Eurographics Symposium on Geometry Processing 2011
Mario Botsch and Scott Schaefer
(Guest Editors)

Volume 30 (2011), Number 5

Deformable 3D Shape Registration Based on Local Similarity Transforms

C. Papazov and D. Burschka

Technische Universität München (TUM), Robotics and Embedded Systems, Germany

Abstract

In this paper, a new method for deformable 3D shape registration is proposed. The algorithm computes shape transitions based on local similarity transforms which allows to model not only as-rigid-as-possible deformations but also local and global scale. We formulate an ordinary differential equation (ODE) which describes the transition of a source shape towards a target shape. We assume that both shapes are roughly pre-aligned (e.g., frames of a motion sequence). The ODE consists of two terms. The first one causes the deformation by pulling the source shape points towards corresponding points on the target shape. Initial correspondences are estimated by closest-point search and then refined by an efficient smoothing scheme. The second term regularizes the deformation by drawing the points towards locally defined rest positions. These are given by the optimal similarity transform which matches the initial (undeformed) neighborhood of a source point to its current (deformed) neighborhood. The proposed ODE allows for a very efficient explicit numerical integration. This avoids the repeated solution of large linear systems usually done when solving the registration problem within general-purpose non-linear optimization frameworks. We experimentally validate the proposed method on a variety of real data and perform a comparison with several state-of-the-art approaches.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—

1. Introduction

Deformable (non-rigid) shape registration is a fundamental problem in computational geometry with applications in the fields of computer vision, computer graphics, medical image processing and many others. In recent years, 3D geometry acquisition techniques have been developed which allow to capture the surface of deforming objects in real time [WLG07]. In order to analyze the motion of the object it is important to register subsequent scans and/or to register a complete geometric model to the scans. Since the object is undergoing a non-rigid motion, rigid registration algorithms [CM91, BM92, GMGP05, PB09] can not be used adequately in this setting.

The problem of deformable shape registration can loosely be defined as follows. Given a source shape $\mathcal{S}$ and a target shape $\mathcal{T}$ find a "reasonable" deformation F that brings $\mathcal{S}$ "close" to $\mathcal{T}$. In this paper, we assume that $\mathcal{S}$ and $\mathcal{T}$ are consisting of a finite set of points with an underlying neighborhood structure. Examples include range images, meshes and volumetric grids, just to name a few. In this case, the deformation we are looking for is a mapping $F : \mathcal{S} \to \mathbb{R}^3$. To choose a reasonable one from the space of all mappings, we have to impose some constraints on the deformation. This is called regularization of F. We use a regularizer that pulls each source shape point $\mathbf{x}_k$ towards its rest position given by the optimal similarity transform which matches the initial (undeformed) neighborhood of $\mathbf{x}_k$ to its current (deformed) neighborhood. This is a generalization of the object deformation technique presented in [MHTG05, RJ07, SOG08], where local rigid shape matching is used. Employing similarity transforms instead of rigid ones allows to model as-rigid-as-possible shape deformations plus local scale. Note that the topic of the papers [MHTG05, RJ07, SOG08] is the generation of physically plausible animations. To the best of our knowledge, this is the first paper which exploits rigid and similarity-based shape matching as regularizers in the context of deformable 3D shape registration.

In order to deform the source shape such that it comes closer to the target shape $\mathcal{T}$, each source point moves towards a corresponding point on $\mathcal{T}$. We use closest-point search to establish preliminary correspondences which are further refined by a simple but effective vector field smoothing procedure.

Considering the regularizer and the correspondence field, we introduce a system of ordinary differential equations (ODEs) which describes the non-rigid motion of the source shape. The iterative solution of the ODEs yields a trajectory for each source point from its initial position to its end position on the target shape. In the case of incomplete data, the points move according to the regularizer and fill in missing regions in a reasonable way. Our method computes a dense correspondence between source and target. Since the initial correspondence estimation is based on closest-point computations we assume that both shapes are roughly pre-aligned. This assumption holds in a variety of situations like, e.g., in the case of scanning a deforming object at high frame rates such that the inter-frame displacements are small.

The rest of the paper is organized as follows. After reviewing previous work in Section 2, we describe our algorithm in Section 3. Important implementation issues are discussed in Section 4. Section 5 presents experimental results. Conclusions are drawn in the final Section 6 of the paper.

2. Related Work

There is a large variety of deformable registration algorithms each one having its advantages and drawbacks. In this paper, our main criterion to judge the methods is the processing time they require.

One class of deformable registration approaches consists of the feature-based methods. Several papers [WAS10, WZL*10, BK10, RBBK10] proposed to use local invariant geometric descriptors to compute a one-to-one mapping between corresponding features on the input shapes. However, detecting feature points and establishing the correspondence can be problematic especially in the presence of noise and missing data. Furthermore, many shapes do not have distinctive features which gives rise to many ambiguous correspondences and the matching algorithm degenerates to a brute force search [AMCO08].

A different strategy is to transform the shapes to a canonical representation in a suitable space in which the correspondence problem is easier to solve. Several papers [EK03, BBK06, WSBA07, WSB09] proposed to compute isometry-invariant embeddings of the original shapes in a low-dimensional Euclidean space and to establish the correspondence using rigid registration algorithms. These methods, however, tend to be costly and, moreover, fail for incomplete data (caused by surface holes, partial views, *etc.*).

The methods cited so far solve the correspondence problem even in the presence of significant deformations and without making any assumptions about an initial alignment of the shapes. However, the deformations are restricted to isometries (an exception is [BK10] which can handle an additional global or local scaling). Furthermore, the actual warp between the shapes has to be computed in an additional step using the established correspondences as constraints [MHTG05, RJ07, BPWG07, SOG08]. In contrast to this, our method is not restricted to a particular family of transformations and it efficiently computes both a dense correspondence and the warp between the shapes.

There is a variety of registration algorithms specialized to articulated shapes. [ACP03] presented a framework for deformable marker-based fitting of a high-resolution template to 3D scans of different humans in the same pose. In [ASK*05] a deformable model was learned that is able to synthesize realistic muscle deformations based on the pose of an articulated human skeleton. Both methods can be used for human shape completion as well. Further shape completion algorithms which use deformable registration modules were presented in [KS05, PMG*05]. In [CZ08], a fully automatic approach for articulated shape registration was proposed. The registration problem is converted to a discrete labeling problem and solved via graph cuts. However, this seems to be very costly since the authors report processing times of more than an hour for shapes consisting of not more than 12,000 points.

A further class of non-rigid registration algorithms consists of iterative solvers. They deform the source shape in an iterative fashion until an “optimal” alignment to the target shape is achieved. Many methods in this class are extensions of the classic ICP algorithm [CM91, BM92]. In [IGL03], a non-rigid registration technique was introduced which decomposes the input scans in a coarse-to-fine hierarchical manner in overlapping rigid pieces which are aligned separately. However, the resulting deformation is not continuous which can lead to artifacts in the overlapping regions. Furthermore, the procedure has a quadratic time complexity in the number of pieces. In [BR04], the discontinuity issue was resolved by incorporating a global thin-plate splines warp which guarantees the smoothness of the solution. A generalization of this method to the simultaneous matching of multiple scans was proposed in [BR07].

Instead of assuming a one-to-one correspondence between the shape points, one-to-many relaxations can be used in order to enlarge the basin of convergence and thus to increase robustness against imprecise initializations. Significant contributions along these lines are the softassign and deterministic annealing technique [CR03] and the coherent point drift algorithm [MS10]. A statistical registration approach without explicitly establishing point-to-point correspondences was proposed in [TK04]. The input point sets are modeled as probability distributions and a distance measure between them is minimized over the transform space. Recently, a Gaussian mixture models-based approach [JV11] was proposed which can be seen as a generalization of [CR03, TK04, MS10]. However, these algorithms compute registration results which are not as precise as ours and are much slower than our method (see the experimental comparisons in Section 5).

A deformable ICP extension was introduced in [ARV07]. The authors formulated a cost function, similar to the one used in [ACP03], which measures the cost of a given non-rigid alignment between the shapes. The deformation is modeled using one affine 3×4 transformation matrix per shape point. This gives rise to a cost function of $12m$ variables, where m is the number of points in the source shape. In order to solve this highly over-determined system, a stiffness term (a regularizer) is introduced. It penalizes differences between the transformation matrices of neighboring points.

A similar strategy was proposed in [SAY*09]. The authors iteratively minimized an error measure which is based on an elastic convolution between the difference of corresponding points in the shapes. This is the 3D surface patch analog to 2D template matching commonly used in image processing. In [LSP08], the deformation is also modeled using one affine transformation matrix per point. The cost function comprises four energy terms and depends on $15m+6$ variables. The authors minimized it with the Levenberg-Marquardt algorithm.

Note that the iterative methods cited above model the deformation in a very redundant way: many more degrees of freedom (DoFs) are introduced than needed to describe an arbitrary motion of a system of m points in $\mathbb{R}^3$. This results in high-dimensional and computationally heavy optimization problems. In contrast to this, our approach is based on a system of ODEs with $3m$ unknowns, which is the least number necessary to model a general motion of a system with $3m$ DoFs. Furthermore, since the proposed ODEs system allows for a very efficient explicit integration we avoid to repeatedly solve large linear systems which is usually done during the minimization of non-linear cost functions. Thus, our method is efficient in terms of both computational complexity and memory.

3. Method Description

Before we describe our deformable shape registration algorithm in detail let us first introduce some notation used in the paper. Consider the source shape $\mathcal{S} = (\mathbf{V}_S, \mathbf{E}_S)$, where $\mathbf{V}_S = \{\mathbf{x}_1^0, \dots, \mathbf{x}_m^0\} \subset \mathbb{R}^3$ is the set of initial positions of the points $\mathbf{x}_1, \dots, \mathbf{x}_m$ and $\mathbf{E}_S = \{\mathbf{N}_1, \dots, \mathbf{N}_m\}$ is the neighborhood structure. Each $\mathbf{N}_k$ contains the indices (including k) of the neighbors of $\mathbf{x}_k$. The position of $\mathbf{x}_k$ at a time t is given by a function $t \mapsto \mathbf{x}_k(t)$, with initial value $\mathbf{x}_k(0) = \mathbf{x}_k^0$. Let $\mathbf{X}_k(t)$ denote the position of $\mathbf{x}_k$ and its neighbors at a time t, i.e., $\mathbf{X}_k(t) = (\mathbf{x}_{k_1}(t), \dots, \mathbf{x}_{k_{N_k}}(t))$, where $\{k_1, \dots, k_{N_k}\} = \mathbf{N}_k$.

Analogously, the target shape is a pair $\mathcal{T} = (\mathbf{V}_T, \mathbf{E}_T)$ with $\mathbf{V}_T = \{\mathbf{y}_1, \dots, \mathbf{y}_n\} \subset \mathbb{R}^3$ being the set of points and $\mathbf{E}_T = \{\mathbf{M}_1, \dots, \mathbf{M}_n\}$ being the neighborhood structure. Note that since $\mathcal{T}$ does not deform we use $\mathbf{y}_1, \dots, \mathbf{y}_n$ to denote both the target points and their positions in $\mathbb{R}^3$.

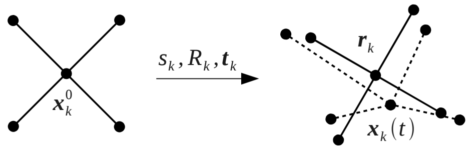

Figure 1: *The initial (undeformed) neighborhood $\mathbf{X}_k(0)$ (shown on the left) is matched to the current (deformed) neighborhood $\mathbf{X}_k(t)$ (indicated by the dashed lines on the right) using the similarity transform which minimizes the sum of squared distances between the corresponding points $\mathbf{x}_i^0 \leftrightarrow \mathbf{x}_i(t)$. The rest position $\mathbf{r}_k$ for $\mathbf{x}_k(t)$ is the transformed point $\mathbf{x}_k^0$.*

3.1. Shape Matching-Based Regularization

The regularizer pulls each point $\mathbf{x}_k$ towards its rest position $\mathbf{r}_k$ which is computed in the following way:

$$\mathbf{r}_k = s_k R_k \mathbf{x}_k^0 + \mathbf{t}_k. \tag{1}$$

$s_k \in \mathbb{R}$ is the scale factor, $R_k \in SO(3)$ is the rotation matrix and $\mathbf{t}_k \in \mathbb{R}^3$ is the translation vector which optimally match the initial (undeformed) neighborhood $\mathbf{X}_k(0)$ of $\mathbf{x}_k$ to its current (deformed) neighborhood $\mathbf{X}_k(t)$. More formally,

$$(s_k, R_k, \mathbf{t}_k) = \underset{s, R, \mathbf{t}}{\operatorname{argmin}} \sum_{i \in \mathbf{N}_k} \|(sR\mathbf{x}_i^0 + \mathbf{t}) - \mathbf{x}_i(t)\|^2. \tag{2}$$

Note that s_k, R_k and $\mathbf{t}_k$ depend on the current positions of the neighbors of $\mathbf{x}_k$. This means that s_k, R_k, $\mathbf{t}_k$ and the rest position $\mathbf{r}_k$ are functions of $\mathbf{X}_k(t)$. To stress this, when necessary, we write $\mathbf{r}_k(\mathbf{X}_k(t))$. Fig. 1 illustrates the idea of the regularization based on similarity shape matching.

The minimization problem (2) is called the absolute orientation problem and is often encountered in different fields as part of different computational problems [BM92, MHTG05, MS09]. Our solution is based on [MS09]. First, a linear deformation matrix A_k is computed:

$$A_k = \sum_{i \in \mathbf{N}_k} (\mathbf{x}_i(t) - \mathbf{c}_k(t))(\mathbf{x}_i^0 - \mathbf{c}_k^0)^T, \tag{3}$$

where the center of mass of the initial and the deformed neighborhood of $\mathbf{x}_k$ are denoted by $\mathbf{c}_k^0$ and $\mathbf{c}_k(t)$, respectively. Next, the optimal rotation matrix R_k is extracted from A_k using its singular value decomposition $A_k = U\Sigma V^T$ in the following way:

$$R_k = UCV^T, \quad C = \operatorname{diag}(1, \dots, 1, \det(UV^T)), \tag{4}$$

where the diagonal matrix C assures that R_k is a rotation and not a reflection. The scale factor is given by

$$s_k = \sqrt{\frac{\sum_{i \in \mathbf{N}_k} \|\mathbf{x}_i(t) - \mathbf{c}_k(t)\|^2}{\sum_{i \in \mathbf{N}_k} \|\mathbf{x}_i^0 - \mathbf{c}_k^0\|^2}} \tag{5}$$

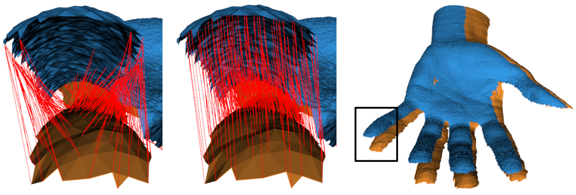

Figure 2: *The correspondence field computed with a closest-point search (a) and with our smoothing procedure (b) for the input shapes on the right. The black rectangle on the right marks the part of the little finger magnified in (a) and (b). Obviously, our procedure estimates the correspondences much more accurate.*

and the translation vector is computed as $\mathbf{t}_k = s_k R_k \mathbf{c}_k^0 - \mathbf{c}_k(t)$.

Using Eq. (1) with $s_k = 1$ results in a rigid shape matching-based regularizer which is well-suited to model as-rigid-as-possible deformations. Computing the scale according to (5) allows us to include a local scale.

The rigid shape matching regularizer was first introduced in [MHTG05] for the generation of physically plausible animations of deforming objects. Further improvements, again, for animating deformations, were proposed in [RJ07, SOG08]. To the best of our knowledge, there is no paper which exploits the rigid shape matching ($s_k = 1$) or similarity-based (s_k according to (5)) regularization in the context of deformable 3D shape registration.

3.2. Correspondence Estimation

In this subsection, we introduce a correspondence field $\mathbf{d} : \mathcal{S} \to \mathbb{R}^3$ which defines a pointwise correspondence between the source shape $\mathcal{S}$ and the target shape $\mathcal{T}$. As already discussed in Section 2, there is a substantial amount of work in the field of non-rigid correspondence estimation [BK10, RBBK10, WZL*10, WAS10, ZWW*10]. These methods solve the problem without making any assumptions about the initial alignment of the shapes but, unfortunately, tend to be costly. In contrast to this, we exploit the fact that consecutive scans of deforming objects exhibit small inter-frame displacements and design a simple but effective correspondence establishment procedure.

Perhaps the most common way of doing this (see [BM92, ARV07, LSP08]) is to compute the vector connecting each source point $\mathbf{x}_k$ to its closest point on $\mathcal{T}$:

$$\mathbf{c}_{\mathcal{T}}(\mathbf{x}_k) = \underset{\mathbf{y}_i \in \mathcal{T}}{\operatorname{argmin}} \|\mathbf{x}_k - \mathbf{y}_i\| - \mathbf{x}_k. \qquad (6)$$

This, however, leads to many wrong correspondences, even for shapes which are not very far from each other (see Fig. 2(a)). In this paper, we use $\mathbf{c}_{\mathcal{T}}$ only as a starting point of a simple and effective smoothing technique which significantly improves the correspondence estimation. In many cases (like the one shown in Fig. 2(a)), $\mathbf{c}_{\mathcal{T}}$ is quite irregular in the sense that it varies too strong when evaluated at neighboring points on the source shape. This should not be so since scanning a deforming object at high frame rates produces a smoothly varying surface which, in turn, results in a smooth correspondence field between consecutive frames. This is the reason why we expect a smoothing of $\mathbf{c}_{\mathcal{T}}$ to improve the correspondence estimation between the shapes.

More precisely, we use $\mathbf{c}_{\mathcal{T}}$ as a starting point for a local optimization procedure which returns a displacement field $\mathbf{d} : \mathcal{S} \to \mathbb{R}^3$ which minimizes the smoothness term

$$E_s(\mathbf{d}) = \sum_{k=1}^{m} \left\| \mathbf{d}(\mathbf{x}_k) - \overline{\mathbf{d}(\mathbf{x}_k)} \right\|^2, \qquad (7)$$

where m is the number of source shape points and $\overline{\mathbf{d}(\mathbf{x}_k)} = \frac{1}{N_k} \sum_{i \in \mathbf{N}_k} \mathbf{d}(\mathbf{x}_i)$ is the mean value of $\mathbf{d}$ over the neighborhood of $\mathbf{x}_k$. Furthermore, E_s has to be minimized under the constraint

$$\mathbf{x}_k + \mathbf{d}(\mathbf{x}_k) \in \mathcal{T}, \quad \forall \mathbf{x}_k \in \mathcal{S}. \qquad (8)$$

Essentially, (7) penalizes displacement fields which vary at neighboring source points and (8) assures that the solution is indeed a correspondence field, i.e., that $\mathbf{d}$ brings each $\mathbf{x}_k$ to a point on the target shape and not to some arbitrary position in $\mathbb{R}^3$. This means that $\mathbf{d}$ has the form $\mathbf{d}(\mathbf{x}_k) = \mathbf{y}_m - \mathbf{x}_k$ for some $\mathbf{y}_m \in \mathcal{T}$ and we have a discrete optimization problem. Having this in mind, we construct the following optimization procedure:

```
1. Initialize $\mathbf{d}_0 := \mathbf{c}_{\mathcal{T}}$ and $j := 0$.
2. Compute $\mathbf{d}_{j+1}$
 [for each $\mathbf{x}_k \in \mathcal{S}$ compute $\mathbf{d}_{j+1}(\mathbf{x}_k)$]
    a. Get the target point $\mathbf{y}_l$ which corresponds to $\mathbf{x}_k$, i.e., $\mathbf{y}_l = \mathbf{x}_k + \mathbf{d}_j(\mathbf{x}_k)$.
    b. Among the neighbors of $\mathbf{y}_l$, choose the target point $\mathbf{y}_m \in \mathcal{T}$ with minimal $\|(\mathbf{y}_m - \mathbf{x}_k) - \overline{\mathbf{d}_j(\mathbf{x}_k)}\|^2$.
    c. Set $\mathbf{d}_{j+1}(\mathbf{x}_k) := \mathbf{y}_m - \mathbf{x}_k$.
 [end for]
3. If $E_s(\mathbf{d}_{j+1}) < E_s(\mathbf{d}_j)$ increment $j$ and go to step 2. Otherwise terminate the procedure.
```

The only parameter of this algorithm is the radius of the neighborhood of the target point $\mathbf{y}_l$ in step 2.b.. The bigger the radius the more global the search, i.e., the greater the chance to overcome local minima in the landscape of the smoothness term. Of course, this comes at the cost of a higher computational load. The algorithm always converges since $\mathbf{y}_l$ is part of its own neighborhood and after a certain number of iterations no further minimization of the terms in step 2.b. is possible which results in $E_s(\mathbf{d}_{j+1}) = E_s(\mathbf{d}_j)$.

Fig. 2(b), shows a correspondence field computed with our smoothing procedure. The improvement compared to the closest-point field $\mathbf{c}_{\mathcal{T}}$ is obvious.

3.3. The ODEs System and its Integration

In contrast to [MHTG05, RJ07, BPWG07], we are not interested in creating physically plausible animations of deforming objects. Thus, our approach is not based on a physical model like, e.g., Newton's second law used in [MHTG05, RJ07, BPWG07].

At each time instance every source shape point should move according to both the regularizer and the correspondence field. In other words, the velocity of each $\mathbf{x}_k$ is a linear combination of the vector pointing to its rest position, namely, $\mathbf{r}_k - \mathbf{x}_k$ and the vector given by the correspondence field evaluated at $\mathbf{x}_k$ which is $\mathbf{d}(\mathbf{x}_k)$. More precisely, we set the velocity to be a convex combination of $\mathbf{r}_k - \mathbf{x}_k$ and $\mathbf{d}(\mathbf{x}_k)$. Writing this down in a formal way, leads to the following system of m ordinary differential equations (with m being the number of source shape points):

$$\dot{\mathbf{x}}_k(t) = \alpha(\mathbf{r}_k(\mathbf{X}_k(t)) - \mathbf{x}_k(t)) + (1-\alpha)\mathbf{d}(\mathbf{x}_k(t)), \qquad (9)$$

$$\mathbf{x}_k(0) = \mathbf{x}_k^0. \qquad (10)$$

Using a convex combination instead of a general linear combination has the advantage of introducing only one parameter $\alpha \in [0,1]$ which can be interpreted as the stiffness of the source shape: the greater the value the more rigid the motion of the shape. (9) together with (10) define an initial value problem which we solve numerically using the following integration scheme:

$$\mathbf{x}_k^{n+1} = \mathbf{x}_k^n + \alpha(\mathbf{r}_k^n - \mathbf{x}_k^n) + (1-\alpha)\mathbf{d}(\mathbf{x}_k^n), \qquad (11)$$

where n is the iteration number. Since this is an explicit method (Euler's method with step size 1) its stability is a-priori not guaranteed. If the step size is chosen too large, explicit integration schemes can overshoot the equilibrium of the system by an amount which increases in each iteration and finally leads to an "explosion" [MHTG05]. In our case, however, this does not happen. Recall from Section 3.2 that for each source point $\mathbf{x}_k^n$ the correspondence field has the form $\mathbf{d}(\mathbf{x}_k^n) = \mathbf{y}_m - \mathbf{x}_k^n$ for some target point $\mathbf{y}_m$. Using this we can rewrite (11) to get

$$\mathbf{x}_k^{n+1} = \alpha\mathbf{r}_k^n + (1-\alpha)\mathbf{y}_m. \qquad (12)$$

This means that the new point $\mathbf{x}_k^{n+1}$ lies on the straight line between $\mathbf{r}_k^n$ and $\mathbf{y}_m$ and thus can not overshoot the equilibrium since it lies on this line as well. Using a step size larger than 1 leads to an $\mathbf{x}_k^{n+1}$ which overshoots the line (and thus the equilibrium) and can lead to instability.

3.4. The Overall Registration Procedure

The overall deformable registration algorithm works as follows. We start the integration of the ODEs system using the numerical scheme described in Section 3.3 with a high stiffness value $\alpha = 0.95$. In each iteration, the rest positions and the correspondences are recomputed as described in Sections 3.1 and 3.2, respectively, using the updated positions of the source shape points. This is repeated until convergence which is detected by checking whether (11) has reached a fix point. The stiffness parameter is then lowered by 0.05 and the integration continues. The registration terminates when the stiffness falls below 0.5.

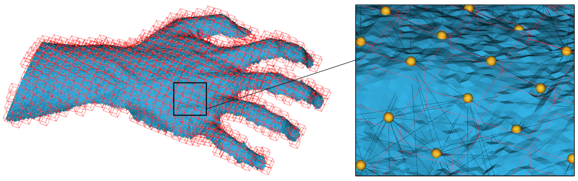

Figure 3: *A deformation graph computed for a range scan of a hand. The magnified part on the right shows the nodes as yellow spheres. Neighboring nodes are connected with black lines.*

Note that this procedure does not cope well with missing data since the correspondence field guides *each* source point $\mathbf{x}_k$ towards a target point no matter if it is the one which semantically corresponds to $\mathbf{x}_k$ or not. What happens to such a source point in the course of the registration is that its neighborhood gets too distorted, especially for lower stiffness values. We detect these points using a simple deformation measure and let them move only according to the regularizer. We use

$$D(n,k) = \frac{1}{N_k - 1} \sum_{\substack{i \in \mathbf{N}_k \\ i \neq k}} \frac{|\|\mathbf{x}_i^n - \mathbf{x}_k^n\| - l_{ik}|}{l_{ik}}, \qquad (13)$$

as the measure of deformation of the source point $\mathbf{x}_k$ in the n-th iteration of the registration algorithm. (This is essentially the definition of strain in mass-spring systems [WOR10].) Recall that $\mathbf{N}_k$ is the set of indices (including k) of the neighbors of $\mathbf{x}_k$ and $N_k = |\mathbf{N}_k|$. l_{ik} denotes the distance between $\mathbf{x}_i$ and $\mathbf{x}_k$ in the undeformed source shape, i.e., $l_{ik} = \|\mathbf{x}_i^0 - \mathbf{x}_k^0\|$. Note that $D(n,k)$ is dimensionless meaning that it does not depend on the units in which the shapes are saved. If $D(n,k)$ exceeds a certain fixed threshold the neighborhood of $\mathbf{x}_k$ is considered too distorted and from the n-th iteration on the point moves only according to the regularizer without being directly attracted to the target shape. In our implementation, we set this threshold to 0.2.

4. Implementation Issues

Note that the registration algorithm introduced in the last section is applicable to all shapes which are represented by a finite set of points plus an underlying neighborhood structure. In this Section, we will briefly discuss two important special cases, namely, range scans and triangular meshes.

A range scan is a 2D image in the xy-plane which stores a depth value along the z-direction [LSP08]. The range scans used in the experimental part of the paper contain around 30,000 points. This introduces a significant computational load and leads to an unpractical registration algorithm. Moreover, many deformations of practical interest like, e.g., articulated motion have much fewer degrees of freedom and can be described in a more efficient way. This is the reason why we decouple the complexity of the registration algorithm from the geometric complexity by using a so-called deformation graph [SSP07]. It consists of nodes connected by undirected edges. We exploit the regularity of the range scans and cover the *xy*-plane with non-overlapping three-dimensional boxes of a certain fixed size. The *z*-coordinate of each box is chosen to be median of the *z*-coordinates of all shape points being covered by this box. Then for each box the shape point closest to its center is defined to be a node in the deformation graph. Boxes having integer *xy*-coordinates which do not differ by more than 2 and having *z*-coordinates which do not differ by a certain amount (we use 30mm) are considered to be neighbors. Fig. 3 illustrates this concept.

If the source shape is represented by a triangular mesh, we use a different strategy. In this case, the deformation graph is an octree of fixed leaf size. If, furthermore, the mesh represents a closed surface, we add all leaves to the octree which are contained within the surface. In this way, a solid 3D lattice is created which results in more stable deformations [BPWG07].

After the deformation graph has been built it is used for the registration instead of the original source shape. However, since we are interested in deforming the shape itself, the deformation computed for the graph has to be transferred to the shape. This is done using thin plate spline (TPS) interpolation of the vector field defined by $\mathbf{g}_i - \mathbf{g}_i^0$, where $\mathbf{g}_i$ is the position of the i-th graph node after the registration and $\mathbf{g}_i^0$ is its initial position. Each source shape point is then transformed using the computed TPS. We chose this interpolation scheme since it produces high-quality vector fields and is easy to compute for scattered data.

5. Experimental Results

In this Section, the proposed registration algorithm is experimentally validated on a variety of real data sets. All tests are performed on a laptop with a 3GHz CPU, 4GB RAM and a Linux operating system. The method is implemented in C++.

Qualitative Tests First, we show two qualitative test results using a doll head model as the source shape and a head model of a girl and a boy as the target shapes. The shapes are represented as closed triangular meshes. Since they were not pre-aligned, we preformed a manual rigid registration based on 8 landmarks. Even after this user intervention the test scenario remains challenging because the models are representing different "subjects" and there is a significant scale difference between the shapes. Figure 4 shows the input data sets and the registration results from several view points.

Range Scan Pairs Next, we run our method on pairs of range scans representing the same object in different poses. In order to evaluate the method quantitatively, we measure the source shape deformation and the RMS error between source and target and plot them versus the iteration number. The source shape deformation is defined using (13) as

$$D(n,\mathcal{S}) = \frac{1}{m}\sum_{k=1}^{m} D(n,k) \tag{14}$$

and the RMS error between $\mathcal{S}$ and $\mathcal{T}$ is given by

$$\mathrm{RMS}(n,\mathcal{S},\mathcal{T}) = \sqrt{\frac{1}{m}\sum_{k=1}^{m} \|\mathbf{c}_{\mathcal{T}}(\mathbf{x}_k^n)\|^2}, \tag{15}$$

where n is the iteration number, m is the number of points in $\mathcal{S}$ and $\mathbf{c}_{\mathcal{T}}(\mathbf{x}_k^n)$ computes the vector connecting $\mathbf{x}_k^n$ to its closest point in $\mathcal{T}$ (see (6)).

Figures 5 to 9 show the data sets used in the test. The scans are shown as they were captured by the scanning devices without any additional alignment. These configurations are used as starting point for our registration algorithm. Fig. 5 shows a range scan pair which is part of a sequence representing a slowly closing hand. The inter-frame displacement is small and mainly caused by the bending finders. The registration computed with our method together with the deformation measure and the RMS error are shown on the right side of the figure.

Fig. 6 shows a further example of a closing hand. This time, there is a larger bending deformation plus an additional global translation. As is to be expected from the initial configuration of the scans the RMS error at the beginning of the registration is bigger. Furthermore, since the fingers bend more than in the last example the amount of deformation required to register the scans is larger. This is confirmed by the plots on the right side of the figure.

Fig. 7 demonstrates the ability of our algorithm to deal with incomplete data. Note that there are many holes in both scans caused by self-occlusion and scan device imperfection. Our method successfully registers the scans even in areas of low overlap as the magnified parts of the figure show.

Fig. 8 shows registration of an articulated object, namely, a bending arm. Note that there is a significant deformation between the scans. This is a challenging example for feature-based methods since the scans are smooth and lack distinctive features. Our method successfully recovers the deformation as depicted in the figure.

In Fig 9, a further example of a moving hand is shown. Additionally to the local deformation caused by the closing fingers this example contains a significant global rotation.

Range Scan Sequence Next, we test our method on a sequence of range scans representing a closing hand. The first frame of the sequence is used as the source shape and is sequentially registered to the other frames. The result for the

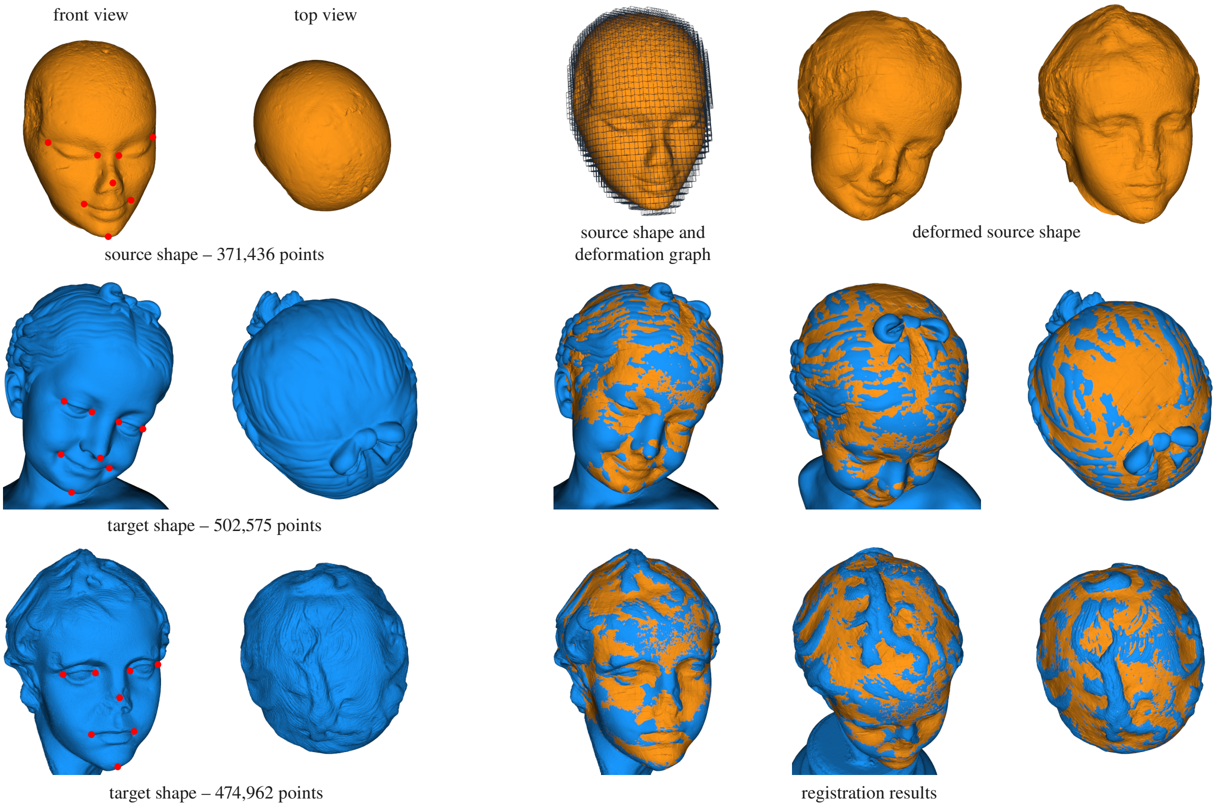

Figure 4: *Registering a doll head (upper left) to a head of a girl and a boy (lower left). The landmarks used for these registration tests are shown as red dots. Note that there is a significant difference in scale between the source and the target shapes. Our algorithm successfully performs the registration as shown on the right. The models were downloaded from the AIM@SHAPE Repository –* `http://shapes.aim-at-shape.net/`

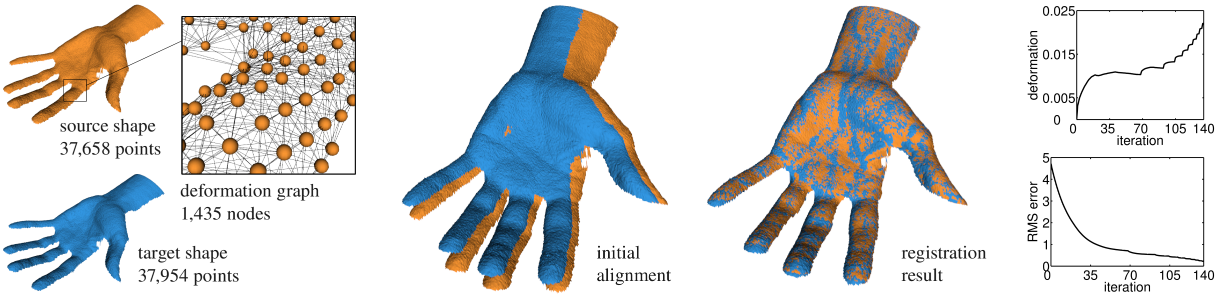

Figure 5: *Registering two range images representing the front part of the same hand in two different poses. The data sets were obtained with a 3D geometry scanner [WLG07] and are publicly available on the authors webpage.*

current frame is used as initialization for the next one. The sequence consists of 100 frames. Fig. 10 exemplary shows some frames and the corresponding registration results.

Comparison Finally, we compare the performance of our method (both registration quality and runtime) with the performance of several state-of-the-art non-rigid registration algorithms: the softassign + deterministic annealing (SDA) approach [CR03], the kernel correlation-based (KC) method [TK04], the coherent point drift (CPD) algorithm [MS10] and the Gaussian mixture models-based (GMM) algorithm [JV11]. Note that the KC [TK04] and the GMM [JV11] methods can perform the registration us-

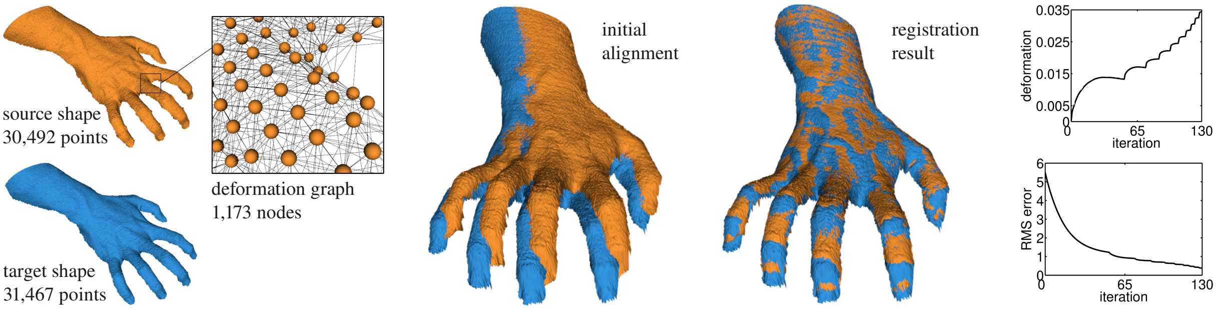

Figure 6: *Range scans representing the back part of the same hand in two different poses. The poses differ not only by the local bending deformation of the fingers but also by a global translation. The data sets were obtained with a 3D geometry scanner [WLG07] and used in [SAY*09].*

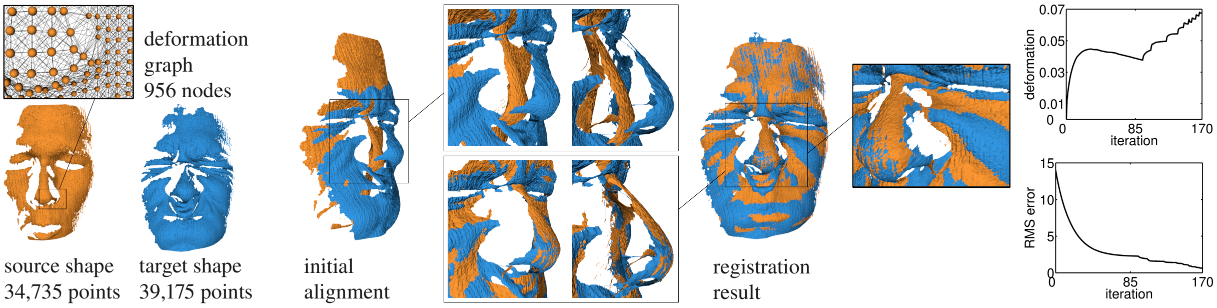

Figure 7: *Registering two facial expressions. Note that the scans are noisy and incomplete. Our methods correctly aligns the shapes even in areas of low overlap.*

ing two different deformation models, namely, thin plate splines (TPS) and Gaussian radial basis functions (GRBF). This effectively results in six different registration methods which we will denote as follows: SDA+TPS is the abbreviation for [CR03], CPD+GRBF stands for [MS10], KC+TPS, KC+GRBF denote [TK04], and GMM+TPS, GMM+GRBF stand for [JV11], depending on which deformation model is employed.

We use the implementation of the above mentioned methods, publicly available on http://gmmreg.googlecode.com, and run them on the same hardware and with the same data sets as our algorithm. The quality of the registration computed by the algorithms is compared using the source shape deformation measure (14) and the RMS error (15). Table 1 shows the results of the quality comparison. Note that our algorithm outperforms the others since it produces a lower RMS error for virtually the same source shape deformation. The results of the runtime comparison are summarized in table 2. Our algorithm clearly outperforms all six methods with the difference in processing time being up to two orders of magnitude.

	data set				
method	Fig. 5	Fig. 6	Fig. 7	Fig. 8	Fig. 9
GMM+TPS	497	315	273	332	453
GMM+GRBF	361	244	237	269	267
SDA+TPS	1004	621	501	642	1033
CPD+GRBF	4389	2443	1368	2269	5952
KC+TPS	491	314	273	331	451
KC+GRBF	361	243	237	267	267
our alg.	14	13	11	18	21

Table 2: *Computation time (in seconds) taken by our algorithm and six state-of-the-art approaches for the registration of the scans presented in the paper. Our method clearly outperforms the others.*

6. Conclusions

In this paper, we proposed an efficient algorithm for deformable registration of 3D shapes. We focused on modeling as-rigid-as-possible shape deformations augmented with local scale. In contrast to many recent methods, our approach is not formulated within a general-purpose optimization framework. The minimization of high-dimensional, nonlinear cost functions is computationally very demanding

	data set									
method	Fig. 5		Fig. 6		Fig. 7		Fig. 8		Fig. 9	
GMM+TPS	0.04	0.6	0.05	0.8	0.10	1.4	0.08	1.4	0.06	2.1
GMM+GRBF	0.03	0.8	0.03	0.9	0.07	2.4	0.05	2.2	0.06	2.5
SDA+TPS	0.09	0.7	0.09	0.7	0.11	1.2	0.08	1.3	0.08	0.9
CPD+GRBF	0.30	1.4	0.30	1.2	0.20	1.5	0.30	3.6	0.20	1.7
KC+TPS	0.04	0.6	0.05	0.8	0.10	1.3	0.08	1.4	0.06	1.8
KC+GRBF	0.03	0.8	0.03	0.9	0.09	2.2	0.06	2.2	0.07	2.4
our alg.	0.02	0.2	0.03	0.4	0.07	0.6	0.04	0.7	0.07	0.5

Table 1: *Comparing the quality of the registration computed by our algorithm and six state-of-the-art approaches for the scans presented in the paper. The first number in each table cell gives the source shape deformation (Eq. (14)) and the second one gives the RMS error in millimeters (Eq. (15)). Our method provides the most precise alignment for a low shape deformation.*

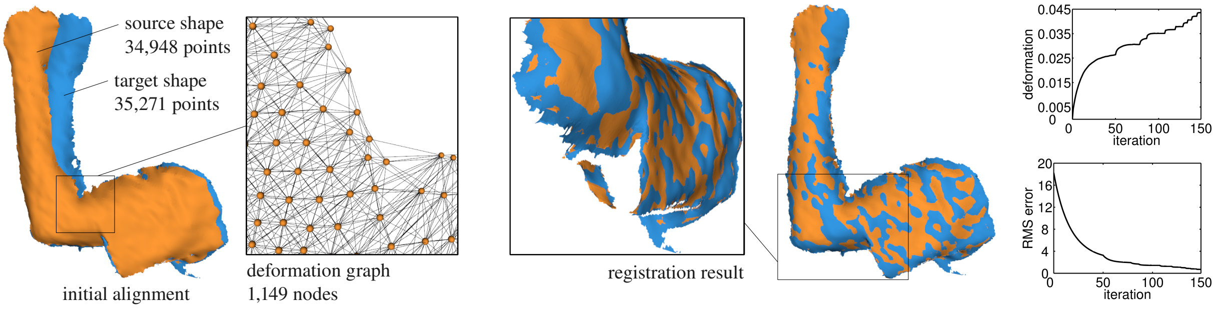

Figure 8: *Registration of a bending arm.*

since it involves the repeated solution of large linear systems. Instead, we rely on a simple and effective numerical integration scheme and solve an ODEs system which models the non-rigid motion of a source shape towards a target shape. The ODE we proposed is based on a correspondence field and a regularization term. Preliminary correspondences are estimated with a closest-point search and further refined with an efficient smoothing procedure. The regularizer is a generalization of the rigid shape matching technique recently developed in the context of physically plausible deformation modeling. We experimentally validated our method on a variety of real range scans and demonstrated that it performs well on noisy and incomplete data. Finally, an experimental comparison to six state-of-the-art approaches showed that the proposed algorithm outperforms them in terms of both registration quality and processing time.

Acknowledgments This work has been funded by the European Commission's Seventh Framework Programme as part of the project GRASP (IST-FP7-IP-215821).

References

[ACP03] Allen B., Curless B., Popovic Z.: The Space of Human Body Shapes: Reconstruction and Parameterization from Range Scans. *ACM TOG 22*, 3 (2003).

[AMCO08] Aiger D., Mitra N. J., Cohen-Or D.: 4-Points Congruent Sets for Robust Pairwise Surface Registration. In *ACM SIGGRAPH* (2008).

[ARV07] Amberg B., Romdhani S., Vetter T.: Optimal Step Nonrigid ICP Algorithms for Surface Registration. In *CVPR* (2007).

[ASK*05] Anguelov D., Srinivasan P., Koller D., Thrun S., Rodgers J., Davis J.: SCAPE: Shape Completion and Animation of People. *ACM TOG 24*, 3 (2005).

[BBK06] Bronstein A. M., Bronstein M. M., Kimmel R.: Generalized Multidimensional Scaling: A Framework for Isometry-Invariant Partial Surface Matching. *PNAS 103*, 5 (2006).

[BK10] Bronstein M. M., Kokkinos I.: Scale-Invariant Heat Kernel Signatures for Non-Rigid Shape Recognition. In *CVPR* (2010).

[BM92] Besl P., McKay N.: A Method for Registration of 3-D Shapes. *IEEE TPAMI 14*, 2 (1992).

[BPWG07] Botsch M., Pauly M., Wicke M., Gross M. H.: Adaptive Space Deformations Based on Rigid Cells. *Comput. Graph. Forum 26*, 3 (2007).

[BR04] Brown B., Rusinkiewicz S.: Non-Rigid Range-Scan Alignment Using Thin-Plate Splines. In *3DPVT* (2004).

[BR07] Brown B., Rusinkiewicz S.: Global Non-Rigid Alignment of 3-D Scans. *ACM TOG 26*, 3 (2007).

[CM91] Chen Y., Medioni G.: Object Modeling by Registration of Multiple Range Images. In *ICRA* (1991).

[CR03] Chui H., Rangarajan A.: A New Point Matching Algorithm for Non-Rigid Registration. *CVIU 89*, 2-3 (2003).

[CZ08] Chang W., Zwicker M.: Automatic Registration for Articulated Shapes. *Comput. Graph. Forum 27*, 5 (2008).

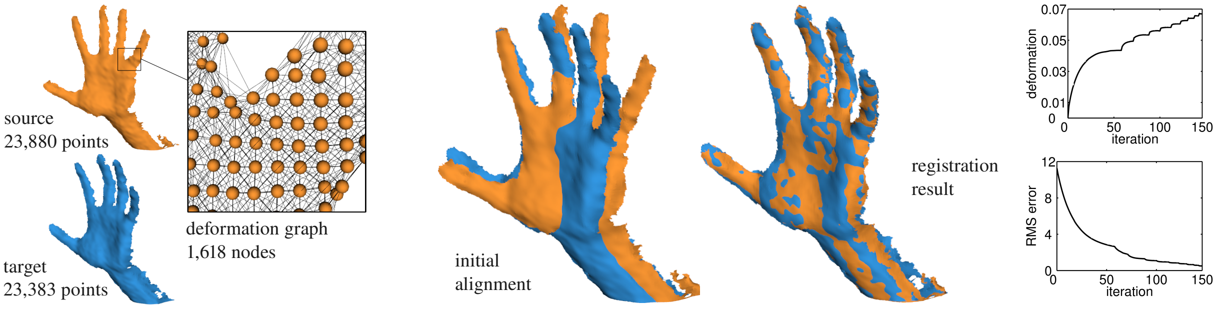

Figure 9: *Registering a closing and rotating hand.*

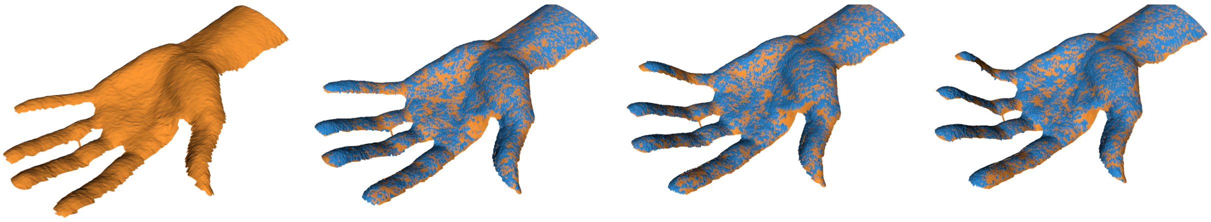
frame 0 frame 60 frame 80 frame 95

Figure 10: *Frame 0 is registered sequentially to the other frames which are part of a range image sequence of a closing hand. The sequence was obtained with a 3D geometry scanner [WLG07] and can be downloaded from the webpage of the authors.*

[EK03] Elad A., Kimmel R.: On Bending Invariant Signatures for Surfaces. *IEEE TPAMI 25*, 10 (2003).

[GMGP05] Gelfand N., Mitra N., Guibas L., Pottmann H.: Robust Global Registration. In *Eurographics Symposium on Geometry Processing* (2005).

[IGL03] Ikemoto L., Gelfand N., Levoy M.: A Hierarchical Method for Aligning Warped Meshes. In *3DIM* (2003).

[JV11] Jian B., Vemuri B.: Robust Point Set Registration Using Gaussian Mixture Models. *IEEE TPAMI* (2011).

[KS05] Kraevoy V., Sheffer A.: Template-Based Mesh Completion. In *Symposium on Geometry Processing* (2005).

[LSP08] Li H., Sumner R. W., Pauly M.: Global Correspondence Optimization for Non-Rigid Registration of Depth Scans. *Comput. Graph. Forum 27*, 5 (2008).

[MHTG05] Müller M., Heidelberger B., Teschner M., Gross M. H.: Meshless deformations based on shape matching. *ACM TOG 24*, 3 (2005).

[MS09] Myronenko A., Song X. B.: On the Closed-Form Solution of the Rotation Matrix Arising in Computer Vision Problems. *CoRR abs/0904.1613* (2009).

[MS10] Myronenko A., Song X. B.: Point Set Registration: Coherent Point Drift. *IEEE TPAMI 32*, 12 (2010).

[PB09] Papazov C., Burschka D.: Stochastic Optimization for Rigid Point Set Registration. In *ISVC* (2009).

[PMG*05] Pauly M., Mitra N. J., Giesen J., Gross M. H., Guibas L. J.: Example-Based 3D Scan Completion. In *Symposium on Geometry Processing* (2005).

[RBBK10] Raviv D., Bronstein M. M., Bronstein A. M., Kimmel R.: Volumetric Heat Kernel Signatures. In *3DOR* (2010).

[RJ07] Rivers A. R., James D. L.: FastLSM: Fast Lattice Shape Matching for Robust Real-Time Deformation. *ACM TOG 26*, 3 (2007).

[SAY*09] Sagawa R., Akasaka K., Yagi Y., Hamer H., Van Gool L.: Elastic Convolved ICP for the Registration of Deformable Objects. In *3DIM* (2009).

[SOG08] Steinemann D., Otaduy M. A., Gross M.: Fast Adaptive Shape Matching Deformations. In *SCA* (2008).

[SSP07] Sumner R. W., Schmid J., Pauly M.: Embedded Deformation for Shape Manipulation. *ACM TOG 26*, 3 (2007).

[TK04] Tsin Y., Kanade T.: A Correlation-Based Approach to Robust Point Set Registration. In *ECCV* (2004).

[WAS10] Wuhrer S., Azouz Z. B., Shu C.: Posture Invariant Surface Description and Feature Extraction. In *CVPR* (2010).

[WLG07] Weise T., Leibe B., Gool L. J. V.: Fast 3D Scanning with Automatic Motion Compensation. In *CVPR* (2007).

[WOR10] Wang H., O'Brien J., Ramamoorthi R.: Multi-Resolution Isotropic Strain Limiting. *ACM TOG 29*, 6 (2010).

[WSB09] Wuhrer S., Shu C., Bose P.: Posture Invariant Correspondence Of Triangular Meshes In Shape Space. In *3DIM* (2009).

[WSBA07] Wuhrer S., Shu C., Bose P., Azouz Z. B.: Posture Invariant Correspondence of Incomplete Triangular Manifolds. *International Journal of Shape Modeling 13*, 2 (2007).

[WZL*10] Wu H.-Y., Zha H., Luo T., Wang X.-L., Ma S.: Global and Local Isometry-Invariant Descriptor for 3D Shape Comparison and Partial Matching. In *CVPR* (2010).

[ZWW*10] Zeng Y., Wang C., Wang Y., Gu X., Samaras D., Paragios N.: Dense Non-Rigid Surface Registration Using High-Order Graph Matching. In *CVPR* (2010).