Variational Level Set Evolution for Non-rigid 3D Reconstruction from a Single Depth Camera

Miroslava Slavcheva, Maximilian Baust, Slobodan Ilic

Abstract—We present a framework for real-time 3D reconstruction of non-rigidly moving surfaces captured with a single RGB-D camera. Based on the variational level set method, it warps a given truncated signed distance field (TSDF) to a target TSDF via gradient flow without explicit correspondence search. We optimize an energy that contains a data term which steers towards voxel-wise alignment. To ensure geometrically consistent reconstructions, we develop and compare different strategies, namely an approximately Killing vector field regularizer, gradient flow in Sobolev space and newly devised accelerated optimization. The underlying TSDF evolution makes our approach capable of capturing rapid motions, topological changes and interacting agents, but entails loss of data association. To recover correspondences, we propose to utilize the lowest-frequency Laplacian eigenfunctions of the TSDFs, which encode inherent deformation patterns. For moderate motions we are able to obtain implicit associations via a term that imposes voxel-wise eigenfunction alignment. This is not sufficient for larger motions, so we explicitly estimate voxel correspondences via signature matching of lower-dimensional eigenfunction embeddings. We carry out qualitative and quantitative evaluation of our geometric reconstruction fidelity and voxel correspondence accuracy, demonstrating advantages over related techniques in handling topological changes and fast motions.

Index Terms—Non-rigid 3D reconstruction, signed distance field evolution, Laplacian eigenfunctions.

1 Introduction

The wide availability of off-the-shelf RGB-D sensors and the growing popularity of virtual and augmented reality have caused great progress in the realm of single-stream 3D reconstruction in recent years. Various techniques for capturing static environments have demonstrated impressive results [1], [2], [3], [4], [5], [6], [7]. However, real-life scenes also include moving people, interacting with objects in their surroundings and with each other. This requires the capture of non-rigidly moving surfaces, which is a highly unconstrained problem that still poses major challenges.

The problem is ill-posed because there are infinitely many warp fields that may deform one frame to the next [8]. While older techniques resorted to the use of multiple cameras [9], [10], [11], [12], [13] or templates [14], [15], [16], [17] in order to better constrain the solution space, nowadays methods that utilize a single RGB-D camera are emerging. DynamicFusion [18] first demonstrated real-time simultaneous tracking and reconstruction of non-rigid surfaces. Several works build over it, incorporating colour features [19], albedo constraints [20] or human-specific priors [21], [22]. Their results are of ever-improving visual quality, however, they are still constrained mainly to contrived motion without interactions or topological changes.

This paper addresses these issues through the use of TSDF evolution. The majority of recent approaches for both dynamic and static reconstruction employ a TSDF for storing the growing reconstruction [4], [18], [19]. One of the main advantages of this representation is its ability to smooth out noise when repeated measurements at the same voxel are averaged [23]. However, these methods intermittently revert back to a mesh representation in order to estimate correspondences for non-rigid alignment, thereby losing accuracy, computational speed and the TSDF capability to conveniently capture topological changes.

Therefore we propose a method that operates entirely within the TSDF representation. It warps a TSDF to a target

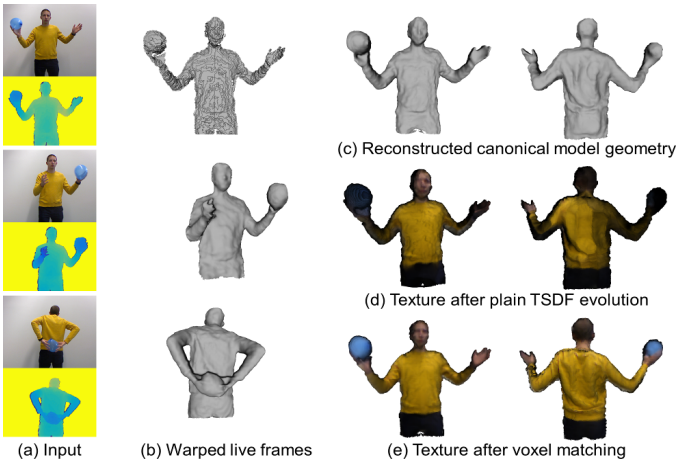


Fig. 1. Reconstruction of a person playing with a balloon. Our system takes a single RGB-D stream as input (a) and warps each frame towards the canonical model in order to grow it. Then the model is warped back towards the live depth for display to the user (b). The final output is a complete 3D model (c), whose colours would diffuse into each other if evolved with the same warp field (d), but become consistent if Laplacian eigenfunction signatures are matched for voxel correspondence (e). This example has been reconstructed with Sobolev gradient flow.

- *During this work M. Slavcheva was with the Chair for Computer Aided Medical Procedures and Augmented Reality, Technische Universität München (TUM CAMP) and Siemens Corporate Technology, Munich, Germany. She has moved to industry after comleting it.*
- *M. Baust is with TUM CAMP and NVIDIA, Munich, Germany.*
- *S. Ilic is with TUM CAMP and Siemens CT, Munich, Germany.*
- *E-mails: see http://campar.in.tum.de*

Manuscript received XX XX, 2018; revised XX XX, 2019.

TSDF via gradient flow without correspondence search, steered by a data term that imposes voxel-wise alignment. Furthermore, we analyze three strategies that ensure geometric plausibility. First, we include an approximately Killing vector field [24] energy term which enforces the estimated deformation field to generate locally nearly isometric motions, acting similar to an as-rigid-as-possible regularizer [25]. Second, instead of adhering to the commonly used gradient defined via an L^2 inner product, we apply gradient flow defined in Sobolev space [26], which acts as a pre-conditioner ensuring a coarse-to-fine evolution behaviour [27]. While the former of these two previously presented approaches [28] is slightly faster and thus allows for the incorporation of additional terms which impose desirable geometric properties, such as unit gradient magnitude, the latter one [29] achieves higher geometric detail without over-smoothing effects. Aiming to attain the advantageous regularity properties of Sobolev gradient flow in a faster setting, we present a new, third warp estimation strategy that builds on recently proposed accelerated gradient descent schemes [30], which achieve considerably faster convergence than standard gradient descent.

As a result, our variational solution is able to handle challenging scenarios such as changing topology and fast motion. However, due to the underlying TSDF evolution, it loses track of correspondences, which are needed for tasks such as animation and texture transfer (see Fig. 1). To recover data association, we propose to utilize the lowest-frequency eigenfunctions of the Laplacian matrices of the TSDFs, as they encode information about the inherent deformation patterns that the shapes can undergo. First, we search for implicit correspondences via an *eigencolour* data term that aligns these representations [31]. As it is robust only up to moderate movements, we suggest an explicit alternative, whereby we match signatures of lower-dimensional embeddings of the eigenfunctions [29].

While this strategy for posterior correspondence estimation is antithetical to traditional approaches, which use data association in order to perform the non-rigid warping, we reckon that it is the most suitable way to incorporate correspondence into the TSDF evolution scheme. As it inherently handles topological changes, which occur whenever objects interact, it paves the way towards capture of arbitrary everyday scenes. We demonstrate advantages over state-of-the-art methods through evaluation of the geometric fidelity of our reconstructions and the precision of voxel correspondences.

2 Related Work

This paper tackles the task of reconstructing a dynamic environment using a single RGB-D sensor without any prior knowledge. In the following we discuss related techniques and do not focus on systems that employ specialized multi-camera set-ups [10], [11], [13], [32], hand [33], [34], face [35], skeleton [21] or human body [14], [22] priors, or that require the acquisition of a static template [17]. We refer the reader to the recent comprehensive overview by Zollhöfer *et al.* [36] for an extensive analysis of the properties of such methods.

Template-free methods for non-rigid fusion from a single depth camera have been on the rise since 2015 with the development of the offline bundle adjustment scheme of Dou *et al.* [37] and the first real-time solution for simultaneous surface tracking and reconstruction, DynamicFusion [18]. Several extensions to this semnial work have been proposed, most notably VolumeDeform [19] which combines the used dense depth-based correspondences with sparse SIFT features to reduce drift and handle tangential motions, and the system of Guo *et al.* [20] which increases robustness by integrating surface albedo constraints. Nevertheless, most examples in these works contain relatively constrained motions without changing topology.

Fast motion, surface merging and splitting are inherently handled by the signed distance field representation [38]. It has been applied for segmentation [39], [40] and registration [41], [42] in medical imaging, where organ shape priors are typically available, and for surface manipulation and animation on complete noise-free models in graphics [43], [44], [45], [46]. In computer vision Paragios *et al.* [47] and Fujirawa *et al.* [8] have used level sets for non-rigid registration on 2D image data and have discussed extensions to 3D. The task of fusion from 2.5D data is more challenging since new data has to be incremented in a consistent manner.

The step before fusion requires estimating a dense warp field between a new frame and the existing reconstruction. This is the objective in scene flow [48], [49], [50], [51], [52]. Related are also approaches that segment the scene into static and dynamic components, then reconstruct them separately [53], [54]. Many of these techniques are variational in nature, combining a data term that imposes similarity between the warped observed data and the target model, and a regularizer that imposes motion smoothness to better constrain the solution space. We thus propose to extend the variational level set method [55] to the setting of incremental fusion from a single depth stream.

As the challenge is to incrementally add new observations instead of erroneously registering them to old data, we investigate additional regularizers. Non-rigid motion tends to be not only smooth, but also volume-preserving. Therefore a prior that enforces the field to be solenoidal, *i.e.* divergence-free, would benefit the fusion. Killing vector fields are of this class and generate locally isometric motions [24], [56], [57]. Thus they offer a way to impose a rigidity prior directly through the warp field, rather than resorting to embedded deformation [58] or as-rigid-as-possible schemes [25].

An important remark is that the L^2-type inner product employed for gradient flow in most variants of the variational level set method [38], [55], [59] assumes a metric that may lead to slow convergence and sub-optimal solutions [60]. Instead, gradient flow in the Sobolev space H^1 has been shown to have a superior performance without changing the global optimum thanks to a desirable coarse-to-fine evolution behaviour that is robust to spurious artifacts [60]. We refer the reader to the book of Neuberger [26] for a thorough mathematical introduction to the topic.

A practical issue is that projecting to Sobolev space entails inversion of differential operators [30]. While it can be approximated by a convolution with the respective impulse response [29], [61], real-time processing might be precluded in 3D. Recently, an optimization acceleration technique that achieves equivalent regularity properties without the use of Sobolev norms has been demonstrated [30]. A mass density

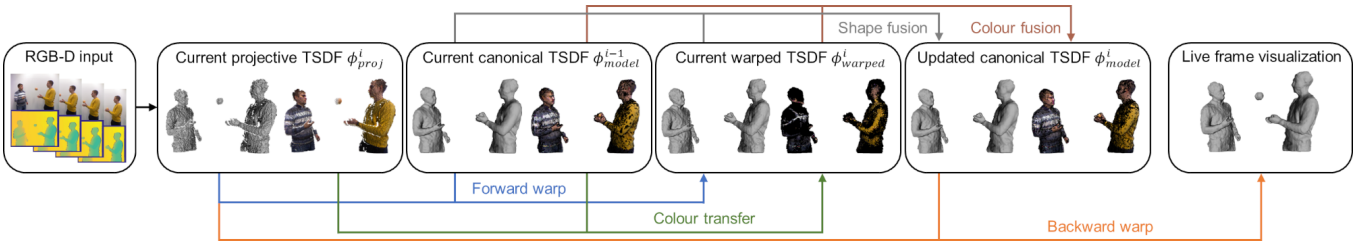


Fig. 2. Non-rigid fusion pipeline. First we generate the projective TSDF ϕ^i_{proj} of an input RGB-D pair from the current camera pose estimate. Then we warp it towards the current canonical model TSDF ϕ^{i-1}_{model} using our variational minimization scheme, obtaining ϕ^i_{warped}. Next, we optionally estimate voxel correspondences between ϕ^i_{proj} and ϕ^i_{warped} in order to transfer colour to the warped TSDF. Afterwards we fuse ϕ^i_{warped} into the canonical model, obtaining its updated state ϕ^i_{model}. Finally, we run a backward warp from ϕ^i_{model} to ϕ^i_{proj} to visualize the live frame to the user.

representing an infinite number of particles is evolved together with the optimization variable. The resulting equations of motion lead to convergence much quicker than standard gradient descent and primal-dual methods [62].

While level sets are beneficial for capturing changing topology, they are unable to track correspondences [63], [64]. Hybrid representations between level sets and meshes have been tried for data association [65], [66], [67], but tend to be slow and numerically unstable [68]. As the graph Laplacian of a shape is invariant to isometric deformations [69], [70], correspondence can be estimated after warping. The approach of Mateus *et al.* [71] matches voxel sets by comparing Laplacian eigenfunction signatures and reducing the problem to rigid alignment in a lower-dimensional embedded space. We modify the technique to handle TSDFs of partial shapes, so that it can be used in non-rigid reconstruction.

3 Preliminaries

Here we introduce our mathematical notation and outline the steps of our variational reconstruction scheme.

3.1 Mathematical Fundamentals

Our system takes an RGB-D stream consisting of pairs (I^i_{RGB}, I^i_D), where i is the frame index, I_{RGB} is the 3-channel colour image and I_D is the aligned depth map. We assume a calibrated camera and a projection function $\pi\colon \mathbb{R}^3 \mapsto \mathbb{N}^2$ from 3D coordinates to pixels.

Our base representation is the truncated signed distance field (TSDF), which associates each point in space with the signed distance to its closest surface location. Surfaces are located at the zero-valued interface between the negative inside and positive outside, so their mesh representation can be easily extracted via marching cubes [72].

We generate TSDFs in a pre-defined bounding volume, which we discretize into cubic voxels of a selected side length. They are indexed by integer tuples $(x, y, z) \in \mathbb{N}^3$. Let $(X, Y, Z) \in \mathbb{R}^3$ be the coordinates of the respective voxel's center in 3D space. A single RGB-D frame allows the generation of a projective TSDF $\phi\colon \mathbb{N}^3 \mapsto \mathbb{R}$. We follow the traditional scaling and truncation scheme [73], [74]:

$$d(x,y,z) = I_D(\pi(X,Y,Z)) - Z\,, \tag{1}$$

$$\phi(x,y,z) = \begin{cases} sgn(d(x,y,z)) & \text{if } |d(x,y,z)| \geq \delta\,, \\ d(x,y,z)/\delta & \text{otherwise}\,, \end{cases} \tag{2}$$

$$\omega(x,y,z) = \begin{cases} 1 & \text{if } d(x,y,z) > -\eta\,, \\ 0 & \text{otherwise}\,. \end{cases} \tag{3}$$

Here d is the directional signed distance, which is truncated to the interval $[-1, +1]$ to disregard voxels that are far away from the surface. In practice we set the responsible parameter δ to 5-10 times the voxel size, while η, which determines the expected object thickness, is set to 2-3 voxels. Voxels outside the object and within this thickness receive a confidence weight ω of 1, while non-observed ones get 0.

TSDFs from multiple views are fused together via the weighted averaging scheme of Curless and Levoy [23], resulting in a true, non-projective TSDF.

Our goal is to determine a vector warp field $\Psi = (U, V, W)\colon \mathbb{N}^3 \mapsto \mathbb{R}^3$ that aligns a pair of TSDFs and has the same resolution as them. U, V and W denote its x-, y- and z-components respectively, each of which is a scalar grid $\mathbb{N}^3 \mapsto \mathbb{R}$. The field assigns a displacement vector (u, v, w) to each voxel (x, y, z).

3.2 Overview

Our fusion pipeline is displayed in Fig. 2. Given the current state of the cumulative model ϕ^{i-1}_{model} and an incoming RGB-D pair (I^i_{RGB}, I^i_D), we iteratively warp the projective TSDF ϕ^i_{proj} generated from I^i_D towards ϕ^{i-1}_{model} by estimating deformation field increments following one of the approaches described in Section 4, resulting in the warped TSDF ϕ^i_{warped}. We then estimate voxel correspondences between the initial and warped TSDFs in order to transfer colour from ϕ^i_{proj} to ϕ^i_{warped}, as explained in Section 5. Then we fuse ϕ^i_{warped} into the global model, obtaining its updated state ϕ^i_{model}. Finally, we run a backward deformation from ϕ^i_{model} towards ϕ^i_{proj} to provide the user a live geometry visualization without colour.

We assume that both the scene and the camera are moving. Therefore we estimate a rigid camera transformation using another purely TSDF-based approach [75] which registers pairs of voxel grids by direct minimization. We prefer this formulation over ICP variants [76], [77], since they would need a very robust norm to discard the many outliers that result from large deformations.

4 Non-rigid 3D Reconstruction

In this section we describe our variational models for non-rigid 3D reconstruction from a single RGB-D stream.

4.1 Signed Distance Field Evolution Energy

As a new RGB-D frame is acquired and we estimate the approximate camera pose, we generate its projective TSDF

ϕ_{proj}. Next, we iteratively warp it towards the canonical TSDF ϕ_{model}. In iteration t, we calculate a deformation field increment $\Psi = (U, V, W)$ and apply it to the current warped TSDF $\phi_{proj}^{(t)}$, obtaining its new state $\phi_{proj}^{(t+1)}$ via tri-linear interpolation. We do this following a variational formulation consisting of a data term and a combination of regularizers:

$$E_{def}(\Psi) = E_{data}(\Psi) + w_{reg} E_{reg}(\Psi)\,, \tag{4}$$

where $w_{reg} > 0$ controls the trade-off between data fidelity and regularity. A solution of this model can be found via a gradient descent scheme with step size $\alpha > 0$:

$$\Psi^{(t+1)} = \Psi^{(t)} - \alpha\, \nabla E_{def}\left(\Psi^{(t)}\right)\,, \tag{5}$$

where $\nabla E_{def}\left(\Psi^{(t)}\right)$ denotes the variational derivative of the energy with respect to the deformation field. As will be explained in Section 4.2, ∇E_{def} depends on the choice of the underlying inner product.

4.1.1 Data term

Our data term is driven by the intuition that under perfect alignment, the warped and the target TSDFs will have identical signed distance values in each overlapping voxel. Therefore the value at each voxel (x, y, z) of the current frame ϕ_{proj}, displaced by its flow vector (u, v, w), will be equal to the value in that voxel in ϕ_{model}. Thus to obtain the warp, we minimize the sum of direct squared voxel-wise differences:

$$E_{data}(\Psi) = \frac{1}{2} \sum_{x,y,z} \big(\phi_{proj}(x+u, y+v, z+w) - \\ -\phi_{model}(x, y, z)\big)^2\,. \tag{6}$$

We obtain the derivative by standard calculus of variations:

$$\nabla E_{data}(\Psi) = \big(\phi_{proj}(\Psi) - \phi_{model}\big)\, \nabla \phi_{proj}(\Psi)\,. \tag{7}$$

Note that we use the symbol ∇ both for the spatial gradient of ϕ and for the variational derivatives of the energy terms.

4.1.2 Regularization

Commonly, non-rigid registration methods impose regularity constraints in order to introduce additional information, thereby reducing the solution space of the problem [36]. In our setting regularity can be enforced through the warp field itself, as well as over the TSDFs. We propose several alternatives in this section and analyze how to best combine them for efficient deformable reconstruction in Section 4.3.

Uniform motion The expected input to our system is noisy Kinect data, which might cause inconsistencies within voxel neighbourhoods that result in holes in the reconstruction. A classical Tikhonov-type regularizer can be used to reduce spurious artifacts and impose motion smoothness, as often done in scene and optical flow [48], [52], [78]:

$$E_{smooth}(\Psi) = \frac{1}{2} \sum_{x,y,z} \big(|\nabla U(x,y,z)|^2 + \\ +|\nabla V(x,y,z)|^2 + |\nabla W(x,y,z)|^2\big)\,. \tag{8}$$

Using calculus of variations we obtain:

$$\nabla E_{smooth}(\Psi) = -(\Delta U, \Delta V, \Delta W)^\top\,, \tag{9}$$

where ΔU denotes the Laplace operator applied to the x-component of the flow field, and similarly for V and W.

Divergence-free flow Another strategy is to prevent uncontrollable deformations via rigidity constraints. Most common are the as-rigid-as-possible [25] and embedded deformation [58] formulations, which ensure that the vertices of a latent control graph move in an approximately rigid manner. Here we propose an alternative, whereby local rigidity is imposed directly through the deformation field.

A 3D flow field that generates locally isometric motions is called a *Killing vector field* [24], [56], [57], named after the German mathematician Wilhelm Killing. It is divergence-free, *i.e.*volume-preserving, and satisfies the *Killing condition* $J_\Psi + J_\Psi^\top = \mathbf{0}$, where J_Ψ is the Jacobian of the field. However, it does not regularize angular motion.

A field which generates only nearly isometric motion and thus balances both volume and angular distortion is an *approximately Killing vector field (AKVF)* [24]. It minimizes the Frobenius norm of the Killing condition:

$$E_{akvf}(\Psi) = \frac{1}{2} \sum_{x,y,z} \left\| J_\Psi + J_\Psi^\top \right\|_F^2\,. \tag{10}$$

Its functional derivative is:

$$\nabla E_{akvf}(\Psi) = -2(\Delta U, \Delta V, \Delta W)^\top - \\ -2\left(\frac{\partial(\mathrm{div}\Psi)}{\partial x}, \frac{\partial(\mathrm{div}\Psi)}{\partial y}, \frac{\partial(\mathrm{div}\Psi)}{\partial z}\right)^\top\,, \tag{11}$$

where $\mathrm{div}\Psi = U_x + V_y + W_z$ is the divergence of the warp field. We refer the reader to the supplementary material for complete derivations of all equations in this section.

However, this constraint might be too strict for surfaces undergoing large deformations. Thus we propose to damp the Killing condition. First, we rewrite Eq. (10) using the column-wise stacking operator $vec(\cdot)$ as follows:

$$E_{akvf}(\Psi) = \frac{1}{2} \sum_{x,y,z} vec(J_\Psi + J_\Psi^\top)^\top vec(J_\Psi + J_\Psi^\top) = \\ = \sum_{x,y,z} vec(J_\Psi)^\top vec(J_\Psi) + vec(J_\Psi^\top)^\top vec(J_\Psi)\,. \tag{12}$$

Next, we notice that the first term can be written as:

$$vec(J_\Psi)^\top vec(J_\Psi) = |\nabla U|^2 + |\nabla V|^2 + |\nabla W|^2 = \\ = 2E_{smooth}(\Psi)\,. \tag{13}$$

Therefore we devise our *damped Killing regularizer* as a damped-down AKVF condition, in which more weight is given to the motion smoothness component:

$$E_{Killing}(\Psi) = \\ = \sum_{x,y,z} \big(vec(J_\Psi)^\top vec(J_\Psi) + \gamma vec(J_\Psi^\top)^\top vec(J_\Psi)\big)\,. \tag{14}$$

The parameter γ controls the trade-off between Killing property and motion uniformity. A value of $\gamma = 1$ corresponds to the AKVF condition from Eq. 10. The derivative is:

$$\nabla E_{Killing}(\Psi) = -2(\Delta U, \Delta V, \Delta W)^\top - \\ -2\gamma\left(\frac{\partial(\mathrm{div}\Psi)}{\partial x}, \frac{\partial(\mathrm{div}\Psi)}{\partial y}, \frac{\partial(\mathrm{div}\Psi)}{\partial z}\right)^\top\,. \tag{15}$$

Level set property One of the characteristic properties of a signed distance field is that its gradient magnitude equals

unity everywhere where it is differentiable [38]. To ensure geometric correctness during the evolution of ϕ_{proj} towards ϕ_{model}, this property has to be conserved [79]:

$$\underset{set}{E_{level}}(\Psi) = \frac{1}{2} \sum_{x,y,z} \left(|\nabla \phi_{proj}(x+u, y+v, z+w)| - 1 \right)^2 . \quad (16)$$

Again, applying the calculus of variations we obtain:

$$\nabla \underset{set}{E_{level}}(\Psi) = \frac{|\nabla \phi_{proj}(\Psi)| - 1}{|\nabla \phi_{proj}(\Psi)|_\epsilon} \, H_{\phi_{proj}(\Psi)} \, \nabla \phi_{proj}(\Psi) \, , \quad (17)$$

where $H_{\phi_{proj}(\Psi)} \in \mathbb{R}^{3\times3}$ is the currrent TSDF's Hessian matrix, composed of second-order partial derivatives. To avoid division by zero we use the expression $|\cdot|_\epsilon$, which equals the norm plus a small constant $\epsilon = 10^{-5}$.

This term is not only suitable for imposing regularity over the warped TSDF, but also for reducing noise in it, since spurious artifacts will get smoothed out when this constraint is applied. However, it does not hold strictly on a discretized signed distance field with a numerically approximated gradient [38], and is not valid at the border of voxel truncation, so it may lead to over-smoothing effects. To overcome these issues, we instead consider preconditioning the gradient flow, as explained next.

4.2 Sobolev Gradient Flow

The concept of Sobolev gradient flow was developed several decades ago in the context of the numerical solutions of partial differential equations (PDEs). The main idea is to compute the variational derivative of an energy with respect to the inner product of a smooth subspace of L^2, *i.e.* a Sobolev space, in order to obtain a gradient, which employed in a descent scheme yields a gradient flow that favours globally consistent solutions and is less susceptible to undesired local minima. To describe this effect Sundaramoorthi *et al.* [27] coined the term *coarse-to-fine evolution*, which accurately captures the fact that coarse-scale changes are favoured over fine-scale ones. In the context of incremental 3D reconstruction, this means that the warped TSDF will first adapt to more global deformations before eventually converging also with respect to fine-scale details.

To compute a Sobolev gradient, it is sufficient to project the original gradient ∇E_{def} to the Sobolev space H^1 [80]. As done in traditional descent schemes, let us define ∇E_{def} from Eq. (5) as the L^2 gradient $\nabla_{L^2} E_{def}$. Thus we obtain:

$$\nabla_{H^1} E_{def} = (Id - \lambda\Delta)^{-1} \, \nabla_{L^2} E_{def} \, , \quad (18)$$

where Id denotes the identity operator. Eq. (18) involves the solution of an equation system, but it is possible to derive an approximate way of obtaining Sobolev gradients. First we note that Eq. (18) can be realized via

$$\nabla_{H^1} E_{def} = S * \nabla_{L^2} E_{def} \, , \quad (19)$$

where the filter S is the impulse response of the operator $(Id - \lambda\Delta)^{-1}$. In practice, we approximate S for chosen λ and filter size s by solving the following system:

$$(Id - \lambda\Delta) S = v \, , \quad (20)$$

where v is a one-hot vector that corresponds to a discretized Dirac impulse of size $s \times s \times s$ voxels, and Δ is the Laplacian matrix discretized via a s-point finite-difference stencil.

However, 3D convolutions might become prohibitively expensive for large values of s. Thus we further approximate the Sobolev kernel S by three separable 1D convolutions. To do so, we calculate the tensor higher-order SVD decomposition [81] of S and retain only the first singular vector from each resulting U matrix, and after normalization to unit sum obtain the 1D s-element filters S_x, S_y and S_z. Note that as their entries are identical, the subscript is used to denote the spatial direction of application. This is an approximation of S with crucial performance advantages.

4.3 Combined Energy Formulations

While any of the energy terms discussed in Section 4.1.2 can be combined into E_{reg} with appropriate balancing weights, and the proposed Sobolev filters can be additionally applied to regularize any energy, each of these components entails an increase in runtime. As we aim for applications at interactive rates, we favour two of the possible combinations.

If we are to use Sobolev gradient flow, a regularizer that imposes smooth motion is sufficient, since the gradient descent will follow a coarse-to-fine evolution that will first recover global motion and then add details [29]:

$$\nabla E_{defSobolev} = \nabla_{H^1} (E_{data} + w_{smooth} E_{smooth}) \, . \quad (21)$$

As the Sobolev gradient flow enforces globally consistent motion without changing the global optimum [60], we do not need to impose additional rigidity constraints or carry out level set re-initialization [79], [82].

However, if the kernel size s is too large, the execution time starts to lag behind near-real-time rates. Therefore we propose another alternative, without Sobolev regularization, which allows for incorporation of more priors into the energy formulation. Due to the lack of pre-conditioning, we need to impose rigidity constraints and ensure that the level set property is conserved throughout the evolution [28]:

$$\nabla E_{defKilling} = \nabla_{L^2} (E_{data} + w_k E_{Killing} + w_{ls} \underset{set}{E_{level}}) \, . \quad (22)$$

As our experiments will demonstrate, the two strategies lead to similar results. While $E_{defKilling}$ is slightly faster, $E_{defSobolev}$ does not suffer from over-smoothing effects and may yield reconstructions with better geometric details.

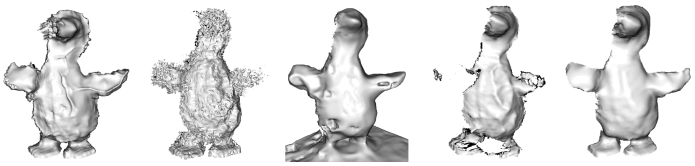
4.4 Accelerated Optimization

The above-described concerns make it desirable to combine the regularization properties of Sobolev optimization with a fast numerical scheme. Speeding up high-dimensional gradient descent problems has been an important topic in machine learning lately, due to the widespread use of convolutional neural networks [83], [84]. The so-called *accelerated gradient descent* methods avoid the local minima that impede optimization by averaging past descent directions. Well-known examples, also referred to as momentum descent, are Polyak's heavy ball method [85] and Nesterov accelerated gradient descent [86]. In particular, Nesterov proved optimal convergence of his scheme among first-order methods.

It has recently been shown [84] that all variants of Nesterov's method are essentially discretizations of the ODE equations of motion for a particular Lagrangian action functional, and can thus be formulated with variational

(a) $s = 3$ (b) $w_{smooth} = 0$ (c) default

Fig. 3. Parameter analysis for $E_{defSobolev}$: (a) a small neigbourhood s is not able to fully overcome the effects of noise; (b) no motion regularization results in inconsistent geometry; (c) the default setting $s = 7$, $w_{smooth} = 0.2$, $\lambda = 0.1$ yields a detailed reconstruction.



(a) $w_{ls} = 0$ (b) $w_k = 0$ (c) $\gamma = 0$ (d) $\gamma = 1$ (e) default

Fig. 4. Parameter analysis for $E_{defKilling}$: (a) no level set property preservation; (b) no motion regularization; (c) conventional motion smoothness without a Killing component; (d) pure AKVF condition; (e) default setting $w_{ls} = 0.2$, $w_k = 0.5$, $\gamma = 0.1$.

principles [30], [84], [87]. Sundaramoorthi and Yezzi have extended this to PDEs, demonstrating superior convergence properties without the use of Sobolev spatial convolution in tasks such as active contours and optical flow [30], [88], [89].

Here we devise the acceleration scheme for non-rigid reconstruction in 3D. Given our deformation energy $E_{def}(\Psi)$, whose optimization we want to speed up, we define an action integral $J_{def}(\Psi)$ that consists of $E_{def}(\Psi)$ as potential energy, in addition to a kinetic energy term $K_{def}(\Psi)$:

$$K_{def}(\Psi) = \frac{1}{2} \sum_{x,y,z} \left(\rho(\Psi, \nabla\Psi)\, \Psi_t^2 \right), \tag{23}$$

$$J_{def}(\Psi) = \int k(t) \left(K_{def}(\Psi) - b(t) E_{def}(\Psi) \right) dt\,. \tag{24}$$

Above $\rho(\Psi, \nabla\Psi)$ is the mass density, $k(t)$ and $b(t)$ are time-dependent weights, and the t-subscript denotes the time derivative. In particular, $k(t)$ ensures dissipation of energy.

The accelerated descent equation is the Euler-Lagrange equation for $J_{def}(\Psi)$. Setting the initial density to a constant $\rho_0 \in \mathbb{R}$ throughout the volume and defining $a(t) = k'(t)/k(t)$, we obtain the following equation of motion [88]:

$$\Psi_{tt} + a(t)\Psi_t = -\frac{b(t)}{\rho_0} \nabla E_{def}(\Psi)\,. \tag{25}$$

While any choice of E_{def} can be optimized via Eq. (25), our goal here is to achieve the robustness properties of Sobolev gradient flow, so we use the same, simpler energy formulation as SobolevFusion rather than KillingFusion:

$$E_{accelerated}(\Psi) = E_{data}(\Psi) + w_{smooth} E_{smooth}(\Psi)\,. \tag{26}$$

4.5 Parameter Analysis

We use the *Andrew-Chair* full-loop sequence from Dou *et al.* [37] in order to determine the most advantageous parameters in case of using Sobolev pre-conditioning with $E_{defSobolev}$, shown in Fig. 3. Our model is robust with regard to the parameter choice and achieves good results with a variety of settings, of which we recommend neighbourhood size $s = 7$, filter parameter $\lambda = 0.1$ and motion smoothness $w_{smooth} = 0.2$ as default.

A Sobolev filter size $s = 3$ is not sufficient to achieve satisfactory results. While a larger kernel would impede the speed, the differences with $s \geq 7$ become negligible.

The parameter λ has an effect on the convergence rate. We estimated empirically that doubling its value reduces the number of iterations by 3-8%. Moreover, motion regularity is essential to overcome noise. The ranges $\lambda \in [0.05; 0.4]$ and $w_{smooth} \in [0.1; 0.5]$ yield high fidelity reconstructions, so we set the default values as the midpoints of those intervals.

For the case without Sobolev regularization, we use the fast-motion *Duck* sequence from the *Deformable 3D Reconstruction Dataset* of KillingFusion [28], since the effect of the damped Killing regularizer is better observable under large motion. As shown in Fig. 4 without level set property preservation the model is not smooth and develops fine-scale artifacts where the property has been violated during the evolution. If all motion regularizers are disabled, the moving parts of the object, such as its wings and head, get destroyed as more frames are fused inconsistently. If only E_{smooth} is used as motion regularization, the reconstruction is somewhat smoother, but holes appear in several regions due to discrepancies. Conversely, if no damping is applied to the AKVF condition, the stronger rigidity prior causes the non-rigidly moving wings to nearly vanish. We empirically determined the parameter ranges that yield geometrically consistent reconstructions to be $w_{ls} \in [0.05; 0.2]$, $w_k \in [0.1; 0.5]$ and $\gamma \in [0.05; 0.25]$.

We use a gradient descent step size $\alpha = 0.1$ in both the standard and accelerated variants.

The accelerated optimizaiton parameters that yield Nesterov's method are $\rho_0 = 1$, $b(t) = 1$, $k(t) = t^3/2$ and thus $a(t) = 3/t$ [88]. There has been no comprehensive study of accelerated descent parameters [88], but a partial investigation that we carried out showed that while this is indeed the most stable value for $a(t)$, decreasing the scalar factor ρ_0 allows the number of iterations until convergence to be further reduced. Plots summarizing this can be found in the supplementary document. The value $\rho_0 = 1/3$ was the closest to the default Nesterov value that always gave stable results in our analysis, so we use it in further experiments.

Eq. (25) can be implemented as a system of first-order evolution equations for velocity, density and warp, but these variables entail auxiliary storage. Our direct implementation of the equation in its second-order form was stable in all considered examples, so we opt for it.

4.6 Implementation Details

One of the main benefits of our correspondence-free variational energy formulation is that it can be applied to each voxel independently, so all displacement vector updated can be computed in parallel. We tested our implementations on a laptop with an Nvidia Quadro K1100M GPU with 2 GB of global memory, and on a desktop PC with an Nvidia Titan Black with 6 GB of memory. Depending on the bounding

(a) reference (b) without E_{eig} (c) with E_{eig}

Fig. 5. Texture transfer from frame #30 to #31 of the *Swing* sequence of the MIT dataset [90]: (a) reference texture; (b) colour propagated with E_{def} from Section 4, showing diffusion around moving parts; (c) colour propagated with E_{def} combined with the *eigencolouring* term E_{eig}.

volume, we used a voxel size in the range 4-12 mm in order to fit the entire regular voxel grid into GPU memory.

On the laptop we achieve 30 frames per second for 64^3 voxels with $E_{defSobolev}$ and for 80^3 voxels with $E_{defKilling}$. On the PC the resolution is approximately doubled, with real-time performance for 128^3 and 150^3 voxels respectively. The runtime with Sobolev regularization can be improved if a smaller kernel size is used, at the risk of certain loss of geometric quality. In particular, a neighbourhood of $s = 5$ achieves similar speed to the L^2-energy formulation. Thanks to the combination of a lightweight energy formulation, no need for spatial convolutions, and at least two times fewer iterations (as shown in the results section), accelerated optimization permits higher resolutions of up to 128^3 to run at 30 fps even on the laptop GPU.

In order to show high resolution results here, human sequences were run on the PC, while smaller toys were tested on the laptop.

5 Looking for Data Association in a Correspondence-free World

Whether regularizing through the deformation field or via Sobolev pre-conditioning, so far we have developed a strategy for reliable 3D reconstruction under non-rigid motion. We may now want to colour or animate the model. However, this is not feasible since level set methods do not preserve correspondences [63], [64]. In particular, if we store an RGB grid containing the colour of each voxel and warp it in the same way as the TSDF, as shown in Fig. 5 (b) colours would diffuse into each other due to displacements to non-integer locations that require trilinear interpolation [64].

Thus we now aim to establish voxel correspondence using the spectrum of the Laplacian matrix of a shape, which is invariant to isometric deformations [91], [92]. Its lower-frequency eigenfunctions, corresponding to its smallest eigenvalues, represent the base shape (*e.g.* a human body) and capture information about its natural non-rigid motion patterns, while the higher-frequency ones account for details (limbs, wrinkles) [69], [70].

5.1 Laplacian Eigencolourings

We first aim to match voxels implicitly in the level set evolution. As the eigenfunction representation results in a colouring of the voxels, which describe the natural deformation modes of the shape, we also call it *eigencolouring*.

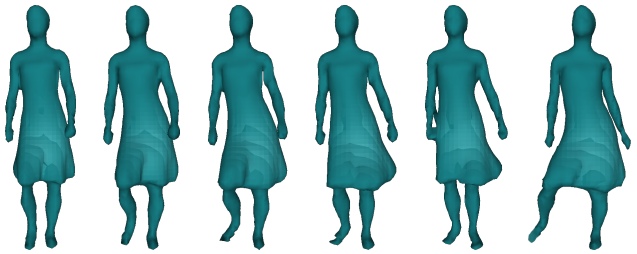


Fig. 6. Lowest-frequency Θ^1-eigencolourings of several poses of the same subject. The contours form similar patterns in all cases and saturate around the skirt folds, which is the most motile region.

To build it we first calculate the normalized graph Laplacian of the respective voxel grid. Let the number of voxels in the narrow band that is not truncated to ± 1 be l - we refer to them as occupied in the current context. This is the main difference to other spectral methods, which typically consider the entire shape. The adjacency matrix W of size $l \times l$ has an entry 1 when adjacent voxels are occupied, and 0 elsewhere. Note that the diagonal entries are 0, as a voxel is not adjacent to itself. The degree matrix D contains the degree of each voxel, *i.e.* the row-wise sums of elements in W, on its diagonal. Then the normalized Laplacian is [71]:

$$L = D^{-\frac{1}{2}}(D - W)D^{-\frac{1}{2}}\,. \tag{27}$$

Next, we calculate its eigendecomposition $L = U\Lambda U^\top$. The full spectrum of the Laplacian (or rather, the Laplace-Beltrami) reflects all possible ways in which the shape can deform isometrically. However, since real-world data contains noise, we discard high-frequency eigenfunctions. Instead, we want to capture only the most significant characteristics of the shape, so we retain only the $K \leq 20$ eigenfunctions with smallest non-zero eigenvalues [71]. Thus we obtain the matrix U^K, which is a lower-dimensional embedding of the shape, whose columns are the K retained eigenvectors, while its l rows are the K-dimensional coordinates of the embedded shape.

As each eigenfunction is an l-element vector, we pad it to the size of the original TSDF and de-linearize its indices, obtaining Θ^e which is the *eigencolouring* of the volume for its e^{th} smallest non-zero eigenvalue. It is a scalar field of the same resolution as the TSDF and if mapped to colour values gives a colour pattern distinctive for the shape, as shown in Fig. 6. We pad with the smallest entry of the eigenfunction so that the gradient is not reversed. Furthermore, we normalize the values to the interval $[-1; 1]$ similar to a TSDF.

Given two TSDFs which we want to align, ϕ_{input} and ϕ_{target}, we expect their K lowest-frequency eigencolourings to be similar, since they stem from the same shape in potentially different poses. However, there is no guarantee that the eigenvalues are reliably ordered in the two embeddings, so we need to determine a $K \times K$ permutation matrix P that aligns the eigenspaces of our two shapes. In addition, due to sign ambiguity, we have to determine a sign matrix M, resulting in an overall transformation $T = MP$.

In case we use only $K = 1$ eigenfunction, it always corresponds to the smallest non-trivial eigenvalue, so there is no ambiguity. For larger K, we determine the transformation T as explained in Section 5.2 and re-order the

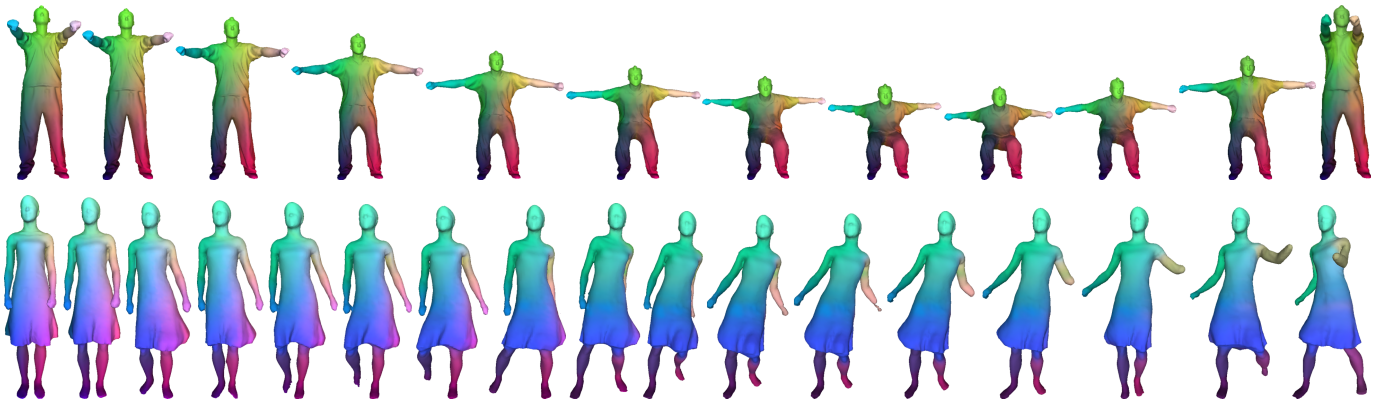


Fig. 7. Texture transfer via implicit correspondence energy on the MIT dataset [90] *Squat* (top) and *Swing* (bottom) sequences. When there is no abrupt motion, E_{eig} is sufficient to preserve a stable texture. However, under larger motion texture diffusion occurs as blue replaces purple on the skirt, and the geometric quality suffers as we cannot recover the arm.

embeddings respectively. Finally, we integrate the **Laplacian eigencolourings term** into our variational formulation:

$$E_{eig}(\Psi) = \frac{1}{2} \sum_{x,y,z} \sum_{t=1}^{K} \left(\Theta_{input}(x+u, y+v, z+w) - \right. \\ \left. - \Theta_{target}(x,y,z)\right)^2 . \quad (28)$$

The complete evolution energy then becomes:

$$E_{def2}(\Psi) = E_{data}(\Psi) + w_{eig} E_{eig}(\Psi) + w_{reg} E_{reg}(\Psi) . \quad (29)$$

As we view the eigencolourings term as another data term, we use $w_{eig} = 1$ in our experiments, but a comprehensive study of the balance between the two data terms is an interesting direction for future work.

Figure 7 shows colour transfer using correspondences estimated implicitly using E_{def2}. It demonstrates that the energy is robust for moderate motion such as a squat, but cannot handle larger deformations such as the turning dancing girl. Next, instead of E_{eig}, we propose explicit voxel matching, to be applied after E_{def} warps the current TSDF.

5.2 Voxel Correspondences

The transformation T discussed in the previous section relates the reduced embeddings of the two shapes as follows:

$$(U^K_{input})^\top = T (U^K_{target})^\top . \quad (30)$$

To calculate it, we seek an optimal assignment between their column eigenvectors $\mathbf{u}^i_{target}$ and $\mathbf{u}^j_{input}$, $i, j \in \{1, ..., K\}$. The approach of Mateus *et al.* [71] suggests to construct histograms from these eigenvectors, since they are invariant to the value ordering and the number of entries l, and to consider them as signatures of the eigenfunctions. We thus build a 200-bin histogram $hist(\cdot)$ from each vector and store the similarity of each eigenvector pair as the ℓ_1 histogram difference in a score matrix A:

$$A_{i,j} = \min(||hist(\mathbf{u}^i_{target}) - hist(\pm\mathbf{u}^j_{input})||_1) . \quad (31)$$

Additionally, a matrix M' stores the sign of $\pm\mathbf{u}^j_{input}$ that yielded the lower score.

This is an assignment problem between eigenfunction signatures, which we solve for the lowest cost via the Munkres algorithm [93] over A. We then build the permutation matrix P according to its output, and look up M' for the appropriate sign in M. We thus obtain the sought transformation matrix $T = MP$ and use it to estimate correspondence, since according to Umeyama's theorem, it can be found through alignment of the two Laplacian eigenspaces [94]. The correspondences between the embeddings are transferred to the voxels of the original shapes via nearest neighbour search between the embedded- and voxel-coordinates. If a near-surface voxel is assigned to an off-surface voxel, we discard the match.

After obtaining initial matches, we use the Weiszfeld algorithm [95] to determine the geometric median in a $3 \times 3 \times 3$ neighbourhood in order to retain only the most likely correspondence. This step is crucial as we are dealing with partial TSDFs, whose Laplacian eigenfunctions might carry information about non-overlapping regions.

5.3 Implementation Details

We use the described strategy to transfer colour from an initial projective TSDF to its warped counterpart. In this way we are able to obtain a reliably coloured cumulative model.

As parallelization of the voxel matching procedure is not straightforward, in practice we run it on the CPU while the next frame(s) are being warped on the GPU. Depending on volume size, it takes 58-500 ms per frame on a 2.80 GHz Intel Core i7 CPU. When done, it continues with the latest warped frame, effectively avoiding temporal overhead.

6 Experimental Evaluation

In this section we carry out various qualitative and quantitative tests of the non-rigid reconstruction and voxel correspondence components of the proposed formulations. All three proposed variational deformation schemes, namely using Sobolev gradient flow (SobolevFusion [29]), using damped AKVF constraints (KillingFusion [28]), and the newly devised one using accelerated gradient descent (AcceleratedFusion), have as basis the same energy term E_{data}. Thus it is expected that their results are similar, so here we discuss the reasons causing differences. Please refer to the supplemental video for additional visual comparisons.

6.1 Convergence

For all methods we stop optimizing when the energy update, as measured by the total SSD error divided by the

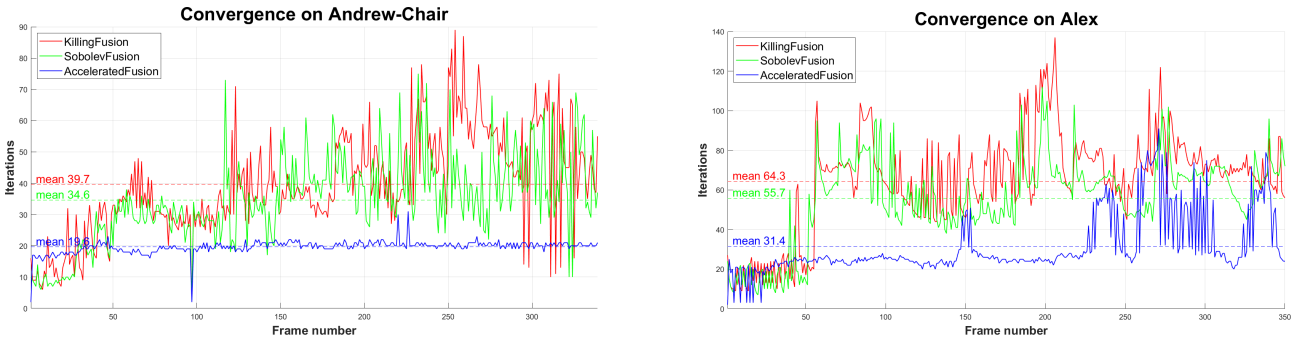


Fig. 8. Comparison of iterations required by our three variational approaches to converge on a slow (left) and fast (right) motion sequence. The average number of iterations per frame for each method is displayed as a dotted line. The number of iterations taken by $E_{defKilling}$ is reduced by 13% with $E_{defSobolev}$ and by more than 50% by $E_{accelerated}$.

Fig. 9. Non-rigid reconstruction from a single depth stream with damped AKVF regularizer ($E_{defKilling}$). We obtain a geometrically consistent model after a 360° loop under topological changes and large motion.

number of voxels, falls below a threshold of 10^{-6}. Theoretically, we expect Sobolev preconditioning to decrease the number of required iterations slightly, while accelerated optimization should decrease them more significantly. Figure 8 displays the iterations taken by the three strategies on the relatively small-motion *Andrew-Chair* sequence of Dou *et al.* [37] and on the large-motion *Alex* sequence from KillingFusion [28], with averages displayed next to the plots. Indeed, $E_{defSobolev}$ decreases the iterations of $E_{defKilling}$ by 12.8% and 13.3% on *Andrew* and *Alex* respectively, while $E_{accelerated}$ decreases them more than twice. As expected, the larger motion sequence requires more iterations on average for any method. It is notable that the iterations with accelerated optimization are rather stable, with prominent increases towards the end of *Alex* when the motion is quick.

6.2 Topological Changes and Fast Motion

A major advantage of our proposed formulation that stays entirely within the TSDF representation is that it can inherently handle topological changes and capture large deformations. Thus we first demonstrate these abilities.

Both Fig. 1 and Fig. 9 show a human turning in a complete 360° loop while undergoing topology changes, such as interacting with a balloon or splitting his hands from the hips. They have been reconstructed with the Sobolev and the AKVF regularization respectively, proving that both variants of our scheme are able to recover a complete 3D model in unconstrained motion.

Likewise, in Fig. 10 we test on the *Umbrella* sequence from VolumeDeform [19]. The damped AKVF scheme consistently over-smooths thin structures such as the tip due to the level set gradient preservation constraint. VolumeDeform occasionally produces artifacts near edges or fuses the strap into the umbrella due to erroneous registration.

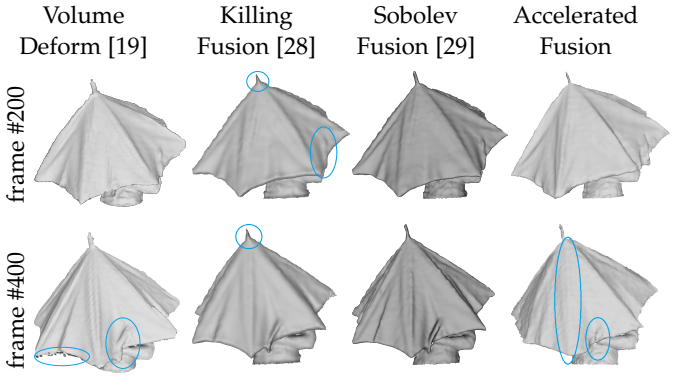


Fig. 10. Warped live frames of the *Umbrella* sequence from VolumeDeform [19]. Damped AKVF, accelerated optimization and VolumeDeform occasionally over-smooth thin elements such as the tip and strap, while Sobolev gradient flow yields similar or higher level of detail as VolumeDeform without artifacts at the edge.

Sobolev gradient flow well captures all geometric details. Accelerated optimization has the least amount of high-frequency noise among the variational methods. However, the second sample frame is over-smoothed due to the fact that accelerated gradient flow may sometimes surpass the energy optimum and then oscillate until reaching stability [30], thereby losing some fine details. This frame is a rare example, but avoiding the effect all-together is subject to further studies of optimal parameters. Likewise, the warped live frames in Fig. 11 show the robustness to fast motion and topology changes of variational level set evolution even in the accelerated scheme, but the ends of the legs in the fourth shown frame are smoothed out. These observations are also reflected in the quantitative evaluation on the *Deformable 3D Reconstruction Dataset* [28], displayed in Fig. 12.

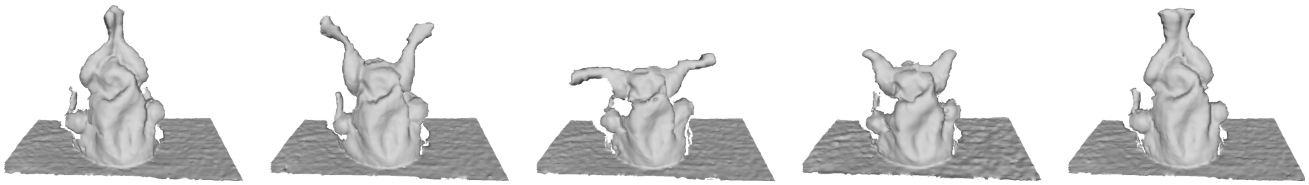


Fig. 11. Reconstruction of fast-moving chicken legs using accelerated optimizaiton ($E_{accelerated}$). Frames are ordered chronologically.

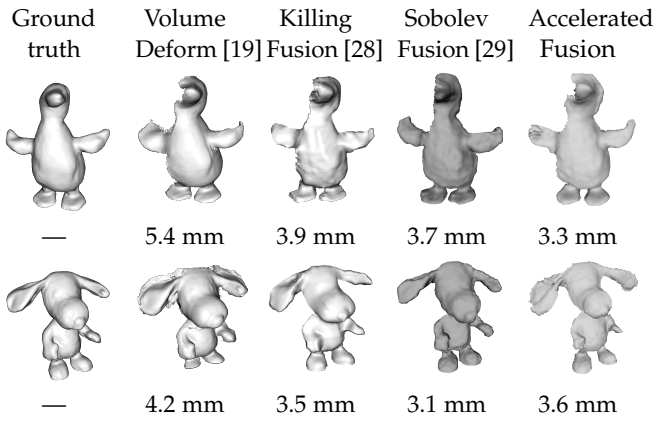


Fig. 12. Geometric error on objects with ground-truth canonical models from the *Deformable 3D Reconstruction Dataset* of KillingFusion [28]. All versions of our variational formulation outperform VolumeDeform [19], as the mechanical toys in the sequences exhibit fast motion. Errors are given below the respective reconstruction.

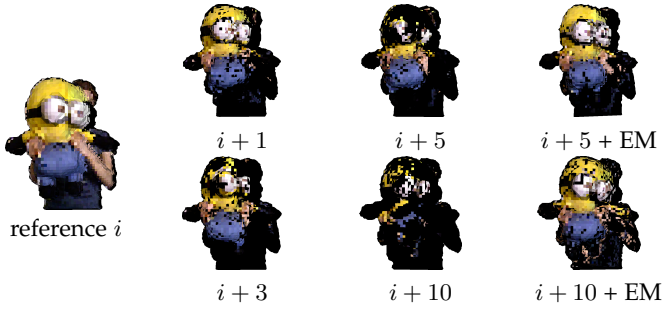


Fig. 13. Colour transfer from reference frame i to target frame $i+n$. With larger distance the amount of transferred colour decreases, but remains correct thanks to our robust voxel correspondence scheme. They can be densified via a posterior EM scheme, as shown on the right.

To conclude, Sobolev gradient flow better captures concavities and defines sharper edges than the damped AKVF strategy, which over-smooths details as the level set term is not valid at the TSDF truncation boundary. But convolving the grid with 7-voxel Sobolev kernels is more computationally demanding. This can be remedied by the accelerated optimization framework, which yields less noisy results without the use of additional regularizers or spatial convolutions, in a shorter time. However, it may over-smooth under larger motion due to its oscillating energy design. Thus selecting the appropriate version of our variational framework depends on the amount of expected motion and on the importance of speed over geometric detail.

6.3 Voxel Correspondences

To evaluate the ability of our system to determine correspondences, we look at texture transfer. If voxel matches are accurately determined, colours will not diffuse into each other over time.

First, we assess the amount of colour that can be transferred depending on the difference in pose. To this end we test on the richly textured *Minion* sequence from VolumeDeform [19]. Fig. 13 shows the results when transferring colour from frame i to the next one, as well as to frames separated by a larger distance. The amount of texture that is being recovered decreases with increasing pose difference, but our scheme manages to determine stable matches even when views are 10 frames apart. Moreover, our match rejection procedure makes sure that only reliable correspondences are returned, and thus there is no transfer of incorrect colours.

Fig. 1 demonstrates results on a full 360° loop sequence. When the RGB values are propagated with the same warp field as the evolving TSDF, the colours on the resulting model diffuse into each other during the interpolation process. In particular, since there is no guarantee that surface voxels remain on the surface during evolution, colours mix not only with their neighbouring ones, but also with the colour-less off-surface voxels, resulting in the observed smoky effect. One possibility to counteract this problem is to propagate colours along the normal direction, but the issue of colour diffusion will still persist.

On the other hand, our voxel matching scheme is able to recover a much clearer texture. Colours on the front are rather crisp, since the difference between the canonical pose and the initial frames is not too large and thus matching is very exact. The back shows more mixed colours, as the poses become more distant and matching becomes more challenging, but the result remains visually pleasing.

Note that our proposed technique is a first solution to combine explicit correspondence information with level set evolution. Thus the main objective has been to reliably colour the reconstructions, rather than to estimate a dense set of correspondences. Nevertheless, we carry out quantitative evaluation on the *yt* sequence with Vicon markers used in BodyFusion [21], which features a human in motion.

We observed that our matching procedure typically returns a low error for markers on the torso of the subject, which is a region where mesh-based correspondences often suffer from sliding. However, since the lower-frequency Laplacian eigenfunctions do not always capture limbs, it is often not possible to find correspondences for markers located on the arms. As 12 out of the 18 Vicon markers are placed on the subject's arms, this dataset is not optimally suited for our method, which on average returns matches for half the markers per frame. Yet, our mean ℓ_1 error of 7.7 cm over the entire sequence is not too far from that of other single-stream methods that do not employ priors, namely 4.4 cm for DynamicFusion [18] and 3.7 cm for VolumeDeform [19]. A reason for the bigger error is

that our method accumulates a higher discretizaiton error, since it always stays in voxel space, while others explicitly determine correspondences for deformation field calculation. Further, Table 1 of BodyFusion [21] allows us to compare the ratios of maximum to average error on the Vicon dataset: 2.0 for BodyFusion, 2.9 for DynamicFusion, 2.4 for VolumeDeform and 2.2 for our approach. This means that for DynamicFusion the maximum error deviates most from the mean, while the error of the skeleton-based BodyFusion stays most uniform throughout the sequence. Our ratio is outperformed only by that of BodyFusion, *i.e.* our algorithm is consistent over all frames and is independent of the amount of motion.

Finally, we devise another quantitative test for voxel correspondences, which allows us to test on locations that are not on limbs. For this purpose we detect SIFT features [96] on well-textured sequences, such as the *Minion* from VolumeDeform [19]. Next, we match them across frames using a very strict outlier rejection policy, so that only very accurate matches are retained. On average we kept 26 SIFT matches per frame pair. Then we carried out our voxel matching scheme as before and compared the 3D locations of the found correspondences to the back-projected SIFT keypoints, obtaining an average ℓ_1 error of 7.2 cm. Since this result is close to that on the Vicon dataset, it confirms the performance of our system. This is a promising result for the incorporation of explicit correspondences into implicit level set frameworks.

7 Limitations and Future Work

Although we achieve interactive frame rates, the resolution, speed and memory consumption of our framework can be improved by replacing the used regular voxel grid TSDF with an appropriate hashing [5] or hierarchical structure [2], or if the warp field is represented at a coarser resolution and interpolated using radial basis functions [97].

Moreover, our voxel matching can be improved, for instance to obtain denser correspondences. One possibility to achieve this is to carry our an expectation-maximization procedure over the spectral matches after the initial estimation that we have proposed, leading to results as those shown on the right of Fig. 13. However, this is currently not feasible in real time [71]. An alternative would be to learn a mapping from sparse to dense flow fields [98], or even to learn correspondences in the spectral embedding [12]. Finally, segmentation can be helpful in the case of multiple objects, so that for each one we can compute a separate, more representative Laplacian matrix.

8 Conclusion

We have presented three variational methods for non-rigid 3D reconstruction of surfaces undergoing free motion, including fast movements, changing topology and interacting subjects. Our framework allows to determine dense deformation flow field updates without correspondence search and to avoid repeated conversion between mesh and TSDF representations. We have proposed several regularization and speed-up alternatives and discussed their advantages and drawbacks. Last but not least, we have devised a voxel correspondence estimation strategy over TSDFs of partial shapes, allowing realistic colouring of the obtained models. We believe that our contribution is a step forward towards making real-time capture of unconstrained motion and 3D avatar creation truly available to the general user.

References

[1] S. Choi, Q.-Y. Zhou, and V. Koltun, "Robust Reconstruction of Indoor Scenes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[2] O. Kähler, V. A. Prisacariu, C. Y. Ren, X. Sun, P. Torr, and D. Murray, "Very High Frame Rate Volumetric Integration of Depth Images on Mobile Devices," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 21, no. 11, pp. 1241–1250, 2015.

[3] C. Kerl, J. Sturm, and D. Cremers, "Dense Visual SLAM for RGB-D Cameras," in *International Conference on Intelligent Robot Systems (IROS)*, 2013.

[4] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-Time Dense Surface Mapping and Tracking," in *10th International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011.

[5] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger, "Real-time 3D Reconstruction at Scale using Voxel Hashing," *ACM Transactions on Graphics (TOG)*, 2013.

[6] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "ElasticFusion: Dense SLAM Without A Pose Graph," in *Robotics: Science and Systems (RSS)*, 2015.

[7] Q. Zhou and V. Miller, S. Koltun, "Elastic Fragments for Dense Scene Reconstruction," in *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[8] K. Fujiwara, K. Nishino, J. Takamatsu, B. Zheng, and K. Ikeuchi, "Locally Rigid Globally Non-rigid Surface Registration," in *IEEE International Conference on Computer Vision (ICCV)*, 2011.

[9] C. Cagniart, E. Boyer, and S. Ilic, "Probabilistic Deformable Surface Tracking from Multiple Videos," in *European Conference on Computer Vision (ECCV)*, 2010.

[10] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, D. Calabrese, H. Hoppe, A. Kirk, and S. Sullivan, "High-Quality Streamable Free-Viewpoint Video," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, 2015.

[11] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor, P. Kohli, V. Tankovich, and S. Izadi, "Fusion4D: Real-time Performance Capture of Challenging Scenes," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, 2016.

[12] M. Dou, P. Davidson, S. R. Fanello, S. Khamis, A. Kowdle, C. Rhemann, V. Tankovich, and S. Izadi, "Motion2Fusion: Real-time Volumetric Performance Capture," in *ACM Transactions on Graphics (TOG)*, 2017.

[13] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic Studio: A Massively Multiview System for Social Motion Capture," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[14] F. Bogo, M. J. Black, M. Loper, and J. Romero, "Detailed Full-Body Reconstructions of Moving People from Monocular RGB-D Sequences," in *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[15] C.-H. Huang, B. Allain, J.-S. Franco, N. Navab, S. Ilic, and E. Boyer, "Volumetric 3D Tracking by Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] H. Li, B. Adams, L. Guibas, and M. Pauly, "Robust Single-View Geometry and Motion Reconstruction," *ACM Transactions on Graphics (TOG)*, vol. 28, no. 5, 2009.

[17] M. Zollhöfer, M. Nießner, S. Izadi, C. Rhemann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, and M. Stamminger, "Real-time Non-rigid Reconstruction using an RGB-D Camera," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, 2014.

[18] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and Tracking of Non-rigid Scenes in Real-Time," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[19] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger, "VolumeDeform: Real-time Volumetric Non-rigid Reconstruction," in *European Conference on Computer Vision (ECCV)*, 2016.

[20] K. Guo, F. Xu, T. Yu, X. Liu, Q. Dai, and Y. Liu, "Real-time Geometry, Albedo and Motion Reconstruction Using a Single RGBD Camera," *ACM Transactions on Graphics (TOG)*, 2017.

[21] T. Yu, K. Guo, F. Xu, Y. Dong, Z. Su, J. Zhao, J. Li, Q. Dai, and Y. Liu, "BodyFusion: Real-time Capture of Human Motion and Surface Geometry Using a Single Depth Camera," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[22] T. Yu, Z. Zheng, K. Guo, J. Zhao, Q. Dai, H. Li, G. Pons-Moll, and Y. Liu, "DoubleFusion: Real-time Capture of Human Performances with Inner Body Shapes from a Single Depth Sensor," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[23] B. Curless and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images," in *23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '96, 1996, pp. 303–312.

[24] J. Solomon, M. Ben-Chen, A. Butscher, and L. Guibas, "As-Killing-As-Possible Vector Fields for Planar Deformation," *Computer Graphics Forum (CGF)*, vol. 30, no. 5, 2011.

[25] O. Sorkine and M. Alexa, "As-Rigid-As-Possible Surface Modeling," in *Fifth Eurographics Symposium on Geometry Processing (SGP)*, 2007.

[26] J. Neuberger, *Sobolev Gradients and Differential Equations*. Springer Science & Business Media, 2009.

[27] G. Sundaramoorthi, A. Yezzi, and A. Mennucci, "Coarse-to-Fine Segmentation and Tracking using Sobolev Active Contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 30, no. 5, pp. 851–864, 2008.

[28] M. Slavcheva, M. Baust, D. Cremers, and S. Ilic, "KillingFusion: Non-rigid 3D Reconstruction without Correspondences," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[29] M. Slavcheva, M. Baust, and S. Ilic, "SobolevFusion: 3D Reconstruction of Scenes Undergoing Free Non-rigid Motion," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[30] G. Sundaramoorthi and A. Yezzi, "Variational PDEs for Acceleration on Manifolds and Application to Diffeomorphisms," in *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[31] M. Slavcheva, M. Baust, and S. Ilic, "Towards Implicit Correspondence in Signed Distance Field Evolution," in *PeopleCap Workshop, IEEE International Conference on Computer Vision (ICCVW)*, 2017.

[32] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H. Seidel, and S. Thrun, "Performance Capture from Sparse Multi-view Video," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, 2008.

[33] D. J. Tan, T. Cashman, J. Taylor, A. Fitzgibbon, D. Tarlow, S. Khamis, S. Izadi, and J. Shotton, "Fits Like a Glove: Rapid and Reliable Hand Shape Personalization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[34] J. Taylor, L. Bordeaux, T. Cashman, B. Corish, C. Keskin, T. Sharp, E. Soto, D. Sweeney, J. Valentin, B. Luff, A. Topalian, E. Wood, S. Khamis, P. Kohli, S. Izadi, R. Banks, A. Fitzgibbon, and J. Shotton, "Efficient and Precise Interactive Hand Tracking Through Joint, Continuous Optimization of Pose and Correspondences," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, 2016.

[35] J. Thies, M. Zollhöfer, M. Nießner, L. Valgaerts, M. Stamminger, and C. Theobalt, "Real-time Expression Transfer for Facial Reenactment," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 6, 2015.

[36] M. Zollhöfer, P. Stotko, A. Görlitz, C. Theobalt, M. Nießner, R. Klein, and A. Kolb, "State of the Art on 3D Reconstruction with RGB-D Cameras," in *Eurographics - State-of-the-Art Reports (STARs)*, vol. 37, no. 2, 2018.

[37] M. Dou, J. Taylor, H. Fuchs, A. Fitzgibbon, and S. Izadi, "3D Scanning Deformable Objects with a Single RGBD Sensor," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[38] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, ser. Applied Mathematical Science. Springer, 2003, vol. 153.

[39] E. Angelini, Y. Jin, and A. Laine, "State of the Art of Level Set Methods in Segmentation and Registration of Medical Imaging Modalities," *Handbook of Biomedical Image Analysis: Registration Models*, vol. III, 2005.

[40] S. Ho, E. Bullitt, and G. Gerig, "Level-Set Evolution with Region Competition: Automatic 3-D Segmentation of Brain Tumors," in *16th International Conference on Pattern Recognition (ICPR)*, 2002.

[41] T. Lee and S. Lai, "3D Non-rigid Registration for MPU Implicit Surfaces," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2008.

[42] J. Maintz and M. Viergever, "A Survey of Medical Image Registration," *Medical Image Analysis*, vol. 2, no. 1, pp. 1–36, 1998.

[43] D. Cohen-Or, A. Solomovic, and D. Levin, "Three-dimensional Distance Field Metamorphosis," *ACM Transactions on Graphics (TOG)*, vol. 17, no. 2, pp. 116–141, 1998.

[44] S. F. Frisken and R. N. Perry, "Designing with Distance Fields," in *ACM SIGGRAPH 2006 Courses*, ser. SIGGRAPH '06, 2006, pp. 60–66.

[45] G. Turk and J. O'Brien, "Shape Transformation Using Variational Implicit Functions," in *26th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '99, 1999.

[46] Y. Weng, M. Chai, W. Xu, Y. Tong, and K. Zhou, "As-Rigid-As-Possible Distance Field Metamorphosis," *Computer Graphics Forum (CGF)*, vol. 32, no. 7, pp. 381–389, 2013.

[47] N. Paragios, M. Rousson, and V. Ramesh, "Non-Rigid Registration Using Distance Functions," *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 142–165, 2003.

[48] F. Huguet and F. Devernay, "A Variational Method for Scene Flow Estimation from Stereo Sequences," in *IEEE International Conference on Computer Vision (ICCV)*, 2007.

[49] J. Quiroga, T. Brox, F. Devernay, and J. Crowley, "Dense Semi-rigid Scene Flow Estimation from RGBD Images," in *European Conference on Computer Vision (ECCV)*, 2014.

[50] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade, "Three-Dimensional Scene Flow," in *IEEE International Conference on Computer Vision (ICCV)*, 1999.

[51] C. Vogel, K. Schindler, and S. Roth, "Piecewise Rigid Scene Flow," in *IEEE International Conference on Computer Vision (ICCV)*, 2013.

[52] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers, "Efficient Dense Scene Flow from Sparse or Dense Stereo Data," in *10th European Conference on Computer Vision (ECCV)*, 2008.

[53] M. Jaimez, C. Kerl, J. Gonzalez-Jimenez, and D. Cremers, "Fast Odometry and Scene Flow from RGB-D Cameras Based on Geometric Clustering," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[54] M. Rünz and L. Agapito, "Co-Fusion: Real-time Segmentation, Tracking and Fusion of Multiple Objects," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[55] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher, "A Variational Level Set Approach to Multiphase Motion," *Journal of Computational Physics*, vol. 127, no. 1, pp. 179–195, 1996.

[56] M. Ben-Chen, A. Butscher, J. Solomon, and L. Guibas, "On Discrete Killing Vector Fields and Patterns on Surfaces," *Computer Graphics Forum (CGF)*, vol. 29, no. 5, 2010.

[57] M. Tao, J. Solomon, and A. Butscher, "Near-Isometric Level Set Tracking," *Computer Graphics Forum (CGF)*, vol. 35, no. 5, 2016.

[58] R. W. Sumner, J. Schmid, and M. Pauly, "Embedded Deformation for Shape Manipulation," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, 2007.

[59] S. Osher and J. Sethian, "Fronts Propagating with Curvature-dependent speed: Algorithms based on Hamilton-Jacobi Formulations," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988.

[60] G. Sundaramoorthi, A. Yezzi, and A. C. Mennucci, "Sobolev Active Contours," *International Journal of Computer Vision (IJCV)*, vol. 73, no. 3, pp. 345–366, 2007.

[61] M. Baust, D. Zikic, and N. Navab, "Variational Level Set Segmentation in Riemannian Sobolev Spaces," in *British Machine Vision Conference (BMVC)*, 2014.

[62] J. Calder and A. Yezzi, "PDE Acceleration: A Convergence Rate Analysis and Applications to Obstacle Problems," *arXiv preprint arXiv:1810.01066*, 2018.

[63] J.-P. Pons, G. Hermosillo, R. Keriven, and O. Faugeras, "How to Deal with Point Correspondences and Tangential Velocities in the Level Set Framework," in *IEEE International Conference on Computer Vision (ICCV)*, 2003.

[64] R. T. Whitaker, "A Level-Set Approach to 3D Reconstruction from Range Data," *International Journal of Computer Vision (IJCV)*, vol. 29, no. 3, pp. 203–231, 1998.

[65] D. Enright, S. Marschner, and R. Fedkiw, "Animation and Rendering of Complex Water Surfaces," *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 736–744, 2002.

[66] B. C. Lucas, M. Kazhdan, and R. H. Taylor, "SpringLS: A Deformable Model Representation to Provide Interoperability between Meshes and Level Sets," in *14th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2011.

[67] V. Mihalef, D. Metaxas, and M. Sussman, "Textured Liquids based on the Marker Level Set," *Computer Graphics Forum (CGF)*, vol. 26, no. 3, pp. 457–466, 2007.
[68] B. C. Lucas, M. Kazhdan, and R. H. Taylor, "Spring Level Sets: A Deformable Model Representation to Provide Interoperability between Meshes and Level Sets," *IEEE Transactions on Visualization and Computer Graphics (VCG)*, vol. 19, no. 5, pp. 852–865, 2013.
[69] B. Levy, "Laplace-Beltrami Eigenfunctions Towards an Algorithm That "Understands" Geometry," in *IEEE International Conference on Shape Modeling and Applications (SMI)*, 2006.
[70] M. Reuter, F.-E. Wolter, and N. Peinecke, "Laplace-Beltrami Spectra As 'Shape-DNA' of Surfaces and Solids," *Computer-Aided Design*, vol. 38, no. 4, pp. 342–366, 2006.
[71] D. Mateus, R. Horaud, D. Knossow, F. Cuzzolin, and E. Boyer, "Articulated Shape Matching using Laplacian Eigenfunctions and Unsupervised Point Registration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
[72] W. E. Lorensen and H. E. Cline, "Marching Cubes: A High Resolution 3D Surface Construction Algorithm," in *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '87, 1987, pp. 163–169.
[73] C. Schroers, H. Zimmer, L. Valgaerts, A. Bruhn, O. Demetz, and J. Weickert, "Anisotropic Range Image Integration," in *Joint German and Austrian Conference on Pattern Recognition (DAGM-OAGM)*, 2012.
[74] C. Zach, T. Pock, and H. Bischof, "A Globally Optimal Algorithm for Robust TV-L^1 Range Image Integration," in *Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV)*, 2007, pp. 1–8.
[75] M. Slavcheva, W. Kehl, N. Navab, and S. Ilic, "SDF-2-SDF: Highly Accurate 3D Object Reconstruction," in *European Conference on Computer Vision (ECCV)*, 2016.
[76] P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 14, no. 2, pp. 239–256, 1992.
[77] S. Rusinkiewicz and M. Levoy, "Efficient Variants of the ICP Algorithm," in *3rd International Conference on 3D Digital Imaging and Modeling (3DIM)*, 2001.
[78] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert, "High Accuracy Optical Flow Estimation Based on a Theory for Warping," in *European Conference on Computer Vision (ECCV)*, 2004.
[79] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level Set Evolution Without Re-initialization: A New Variational Formulation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
[80] J. Calder, A. Mansouri, and A. Yezzi, "Image Sharpening via Sobolev Gradient Flows," *SIAM Journal on Imaging Sciences*, vol. 3, no. 4, pp. 981–1014, 2010.
[81] J. B. Kruskal, "Multiway Data Analysis," R. Coppi and S. Bolasco, Eds., 1989, ch. Rank, Decomposition, and Uniqueness for 3-way and N-way Arrays.
[82] C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance Regularized Level Set Evolution and Its Application to Image Segmentation," *IEEE Transaction on Image Processing (TIP)*, vol. 19, no. 12, pp. 3243–3254, 2010.
[83] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the Importance of Initialization and Momentum in Deep Learning," in *International Conference on Machine Learning (ICML)*, 2013.
[84] A. Wibisono, A. C. Wilson, and M. I. Jordan, "A Variational Perspective on Accelerated Methods in Optimization," *Proceedings of the National Academy of Sciences*, vol. 113, no. 47, pp. E7351–E7358, 2016.
[85] B. T. Polyak, "Some Methods of Speeding up the Convergence of Iteration Methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1–17, 1964.
[86] Y. E. Nesterov, "A Method for Solving the Convex Programming Problem with Convergence Rate $O(1/k^2)$," *Soviet Mathematics Doklady*, vol. 269, pp. 543–547, 1983.
[87] M. Benyamin, J. Calder, G. Sundaramoorthi, and A. Yezzi, "Accelerated PDE's for Efficient Solution of Regularized Inversion Problems," *arXiv preprint arXiv:1810.00410*, 2018.
[88] G. Sundaramoorthi and A. Yezzi, "Accelerated Optimization in the PDE Framework: Formulations for the Manifold of Diffeomorphisms," *arXiv preprint arXiv:1804.02307*, 2018.
[89] A. Yezzi and G. Sundaramoorthi, "Accelerated Optimization in the PDE Framework: Formulations for the Active Contour Case," *arXiv preprint arXiv:1711.09867*, 2017.
[90] D. Vlasic, I. Baran, W. Matusik, and J. Popović, "Articulated Mesh Animation from Multi-view Silhouettes," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, 2008.
[91] M. Belkin and P. Niyogi, "Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering," in *International Conference on Neural Information Processing Systems: Natural and Synthetic (NIPS)*.
[92] V. Jain and H. Zhang, "Robust 3D Shape Correspondence in the Spectral Domain," in *IEEE International Conference on Shape Modeling and Applications (SMI)*, 2006.
[93] A. Frank, "On Kuhn's Hungarian Method - A Tribute from Hungary," Egérvary Research Group on Combinatorial Optimization, Budapest, Hungary, Tech. Rep. 2004-14, 2004.
[94] S. Umeyama, "An Eigendecomposition Approach to Weighted Graph Matching Problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 10, no. 5, pp. 695–703, 1988.
[95] E. Weiszfeld and F. Plastria, "On the Point for Which the Sum of the Distances to n Given Points is Minimum," *Tôhoku Mathematical Journal*, 1937.
[96] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.
[97] X. Xie and M. Mirmehdi, "Radial Basis Function Based Level Set Interpolation and Evolution for Deformable Modelling," *Image and Vision Computing (IVC)*, vol. 29, no. 2-3, pp. 167–177, 2011.
[98] J. Wulff and M. J. Black, "Efficient Sparse-to-Dense Optical Flow Estimation using a Learned Basis and Layers," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Miroslava Slavcheva recently completed her PhD (summa cum laude) at the Technical University of Munich (TUM) and Siemens Corporate Technology. Previously, she obtained her MSc degree in Computational Science and Engineering from TUM in 2015 and a BSc degree in Computer Science from Jacobs University Bremen in 2012. In the meantime she spent research internships at Oculus Research, Realtime Technology and Carl Zeiss. Her research interests are in the area of 3D reconstruction from RGB-D data, where her recent focus on non-rigid fusion has been recognized with two CVPR spotlight presentations and the best paper award at the PeopleCap workshop at ICCV'17, namely the three works upon which this paper is based.

Maximilian Baust studied mathematics for science and engineering at the Technical University of Munich and the Swiss Federal Institute of Technology Zurich from 2003 to 2008. He then joined the research group for computer aided medical procedures and augmented reality at TUM. In his PhD thesis, he investigated variational methods for image segmentation and deformable image registration. In 2012, he joined ESG Elektroniksystem- und Logistik-GmbH, an SME in the area of defense and automotive technology, as a system engineer for camera-based driver assistance systems. From 2013 to 2017, Maximilian was working at TUM as a postdoctoral associate, where he supervised doctoral students and managed research projects in the areas of computer aided medical procedures, computer vision and machine learning. Between 2017 and 2019 he was a senior R&D engineer and research specialist at Konica Minolta Laboratory Europe in the areas of medical image analysis, deep learning and robotics. In 2019 Maximilian joined NVIDIA in Munich as a senior deep learning solution architect industry manager.

Slobodan Ilic is a senior key expert research scientist at Siemens Corporate Technology in Munich, Perlach. He is also a visiting researcher and lecturer at the Computer Science Department of TUM and closely works with the CAMP Chair. From 2009 until the end of 2013 he was leading the Computer Vision Group of CAMP at TUM, and before that he was a senior researcher at Deutsche Telekom Laboratories in Berlin. He obtained his PhD in 2005 from EPFL in Switzerland under the supervision of Pascal Fua. His research interests include: 3D reconstruction, deformable surface modelling and tracking, real-time object detection and tracking, human pose estimation and semantic segmentation.