Eurographics Workshop on 3D Object Retrieval (2012)
M. Spagnuolo, M. Bronstein, A. Bronstein, and A. Ferreira (Editors)

Articulated Motion Segmentation of Point Clouds by Group-Valued Regularization

G. Rosman†[1], A. M. Bronstein[2], M. M. Bronstein[3] and R. Kimmel[1]

[1]Dept. of Computer Science, Technion – Israel Institute of Technology, Haifa, 32000, Israel
[2]School of Electrical Engineering, Faculty of Engineering, Tel Aviv University, Ramat Aviv 69978, Israel
[3]Institute of Computational Science, Faculty of Informatics, Universitá della Svizzera italiana, CH - 6904 Lugano, Switzerland

Abstract

Motion segmentation for articulated objects is an important topic of research. Yet such a segmentation should be as free as possible from underlying assumptions so as to fit general scenes and objects.

In this paper we demonstrate an algorithm for articulated motion segmentation of 3D point clouds, free of any assumptions on the underlying model and yet firmly set in a well-defined variational framework.

Results on scanned images show the generality of the proposed technique and its robustness to scanning artifacts and noise.

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

1. Introduction

Surface segmentation is a well-known research topic in the computer graphics and computer vision communities [AF98, AKM*06, CGF09, SF11]. Examples for the use of surface segmentation include 3D scene analysis [SF11], part-based recognition [HKDH04] and 3D video compression [Len99], among others.

In many cases, motion cues provide us with a very strong hint on the structure and association of object parts in the scene. Thus, they serve a fundamental role in 3D object analysis and scene interpretation, which are important for many computer vision tasks. It comes as little surprise that motion based segmentation of surfaces is in itself an active branch of surface segmentation [AF98, AKP*04, JT05, LWC06, TVD08, WB10, ABH*10].

In computer graphics algorithms for motion-based segmentation are known as dynamic mesh algorithms or skeletonization algorithms. This term is often used with a presumed correspondence, known to some extent between the surfaces. As this assumption is not plausible in many cases, we wish to avoid it in motion-based segmentation.

Moreover, the raw input in most applications is based on depth scanners such as structured light systems, laser range sensors, or time-of-flight sensors. This is especially true with the introduction of commodity depth scanners. Data from depth sensors does not have a meaningful predetermined topology – determining this topology is in itself scene segmentation, which should not be a preprocessing step, but rather part of scene understanding as a whole. The most basic representation of the input is that of a cloud of points without any sampling or mesh structure.

Lie group theory plays an important part in motion understanding [PBP95,TPM08,RS10], analysis [LGF09,BaAP10] and synthesis [PBP95, vKC98, PLZ*08, KCD09]. They provide a well-defined axiomatic approach for motion interpretation. It is only natural to find uses for them in variational schemes for 3D motion analysis [RBB*11], where they provide a natural tool for motion understanding. Such tools should also play a role in analysis of point cloud data.

In this paper we treat the topic of variational motion-based segmentation for articulated objects sampled as point clouds. We propose a general framework for motion segmentation, that is based on a minimal set of assumptions, using diffusion of Lie-group elements on point clouds. We show that with reasonable discretization schemes, this framework can apply to detection of articulated parts in noisy range scans

† This research was supported by European Community's FP7-ERC program, grant agreement no. 267414.

available from commodity range scanners as well as other sources.

Specifically, we do not make in this paper any assumptions on the topology of the object to be segmented beyond the notion that it is a surface, or several surfaces, sampled in a reasonably consistent manner. Unlike many algorithms for pose estimation and articulated object segmentation, we do not assume that the object consists solely of rigidly-moving parts. Rather, we merely assume that if points do belong to a rigid part they should undergo the same rigid transformation between object poses. We allow for points not to belong to rigid parts, but prefer a piecewise-rigid interpretation of the motion if one exists.

Recently, Rosman et al. [RBB*11] suggested a framework for variational motion segmentation that is independent of assumptions on the topology of the parts. Yet, the proposed method was still based on triangulated surfaces for the diffusion of rigid transformations. In this paper we free ourselves of such assumptions, providing a complete pipeline for motion estimation based on point clouds with a general structure, and show that the framework works with noisy data scanned using off-the-shelf equipment.

We formulate a diffusion process on point clouds involving group-valued functions defined on the points, and use this process in order to regularize in a piecewise smooth manner the transformations between several sampled poses of the same surface. This allows us to analyze the surface and detect rigidly-moving parts in the $3D$ scene in the most general way possible.

We formulate our assumptions and resulting model in Section 2. The algorithm, along with discussion of relevant numerical schemes are given in Section 3. The results of our algorithm are shown in Section 4. Section 5 concludes the paper.

2. Model Formulation

We now develop the proposed model, while stating our underlying assumptions, which are as lenient as possible. We assume that we are given a set of point clouds sampled from a surface at different poses. These samples may be partial, and with sampling errors as well as topological noise. We expect the surface to be regular in most places, allowing us to discuss smoothness and continuity of functions, given a reasonable sampling of the surface. Moreover, since our algorithm is defined with implicit segmentation in mind, we do not wish to presume a specific structure of the object, but assume the surface has a reasonable structure, whose parts are not too delicate compared to the motions involved and the sampling of the surface. If the sampling is dense enough, we wish for this structure to be inferred from local neighborhoods in the given point clouds in an implicit manner, without specific surface fitting steps, save for local steps often taken as part of mesh-free discretization methods.

Furthermore, since the segments to be detected are not known in advance, some of the points may not even belong to well defined rigidly moving parts. Instead, we merely expect points that belong to the same part to be moving together between different scans of the same object. Even when rigid parts do exist, the contours of each part need not form a closed simple curve.

One simplification we make in this paper is that of a relatively complete common surface which is mapped into the other surface poses. This allows us to describe the detection of rigid parts as a diffusion process defined on this common surface, while still allowing for small partially occluded parts to be handled by carefully defining our notion of registration between poses. This assumption is valid, however, since the motions allowed are still large enough to provide cues for segmentation, as will be shown in Section 4.

The small set of assumptions we have made and our search for an axiomatic approach for the problem of motion segmentation lead us to the model we now describe, along with its associated cost function and minimizing PDEs. We first make a short detour and describe the relevant notion of Lie-groups. Describing rigid motion using Lie-groups allows us to discuss regularity of motion in a well-defined way, and will allow us to describe smoothing and piecewise-smoothness of transformations fields on surfaces. We will then proceed to detail our model.

2.1. A Brief Introduction to Lie Groups

We now give a brief description of Lie-groups, and refer the interested reader to the literature (see [Hal04], for example). A Lie-group is a manifold endowed with an algebraic group structure. Due to its manifold structure, a Lie-group can be locally described by a tangent plane. Because of the group structure of the Lie-group, the tangent plane can be homeomorphically mapped onto the tangent plane at the identity element.

The resulting linear space, known as a Lie-algebra, provides us with a uniform way of treating smoothness and similarity between neighboring elements, and more specifically, allows us to define differential regularization of transformation fields mapping some smooth domain onto the Lie-group.

In our case, the Lie-group that describes rigid transformations in an n-dimensional space is the special-Euclidean group $SE(n)$. This group is defined in matrix form as the group

$$SE(n) = \left\{ \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix}, \mathbf{R}^T\mathbf{R} = I_{n\times n}, \mathbf{t} \in \mathbb{R}^n \right\}, \quad (1)$$

with matrix multiplication as the group action.

The associated Lie-algebra is the space $se(n)$

$$se(n) = \left\{ \begin{pmatrix} \mathbf{A} & \mathbf{v} \\ \mathbf{0} & 0 \end{pmatrix}, \mathbf{A}^T = -\mathbf{A}, \mathbf{v} \in \mathbb{R}^n \right\}, \quad (2)$$

2.2. Motion Segmentation via Group-valued Regularization

We now describe the proposed model. We assume each pose of the object to be a surface, or a set of surface fragments due to occlusions, self-occlusions, or sampling artifacts. We choose one of these poses to be a common domain (removing this assumption is possible but is beyond the scope of this work). Let $\mathcal{S}_1$ be the surface pose chosen as our parameterization domain. Several additional poses are available of the object. These are denoted by $\{\mathcal{S}_i\}, i = 2, \ldots N$, and need not have the same topological structure as $\mathcal{S}_1$.

We do, however, expect a large overlap between surface $\mathcal{S}_1$ and the other poses $\{\mathcal{S}_i\}$. Furthermore, for the type of objects and scenes we are interested in, we expect the transformation between $\mathcal{S}_1$ and $\mathcal{S}_i$ to be locally-rigid on much of the surface $\mathcal{S}_1$. Such transformations retain the distances inside a local neighborhood, and can generally be described by a rotation, followed by a translation. Regions where the assumption of a single rigid transformation between each two poses holds are known as *rigid parts*. Detecting rigid parts is important in many understanding and recognition tasks, due to the large number of approximately articulated objects around us, and is the focus of this paper.

We describe the local transformation between $\mathcal{S}_1$ and each other pose surface $\mathcal{S}_i$ as a map $g_i(x) : \mathcal{S}_i \to \mathcal{G}$ from the surface onto a transformation group of all the rigid transformations. Ideally, in the context of articulated motion segmentation we expect the same transformation to apply to each of the points in a rigidly moving part. Thus, we wish to define a model describing the detection of rigidly moving parts as piecewise smooth regularization of maps on the surface $\mathcal{S}_1$.

A natural choice for parameterizing rigid transformations is the Lie-group $SE(3)$. This Lie-group, along with the corresponding Lie-algebra provides us with a representation on which we can define smoothness measures for maps through the generalized Dirichlet functional [ES64],

$$E_{Dir} = \int_{\mathcal{S}_1} \|g^{-1}\nabla g\|_F^2 dx, \quad (3)$$

where dx denotes the area element on the surface and $g^{-1}\nabla g$ describes the Jacobian in terms of local coordinates on $SE(3)$ of the map g with respect to points on $\mathcal{S}_1$.

This functional provides us with a tool for discussing the smoothness of maps onto manifolds such as the transformation groups. Yet, our goal is to find a segmentation of the surface, which is is strongly related to piecewise-smooth regularization via the Mumford-Shah functional [MS89]. This model can be approximated (via a Γ-convergence process) by the Ambrosio-Tortorelli scheme using optimization functions of the form

$$E_{AT} = \int_{\mathcal{S}_1} v^2 \|g^{-1}\nabla g\|_F^2 + \varepsilon \|\nabla v\|^2 + \frac{(1-v)^2}{4\varepsilon} dx, \quad (4)$$

where v is a diffusivity function accepting values in the interval $[0,1]$. This function is easily extended to the case of multiple surfaces with transformation maps describing the transformation between pose $\mathcal{S}_1$ and each of the other poses $\mathcal{S}_i$,

$$E_{AT} = \int_{\mathcal{S}_1} v^2 \sum_{i=2}^{N} \|g_i^{-1}\nabla g_i\|_F^2 + \varepsilon \|\nabla v\|^2 + \frac{(1-v)^2}{4\varepsilon} dx, \quad (5)$$

The optimality condition for the regularity term is given by its Euler-Lagrange equations. In order to avoid computing the Christoffel symbols and the associated high-order derivative operators on noisy sampled surfaces we first transform the data to a locally Euclidean approximation using the Rodrigues formula, effectively flattening the manifold of $SE(3)$ into a local representation. In this locally-Euclidean parameterization, the optimality conditions with respect to g_i and v become the diffusion equations,

$$\frac{\delta E_{AT}}{\delta g_i} = v^2 \Delta_{\mathcal{S}_1} g_i \quad (6)$$

$$\frac{\delta E_{AT}}{\delta v} = 2v \sum_{i=2}^{N} \|g_i^{-1}\nabla g_i\|_F^2 + 2\varepsilon \Delta_{\mathcal{S}_1} v + \frac{(v-1)}{2\varepsilon}. \quad (7)$$

In order for this regularity measure to make sense, we must require the transformations to fit the surface $\mathcal{S}_1$ into the other poses $\mathcal{S}_i$ in some sense. On the other hand, constrained minimization is not appropriate in our case, where the correspondences are not know. We therefore incorporate a second energy term favoring correct data fitting,

$$E_{DATA} = \int_{\mathcal{S}_1} \sum_{i=2}^{N} \Psi\left(\|g_i(x)(x) - y_i(x)\|^2 \right) dx, \quad (8)$$

where $y_i(x)$ is a latent variable signifying the assumed correspondence of point x on surface i, according to the transformation $g_i(x)$, and $\Psi(\cdot)$ is a robust fitting function. Given x and $g_i(x)$ the update of $y_i(x)$ is a corresponding point search, similar to the ones encountered in the context of iterative closest point (ICP) algorithms [BM92, CM92].

Rigid transformations have locally 6 degrees of freedom whereas a single point matching only provides 3 contraints. This constitutes an overparameterized motion estimation process [NBK08], with the missing constraints provided either by fitting a finite neighborhood around the point, or by forcing overall smoothness of the resulting transformation field. The former case is clearly analogous to the Lucas-Kanade registration algorithm [LK81], and the latter resembles the Horn-Schunck algorithm [HS81].

We suggest to use a combined global-local approach

[BWS05] approach. Updating the transformations so as to locally match the other surfaces $\mathcal{S}_i$ results in an ICP-like fitting term at each point x,

$$E_{ICP} = \int_{\mathcal{S}_1} \int_{x' \in \mathcal{N}(x)} \Sigma_{i=2}^{N} \Psi \left(\| g_i(x)\left(x'\right) - y_i\left(g_i(x)x'\right) \|^2 \right) dx' dx, \tag{9}$$

Weighting the ICP data term with the Ambrosio-Tortorelli smoothness term gives us a cost function suitable to the model we have describe,

$$E = \lambda E_{AT} + E_{ICP}. \tag{10}$$

We will discretize this functional over point clouds and minimize it in Section 3.

2.3. Group-valued Regularization on Point Clouds

As mentioned before, in order to discuss group-valued regularization and motion segmentation, the continuous model defined above should be made relevant for the data at hand. In this context we acknowledge the fact that the sampled surfaces are usually incomplete and noisy, and that the sampling will not be uniform. The discretization of the operators and flows we will choose should therefore be robust to topological errors and occlusions. Luckily, working directly with the point clouds allows us to discretize the PDEs given in equations 6,7 without any assumptions on the global scene structure. In addition, the data term should take into account the possibility of partial matching between incomplete scans of an object. In general, such concerns have been thoroughly investigated in the implementation of ICP algorithms [RL01]. If the missing parts are relatively small, we can easily use robust data fitting in order to avoid the influence of this incorrect matches by treating them as outliers. This approach suffices in the case of scans taken from short video sequences where the motions are relatively small, and are used to detect candidates for rigid parts. In more general cases, a more complete approach for partial ICP should be incorporated, but this is beyond the scope of this work. Let $\mathcal{S}_i$ be sampled as a point clouds $\{(\mathbf{s}_i)_j\}_{i=1}^{N}$. The maps g_i are now defined discretely by elements of $SE(3)$ at each sampled surface point. Discrete operators are defined on functions on $(\mathbf{s}_1)$. Specifically, we require a definition of the Laplacian, gradient, and a semilocal fitting process on point clouds. These are discussed in Section 3.

3. Algorithmic Description

We now detail our algorithm for motion segmentation of point clouds. This algorithm minimizes the functional described in Section 2 with respect to the diffusivity function v and the transformation elements $g_i(x)$. The complete algorithm is described as Algorithm 1.

Algorithm 1 Fast TV regularization of matrix-valued data

Require: Point clouds $\mathbf{s}_i, i = 1, \ldots, N$
1: Initialize correspondences between point clouds based on tracking or motion capture markers.
2: **for** $k = 1, 2, \ldots,$ until convergence **do**
3: Update functions g_i,v in an alternating minimization fashion,
- Update $g_i(x)$ according to Equation 6 and according to Equation 12 in a fractional-step manner.
- Update $v(x)$ according to Equation 7.

4: **end for**

3.1. Differential operators on point clouds

In order to minimize functionals on surfaces described by point clouds, differential operators on point clouds must first be defined. Several techniques are available for computing differential operators on point clouds. One set of methods approaches the problem by approximating the tangent plane at each point, and then reconstructing a local operator on the surface based on this approximation. Discretization schemes based on this approach include the work of Belkin et al. [BSW09], and are strongly related to moving least squares approaches for surface estimation [Lev03].

Yet another approach avoids the need for tangent plane estimation by looking at a narrow-band around the surface. Techniques in this group include the *closest point method* [RM08] as well as other mesh-free techniques often used in physical simulations.

In our algorithm we chose approximations of the local tangent plane for the differential operators involved. We used the scheme suggested by Belkin et al. [BSW09] in order to compute the Laplacian weights. We use polynomial fitting weights in order to approximate function derivatives at each neighborhood. For this purpose we take the nearest neighboring points of each point without any outlier rejection or reweighting. We note that better results are to be expected, especially in the case of noisy data, but a thorough investigation of such scheme would be data dependent and is beyond the scope of this work.

3.2. An Ambrosio Tortorelli Scheme for Transformations on Point Clouds

Once the gradient and the Laplacian are defined on a point cloud, it is quite simple to use an Ambrosio-Tortorelli scheme on a point cloud. In our case, we take the same approach as suggested in [RBB*11], of first transforming neighboring points onto the tangent plane at point $g(x)$, performing the diffusion step, and transforming back. We write a diffusion step in the same notation as for fractional steps approach [Yan71],

$$(g_i)_j^{(n+\frac{1}{2})} = exp_{(g_i)_j^{(n)}}\left(\frac{\Delta t}{2}\sum_{k\in\mathcal{N}(j)} w_{jk} log_{(g_i)_j^{(n)}}(g_i)_k^{(n)}\right), \tag{11}$$

where w_{jk} are the weights given by the Laplacian matrix for the point cloud $\mathbf{s}_1$. As previously mentioned, our Laplacian operator based on the scheme suggested by Belkin et al. [BSW09], although other schemes are applicable.

We then optimize with respect to the registration of neighborhood $\mathcal{N}(j)$ by taking a partial ICP step,

$$(g_i)_j^{(n+1)} = (g_i)_j^{(n+\frac{1}{2})} - \frac{\Delta t}{2}\frac{\delta E_{ICP}(g_i^{(n+\frac{1}{2})})}{\delta g_i^{(n+\frac{1}{2})}}, \tag{12}$$

where the optimization step is done by gradient descent on the linearized rotation matrix and translation coefficient, followed by projection onto $SE(3)$. This update is in the spirit of fractional steps algorithms [Yan71], and specifically the registration-regularization cycle is akin to demons algorithms [Thi98, PCA99].

The update with respect to the ICP term requires a notion of a surface patch on the surface $\mathcal{S}_i$ that corresponds to the neighborhood of point x. Choosing the corresponding point in a robust manner strongly affects the convergence of ICP algorithms. Removal of outliers in the correspondence and pruning correspondences to prevent bias has been studied extensively, see for example [RL01]. Choosing the right expression for a surface-to-surface distance is also known to be crucial, see for example works by Mitra et al. [MGPG04]. In our implementation, due to the relatively good initialization, simple point-to-point distance function with only distance-based rejection of correspondences proved sufficient.

In addition we optimize the functional with respect to v based on Equation 7. We use the same Laplacian approximation as for the update of the functions $\{g_i\}$. Optimization with respect to v requires, however, the computation of the gradient norm on the surface. As mentioned previously, we approximate the gradient of each map by building a local polynomial approximation for functions using neighboring point values and differentiating with respect to the locally-estimated tangent coordinates.

3.3. Initialization for Motion Segmentation

Since the functionals we minimize are not convex, special care must be taken to assure reasonable initialization of the algorithm. In order to initialize the algorithm, the functions g_i should be estimated for every point in the point cloud $\mathbf{s}_1$. This can be done in several ways, as we now describe.

Initialization based on motion capture markers In a scenario where motion capture markers or nonrigid descriptors are available, we can propagate the sparse motion data into a dense correspondence. This correspondence will be used to estimate the initial solution for our algorithm. One way to obtain such an initial dense motion field is by Laplacian interpolation on the point cloud [Rus11]. We used the same Laplacian operator as we use in the rest of our algorithm in order to obtain an initial dense flow field using the motion capture markers as a (Dirichlet) boundary condition and solving the heat equation on the point cloud for each of the $3D$ motion components. This field is then used to initialize a local ICP search.

Initialization based on tracking In the case where the input is a $3D$ video, we can use temporal consistency to initialize the correspondence. While a local ICP process without spatial coherence can be used for motion analysis [RWT*11] for short sequences, for longer sequences, spatial coherence of the transformation can be crucial in order to avoid gross errors in the initialization. As initialization in our experiments on Kinect sensor data we use two approaches. One is a simple a local-ICP process from frame to frame that proved to be relatively error-prone, as seen in Figure 2. Another approach is initialization based on the coherent point drift [MS10] algorithm, which was used in Figure 3. Other, more elaborate approaches are also possible, including nonrigid registration algorithms [LSP08]. The use of such techniques was not necessary for reasonably slow motions, and is beyond the scope of this paper.

4. Results

We now show a few results of our algorithm. We demonstrate the segmentation of real point clouds obtained from laser scanners and Microsoft Kinect depth sensors. The examples are implemen In Figures 1–3 we use vector quantization (VQ, [Max60, Llo82]), in terms of the embedding $SE(3) \subset \mathbb{R}^{12}$, with multiple initializations in order to demonstrate the resulting transformations. In the examples shown, 40 initializations of vector quantization are used, at which point a minimal quantization cost is practically achieved and new hypotheses do not feature lower costs.

While vector quantization can be used in itself to provide segmentation of motion, using it over the raw estimated transformation create various artifacts. These are seen in the examples, where our piece-wise smooth regularization solution manages to fix these artifacts.

In addition, we show the Ambrosio-Tortorelli diffusivity field, where several of the main boundaries between parts can be seen.

In Figure 1 we demonstrate results from the SCAPE dataset [ASK*05]. The results are based on the algorithm with initialization using 200 initial matches, and use the first 5 frames of the dataset.

In Figures 2,3 we demonstrate results from a Kinect sensor. The transformation maps were initialized using frame-

Figure 1: *Visualization of the detected transformations before and after smoothing, using 6 frames from the SCAPE dataset. Colors show the vector quantization results on the transformations embedded into* $\mathbb{R}^{12}$

to-frame 3*D* tracking. Figure 2 demonstrates results on arbitrarily segmented part of the upper arm, with initialization based on local, patch-based, ICP between frames. For this experiment, 4 frames were taken. Figure 3 demonstrates results on a human hand doing a waving motion, with initialization based on the coherent point drift algorithm [MS10], with 6 frames taken for the segmentation. These results show the applicability of the proposed framework also for analysis of depth video from noisy data sources in an automated manner.

5. Conclusions

In this paper we demonstrate the possibility of using a general variational model involving a piecewise smooth regularization model and a registration-based data term for analysis of articulated objects and for finding the articulated parts using motion cues alone.

Specifically, we have shown that this very general framework, despite its differential formulation, is useful also for real data coming fron noisy data sources such as commodity range scanners.

In future work we intend to take this framework and incorporate the optimization scheme into a more general online analysis algorithm, as well as utilize additional segmentation priors in order to robustify the algorithm.

References

[ABH*10] ARCILA R., BUDDHA S. K., HÉTROY F., DENIS F., DUPONT F.: A framework for motion-based mesh sequence segmentation. In *Int. Conf. on Comp. Graphics, Visual. and Comp. Vision* (Plzen, Tchéquie, 2010), pp. 33–40. 1

[AF98] ASHBROOK A., FISHER R. B.: Segmentation of range data into rigid subsets using surface patches. In *ICCV* (1998), pp. 201–206. 1

[AKM*06] ATTENE M., KATZ S., MORTARA M., PATANE G., SPAGNUOLO M., TAL A.: Mesh segmentation - a comparative study. In *Proc. IEEE Int. Conf. on Shape Modeling and Applications* (Washington, DC, USA, 2006), IEEE Computer Society, pp. 7–18. 1

[AKP*04] ANGUELOV D., KOLLER D., PANG H.-C., SRINIVASAN P., THRUN S.: Recovering articulated object models from 3D range data. In *Proc. Conf. on Uncertainty in Artificial Intelligence* (Arlington, Virginia, United States, 2004), AUAI Press, pp. 18–26. URL: http://portal.acm.org/citation.cfm?id=1036843.1036846. 1

[ASK*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: shape completion and animation of people. *ACM Trans. Graph. 24*, 3 (2005), 408–416. 5

[BaAP10] BUE A. D., AO X. J., AGAPITO L., PALADINI M.: Bilinear factorization via augmented Lagrange multipliers. In *ECCV* (2010), Springer-Verlag, pp. 283–296. 1

[BM92] BESL P. J., MCKAY N. D.: A method for registration of 3D shapes. *IEEE Trans. Pattern Anal. Mach. Intell. 14*, 2 (1992), 239–256. 3

[BSW09] BELKIN M., SUN J., WANG Y.: Constructing Laplace operator from point clouds in R^d. In *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms* (Philadelphia, PA, USA, 2009), SODA '09, Society for Industrial and Applied Mathematics, pp. 1031–1040. 4, 5

[BWS05] BRUHN A., WEICKERT J., SCHNÖRR C.: Lucas/Kanade meets Horn/Schunck: combining local and global optic flow methods. *IJCV 61*, 3 (2005), 211–231. 4

[CGF09] CHEN X., GOLOVINSKIY A., FUNKHOUSER T.: A benchmark for 3D mesh segmentation. *ACM Trans. on Graphics 28*, 3 (Aug. 2009). 1

[CM92] CHEN Y., MEDIONI G.: Object modelling by registration of multiple range images. *Image Vision Comput. 10* (April 1992), 145–155. 3

[ES64] EELLS J. J., SAMPSON J. H.: Harmonic mappings of Riemannian manifolds. *American J. of Math 86*, 1 (1964), 106–160. 3

[Hal04] HALL B. C.: *Lie Groups, Lie Algebras,and Representations, An Elementary Introduction*. Springer Verlag, 2004. 2

[HKDH04] HUBER D., KAPURIA A., DONAMUKKALA R., HEBERT M.: Parts-based 3D object classification. In *CVPR* (2004), vol. 2, pp. 82–89. 1

[HS81] HORN B. K., SCHUNCK B. G.: Determining optical flow. *Artificial Intelligence 17* (1981), 185–203. 3

[JT05] JAMES D. L., TWIGG C. D.: Skinning mesh animations. *SIGGRAPH 24*, 3 (Aug. 2005), 399–407. 1

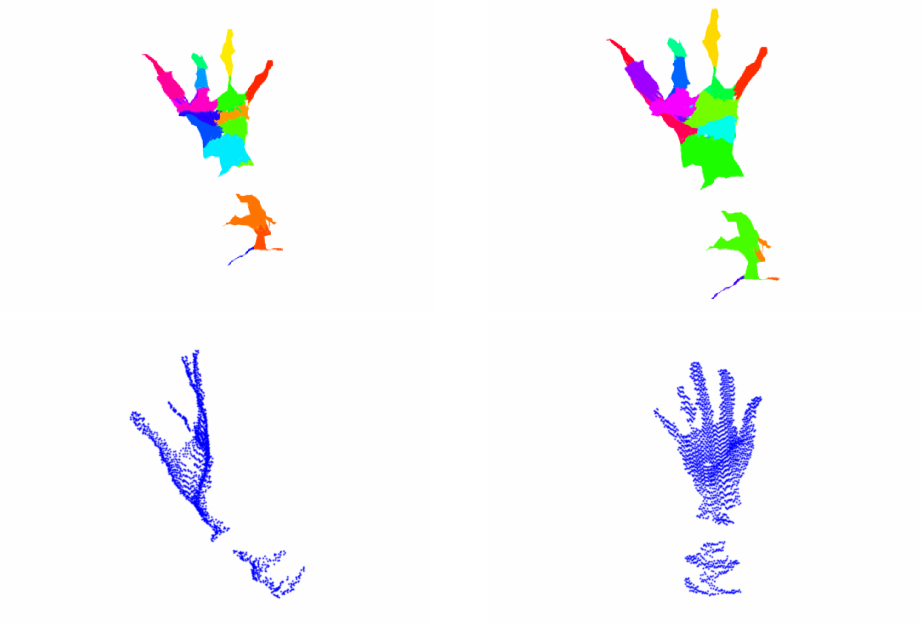

Figure 2: *Top row: Visualization of the detected transformations before and after regularization, based on a point cloud from a Kinect sensor at 70cm, using local ICP for initialization. Colors show the vector quantization results on the transformations embedded into $\mathbb{R}^{12}$. Left: visualization of the initial solution based on local-ICP between frames. Right: the result after optimization. Bottom row: the first input frame from the front/side. Note the fragmented surface*

[KCD09] KOBILAROV M., CRANE K., DESBRUN M.: Lie group integrators for animation and control of vehicles. *ACM Trans. Graph. 28*, 2 (2009), 1–14. 1

[Len99] LENGYEL J. E.: Compression of time-dependent geometry. In *Symposium on Interactive 3D Graphics* (1999), pp. 89–95. 1

[Lev03] LEVIN D.: Mesh-independent surface interpolation. *Geometric Modeling for Scientific Visualization 3* (2003). 4

[LGF09] LIN D., GRIMSON W., FISHER J.: Learning visual flows: A Lie algebraic approach. In *CVPR* (2009), pp. 747–754. 1

[LK81] LUCAS B. D., KANADE T.: An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence* (april 1981), pp. 674–679. 3

[Llo82] LLOYD S. P.: Least squares quantization in PCM. *IEEE Transactions on Information Theory IT-28*, 2 (1982), 129–137. 5

[LSP08] LI H., SUMNER R. W., PAULY M.: Global correspondence optimization for non-rigid registration of depth scans. *Computer Graphics Forum (Proc. SGP'08) 27*, 5 (July 2008). 5

[LWC06] LEE T.-Y., WANG Y.-S., CHEN T.-G.: Segmenting a deforming mesh into near-rigid components. *Vis. Comput. 22*, 9 (2006), 729–739. 1

[Max60] MAX J.: Quantizing for minimum distortion. *IRE Transactions on Information Theory IT-6*, 1 (1960), 7–12. 5

[MGPG04] MITRA N. J., GELFAND N., POTTMANN H., GUIBAS L.: Registration of point cloud data from a geometric optimization perspective. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (New York, NY, USA, 2004), SGP '04, ACM, pp. 22–31. 5

[MS89] MUMFORD D., SHAH J.: Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics 42*, 5 (1989), 577–685. 3

[MS10] MYRONENKO A., SONG X.: Point set registration: Coherent point drift. *IEEE Trans. Pattern Anal. Mach. Intell. 32*, 12 (Dec. 2010), 2262–2275. 5, 6

[NBK08] NIR T., BRUCKSTEIN A. M., KIMMEL R.: Over-parameterized variational optical flow. *IJCV 76*, 2 (2008), 205–216. 3

[PBP95] PARK F. C., BOBROW J. E., PLOEN S. R.: A Lie group formulation of robot dynamics. *Int. J. Rob. Res. 14* (December 1995), 609–618. 1

[PCA99] PENNEC X., CACHIER P., AYACHE N.: Understanding the "demon's algorithm": 3D non-rigid registration by gradient descent. In *MICCAI* (1999), pp. 597–605. 5

[PLZ*08] PARK W., LIU Y., ZHOU Y., MOSES M., CHIRIKJIAN G. S.: Kinematic state estimation and motion planning for stochastic nonholonomic systems using the exponential map. *Robotica 26* (July 2008), 419–434. 1

[RBB*11] ROSMAN G., BRONSTEIN M. M., BRONSTEIN A. M., WOLF A., KIMMEL R.: Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In *SSVM* (2011), vol. 6667 of *Lecture Notes on Computer Science*, pp. 616–627. 1, 2, 4

[RL01] RUSINKIEWICZ S., LEVOY M.: Efficient variants of the

Figure 3: *Top row: Visualization of the detected transformations before and after regularization, based on a point cloud from a Kinect sensor at 70cm. Colors show the vector quantization results on the transformations embedded into* $\mathbb{R}^{12}$*, greyscale shows the depth in regions that were not subject to the algorithm. Left: VQ visualization of the initial state obtained by the CPD algorithm. Right: visualization of the resulting state after optimization. Note the merged sections of the ring and middle finger, as well as additional artifacts vector quantization before the regularization. Bottom row: Left: Two surface reconstructions of the point cloud obtained from the Kinect. Note the relatively high noise level in the surface reconstructions. Right: The diffusivity function v.*

ICP algorithm. In *Third International Conference on 3D Digital Imaging and Modeling (3DIM)* (June 2001). 4, 5

[RM08] RUUTH S. J., MERRIMAN B.: A simple embedding method for solving partial differential equations on surfaces. *J. Comput. Phys. 227* (January 2008), 1943–1961. 4

[RS10] RAPTIS M., SOATTO S.: Tracklet descriptors for action modeling and video analysis. In *ECCV* (Sep. 2010), pp. 577–590. 1

[Rus11] RUSTAMOV R. M.: Average interpolating wavelets on point clouds and graphs. *CoRR abs/1110.2227* (2011). 5

[RWT*11] ROSMAN G., WANG Y., TAI X.-C., KIMMEL R., BRUCKSTEIN A. M.: *Fast Regularization of Matrix-Valued Images*. CAM-Report 11-87, UCLA, 2011. 5

[SF11] SILBERMAN N., FERGUS R.: Indoor scene segmentation using a structured light sensor. In *ICCV* (2011), pp. 601–608. 1

[Thi98] THIRION J. P.: Image matching as a diffusion process: an analogy with Maxwell's demons. *Medical Image Analysis 2*, 3 (Sept. 1998), 243–260. 5

[TPM08] TUZEL O., PORIKLI F., MEER P.: Learning on lie groups for invariant detection and tracking. In *CVPR* (2008). 1

[TVD08] TIERNY J., VANDEBORRE J.-P., DAOUDI M.: Fast and precise kinematic skeleton extraction of 3D dynamic meshes. In *ICPR* (2008), pp. 1–4. 1

[vKC98] ŽEFRAN M., KUMAR V., CROKE C.: On the generation of smooth three-dimensional rigid body motions. *Robotics and Automation, IEEE Transactions on 14*, 4 (Aug. 1998), 576 –589. 1

[WB10] WUHRER S., BRUNTON A.: Segmenting animated objects into near-rigid components. *The Visual Computer 26* (2010), 147–155. 1

[Yan71] YANENKO N. N.: *The method of fractional steps: solution of problems of mathematical physics in several variables*. Springer Verlag, 1971. Translated from Russian. 5