Group-valued Regularization for Motion Segmentation of Articulated Shapes

Guy Rosman, Michael M. Bronstein, Alex M. Bronstein, Alon Wolf and Ron Kimmel

Abstract Motion-based segmentation is an important tool for the analysis of articulated shapes. As such, it plays an important role in mechanical engineering, computer graphics, and computer vision. In this chapter, we study motion-based segmentation of 3D articulated shapes. We formulate motion-based surface segmentation as a piecewise-smooth regularization problem for the transformations between several poses. Using Lie-group representation for the transformation at each surface point, we obtain a simple regularized fitting problem. An Ambrosio-Tortorelli scheme of a generalized Mumford-Shah model gives us the segmentation functional without assuming prior knowledge on the number of parts or even the articulated nature of the object. Experiments on several standard datasets compare the results of the proposed method to state-of-the-art algorithms.

Key words: Motion Segmentation, Lie-groups, Surface Diffusion, Ambrosio-Tortorelli

Guy Rosman and Ron Kimmel
Dept. of Computer Science, Technion, 32000, Haifa, Israel e-mail: rosman,ron@cs.technion.ac.il

Michael M. Bronstein
Institute of Computational Science, Faculty of Informatics, Università della Svizzera italiana, Lugano, Switzerland michael.bronstein@usi.ch

Alex M. Bronstein
School of Electrical Engineering, Faculty of Engineering Tel Aviv University, Ramat Aviv 69978, Israel bron@eng.tau.ac.il

Alon Wolf
Dept. of Mechanical Engineering, Technion – Israel Institute of Technology, Haifa, 32000, Israel alonw@tx.technion.ac.il

1 Introduction

Articulated objects segmentation is a key problem in biomechanics [1], mechanical engineering, computer vision [8, 36, 39, 28, 52], and computer graphics [33, 37, 57, 35, 65, 6, 64]. Related problems of deformation analysis [63, 4] and motion segmentation [5, 20] have also been studied extensively in these disciplines. Algorithms solving these problems try to infer the articulated motion of an object, given several instances of the object in different poses. Simultaneously, the segmentation of the object into rigid parts takes place along with motion estimation between the corresponding parts in the various poses.

Most motion analysis techniques make some assumptions on the object to be segmented. These usually concern the number or location of rigid parts in the articulated object. This can be in the form of a skeleton describing the topology of the shape, or some other prior on the object structure. Such priors are usually formulated in an *ad hoc* manner, but not based on the kinematic model commonly assumed for near-rigid objects [1, 4]. In cases where such a prior is not available for the objects in question, or where assumptions about the data are only approximate, this can lead to errors in the segmentation and motion estimation.

Another common assumption, especially in graphics applications, is that of known correspondences. In computer graphics, the problem is usually referred to as *dynamic mesh segmentation*. While a matching technique between poses can be combined with existing motion segmentation tools, a more complete formulation for motion segmentation should handle the correspondence problem implicitly.

Clearly, the above assumptions are often too limiting in real-world applications, and should be avoided as part of the basic problem formulation. We would like instead to apply the intuition often used when studying real-life near-rigid objects, about the existence of a representative rigid motion existing for each body part. We wish, however, to avoid detecting the articulated parts in advance. Furthermore, in some object, a clear partition into rigid parts may not exist for all of the surface. We wish to obtain reasonable results in such a case. In other words, we would like to obtain a "soft" segmentation of the surface, without knowing the number or location of regions in advance, an explicit analysis of the surface features, or having additional priors on the various object parts. Also, we strive towards a formulation of motion segmentation that incorporates an implicit handling of the correspondence problem, given a reasonable initialization.

1.1 Main contribution.

In this chapter we try to remedy the shortcoming of existing approaches to articulated motion estimation by combining the two tasks of motion estimation and segmentation into a single functional. This scheme has been described in a recent conference paper [48] and we now slightly expand upon it. Unlike existing methods, we propose a principled variational approach, attempting to find a rigid transformation at each surface point, between the instance surfaces, such that the overall transformation is described by a relatively sparse set of such transformations, each matching a rigid part of the object. The functional we propose regularizes the motion between the surfaces, and is guided by the fact that the parameters of the motion transformations

(i) should describe the motion at each point with sufficient accuracy.
(ii) should vary smoothly within the (unknown) rigid parts.
(iii) can vary abruptly between rigid parts.

We see our main contribution in these :

A new framework: First, we propose an axiomatic variational framework for articulated motion segmentation. While focusing on the segmentation problem in this chapter, our framework is more general and the proposed functionals can be easily incorporated into other applications such as motion estimation, tracking, and surface denoising.

Variational segmentation: We claim that using the right parameterization, taken from the specific domain of rigid motion analysis, we can formulate the articulated motion segmentation problem as a generalization of classical tools in variational computer vision. This allows for an elegant and simple solution within the proposed framework, obtaining results competitive with domain-specific state-of-the-art tools.

A novel visualization algorithm: Third, we suggest a spatially-coherent algorithm for spatial visualization of group valued data on manifolds, which draws from the same variational principles.

1.2 Relation to prior work.

Several previous works have attempted motion based segmentation of surfaces. We mention but a few of these. Kompatsiaris et al. [36] use an estimation of the rigid motion at each segment in order to segment the visible surface in a coarse-to-fine manner. Arcila et al. [6] iteratively refine the segmentation for segments whose transformation error is too large. Wuhrer and Brunton [64] use a dual tree representation of the surface with weights between triangles set according to the dihedral angles. Lee et al. [37] use a similar graph-based formulation, looking at deformation matrices around each triangle.

The scheme we propose involves diffusing the transformations between poses along the surface, in the spirit of the Ambrosio-Tortorelli scheme [2] for Mumford-Shah segmentation [41]. The diffusion component of our scheme is a diffusion process of Lie-group elements, which has recently attracted significant attention in other applications [23, 54, 26]. In diffusing transformations on the surface, our work is similar to that of Litke et al. [38], although the parameterization of the motion and of the surface is different. In addition, we do not make an assumption on the surface topology; to that end, the proposed method diffuses transformations along the surface, rather than representing the surface in an evenly sampled 2D parametrization plane. When dealing with real-life deformable objects that seldom admit regular global parametrization, such an assumption could be too restrictive.

The idea of combining soft segmentation and motion estimation has been attempted before in the case of optical flow computation (see, e.g., [3, 16]). In optical flow fields, however, the motion field is merely expected to be piecewise smooth. For truly articulated objects one would expect piecewise-constant flow fields, when expressed in the correct parametrization.

Finally, the functional can be extended with priors from general mesh segmentation techniques. These are usually based on the geometry of the surface itself, and obtain remarkable results for a variety of objects. We point the reader to [9, 53, 18, 34], and references therein, for additional examples of mesh segmentation algorithms. We do not, however, use an additional prior as such an addition will prevent the isolated examination of the principles shown in this chapter.

2 Problem Formulation

We now proceed to define the problem we try to solve and the proposed model.

2.1 Articulation model.

We denote by X a 2-manifold representing a three-dimensional shape. We assume X to have several embeddings into $\mathbb{R}^3$. Each of these embedding constitutes a pose of the surface. In the following, we will denote by $\mathbf{x} : X \to \mathbb{R}^3$ the embedding of X into $\mathbb{R}^3$, and use synonymously the notation x and $\mathbf{x}$ referring to a point on the manifold and its Euclidean embedding coordinates, for a specific pose.

In the setting of rigid motion segmentation, we assume that X represents an *articulated* shape, i.e., it can be decomposed into *rigid parts* $S_1, \ldots, S_p$. These are transformed between different poses of the objects by a rigid transformation. This transformation, a rotation and a translation, is an isometry of $\mathbb{R}^3$. The rigid parts are connected by nonrigid *joints* $J_1, \ldots, J_q$, such that $X = \bigcup_{i=1}^{p} S_i \cup \bigcup_{k=1}^{q} J_k$. An *articulation* $Y = \mathbf{A}X$ is obtained by applying rigid motions $\mathbf{T}_i \in \mathrm{Iso}(\mathbb{R}^3)$ to the rigid parts, and non-rigid deformations $\mathbf{Q}_k$ to the joints, such that $\mathbf{A}X = \bigcup_{i=1}^{p} \mathbf{T}_i S_i \cup \bigcup_{k=1}^{q} \mathbf{Q}_k J_k$.

2.2 Motion segmentation.

The problem of *motion-based segmentation* can be described as follows: given two articulations of the shape, X and Y, extract its rigid parts. An extension to the case of multiple shape poses is straightforward. We therefore consider in the following only a pair of shapes for the sake of simplicity and without loss of generality. A strongly related question attempts to determine, given these articulations, the motion parameters linking the poses of the object.

Assuming that the correspondence between the two poses X and Y is known, given a point $x \in X$ and its correspondent point $y(x) \in Y$, we can find a motion $g \in \mathscr{G}$ such that $g\mathbf{x} = \mathbf{y}$, where $\mathscr{G}$ is some representation of coordinate transformations in $\mathbb{R}^3$. This motion g may change, in the setting described above, for each surface point. We therefore consider g to be a function $g : X \to \mathscr{G}$. We will simultaneously use $g\mathbf{x} \in \mathbb{R}^3$ to denote the action of $g(x)$ on the coordinates of the point x, as well as consider the mapping given by $g : X \to \mathscr{G}$ and its properties.

We note that typical representations of motion in $\mathbb{R}^3$ contain more than 3 degrees of freedom. In this sense, they are over-parameterized [44], and thus some measure of regularity is required in order to avoid ambiguity as well as favor a meaningful solution. On the other hand, we note that since the articulated parts of the shape move rigidly, if we choose an appropriate motion representation (as detailed below), two points $x, x' \in S_i$ will undergo the same transformation, from which it follows that $g(x)|_{x \in S_i} = \text{const}$. One possibility is to adopt a constrained minimization approach, forcing $g(X) = Y$, where $g(X)$ is a notation for the set $g(x)\mathbf{x}(x)$ for all $x \in X$. This approach, however, needs to somehow handle the set of joints, for which such a constraint may be meaningless. In general, restricting the feasible set of solutions by such constraints or even constraints involving an error in the data may be harmful for the overall result. In order to avoid this, another possible approach is to take an unconstrained, yet regularized, variational formulation,

$$\min_{g : X \to \mathscr{G}} \lambda E_{\mathrm{D}}(g) + \rho(g), \tag{1}$$

where ρ denotes a smoothness term operating on the motion parameters field. This term is expected to be small for fields g which are piecewise constant on the manifold X. While an appropriate parameterization of motion g, and regularization term $\rho(g)$ are crucial, we also require a data term that will encourage consistency of the transformation field g with the known surface poses. Specifically, we wish to favor a transformation field where the point $\mathbf{x}$ is taken by its transformation $g(x)$ to a point on the other surface. $E_{\mathrm{D}}(g)$ is our fitting term which measures this consistency with the data.

$$E_{\mathrm{D}}(g) = \int_X \|g(x)\mathbf{x} - \mathbf{y}(\mathbf{x})\|^2 da, \tag{2}$$

where $\mathbf{y}(x) \in \mathbb{R}^3$ denotes the coordinate of the point $y(x) \in Y$ corresponding to x, $g(x)$ is the transformation at x, and da is a measure on X. We have assumed in the discussion so far that the correspondence between X and Y is known, which is usually not true. We can solve for the correspondence as part of the optimization in an efficient manner. We will mention this issue in Section 4.1. We use the term corresponding point $y(x)$ since, as in the case of *iterative closest point* (ICP) algorithms [19, 11], several approaches for pruning erroneous or ineffective matches exist [51].

Minimizing the functional with respect to $g, y(x)$ from a reasonable initial solution allows recovery of the articulated parts by clustering g into regions of equal value. Yet another choice of a data term is a semi-local fitting term, is a semi-local one,

$$E_{\mathrm{D,SL}}(g) = \int_X \int_{y \in \mathscr{N}(x)} \|g(x)\mathbf{x}' - \mathbf{y}(\mathbf{x}')\|^2 da' da, \tag{3}$$

where $\mathscr{N}(x)$ denotes a small neighborhood around the point x (we took $\mathscr{N}(\mathbf{x})$ to be the 12 nearest neighbors). This fitting term, by itself, formulates a local ICP process. The functional (1) equipped with the semi-local data term can be considered as the geometrical fitting equivalent of the combined global-local approach for optic flow estimation [15].

The simplest representation of motion is a *linear motion* model, affectively setting $\mathscr{G}$ to be the group of translation, or $\mathscr{G} = \mathbb{R}^3$. This results in the motion model $g\mathbf{x} = \mathbf{x} + \mathbf{t} = \mathbf{y}$ for some $\mathbf{t} \in \mathbb{R}^3$. However, such a simplistic model fails to capture the piecewise constancy of the motion field in most cases. Instead of turning to a higher order approximation model such as the affine over-parameterized model [43], or to more elaborate smoothness priors [58], we look for a relatively simple model that will capture natural motions with a simple smoothness prior. Thus we turn to a slightly different motion model, naturally occuring in motion research.

2.3 Lie-groups.

One parametrization often used in computer vision and robotics [59, 40, 35, 26] is the representation of rigid motions by the Lie-group $SE(3)$ and the corresponding Lie-algebra $se(3)$, respectively. Works by Brockett [12], Park et al. [45] and Zefran et al. [60, 61] strongly relate Lie-groups, both in their global and differential description, to robotics and articulated motions. We give a very brief introduction to the subject and refer the reader to standard literature on the subject (e.g., [42, 27]) for more information.

Lie-groups are topological groups with a smooth manifold structure such that the group action $\mathscr{G} \times \mathscr{G} \mapsto \mathscr{G}$ and the group inverse are differentiable maps.

For every Lie-group, we can canonically associate a *Lie-algebra* $\mathfrak{g}$. A Lie-algebra is as a vector space endowed with a *Lie brackets operator* $[\cdot,\cdot] : \mathscr{G} \times \mathscr{G} \to \mathscr{G}$, describing the local structure of the group. The Lie-algebra associated with a Lie-group can be mapped diffeomorphically via the *exponential map* onto a neighborhood of the identity operator and its tangent space.

This property will allow us express neighboring elements in the Lie-group in a vector space, and thereby define derivatives, regularity, and diffusion operators on the group valued data.

In this chapter, we are specifically interested in the special orthogonal (rotation) matrix group $SO(3)$ and the Euclidean group $SE(3)$ to represent rigid motions. These can be represented in matrix forms, where $SO(3)$ is given as

$$SO(3) = \left\{ \mathbf{R} \in \mathbb{R}_{3\times 3}, \mathbf{R}^T \mathbf{R} = \mathbf{I} \right\}, \tag{4}$$

and $SE(3)$ is given by

$$SE(3) = \left\{ \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix}, \mathbf{R} \in SO(3), \mathbf{t} \in \mathbb{R}^3 \right\}. \tag{5}$$

The Lie-algebra of $SO(3)$, $so(3)$ consists of skew-symmetric matrices,

$$so(3) = \left\{ \mathbf{A} \in \mathbb{R}_{3\times 3}, \mathbf{A}^T = -\mathbf{A} \right\}, \tag{6}$$

whereas the Lie-algebra of $SE(3)$ can be identified with the group of 4×4 matrices of the form

$$se(3) = \left\{ \begin{pmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0} & 0 \end{pmatrix}, \mathbf{A} \in so(3), \mathbf{t} \in \mathbb{R}^3 \right\}, \tag{7}$$

where $so(3)$ is the set of 3×3 skew-symmetric matrices.

In order to obtain piecewise constant description over the surface for the relatively simple case of articulated object, we would like the points at each object part to have the same representative. Under the assumption of $\mathcal{G} = SE(3)$, this desired property holds. We note, however, that the standard parameterization of small rigid motions has 6 degrees of freedom, while the number of degrees of freedom required to describe point motion is mere 3. Thus, this parameterization clearly constitutes an over-parameterized motion field [43] for articulated surfaces.

We now turn to the regularization term, $\rho(g)$, and note that the formulation given in Equation 1 bears much resemblance to *total variation* (TV) regularization common in signal and image processing [49]. Total variation regularization does not, however, favor distinct discontinuity sets. This property of TV regularization is related to the *staircasing effect*. Furthermore, in the scalar case, discontinuity sets form closed curves, which may not be the case in some surfaces with large joint areas. Instead, a model that better suits our segmentation problem is the Mumford-Shah segmentation model [41]. This model can be implemented using an Ambrosio-Tortorelli scheme [2], which can be easily generalized for the case of maps between general manifolds such as maps from surfaces into motion manifolds. We further describe the regularization chosen in Section 3.

We also note that due to the non-Euclidean structure of the group, special care should be taken when parameterizing such a representation [40, 26, 54, 35], as discussed in Section 4.2.

3 Regularization of Group-Valued Functions on Surfaces

Ideally, we would like the transformation field defined on the articulated surface to be piecewise smooth, if not piecewise constant. Therefore, a suitable regularization of the transformation parameters is required. Since the Lie-group $\mathscr{G}$ as a Riemannian manifold, it is only natural to turn to regularization functionals defined on maps between manifolds of the form $g : X \to \mathscr{G}$.

A classical functional defined over such maps is the well-known *Dirichlet energy* [24],

$$\rho_{\mathrm{DIR}}(g) = \frac{1}{2}\int_X \langle \nabla g, \nabla g \rangle_{g(x)} da = \frac{1}{2}\int_X tr\left(g^{-1}\nabla g\right)^2 da, \tag{8}$$

where ∇g denotes the *intrinsic gradient* of g on X, $\langle \cdot, \cdot \rangle_{g(x)}$ is the Riemannian metric on $\mathscr{G}$ at a point $g(x)$, and da is the area element of X. This functional is the more general form of the Tikhonov regularization (for Euclidean spaces X and $\mathscr{G}$), and its properties are well defined for general manifolds, as studied by Eells [24].

Minimizers of the Dirichlet energy are called *harmonic maps*. These result from a diffusion process, and are often used for surface matching [67, 62].

3.1 Ambrosio-Tortorelli scheme.

Unfortunately, the Dirichlet energy favors smooth maps defined on X, whereas our desired solution has discontinuities at the boundaries of rigid parts. We would, intuitively, want to prevent diffusion across these discontinuity curves. This can be obtained by adding a diffusivity function $v : X \to [0,1]$ to the Dirichlet functional, leading to the generalized Ambrosio-Tortorelli scheme [2] for Mumford-Shah regularization [41].

$$\rho_{\mathrm{AT}}(g) = \int_X \left(\frac{1}{2} v^2 \langle \nabla g, \nabla g \rangle_g + \varepsilon \langle \nabla v, \nabla v \rangle + \frac{(1-v)^2}{4\varepsilon} \right) da, \tag{9}$$

where ε is a small positive constant. This allows us to extend our outlook in several ways. The Mumford-Shah functional replaces the notion of a set of regions with closed simple boundary curves with general discontinuity sets. It furthermore generalizes our notion of constant value regions with that of favored smoothness inside the areas defined by these discontinuity curves. This is in order to handle objects which deviate from articulated motion, for example in flexible regions or joints.

Furthermore, the generalized Ambrosio-Tortorelli scheme allows us to explicitly reason about places in the flow where the nonlinear nature of the data manifold manifests itself. Suppose we have a solution (g^*, v^*) satisfying our piecewise-constancy assumptions of g, and a diffusivity function with 0 at region boundaries and 1 elsewhere. At such a solution, we expect two neighboring points which belong to different regions to have a very small diffusivity value v connecting them, effectively nullifying the interaction between far-away group elements which is dependent on the mapping used for the logarithm map at each point, and hence can be inaccurate [30, 40]. While such a solution (g^*, v^*) may not be a minimizer of the functional, it serves well to explain the intuition motivating the choice of the functional.

3.2 Diffusion of Lie-group elements.

In order to efficiently compute the Euler-Lagrange equation corresponding to the generalized Ambrosio-Tortorelli functional (9), we transform the neighborhood of each point into the corresponding Lie-algebra elements before applying the diffusion operator. Using Lie-algebra representation of differential operators for rigid motion has been used before in computer vision [54], numerical PDE computations [30], path planning and optimal control theory [40, 35].

The Euler-Lagrange equation for the generalized Dirichlet energy measuring the map between two manifolds is given as [24]

$$\Delta_X g^{\alpha} + \Gamma^{\alpha}_{\beta\gamma} \left\langle \nabla g^{\beta}, \nabla g^{\gamma} \right\rangle_{g(x)} = 0, \tag{10}$$

where $\alpha, \beta\, \gamma$ enumerate the local coordinates of our group manifold, $se(3)$, and we use Einstein's notation according to which corresponding indices are summed over. $\Gamma^{\alpha}_{\beta\gamma}$ are the *Christoffel symbols* of $SE(3)$, which express the Riemannian metric's local derivatives. We refer the reader to [22] for an introduction to Riemannian geometry. Finally, Δ_X denotes the Laplace-Beltrami operator on the surface X.

In order to avoid computation of the Christoffel symbols, we transform the point and its neighbors using the logarithm map at that point in $SE(3)$. The diffusion operation is now affected only by the structure of the surface X. After applying the diffusion operator, we use the exponential map in order to return to the usual representation of the transformation. While this approach may suffer at discontinuities, where the logarithm and exponential maps are less accurate, it is at these continuities that we expect the diffusivity function v to be very small, perventing numerical instability. In practice, as we will demonstrate, this did not a significant problem.

4 Numerical Considerations

We now describe the algorithm for articulated motion estimation based on the minimization of the functional

$$E(g,v) = \lambda E_{\text{DATA}}(g) + \rho_{AT}(g,v), \tag{11}$$

where $E_{\text{DATA}}(g)$ is the matching term defined by Equation 2, and $\rho_{AT}(g,v)$ is defined in Equation 9. The main steps of the algorithm are outlined as Algorithm 1. Throughout the algorithm we parameterize $g(x)$ based on the first surface, given as a triangulated mesh, with vertices $\{x_i\}_{i=1}^{N}$, and an element from $SE(3)$ defined at each vertex. The triangulation is used merely to obtain a more consistent numerical diffusion operator, and can be avoided, for example by point-cloud based Laplacian approximations [10]. Special care is made in the choice of coordinates during the optimization as explained in Section 4.2.

4.1 Initial Correspondence Estimation

As in other motion segmentation and registration algorithms, some initialization of the matching between the surfaces must be used. One approach [6] is to use nonrigid surface matching for initialization. Another possibility, in the case of high framerate range scanners [65], is to exploit temporal consistency by $3D$ tracking. Yet another possible source for initial matches incorporates motion capture marker systems. Such sparse initial correspondence lends itself to interpolation of the motion field, in order to initialize a local ICP algorithm, and match the patch around each source point to the target mesh. In Figure 4, we use 30 matched points for initialization. This number of points is within the scope of current motion capture marker systems, or of algorithms for global nonrigid surface matching such as spectral methods [32, 39, 50, 47], or the *generalized multidimensional scaling* (GMDS) algorithm [13].

We expect that a better initial registration, as can be obtained e.g. using a smoothness assumption, or by pruning unsuitable candidates [51], will reduce the number of markers needed.

4.2 Diffusion of Lie-Group Elements

Rewriting the optimization over the functional in Equation 11 in a fractional step approach [66], we update the parameters w.r.t. each term of the functional in a suitable representation. The treatment of regularized data fitting in a fractional step approach is also similar to the approach taken by Thirion's demons algorithm [56, 46].

Using the transformation described in Section 3, the update step with respect to the regularization now becomes [26]

$$g^{k+1/2} = \exp\left(-dt\frac{\delta\rho_{\mathrm{AT}}}{\delta\tilde{g}}\right)g^k, v^{k+1} = v^k - dt\frac{\delta\rho_{\mathrm{AT}}}{\delta v} \tag{12}$$

where $\exp(A) = I + A + A^2/2! + A^3/3! + \ldots$ denotes the matrix exponential, $\tilde{g}$ denotes the logarithm transform of g, and dt denotes the time step. $\frac{\delta\rho_{\mathrm{AT}}}{\delta\tilde{g}}$ denotes the variation of the regularization term $\rho_{AT}(g)$ w.r.t. the Lie-algebra local representation of the solution, describing the Euler-Lagrange descent direction. $g(x)$ and the neighboring transformations are parameterized by a basis for matrices in $se(3)$, after applying the logarithm map at $g(x)$. The descent directions are given by

$$\begin{aligned}
\frac{\delta\rho_{\mathrm{AT}}}{\delta\tilde{g}_i} &= v^2\Delta_X(\tilde{g}_i) + v\langle\nabla v, \nabla\tilde{g}_i\rangle \\
\frac{\delta\rho_{\mathrm{AT}}}{\delta v} &= \langle\nabla g, \nabla g\rangle_{g(x)}v + 2\varepsilon\Delta_X(v) + \frac{(v-1)}{2\varepsilon},
\end{aligned} \tag{13}$$

where $\tilde{g}_i$ denote the components of the logarithmic representation of g. The discretization we use for Δ_X is a cotangent one suggested by [21], which has been shown to be convergent for relatively smooth and well-parameterized surfaces. It is expressed as

$$\Delta_X(u) \approx \frac{3}{\mathscr{A}_i} \sum_{j \in \mathscr{N}_1(i)} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} [u_j - u_i], \tag{14}$$

for a given function u on the surface X, where $\mathscr{N}_1(i)$ denotes the mesh neighbors of point i, and α_{ij}, β_{ij} are the angles opposing the edge ij in its neighboring faces. $\mathscr{A}_i$ denotes the area of the 1-ring around i in the mesh. After a gradient descent step w.r.t. the diffusion term, we take a step w.r.t. the data term.

$$g^{k+1} = P_{SE(3)}\left(g^{k+1/2} - dt \frac{\delta E_{DATA}}{\delta g} \right), \tag{15}$$

where $P_{SE(3)}(\cdot)$ denotes a projection onto the group $SE(3)$ obtained by correcting the singular values of the rotation matrix. We compute the gradient w.r.t. a basis for small rotation and translation matrices comprised of the regular basis for translation and the skew-matrix approximation of small rotations. We then reproject the update onto the manifold. This keeps the inaccuracies associated with the projecting manifold-constrained data [17, 30, 40, 26] at a reasonable level.

Finally, we note that we may not know in advance the points $y(x)$ which match X in Y. The correspondence can be updated based on the current transformations in an efficient manner similarly to the ICP algorithm. In our implementation we used the ANN library [7] for approximate nearest-neighbor search queries. We did not incorporate, however, any selective pruning of the matching candidates. These are often used in order to robustify such the ICP algorithm against ill-suited matches but are beyond the scope of this chapter.

Algorithm 1 Articulated Surface Segmentation and Matching

1: Given an initial correspondence.
2: **for** $k = 1, 2, \ldots$, until convergence **do**
3: Update $g^{k+1/2}, v^{k+1}$ w.r.t. the diffusion term, according to Equation 12.
4: Obtain g^{k+1} according to the data term, using Equation 15.
5: Update $y^{k+1}(x)$, the current estimated correspondence of the deformed surface.
6: **end for**

4.3 Visualizing Lie-Group Clustering on Surfaces

Finally, we need to mention the approach taken to visualize the transformations as the latter belong to a six-dimensional non-Euclidean manifold. Motivated by the widespread use of vector quantization in such visualizations, we use a clustering algorithm with spatial regularization. Instead of minimizing the Lloyd-Max quantization [31] cost function, we minimize the function

$$E_{VIS}(g_i, R_i) = \sum_i \int_{R_i} \|g - g_i\|^2 da + \int_{\partial R_i} v^2(s) ds, \tag{16}$$

where ∂R_i denotes the set of boundaries between partition regions $\{R_i\}_{i=1}^N$, g_i are the group representatives for each region, and $v^2(s)$ denotes the diffusivity term along the region boundary. The representation of members in $SE(3)$ is done via its embedding into $\mathbb{R}^{12}$, with some weight given to spatial location, by looking at the product space $\mathbb{R}^3 \times SE(3) \subset \mathbb{R}^{15}$. Several (about 50) initializations are performed, as is often customary in clustering, with the lowest cost hypothesis kept. The visualization is detailed as Algorithm 2

Algorithm 2 Spatially-consistent clustering algorithm

1: **for** $j = 1, 2, \ldots,$ for a certain number of attempts **do**
2: Use k-means on the spatial-feature space embedding, $\mathbb{R}^3 \times SE(3) \subset \mathbb{R}^{15}$, to get an initial clustering.
3: Use the clusters in order to optimize a spatially-regularized vector quantization measure,

$$C = \min_{g_i, \partial R_i} \int_X \|g - g_i\|^2 da + \int_{\partial R_i} v^2(s) ds,$$

where ∂R_i denotes the set of boundaries between clustered regions, g_i are the transformation representatives for each region, and $v^2(s)$ denotes the diffusivity term along the region boundary.
4: If C is lower than the lowest C found so far, keep the hypothesis.
5: **end for**
6: **return** current best hypothesis.

While this visualization algorithm coupled with a good initialization at each point can be considered as a segmentation algorithm in its own right, it is less general as it assumes a strict separation between the parts. One possible question that can be raise concerned the meaning behind vector quantization of points belonging to a manifold through its embedding into Euclidean space. In our case, since we are dealing with relatively well-clustered points (most of the points in a part move according to a single transformation in $SE(3)$), the distances on the manifold are not large and are therefore well-approximated by Euclidean ones. We further note, however, that the diffusion process lowered the score obtained in Equation 16 in the experiments we conducted, indicating a consistency between the two algorithms in objects with well-defined rigid parts.

5 Results

We now show the results of our method, in terms of the obtained transformations clusters and the Ambrosio-Tortorelli diffusivity function. In Figure 1 we show the segmentation obtained by matching two human body poses taken from the TOSCA dataset [14]. We visualize the transformations obtained using the clustering algorithm described in subsection 4.3. We initialized the transformations on the surface by matching the neighborhood of each surface point to the other poses using the true initial correspondence. The results of our method seem plausible, except for the missing identification of the right leg, which is due to the fact that its motion is limited between the two poses.

Figure 1 also demonstrates the results of comparing four poses of the same surface, this time with the patch-based data term described by (3). In our experiments the patch-based term gave a cleaner estimation of the motion, as is observed in the diffusivity function. We therefore demonstrate the results of minimizing the functional incorporating this data term. We also show the diffusivity function, which hints at the location of boundaries between parts, and thus justifies the assumption underlying Algorithm 2.

In Figure 2,3 we show the results of our algorithm on a set of 6 poses of a horse and camel surfaces taken from [55]. In this figure we compare our results to those of Wuhrer and Brunton [64], obtained on a similar set of poses with 10 frames. The results of our method seem to be quite comparable to those obtained by Wuhrer and Brunton, despite the fact that we use only 6 poses. We also note that both the diffusion scheme and the visualization algorithm gave a meaningful result for the tail part, which is not rigid and does not have a piecewise-rigid motion model.

In Figure 4 we demonstrate our algorithm, with an initialization of 30 simulated motion capture marker points, where the displacement is known. The relatively monotonous motion range available in the dynamic mesh sequence leads to a less complete, but still quite meaningful, segmentation of the horse, except for its head.

We also note the relatively low number of poses required for segmentation – in both Figure 2 and Figure 4 we obtain good results despite the fact that we use only a few poses, six and eight respectively.

Finally, in Figure 4 we demonstrate initialization of our method based on a sparse point set, with 30 known correspondence points. The points are arbitrarily placed using farthest point sampling [25, 29]. This demonstrates a possibility of initializing the algorithm using motion capture markers, coupled with a 3D reconstruction pipeline, for object part analysis. While the examples shown in this chapter are synthetic, this example shows that the algorithm can be initialized with data obtained in a realistic setup.

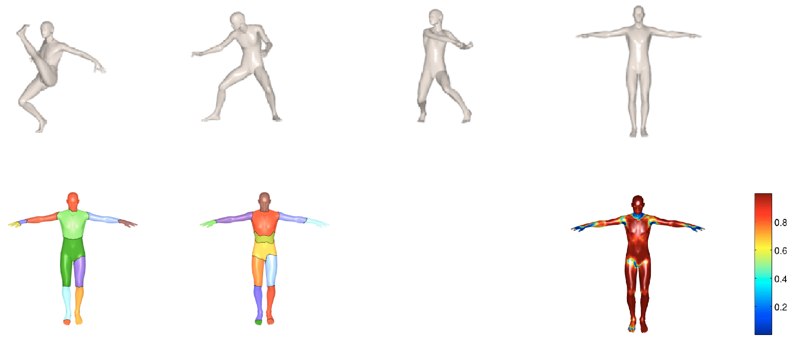

Fig. 1 Segmenting a human figure. Top row: the set of poses used. Bottom row, left to right: the transformations obtained from the two left most poses, the transformations obtained from all four poses using Equation 3 as a data term, and the Ambrosio-Tortorelli diffusivity function based on four poses.

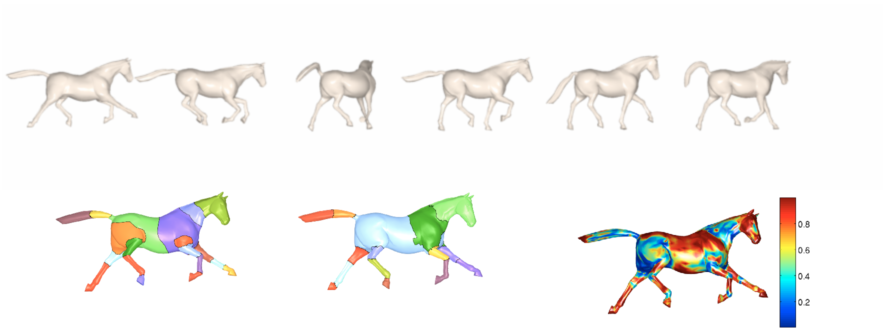

Fig. 2 Segmenting a horse dynamic surface motion based on six different poses. Top row: the poses used. Bottom row, left to right: a visualization of the transformations of the surface obtained by our method, and the segmentation results obtained by [64], and the diffusivity function v.

6 Conclusion

In this chapter we present a new method for motion-based segmentation of articulated objects, in a variational framework. The method is based on minimizing a generalized Ambrosio-Tortorelli functional regularizing a map from the surface onto the Lie-group $SE(3)$. The results shown demonstrate the method's effectiveness, and compare it with state-of-the-art articulated motion segmentation algorithms. The functional we suggest can be easily tailored to specific problems where it can be contrasted and combined with domain-specific algorithms for articulated object analysis. In future work we intend to adapt the proposed algorithm to real data from range

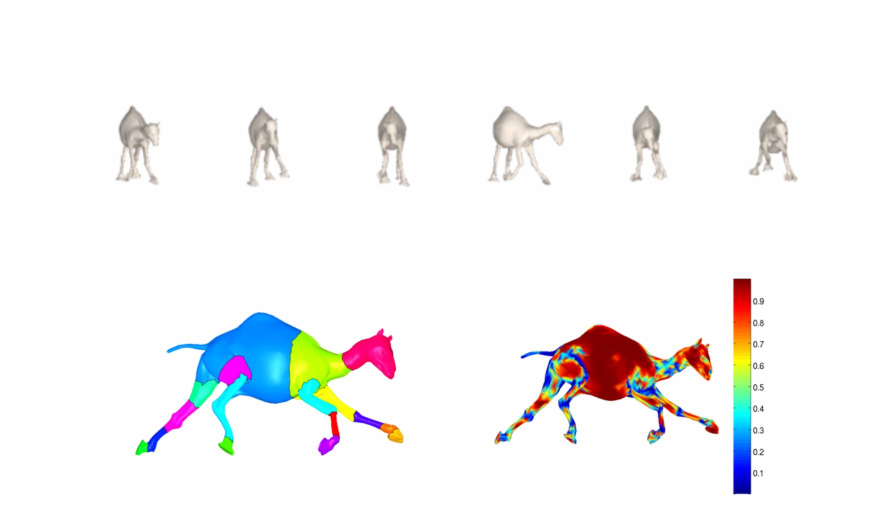

Fig. 3 Segmenting a camel dynamic surface motion based on six different poses. Top row: the poses used. Bottom row, left to right: a visualization of the transformations of the surface obtained by our method and the diffusivity function v.

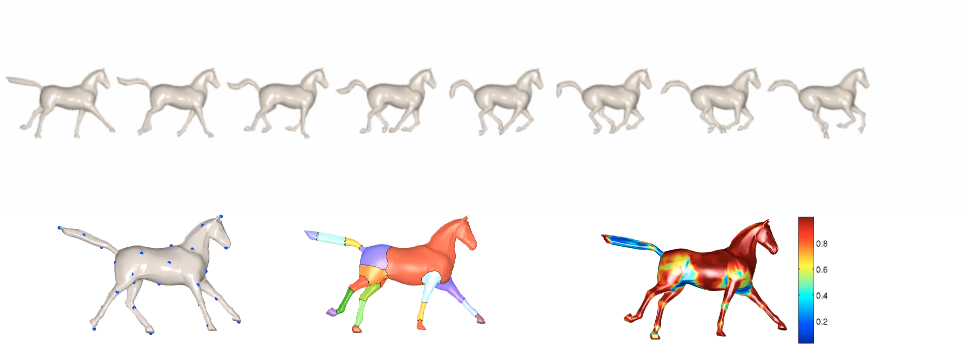

Fig. 4 Segmenting a horse dynamic surface motion with a given sparse initial correspondences. Top row: the eight random poses used. Bottom row, left to right: the set of points used for initializing the transformations, and a visualization of the transformations obtained, and the diffusivity function v.

scanners, and explore initialization methods as well as use the proposed framework in other applications such as articulated surfaces tracking and denoising.

References

1. E. J. Alexander, C. Bregler, and T. P. Andriacchi. Non-rigid modeling of body segments for improved skeletal motion estimation. *Computer Modeling in Engineering and Science*, 4:351–364, 2003.
2. L. Ambrosio and V. M. Tortorelli. Approximation of functional depending on jumps by elliptic functional via Γ-convergence. *Communications on Pure and Applied Mathematics*, 43(8):999–1036, 1990.
3. T. Amiaz and N. Kiryati. Piecewise-smooth dense optical flow via level sets. *Int. J. of Comp. Vis.*, 68(2):111–124, 2006.
4. M. S. Andersen, D. L. Benoit, M. Damsgaarda, D. K. Ramsey, and J. Rasmussen. Do kinematic models reduce the effects of soft tissue artefacts in skin marker-based motion analysis? an in vivo study of knee kinematics. *J. of Biomechanics*, 43:268–273, 2010.
5. D. Anguelov, D. Koller, H.-C. Pang, P. Srinivasan, and S. Thrun. Recovering articulated object models from 3D range data. In *Proc. Conf. on Uncertainty in Artificial Intelligence*, pages 18–26, Arlington, Virginia, United States, 2004. AUAI Press.
6. R. Arcila, S. K. Buddha, F. Hétroy, F. Denis, and F. Dupont. A framework for motion-based mesh sequence segmentation. In *Int. Conf. on Comp. Graphics, Visual. and Comp. Vision*, Plzeň, Czech Republic, 2010.
7. S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *J. ACM*, 45(6):891–923, 1998.
8. A. P. Ashbrook, R. B. Fisher, C. Robertson, and N. Werghi. Segmentation of range data into rigid subsets using surface patches. In *Int. Conf. on Comp. Vis.*, pages 201–206, 1998.
9. M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal. Mesh segmentation - a comparative study. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, pages 7–18, Washington, DC, USA, 2006. IEEE Computer Society.
10. M. Belkin, J. Sun, and Y. Wang. Constructing Laplace operator from point clouds in $\mathbb{R}^d$. In *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '09, pages 1031–1040, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.
11. P. J. Besl and N. D. McKay. A method for registration of 3D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.
12. R. Brockett. Robotic manipulators and the product of exponential formula. In P. Fuhrman, editor, *Proc. Mathematical Theory of Networks and Systems*, pages 120–129, 1984.
13. A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proc. Natl. Acad. Sci. USA*, 103(5):1168–1172, January 2006.
14. A. M. Bronstein, M. M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer-Verlag New York Inc, 2008.
15. A. Bruhn, J. Weickert, and C. Schnörr. Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods. *Int. J. of Comp. Vis.*, 61(3):211–231, 2005.
16. C. Brune, H. Maurer, and M. Wagner. Detection of intensity and motion edges within optical flow via multidimensional control. *SIAM J. Imag. Sci.*, 2(4):1190–1210, 2009.
17. E. Celledoni and B. Owren. Lie group methods for rigid body dynamics and time integration on manifolds. *Computer Methods in Applied Mechanics and Engineering*, 19:421–438, 1999.
18. X. Chen, A. Golovinskiy, and T. Funkhouser. A benchmark for 3D mesh segmentation. *ACM Trans. on Graphics*, 28(3), Aug. 2009. Article 73.
19. Y. Chen and G. Medioni. Object modelling by registration of multiple range images. *Image Vision Comput.*, 10:145–155, April 1992.
20. D. Cremers and S. Soatto. Motion competition: A variational framework for piecewise parametric motion segmentation. *Int. J. of Comp. Vis.*, 62(3):249–265, 2005.
21. M. Desbrun, M. Meyer, P. Schroder, and A. H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. *SIGGRAPH*, pages 317–24, 1999.
22. M. P. do Carmo. *Riemannian Geometry*. Birkhäuser Verlag, Boston, MA, 1992.

23. R. Duits and B. Burgeth. Scale spaces on Lie groups. In *Scale Space and Var. Methods in Comp. Vis.*, volume 4485 of *LNCS*, pages 300–312, Berlin, Heidelberg, 2007. Springer-Verlag.
24. J. J. Eells and J. H. Sampson. Harmonic mappings of Riemannian manifolds. *American J. of Math*, 86(1):106–160, 1964.
25. T. F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theor. Comput. Sci.*, 38:293–306, 1985.
26. Y. Gur and N. A. Sochen. Regularizing flows over lie groups. *J. of Math. in Imag. and Vis.*, 33(2):195–208, 2009.
27. B. C. Hall. *Lie Groups, Lie Algebras,and Representations, An Elementary Introduction*. Springer-Verlag, 2004.
28. S. Hauberg, S. Sommer, and K. S. Pedersen. Gaussian-like spatial priors for articulated tracking. In *European Conf. Comp. Vis.*, volume 6311 of *LNCS*, pages 425–437. Springer-Verlag, 2010.
29. D. Hochbaum and D. Shmoys. A best possible approximation for the k-center problem. *Mathematics of Operations Research*, 10(2):180–184, 1985.
30. A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna. Lie group methods. *Acta Numerica*, pages 215–365, 2000.
31. A. K. Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.
32. V. Jain and H. Zhang. Robust 3D shape correspondence in the spectral domain. In *Proc. of Shape Modeling International*, pages 118–129. IEEE Computer Society, 2006.
33. D. L. James and C. D. Twigg. Skinning mesh animations. *SIGGRAPH*, 24(3):399–407, Aug. 2005.
34. E. Kalogerakis, A. Hertzmann, and K. Singh. Learning 3D Mesh Segmentation and Labeling. *ACM Trans. on Graphics*, 29(4), 2010. Article 102.
35. M. Kobilarov, K. Crane, and M. Desbrun. Lie group integrators for animation and control of vehicles. *ACM Trans. Graph.*, 28(2):1–14, 2009.
36. I. Kompatsiaris, D. Tzovaras, and M. G. Strintzis. Object articulation based on local 3D motion estimation. In *Proc. of ECMAST*, pages 378–391, London, UK, 1999. Springer-Verlag.
37. T.-Y. Lee, Y.-S. Wang, and T.-G. Chen. Segmenting a deforming mesh into near-rigid components. *Vis. Comput.*, 22(9):729–739, 2006.
38. N. Litke, M. Droske, M. Rumpf, and P. Schröder. An image processing approach to surface matching. In *Proc. Eurographics Symp. on Geom. Proc*, pages 207–216, Aire-la-Ville, Switzerland, 2005. Eurographics Association.
39. D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer. Articulated shape matching using Laplacian eigenfunctions and unsupervised point registration. In *CVPR*, 2008.
40. A. Müller and Z. Terze. Differential-geometric modelling and dynamic simulation of multibody systems. *Strojarstvo: Journal for Theory and Application in Mechanical Engineering*, 51(6):597–612, 2009.
41. D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.
42. R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC, 1 edition, March 1994.
43. T. Nir, A. M. Bruckstein, and R. Kimmel. Over-parameterized variational optical flow. *Int. J. of Comp. Vis.*, 76(2):205–216, 2008.
44. T. Nir, R. Kimmel, and A. M. Bruckstein. Variational approach for joint optic-flow computation and video restoration. Technical Report CIS-05, Technion, 2005.
45. F. C. Park, J. E. Bobrow, and S. R. Ploen. A Lie group formulation of robot dynamics. *Int. J. Rob. Res.*, 14:609–618, December 1995.
46. X. Pennec, P. Cachier, and N. Ayache. Understanding the "demon's algorithm": 3D non-rigid registration by gradient descent. In *MICCAI*, pages 597–605, 1999.
47. D. Raviv, A. Dubrovina, and R. Kimmel. Hierarchical shape matching. In A. M. Bruckstein, B. ter Haar Romeny, A. M. Bronstein, and M. M. Bronstein, editors, *Scale Space and Var. Methods in Comp. Vis.*, volume 6667 of *LNCS*, 2011. Accepted.

48. G. Rosman, M. M. Bronstein, A. M. Bronstein, A. Wolf, and R. Kimmel. Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In *Scale Space and Var. Methods in Comp. Vis.*, 2011. submitted.
49. L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D Letters*, 60:259–268, 1992.
50. M. R. Ruggeri, G. Patanè, M. Spagnuolo, and D. Saupe. Spectral-driven isometry-invariant matching of 3D shapes. *Int. J. of Comp. Vis.*, 89(2-3):248–265, 2010.
51. S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *Third International Conference on 3D Digital Imaging and Modeling (3DIM)*, pages 145–152, June 2001.
52. B. Sapp, A. Toshev, and B. Taskar. Cascaded models for articulated pose estimation. In *European Conf. Comp. Vis.*, pages 406–420. Springer-Verlag, 2010.
53. A. Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, 2008.
54. R. Subbarao and P. Meer. Nonlinear mean shift over Riemannian manifolds. *Int. J. of Comp. Vis.*, 84(1):1–20, 2009.
55. R. W. Sumner and J. Popović. Deformation transfer for triangle meshes. In *SIGGRAPH*, pages 399–405, New York, NY, USA, 2004. ACM.
56. J. P. Thirion. Image matching as a diffusion process: an analogy with Maxwell's demons. *Medical Image Analysis*, 2(3):243–260, Sept. 1998.
57. J. Tierny, J.-P. Vandeborre, and M. Daoudi. Fast and precise kinematic skeleton extraction of 3D dynamic meshes. In *ICPR*, 2008.
58. W. Trobin, T. Pock, D. Cremers, and H. Bischof. An unbiased second-order prior for high-accuracy motion estimation. pages 396–405. Springer-Verlag, 2008.
59. O. Tuzel, F. Porikli, and P. Meer. Learning on lie groups for invariant detection and tracking. In *CVPR*, 2008.
60. M. Žefran, V. Kumar, and C. Croke. On the generation of smooth three-dimensional rigid body motions. *Robotics and Automation, IEEE Transactions on*, 14(4):576 –589, Aug. 1998.
61. M. Žefran, V. Kumar, and C. Croke. Metrics and connections for rigid-body kinematics. *I. J. Robotic Res.*, 18(2):242, 1999.
62. S. Wang, Y. Wang, M. Jin, X. D. Gu, and D. Samaras. Conformal geometry and its applications on 3D shape matching, recognition, and stitching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(7):1209–1220, 2007.
63. A. Wolf, I. Sharf, and M. B. Rubin. Using Cosserat point theory for estimating kinematics and soft-tissue deformation during gait analysis. In J. Lenarcic and M. M. Stanisic, editors, *Advances in Robot Kinematics: Motion in Man and Machine*, pages 63–70. Springer Netherlands, 2010.
64. S. Wuhrer and A. Brunton. Segmenting animated objects into near-rigid components. *The Visual Computer*, 26:147–155, 2010.
65. T. Yamasaki and K. Aizawa. Motion segmentation for time-varying mesh sequences based on spherical registration. *EURASIP Journal on Applied Signal Processing*, pages 1–9, 2009. Article 346425.
66. N. N. Yanenko. *The method of fractional steps: solution of problems of mathematical physics in several variables*. Springer-Verlag, 1971. Translated from Russian.
67. D. Zhang and M. Hebert. Harmonic maps and their applications in surface matching. In *CVPR*, volume 2, pages 524–530, 1999.