Y$.

Discete version of (ii): As in the continous setting we can project product triangles to the surfaces X and Y.

Definition 4. *The projection $\pi_X : F \to \{0, 1\}^{|F_X|}$ is defined by*

$$\pi_X(f) := \begin{cases} e_a & \text{if } a = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \text{ is non-deg.} \\ (0, \ldots, 0) & \text{else} \end{cases} \tag{13}$$

for each face $f = \begin{pmatrix} a_1, b_1 \\ a_2, b_2 \\ a_3, b_3 \end{pmatrix} \in F$. Here, e_a is the vector with 1 in the a-entry and 0 in all other entries.

We extend the projection π_X linearly to a map $\pi_X : \text{surf}(X \times Y) \to \mathbb{Z}^{|F_X|}$. The projection $\pi_Y : F \to \{0, 1\}^{|F_Y|}$ and its linear extension $\pi_Y : \text{surf}(X \times Y) \to \mathbb{Z}^{|F_Y|}$ are defined similarly.

Let now Γ be a discrete surface in $X \times Y$. Then we say that the projections of Γ to X and Y are discrete diffeomorphisms if and only if

$$\begin{aligned} \pi_X(\Gamma) &= (1, \ldots, 1) \in \mathbb{Z}^{|F_X|} \text{ and} \\ \pi_Y(\Gamma) &= (1, \ldots, 1) \in \mathbb{Z}^{|F_Y|}. \end{aligned} \tag{14}$$

This gives a discrete version of (ii).

Note that in this definition we do not ask for injectivity on the vertices set. This is necessary for modelling discretely strong compressions. However, conditions (14) ensure a global bijectivity property which is sufficient in our context.

Discrete version of (iii): By definition, the set of surfaces in $X \times Y$ only contains surface patches which are consistently oriented. Therefore any surface in $\text{surf}(X \times Y)$ satisfies condition (iii).

Definition 5. *Let $\Gamma \in \{0, 1\}^{|F|}$ be a discrete surface in $X \times Y$, represented by its indicator vector. Then Γ is a* **discrete graph surface** *in $X \times Y$ if*

$$\begin{pmatrix} \partial \\ \pi_X \\ \pi_Y \end{pmatrix} \cdot \Gamma = \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \\ \mathbf{1} \end{pmatrix}. \tag{15}$$

3.3. Discrete Surface Energy

Now we introduce a discrete energy on the set of product triangles in $X \times Y$. For the membrane energy in (2) we adopt the term proposed by Delingette [4]. Given two triangles $T_1, T_2 \subset \mathbb{R}^3$, Delingette computes the stretch energy $E_{\text{mem}}(T_1 \to T_2)$ necessary for deforming T_1 in T_2. In our framework we associate with each product triangle $(a, b) = \begin{pmatrix} a_1, b_1 \\ a_2, b_2 \\ a_3, b_3 \end{pmatrix} \in F$ the membrane cost

$$\begin{aligned} E_{\text{mem}}(a, b) := & E_{\text{mem}}\left(\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \to \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}\right) + \\ & E_{\text{mem}}\left(\begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \to \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}\right). \end{aligned} \tag{16}$$

For the bending term we proceed similarly associating with each product triangle (a, b) the cost

$$E_{\text{bend}}(a, b) = \int_a (H_X - H_Y)^2 + \int_b (H_Y - H_X)^2. \tag{17}$$

In practice we discretize the mean curvature following [14].

Next, we extend the energy linearly from discrete surface patches to discrete surfaces in $X \times Y$. Identify a discrete surface with its indicator vector $\Gamma \in \{0, 1\}^{|F|}$. Define the vector $E \in \mathbb{R}^{|F|}$ whose f-th entry is

$$E_f = E_{\text{mem}}(f) + E_{\text{bend}}(f). \tag{18}$$

Then the **discrete energy** of Γ is given by the vector product

$$E^t \cdot \Gamma. \tag{19}$$

4. Linear Programming Solution

In the previous section we have introduced a discrete notion of graph surfaces (15) and a discrete deformation energy (19) for such graph surfaces. This enables us to state the discrete version of (5):

$$\begin{array}{ll} \min & E^t \cdot \Gamma \\ \text{subject to} & \Gamma \in \{0,1\}^{|F|} \text{ and} \\ & A \cdot \Gamma = b \end{array} \tag{20}$$

where A is the matrix $\begin{pmatrix} \partial \\ \pi_X \\ \pi_Y \end{pmatrix}$ and $b = \begin{pmatrix} \mathbf{0} \\ \mathbf{1} \\ \mathbf{1} \end{pmatrix}$. This is a binary linear programm.

For solving (20), we relax the binary constraints to $\Gamma \in [0,1]^{|F|}$. This relaxed version can be solved globally optimally in polynomial time. We employed an alternating direction method developped by Eckstein et al. [6]. This algorithm is parallelizable which allowed us an efficient implementation on the GPU.

Since the constraint matrix of the relaxed problem is not totally unimodular, we are not guaranteed an integral solution. A simple thresholding scheme would destroy the geometric consistency of the solution. Therefore, for obtaining an integral solution we revert to a hyperplane cutting method, that is we successively fix the maximum value to 1 and solve the residual problem until an integral solution is reached.

5. Experimental Results

We have introduced a framework for computing dense matchings between $3D$ shapes using LP relaxation. The following experimental results are aimed at highlighting various properties of the proposed method.

5.1. Matching of Articulated Shapes

A common problem in shape matching is that the same shape may undergo substantial deformation and articulation. Nevertheless, one would like to reliably identify corresponding structures. Figure 2 shows the matching computed for two different articulations of a dancer. Although arms and legs are in very different positions and the skirt of the dancer is deformed the proposed method identified the correct matching. Since the proposed framework enforces geometric consistency matching errors occur only on a small spatial scale. In contrast to methods without spatial regularization strong outliers such as single points matched to the wrong leg do not arise.

5.2. Matching Near-Symmetrical Shapes

In the upper image of figure 3 we match on left hand to another left hand. The algorithm determines the desired matching. In the lower image a right hand is matched to a left hand. Interestingly, these two shapes are near-symmetric but not isometric. Rather than matching thumb-to-thumb and thus the back of one hand to the front of the other, the algorithm determined that the matching of thumb to pinkie provides a smaller deformation energy.

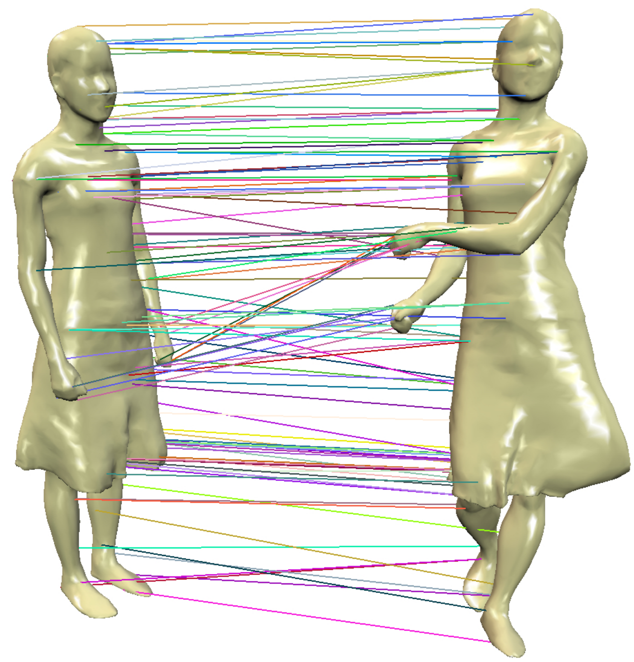


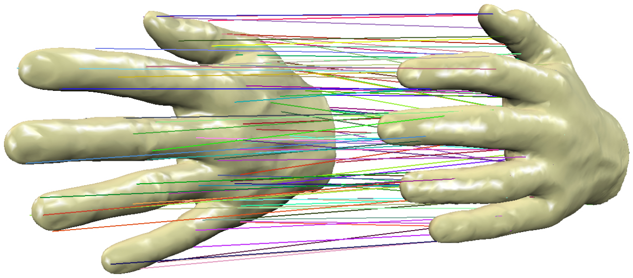
Figure 2. **Matching of a person in different poses.** Despite large gestures of the dancer and a deformed skirt the proposed method matches arms and legs correctly. The framework guarantees geometric consistency and thereby prevents strong outliers.

5.3. On the Influence of Feature Descriptors

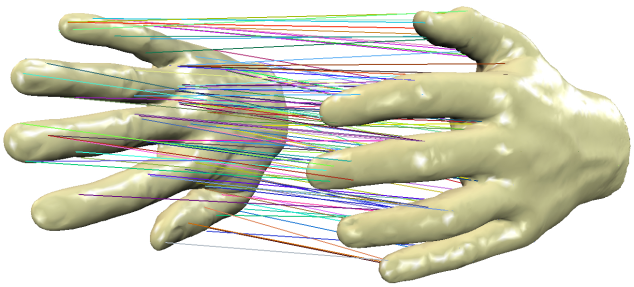
As the cost coefficients in the proposed linear programming approach can be arbitrarily chosen, we are not restricted to the proposed physical deformation model. An alternative can be a feature descriptor which assigns a cost to each possible match between vertices. The upper image in Figure 4 shows the result computed with the physical deformation model. In the lower image we added the Heat Kernel Signature [16] to the cost function so as to favor local feature similarity of corresponding surface elements. Interestingly there is no substantial visible effect of the feature descriptor showing that the proposed physical deformation model is indeed quite powerful. In some experiments, however, we found that imposing feature similarity leads to faster convergence of the algorithm.

6. Conclusion

We proposed to cast the spatially dense matching of $3D$ shapes as a minimal surface problem in a higher-dimensional space spanned by the two shapes. We consider two alternative cost functionals, the first one measuring the physical energy associated with deforming one shape into the other, the second one imposing a local feature similarity of corresponding surface elements. We showed that a consistent discretization leads to an integer linear program. As



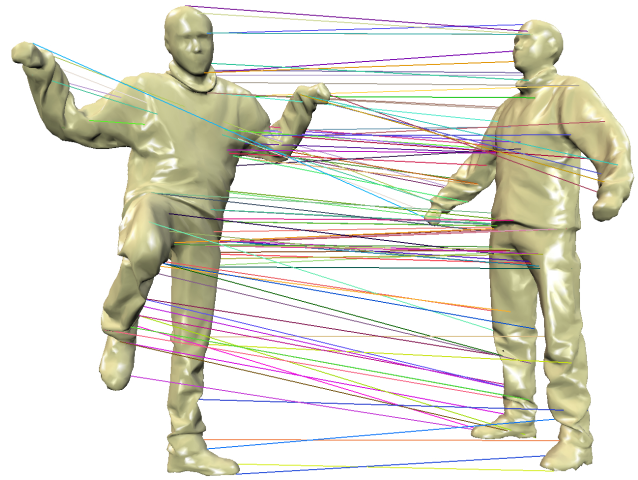
Matching two left hands



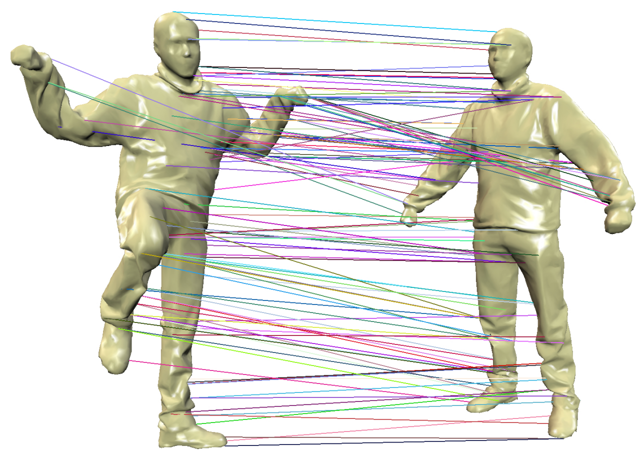
Matching a right hand and a left hand

Figure 3. **Matching Near-Symmetrical Shapes** While matching two left hands provides the desired matching of thumb-on-thumb, for the case of matching right hand to left hand, the algorithm determines that matching thumb-to-pinkie gives rise to a smaller deformation energy than trying to match the front of one hand to the back of the other.

a consequence, we can compute high-quality solutions to the matching problem which are independent of initialization by means of LP relaxation. Experimental results confirm that the proposed method generates reliable dense correspondences for a variety of articulated real-world shapes.



Matching with thin shell energy



Matching with additional local feature similarity

Figure 4. **Comparison of different matching energies.** Surprisingly, the matching obtained using a highly informative local feature descriptor is essentially identical with the matching computed based on the deformation deformation energy only, indicating that the proposed thin shell energy alone provides for reliable matching results.

Acknowledgment

All 3D shape data is courtesy of [18], except for the hand data which is courtesy of `AimAtShape.net`.

References

[1] A. Bronstein, M. Bronstein, and R. Kimmel. Efficient computation of isometry-invariant distances between surfaces. *SIAM J. Sci. Comput.*, 28(5):1812–1836 (electronic), 2006. 1

[2] A. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro. A gromov-hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *International Journal of Computer Vision*, 89(2-3):266–286, 2010. 2

[3] P. Ciarlet. *An introduction to differential geometry with applications to elasticity*. Springer, Dordrecht, 2005. Reprinted from J. Elasticity **7**8/79 (2005), no. 1-3 [MR2196098]. 2

[4] H. Delingette. Triangular springs for modeling nonlinear membranes. *IEEE Transactions on Visualisation and Computer Graphics*, 14(2), March/April 2008. 4

[5] M. Desbrun, A. N. Hirani, M. Leok, and J. E. Marsden. Discrete exterior calculus. preprint, arxiv:0508341v2, 2005. 4

[6] J. Eckstein and D. Bertsekas. An alternating direction method for linear programming. Technical report, Harvard Business School, 1990. 5

[7] M. Giaquinta, G. Modica, and J. Souček. *Cartesian currents in the calculus of variations. II*, volume 38 of *Ergebnisse der Mathematik und ihrer Grenzgebiete.* Springer-Verlag, Berlin, 1998. Variational integrals. 3

[8] W. Koiter. On the nonlinear theory of thin elastic shells. I, II, III. *Nederl. Akad. Wetensch. Proc. Ser. B*, 69:1–17, 18–32,

33–54, 1966. 3

[9] S. Kurtek, E. Klassen, Z. Ding, and A. Srivastava. A novel riemannian framework for shape analysis of 3d objects. In *CVPR*, pages 1625–1632, 2010. 2

[10] Y. Lipman and I. Daubechies. Surface comparison with mass transportation. preprint, arxiv:0912.3488, 2009. 2

[11] N. Litke, M. Droske, M. Rumpf, and P. Schröder. An image processing approach to surface matching. In *Symposium on Geometry Processing*, pages 207–216, 2005. 2

[12] L. Lui, T. Wong, P. Thompson, T. Chan, X. Gu, and S. Yau. Shape-based diffeomorphic registration on hippocampal surfaces using beltrami holomorphic flow. In *MICCAI (2)*, pages 323–330, 2010. 2

[13] F. Mémoli and G. Sapiro. A theoretical and computational framework for isometry invariant recognition of point cloud data. *Foundations of Computational Mathematics*, 5(3):313–347, 2005. 1

[14] M. Meyer, M. Desbrun, P. Schröder, and A. Barr. Discrete differential-geometry operators for triangulated 2-manifolds, 2002. 4

[15] F. R. Schmidt, D. Farin, and D. Cremers. Fast matching of planar shapes in sub-cubic runtime. In *IEEE Int. Conf. on Computer Vision*, Rio de Janeiro, October 2007. 1

[16] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Comput. Graph. Forum*, 28(5):1383–1392, 2009. 5

[17] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. In *ECCV (2)*, pages 596–609, 2008. 2

[18] D. Vlasic, I. Baran, W. Matusik, and J. Popović. Articulated mesh animation from multi-view silhouettes. In *ACM SIGGRAPH 2008 papers*, SIGGRAPH '08, pages 97:1–97:9, New York, NY, USA, 2008. ACM. 6

[19] Y. Wang, X. Gu, K. Hayashi, T. Chan, P. Thompson, and S. Yau. Brain surface parameterization using riemann surface structure. In *MICCAI (2)*, pages 657–665, 2005. 2

[20] B. Wirth, L. Bar, M. Rumpf, and G. Sapiro. Geodesics in shape space via variational time discretization. In *Proceedings of the 7th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR'09)*, volume 5681 of *LNCS*, pages 288–302, 2009. 2

[21] L. Younes. *Shapes and diffeomorphisms*, volume 171 of *Applied Mathematical Sciences*. Springer-Verlag, Berlin, 2010. 2

[22] W. Zeng, Y. Zeng, Y. Wang, X. Yin, X. Gu, and D. Samaras. 3d non-rigid surface matching and registration based on holomorphic differentials. In *ECCV (3)*, pages 1–14, 2008. 2

[23] Y. Zeng, C. Wang, Y. Wang, X. Gu, D. Samaras, and N. Paragios. Dense non-rigid surface registration using high-order graph matching. In *CVPR*, pages 382–389, 2010. 2