2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

Piecewise Rigid Scene Flow with Implicit Motion Segmentation

Andreas Görlitz[1], Jonas Geiping[2] and Andreas Kolb[1]

Abstract— **In this paper, we introduce a novel variational approach to estimate the scene flow from RGB-D images. We regularize the ill-conditioned problem of scene flow estimation in a unified framework by enforcing piecewise rigid motion through decomposition into rotational and translational motion parts. Our model crucially regularizes these components by an L_0 "norm", thereby facilitating implicit motion segmentation in a joint energy minimization problem. Yet, we also show that this energy can be efficiently minimized by a proximal primal-dual algorithm. By implementing this approximate L_0 rigid motion regularization, our scene flow estimation approach implicitly segments the observed scene of into regions of nearly constant rigid motion. We evaluate our joint scene flow and segmentation estimation approach on a variety of test scenarios, with and without ground truth data, and demonstrate that we outperform current scene flow techniques.**

I. INTRODUCTION

Motion estimation, be it the motion of a camera or of entire, rigid or deformable objects is of high interest in numerous fields of research, because of its manifold applications - especially in robotics. Object tracking, gesture recognition or any variety of robotic interaction with the dynamical real world necessitate a thorough knowledge of these motions. Yet, although approaches to estimate camera motion and motion of 3D points arise from mutual problems, they are solved using different approaches that have distinct advantages and disadvantages.

Estimation of the camera pose from images is referred to as visual odometry. This is one of the central techniques in robotics for simultaneous localization and mapping (SLAM) [9], allowing online 3D scene reconstruction and navigation of unmanned vehicles in unknown terrains. In the case of scene dynamics, however, the assumption that the scene is static leads to faulty pose estimations, which is a prevailing drawback of state-of-the-art SLAM approaches [33].

In contrast to visual odometry, scene flow describes the inter-frame three-dimensional motion for each single 3D point in the input data. The scene flow problem is inherently ill-posed, and it is thus necessary to incorporate prior knowledge via regularization [32]. Smoothness regularization is commonly applied, yet as all points are handled equally, static and dynamic scene-parts become indistinguishable and, thus, camera motion, i.e. visual odometry, cannot be directly obtained using these algorithms.

*This work was partially funded by the German Research Foundation (DFG) under grants GRK-1564 and KO-2960-13/1.

[1] Computer Graphics and Multimedia Group, University of Siegen, Germany {andreas.goerlitz, andreas.kolb}@uni-siegen.de

[2] Visual Scene Analysis Group, University of Siegen, Germany jonas.geiping@uni-siegen.de

Alternative approaches exploit the spatial coherence of the scene flow by segmenting the range and/or color information and assuming a rigid motion for each of these segments [11], [14], [21]. Golyanik et al. [11] show that such scene flow methods outperform classical point-wise algorithms in terms of accuracy. Furthermore, assuming the largest cluster to be the static scene background, these methods enable visual odometry [14]. The major drawback of incorporating explicit segmentation into scene flow is the dependency between both, the scene flow and the segmentation. That is, per-image segmentation is lacking in temporal coherence and, thus cluster correspondences are difficult to establish. Furthermore, irregular or coarse segmentation results in false scene flow estimations while too fine a segmentation yields unstable and thus globally incoherent scene flow estimations.

In this paper we propose a novel scene flow estimation algorithm that *jointly* estimates the scene flow and enforces an implicit motion segmentation. This novel contribution facilitates two important improvements. Firstly, failure cases of previous methods, where the segmentation does not match the underlying scene flow, are eliminated. Segmentation and scene flow estimation are now considered not as separate instances, but as directly interdependent. Secondly, scene flow clustering is incorporated into the model with negligible cost, since our method computes the clusters implicitly by L_0 regularization of the rigid-motion estimates. The L_0 "norm", defined as the amount of non-zero components of a given vector, can be understood as a total measure of sparsity of a vector. The use of this regularization leads to solutions containing unbiased, piecewise-constant rigid-motion components. We use a primal dual hybrid gradient (PDHG) algorithm, Chambolle and Pock [7], and the Moreau decomposition as discussed in [26] to construct an efficient implementation that lends itself especially well to parallelization. This strategy essentially absorbs the unsupervised segmentation capabilities of the real-time Mumford-Shah approach [26] into a unified scene flow estimation.

We evaluate our approach on the Bonn multi-body dataset [27], the TUM dataset [28] and the dataset provided by Jaimez et al. [14], all comprising real-world RGB-D image sequences. We evaluate photometric and geometric errors on these realistic datasets. Furthermore, we compare the endpoint-errors of static scenes directly, using reference scene flow computed from ground truth camera poses. The results reveal a significant improvement of scene flow accuracy over existing approaches such as [14], [15], [18].

978-1-7281-4004-9/19/$31.00 ©2019 IEEE

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

II. RELATED WORK

The estimation of the three-dimensional motion field, called scene flow, has its roots in the pioneering work of Vedula et al. [30]. Their work and ensuing publications had their focus on multi-view camera systems. However, this was changed by the emergence of affordable RGB-D cameras, able to measure both color and depth information simultaneously. Subsequently we will focus on approaches that work on color and depth information and refer to the survey of Yan and Xiang [32] for other input types.

One of the first RGB-D scene flow estimations based on variational approaches, i.e. formulated as energy minimizations was proposed by Herbst et al. [12]. Their work shows how techniques from optical flow estimation such as [5] can be transferred into the scene flow setting. They build a variational model consisting of linearized color consistency, depth consistency and a regularization term that penalizes the (smoothed) total variation of related points.

Quiroga et al. [19] apply a combined local and global constraint in their variational scene flow estimation, where a set of 3D correspondences is used to deal with large displacements. They likewise use total variation (TV) to preserve motion discontinuities.

Jaimez et al. [15] apply a variational formulation including intensity and geometric consistency to RGB-D camera data. The main idea is to regularize the flow field with respect to the 3D geometry observed in the depth image instead of on the image plane using total variation, in order to achieve geometrically consistent results. Quiroga et al. [18] propose a scene flow reconstruction using local rigidity by modeling the 3D motion as a field of twists that consists of the three Euler rotation angles and a translation. This approach is difficult to steer, since translational and rotational motion can hardly be distinguished on a local level.

As scene flow estimation is an ill-posed problem, algorithms using a segmentation of the image plane have been developed in order to stabilize the estimation and further regularizing the per-segment flow.

Several approaches apply a *single, separate segmentation step*, which is especially helpful for online applications such as scene reconstruction [33]. Ghuffar et al. [10] estimate the scene flow in a two-stage process using local scene flow estimation that integrates range flow and optical flow constraints into a global solution. They, furthermore, regularize the sum of differences of neighboring 3D velocities, which can be understood as an approximate total variation, and perform a graph-based motion segmentation in order to group trajectories based on depth and motion similarity.

Rünz and Agapito [21] propose a method that allows for the independent reconstruction of multiple rigidly moving objects. In their scene reconstruction approach, the current set of models is tracked for each new frame. The motion segmentation is formulated as a labeling problem using a fully connected Conditional Random Field, where the image is initially segmented into *simple linear iterative clustering (SLIC)* super-pixels [1], using the geometric *iterative closest point (ICP)* cost [3], [8] to assign a pixel to the rigid motion models as unary potentials.

Instead of separating segmentation and scene flow entirely, alternative approaches *alternate between scene segmentation and scene flow estimation*. Golyanik et al. [11] describe a multi-frame scene flow approach that models scene flow as a global non-linear least squares problem using an ICP-like formulation, whereas Jaimez et al. [14] perform a segmentation of the scene and classify the resulting motion clusters as static or moving elements, which allows for the separate estimation of the camera motion and the individual motions clusters observed in the scene. Initially, the scene is segmented using a K-means clustering on the depth channel, while the following optimization alternates between motion field estimation and smooth rigid segmentation.

Furthermore, a third class of methods model segmentation and motion estimation as a *joint variational approach*. Sun et al. [29] estimate scene flow on the level of a finite number of depth layers extracted from the depth measurements. However, this yields a very strong constraint for the estimation of scene segmentation and flow. The initial segmentation is given by a K-means clustering of the depth values and the approach assumes a rigid motion for each depth layer. Innmann et al. [13] instead propose a method that reconstructs dynamic geometric shapes and operates on a fine volumetric grid that defines a piecewise rigid motion field. In this approach, geometry is implicitly modeled using truncated signed distance functions (TSDFs). In order to tackle the highly under-constrained non-rigid motion estimation, an as-rigid-as-possible (ARAP) [24] regularization prior

$$E_{\mathrm{ARAP}} = \sum_{x} \sum_{y \in \mathcal{N}(x)} \left\| (x_{i+1} - y_{i+1}) - R_i (x_i - y_i) \right\|_2^2$$

is applied on a regular volumetric grid, which is derived from the TSDF. The indices i and $i+1$ refer to two consecutive RGB-D image frames, $\mathcal{N}(x)$ is the neighborhood of scene point x and $R_i \in SO(3)$ are rotation matrices. Similarly, Slavcheva et al. [22] address dynamic scenes reconstruction, using an implicit geometry-driven approach based on TSDFs. They utilize level set evolution without explicit correspondence search and estimate a dense, locally nearly isometric motion field using an approximate killing vector field (AKVF) [23] as regularization prior

$$E_{\mathrm{AKVF}} = \sum_{v} \left\| J_\Psi(v) + J_\Psi(v)^T \right\|_F^2,$$

where J_Ψ is Jacobian of the required flow field in the voxels v close to the object's surface.

In contrast to the existing scene flow estimation approaches discussed so far, our technique formulates rigid motion by regularizing the translational part t and the rotational part α of the rigid motion. The implicit motion segmentation via the L_0 formulation leads to a considerable stabilization of the flow estimation that cannot be reached by the previous techniques that considered the direct total variation, i.e. L_1 regularization of the scene flow gradient.

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

III. PROPOSED METHOD

A. Preliminaries

The input to our method is a sequence of RGB-D frames consisting of color (or grayscale) images $\mathcal{I}_i$ and depth maps $\mathcal{D}_i$. The corresponding color image and depth map are assumed to be temporally synchronized as well as spatially registered. Furthermore, we assume temporally dense RGB-D streams.

We define the projection operator $\pi : \mathbb{R}^3 \to \mathbb{R}^2$:

$$\pi((x,y,z)^\top) = \left(f_x \frac{x}{z} + c_x, f_y \frac{y}{z} + c_y \right)^\top \tag{1}$$

and the inverse projection operator $\pi^{-1} : \mathbb{R}^2 \times \mathbb{R} \to \mathbb{R}^3$:

$$\pi^{-1}((p_x,p_y)^\top,z) = \left(z\frac{p_x - c_x}{f_x}, z\frac{p_y - c_y}{f_y}, z \right)^\top , \tag{2}$$

where $f_x, f_y, c_x, c_y \in \mathbb{R}$ are the intrinsic camera parameters. These operators formalize the conversion between *real world* points in 3D space, denoted by $(x,y,z)^\top \in \mathbb{R}^3$, and their projection onto the 2D camera plane, $p = (p_x, p_y) \in \Omega \subset \mathbb{N}^2$.

Given the intrinsic camera parameters, the inverse projection operator and the depth maps $\mathcal{D}_i$, we compute the corresponding vertex maps $\mathcal{V}_i(p) = \pi^{-1}(p^\top, \mathcal{D}_i(p))$.

Subsequently we will make use of matrix norms $|\cdot|_{p,q}: \mathbb{R}^d \times \Omega \to \mathbb{R}^+$, which are applied for instance to gradients of the scene flow, $\nabla s \in \mathbb{R}^d \times \Omega$. For a matrix $A \in \mathbb{R}^{m \times n}$ the norm $||A||_{p,q}$ denotes the q norm of the p-norms on the elements of matrix A, i.e. $||A||_{p,q} = \left\| (\|A_{:1}\|_p, \ldots, \|A_{:n}\|_p) \right\|_q$.

B. Scene Flow

The goal of scene flow estimation is to compute the three-dimensional motions of three-dimensional points. To be more precise, we want to find the scene flow $s(p)$ that maps the three-dimensional point $\mathcal{V}_i(p)$ to its corresponding point in the subsequent frame $\mathcal{V}_{i+1}(p')$, i.e.

$$\mathcal{V}_i(p) + s(p) = \mathcal{V}_{i+1}(p') \, , \, p, p' \in \Omega \, , \tag{3}$$

with $p' = \pi(\mathcal{V}_i(p) + s(p))$. Pixel p' is thus the corresponding location in $\mathcal{V}_{i+1}$ of pixel p in $\mathcal{V}_i$, which means that the scene flow needs to be a consistent mapping between the successive vertex maps $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$. This equivalence is the central scene flow assumption and can also be understood as a form of *depth constancy* as discussed in [12], due to the fact that the vertex maps are directly computed from the depth measurements.

C. Rigid Scene Flow

We assume that per-pixel motions in three-dimensional space can be represented as per-pixel rigid transformations, i.e. rotations $R(p) \in SO(3)$ and translations $t(p) \in \mathbb{R}^3$. This transformation field is referred to as *rigid scene flow*. Similar to Eq. (3) we can derive

$$R(p)\mathcal{V}_i(p) + t(p) = \mathcal{V}_{i+1}(p') \quad p, p' \in \Omega \ , \tag{4}$$

with $p' = \pi\left(R(p)\mathcal{V}_i(p) + t(p)\right)$.

The relation between *scene flow* and *rigid scene flow* can be observed by equating the left-hand sides of Eqs. (3) and (4), which leads to

$$s(p) = \left(R(p) - I_{3\times3}\right)\mathcal{V}_i(p) + t(p) \ , \tag{5}$$

where $I_{3\times3}$ is the 3×3 identity matrix. We can analyze the variational regularization of the scene flow by considering the absolute deviation of two neighboring pixels $p, \bar{p} \in \Omega$ with equal rigid transformations, $R(p) = R(\bar{p})$ and $t(p) = t(\bar{p})$:

$$\|s(p) - s(\bar{p})\| = \|\left(R(p) - I_{3\times3}\right)\left(\mathcal{V}_i(p) - \mathcal{V}_i(\bar{p})\right)\| \ . \tag{6}$$

The left-hand side of Eq. (6) is minimized by $s(p) = s(\bar{p})$, which in turn leads to $R(p) = I_{3\times3}$ on the right-hand side. This examination shows that variational regularization directly on the scene flow $s(p)$, e.g. $||\nabla s||_{2,1}$, actually favors solutions with vanishing rotations. We can thus significantly stabilize the computation by regularizing the parameters of the *rigid* scene flow estimates instead.

Due to the dense temporal acquisition of the RGB-D data stream, we can assume the rigid motions to be *small*. Applying the Rodrigues' rotation formula [20] and the small-angle approximation allows us to rewrite the rotation as

$$R \approx I_{3\times3} + [\alpha]_\times \ , \tag{7}$$

with rotation angle $\theta = \|\alpha\|$, rotation-axis $r = \frac{\alpha}{\|\alpha\|} \in \mathfrak{so}(3)$, and the skew-symmetric cross-product matrix $[\cdot]_\times$. Using this approximation for rotations and the anticommutativity of the cross product, Eq. (5) becomes

$$s(p) = -[\mathcal{V}_i(p)]_\times \alpha(p) + t(p) \ . \tag{8}$$

Subsequently, we will refer to this linearized version as *rigid scene flow* and derive an algorithm that estimates the values of $\alpha(p)$ and $t(p)$.

D. Energy Formulation

After describing the scene flow in terms of local angle α and local translation t in Sec. III-C, we can now formulate the problem of scene flow estimation as an energy minimization, defining the optimal rigid scene flow as the minimizing arguments of the energy

$$E_{\text{scene flow}}(\alpha, t) = E_{\text{data}}(\alpha, t) + E_{\text{reg}}(\alpha, t). \tag{9}$$

The first part of this energy, the data term E_{data}, penalizes deviations from the rigid scene flow characterizations in our input data. The second term E_{reg} is a regularizer, that penalizes globally inconsistent solutions based on prior knowledge about our desired solutions.

a) Data term: The data term consists of two components $E_\mathcal{V}$ and $E_\mathcal{I}$, which describe the *depth consistency* related to the vertex maps $\mathcal{V}_i$ and the *brightness consistency assumption* with respect to the RGB images $\mathcal{I}_i$.

The first premise is the definition of rigid scene flow in Eq. (4), which can be formulated as

$$\mathcal{V}_i(p) - [\mathcal{V}_i(p)]_\times \alpha(p) + t(p) = \mathcal{V}_{i+1}\left(p'(\alpha(p), t(p))\right) \ , \tag{10}$$

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

given the linearization in Eq. (7) and anticommutativity of the cross product. The corresponding pixel p' is

$$p'(\alpha(p),t(p)) = \pi\left(\mathcal{V}_i(p) - [\mathcal{V}_i(p)]_\times \alpha(p) + t(p)\right) \quad . \tag{11}$$

Rewriting Eq. (10), we finally obtain the energy functional $E_\mathcal{V}$

$$\begin{aligned} E_\mathcal{V}(\alpha,t) = \sum_{p\in\Omega} \Big\| & \mathcal{V}_{i+1}\big(p'(\alpha(p),t(p))\big) \\ & - \mathcal{V}_i(p) + [\mathcal{V}_i(p)]_\times \alpha(p) - t(p) \Big\|_1 \quad . \end{aligned} \tag{12}$$

This component of the energy functional can be understood as a form of the *depth constancy assumption*.

The second premise is that colors do not change between two consecutive images $\mathcal{I}_i$ and $\mathcal{I}_{i+1}$ for corresponding pixels, i.e.

$$\mathcal{I}_i(p) = \mathcal{I}_{i+1}\big(p'(\alpha(p),t(p))\big) \quad . \tag{13}$$

We formulate this so called *brightness consistency assumption* implicitly through

$$E_\mathcal{I}(\alpha,t) = \sum_{p\in\Omega} \Big\| \mathcal{I}_{i+1}\big(p'(\alpha(p),t(p))\big) - \mathcal{I}_i(p) \Big\|_1 \quad . \tag{14}$$

The overall data term is given by

$$E_{\text{data}}(\alpha,t) = E_\mathcal{I}(\alpha,t) + \lambda_\mathcal{V} E_\mathcal{V}(\alpha,t). \tag{15}$$

The non-negative parameter $\lambda_\mathcal{V}$ weights the different components of the data term. We will discuss the choice of this hyper-parameter in the Sec. IV.

b) Regularization term: Solely optimizing for the data term described above would lead to incoherent and noisy point-wise solutions. Therefore, regularization terms are indispensable to construct plausible solutions. We formulate our piecewise rigid motion regularization directly as a penalization of the rotational and translational variables α and t, respectively.

Merely minimizing $\|\nabla\alpha\|_{2,1}$ and $\|\nabla t\|_{2,1}$ would, however, interfere with our primary goal to model piecewise-constant rigid motion, as it penalizes the magnitude of change in α and t, and, thus, only approximately favors piecewise-constant solutions. Therefore, we switch from the convex L_1 norm to the non-convex L_0 "norm" [6], defined for a vector $x \in \mathbb{R}^d$ by $||x||_0 = \#\{i \in 1,\ldots,d \mid x_i \neq 0\}$ and choose the regularizing terms

$$E_{d\alpha}(\alpha) = \sum_{p\in\Omega} \|\nabla\alpha(p)\|_{2,0} \tag{16}$$

and

$$E_{dt}(t) = \sum_{p\in\Omega} \|\nabla t(p)\|_{2,0} \tag{17}$$

This regularization, sometimes denoted as *Pott's* model [25], is a special case of the piecewise-constant Mumford-Shah model [2]. It is important to note that this form of L_0 regularization, can not be applied to the scene flow s directly, as it would favor an unrealistic, piecewise-constant scene flow. In contrast, our regularization of α and t favors the intended *piece-wise rigid scene flow*.

E. Energy Minimization

In this section we discuss algorithmic approaches capable of efficiently minimize the energy functional $E_{\text{scene flow}}(\alpha,t)$ introduced in Sec. III-D. Our first observation is that the described data terms are nonlinear and non-convex, prohibiting efficient convex optimization algorithms. As such we consider the well-known strategy of *successive linearizations* and *warpings* [5]. We start by downsampling the RGB-D images to a small resolution, at which the nonlinear parts of E_{data} can be linearly approximated with sufficient precision. Afterwards, we iteratively upsample the current solution to a higher resolution and refine the linearization. In the following, we discuss the linearization process, as well as the optimization approach for the linearized subproblems.

a) Linearization: In the following discussion, we drop the argument of the pixel position p to improve readability. Note that p' is the mapping function given in Eq. (3). The data energy functional term linearization $\mathcal{M} \in \{\mathcal{I},\mathcal{V}\}$ is developed at an initial rigid scene flow guess (α_0,t_0) as

$$\begin{aligned} \mathcal{M}_{i+1}\big(p'(\alpha,t)\big) \approx\ & \mathcal{M}_{i+1}\big(p'(\alpha_0,t_0)\big) \\ & + \nabla\mathcal{M}_{i+1}\big(p'(\alpha_0,t_0)\big) \cdot \nabla p'(\alpha_0,t_0) \\ & \cdot \Big((\alpha,t)^\top - (\alpha_0,t_0)^\top\Big) \quad . \end{aligned} \tag{18}$$

The corresponding pixel p' is given by Eq. (11), and its gradient is

$$\nabla p'(\alpha,t) = \nabla\pi(p'(\alpha,t)) \cdot [-\mathcal{V}_i | I_{3\times3}] \quad . \tag{19}$$

Applying this linearization to Eqs. (12) and (14), we get

$$E_\mathcal{I}(\alpha,t) = \Big\| A_\mathcal{I}(\alpha,t)^\top - b_\mathcal{I} \Big\|_{1,1} \tag{20}$$

$$E_\mathcal{V}(\alpha,t) = \Big\| A_\mathcal{V}(\alpha,t)^\top - b_\mathcal{V} \Big\|_{1,1} \tag{21}$$

where

$$A_\mathcal{I} = \nabla\mathcal{I}\big(p'(\alpha_0,t_0)\big) \cdot \nabla p'(\alpha_0,t_0) \tag{22}$$

$$A_\mathcal{V} = \nabla\mathcal{V}\big(p'(\alpha_0,t_0)\big) \cdot \nabla p'(\alpha_0,t_0) + [\mathcal{V}_i | -I_{3\times3}] \quad , \tag{23}$$

and $b_\mathcal{M} = \mathcal{M}_i - \mathcal{M}_{i+1} + A_\mathcal{M} \cdot (\alpha,t)^\top$.

As we use the L_1 norm in the entire data term, we can rewrite our problem as

$$E_{\text{datalin}}(\alpha,t) = \left\| A_{\text{data}} \begin{pmatrix} \alpha \\ t \end{pmatrix} - b_{\text{data}} \right\|_{1,1} \quad , \tag{24}$$

with $A_{\text{data}} = \left[\mathbf{A}_\mathcal{I}^\top \middle| \lambda_\mathcal{V} \mathbf{A}_\mathcal{V}^\top\right]^\top$ and $b_{\text{data}} = (b_\mathcal{I}, \lambda_\mathcal{V} b_\mathcal{V})^\top$.

b) Regularizer: Similar to $E_{datalin}$ we rewrite the regularization term in matrix notation as

$$E_{\text{reg}}(\alpha,t) = \left\| A_{\text{reg}} \cdot \begin{pmatrix} \alpha \\ t \end{pmatrix} \right\|_{2,0} \quad , \tag{25}$$

with

$$A_{\text{reg}} = \begin{pmatrix} \lambda_{dt}\nabla & 0 \\ 0 & \lambda_{d\alpha}\nabla \end{pmatrix} \quad . \tag{26}$$

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

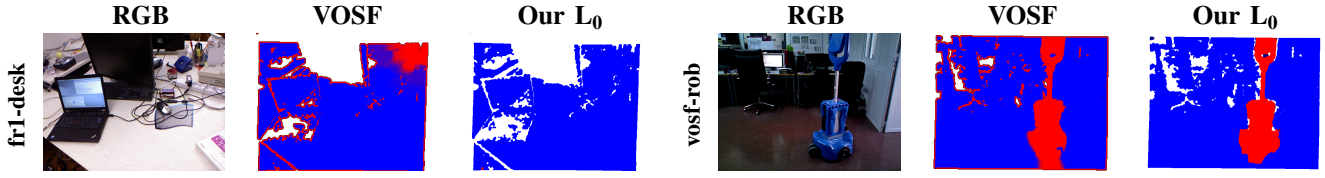

Fig. 1. Visual comparison of motion segments for **VOSF** and **Our L_0** approach. The left example shows the static scene **fr1-desk**, i.e. a scene with one single motion cluster. This is well approximated by both, although **VOSF** is not as robust in the upper right corner. The right example shows a dynamic scene, where both approaches achieve comparable results.

c) An Efficient Subproblem Solver: To solve the energy functional consisting of the linearized data term in Eq. (24) and the regularization terms in Eq. (16) and (17), we make use of *Algorithm 2* of the *primal-dual hybrid gradient* (PDHG) approach discussed in [7]. This algorithm minimizes energy functionals of the following kind

$$\min_u G(u) + F(Ku) \ . \tag{27}$$

The solution to this problem under the primal-dual framework is the iterative procedure, where u_0 and $\hat{u}_0$ are taken from the upsampled, previous solution and $y_0 = 0$

$$y_{k+1} = \text{prox}_{\sigma_n F^*}(y_k + \sigma_n K u_k) \tag{28}$$

$$\hat{u}_{k+1} = \text{prox}_{\tau_n G}\left(\hat{u}_k - \tau_n K^\top y_{k+1}\right) \tag{29}$$

$$\theta_n = \frac{1}{\sqrt{1+2\gamma\tau_n}},\ \tau_{n+1} = \theta_n \tau_n,\ \sigma_{n+1} = \frac{\sigma_n}{\theta_n} \tag{30}$$

$$u_{k+1} = \hat{u}_{k+1} + \theta_n(\hat{u}_{k+1} - \hat{u}_k) \ , \tag{31}$$

with $\sigma_0, \tau_0 > 0$, $\sigma_0 \tau_0 L^2 < 1$ and $L = \|K\|$. As proposed in [26], see also [17], we need to reformulate this algorithm to use only primal proximal operators, as the non-convexity of the L_0 "norm" invalidates a direct application of the algorithm. To do so, we apply Moreau's decomposition [4] and rewrite Eq. (28) to

$$y_{k+1} = y_k + \sigma_n K u_k - \sigma_n \text{prox}_{\frac{1}{\sigma_n}F}\left(\frac{y_k}{\sigma_n} + K u_k\right). \tag{32}$$

This transforms the primal-dual structure into a primal-proximal algorithm, whose convergence properties are closely connected to non-convex ADMM approaches [31].

d) Primal-Dual formulation: For the primal-dual framework, we define $\widehat{E}_\text{d}(v) = \|v - b_{data}\|_1$ and $\widehat{E}_\text{r}(v) = \|v\|_0$, such that $\widehat{E}_\text{d}(A_\text{data}u) = E_\text{datalin}(u)$ and $\widehat{E}_\text{r}(A_\text{reg}u) = E_\text{reg}(u)$. Given these definitions, we choose $G(u) = 0$ and set

$$\mathbf{u} = \begin{pmatrix}\alpha \\ t\end{pmatrix}, \qquad \mathbf{K} = \begin{pmatrix}A_\text{data} \\ A_\text{reg}\end{pmatrix} \tag{33}$$

$$F(Ku) = \widehat{E}_\text{d}(A_\text{data}u) + \widehat{E}_\text{r}(A_\text{reg}u) \ . \tag{34}$$

The separable sum property of the proximity operator leads to

$$\text{prox}_{\frac{1}{\sigma_n}F}(q,r) = \begin{pmatrix}\text{prox}_{\frac{1}{\sigma_n}\widehat{E}_\text{d}}(q) \\ \text{prox}_{\frac{1}{\sigma_n}\widehat{E}_\text{r}}(r)\end{pmatrix}, \tag{35}$$

which finally leads to

$$\text{prox}_{\frac{1}{\sigma_n}\widehat{E}_\text{d}}(q) = \text{sign}(q - b_{data}) \max\left(|q| - \frac{1}{\sigma_n}, b_{data}\right) \tag{36}$$

$$\text{prox}_{\frac{1}{\sigma_n}\widehat{E}_\text{r}}(r) = \begin{cases}0 & |r| \le \sqrt{2\frac{1}{\sigma_n}} \\ r & \text{else}\end{cases} \tag{37}$$

$$\text{prox}_{\tau_n G}(u) = u, \tag{38}$$

which are the *soft thresholding* function, the *hard thresholding* function and the *identity* function, respectively.

This demonstrates the great attractiveness of the primal dual algorithm. The intricate non-convex energy constructed in Eq. (9) in Sec. III-D can be rewritten as a series of linear operators and simple, point-wise nonlinear operations. This allows for an efficient implementation that lends itself well to pixel-level parallel executions interlaced with well-developed linear algebra calls. Inserting these proximity operators in Eqs. (28)-(31) yields an primal-dual solver for the linearized subproblems.

IV. EVALUATION

A. Evaluation Setup

a) Implementation: We implemented our approach in the *PROST* framework [16], which is a framework designed for solving large-scale problems with proximal structure. This general purpose optimization framework however, does not allow for a real-time implementation. We note that the complexity of our optimization algorithm is effectively equivalent to the **PDFlow** approach of Jaimez et al. [15], which runs in real-time.

As shown in the previous section, the behavior of our algorithm can be controlled through a set of parameters. Throughout all our experiments, we set these parameters to $\lambda_V = 2.0, \lambda_{dt} = 2.0, \lambda_{da} = 0.5, \tau_0 = 0.25, \sigma_0 = 1.5, \gamma = 140$. We found these parameters to yield a good trade-off between robustness and accuracy of our approach.

Our approach segments the rigid motions implicitly. However, we can compute the segmentation from the final rigid motion parameters α and t by thresholding and point-wise multiplication of these values. Fig. 1 shows that these implicitly computed motions clusters are competitive to explicitly computed segmentations.

b) Datasets and Competitors: For comparison and evaluation, we use RGB-D image pairs from three different real world datasets, the Bonn multi-body benchmark [27], the TUM benchmark [28], and the supplemented datasets

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

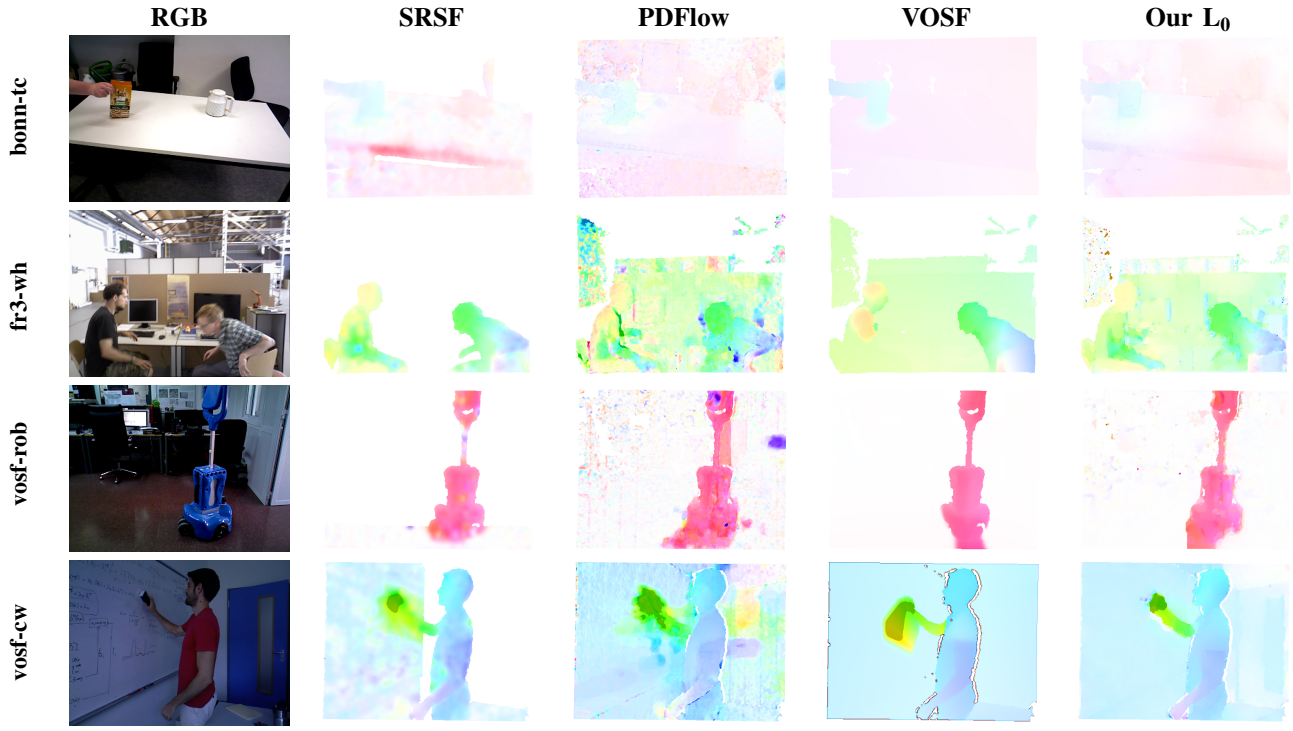

Fig. 2. Color-coded projections of the scene flow estimates of **SRSF** [18], **PDFlow** [15], **VOSF** [14] and our L_0 approach on dynamic scenes. Note how our approach combines both, robustness and accuracy. Consider especially the right arm of the left person in **tum fr3-wh** or the white board eraser in **vosf-cw**.

from Jaimez et al. [14]. Tab. I gives an overview over the used image pairs. Further examples are presented in the supplementary material.

For evaluation we compare our method (**Our L_0**) to the **SRSF** approach [18] (rigid scene flow with L_1 regularization), the **PDFlow** [15] (scene flow with L_1 regularization) and **VOSF** [14] (alternating odometry and rigid scene flow).

ID	Full Name	Dyn	Bench.	Frames
bonn-tc	Tea can	y	Bonn	700-701
fr1-desk	freiburg1_desk	n	TUM	200-201
fr1-xyz	freiburg1_xyz	n	TUM	203-204
fr3-wh	freiburg3_Walk_Halfsphere	y	TUM	1-2
vosf-rob	Robot	y	[14]	1-2
vosf-cw	Cleaning whiteborad	y	[14]	1-2

TABLE I

RGB-D IMAGE PAIRS USED FOR EVALUATION: ABBREVIATION USED IN THE PAPER (**ID**), FULL NAME OF DATASET (**FULL NAME**), DYNAMIC SCENE (**DYN**), BENCHMARK (**BENCH.**), AND FRAMES USED (**FRAMES**).

photometric and geometric error measures of valid pixels shown in Table II, yields that the actual accuracy of our approach is significantly better. Our indirect regularization of the scene flow via the implicit L_0 segmentation of the rigid motion components avoids over-smoothing in contrast to the direct regularization of the **VOSF** approach.

ID	metric	SRSF	PDFlow	VOSF	Our L0
bonn-tc	*PE*	0.032	0.031	0.033	**0.027**
	GE	0.023	0.029	0.033	**0.018**
fr3-wh	*PE*	0.187	0.106	0.094	**0.069**
	GE	0.564	0.634	0.496	**0.418**
vosf-rob	*PE*	0.054	0.067	0.050	**0.040**
	GE	0.130	0.146	0.127	**0.093**
vosf-cw	*PE*	0.048	0.037	0.041	**0.029**
	GE	0.068	0.068	0.074	**0.045**

TABLE II

PHOTOMETRIC ERRORS (PE) AND GEOMETRIC ERRORS (GE) FOR THE DATASETS SHOWN IN FIG. 2.

B. Evaluation without reference flows

A visual comparison of the previously mentioned approaches shown in Fig. 2. We show a color-coded projection of the three-dimensional scene flow estimations onto the image plane, which demonstrates the accuracy and robustness of our approach on real-world data. Compared to **SRSF** and **PDFlow**, our flow estimation contains significantly less noise and the moving objects in the scene are consistently estimated. Compared to our approach, **VOSF** produces visually smoother scene flows. However, incorporating the

C. Evaluation with Reference Flows

The static scenes in Fig. 3 allow for an estimation of the reference scene flow from the given ground truth camera pose. This allows for a comparison of the endpoint error (EPE), i.e. the L_2 norm of the differences of ground truth and estimated scene flow. The EPEs of these two scenes are shown in Table III, while the color-coded projections of the scene flows are shown in Fig. 3. It turns out that our approach leads to a lower EPE. This is due to the fact that the static scene can be described by a single rigid motion, the motion

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

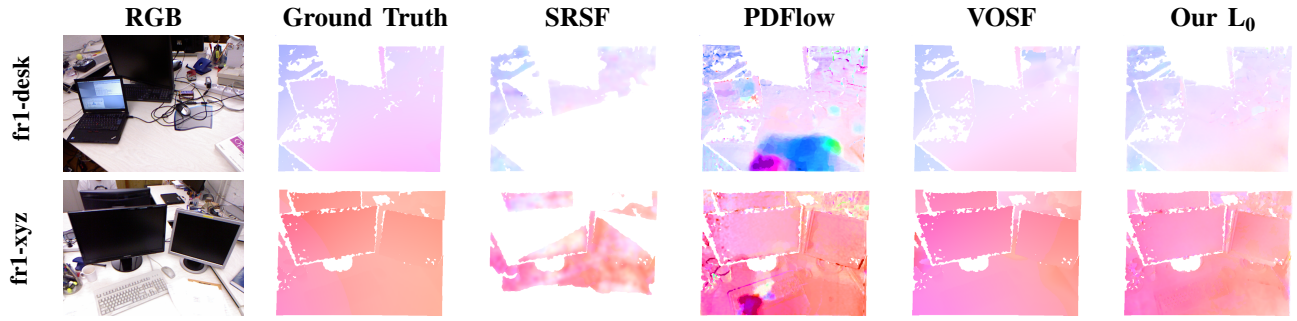

Fig. 3. Projection of the scene flow estimates of **SRSF** [18], **PDFlow** [15], **VOSF** [14] and our L_0 approach for static scenes with reference scene flow from the TUM dataset [28]. The reference scene flow was estimated from ground truth camera pose. Note how the smoothing effect of the **VOSF** approach is useful in this mostly planar scene.

of the camera. Our L_0 approach is well-suited for this kind of scene flow estimation, as we directly optimize for rigid motions. This effect is visualized in Fig. 1, where we find that our approach leads to a single consistent motion cluster.

ID	metric	SRSF	PDFlow	VOSF	Our L0
fr1-desk	*PE*	0.112	0.082	0.070	**0.048**
	GE	0.087	0.117	0.115	**0.065**
	EPE	5.332	7.132	4.385	**4.236**
fr1-xyz	*PE*	0.117	0.076	0.083	**0.046**
	GE	0.099	0.112	0.114	**0.065**
	EPE	7.249	5.692	4.110	**3.747**

TABLE III

PHOTOMETRIC ERRORS (PE), GEOMETRIC ERRORS (GE) AND END-POINT ERRORS (EPE) FOR THE DATASETS SHOWN IN FIG. 3.

D. Discussion

We formulate a model of piece-wise rigid scene flow via L_0 regularization of the rotational and translational motion components. This formulation strongly contributes to the improvements in terms of accuracy of the scene flow estimation.

Failure cases of our model are RGB-D frames, where the rigid-motion approximation in (7) is violated by rapid rotations. Such cases still produce a globally reasonable scene flow, but the motion of these objects is not recovered. This can be observed for the dataset **vosf-cw** in Fig. 2. Here, the fast moving white board eraser is not precisely reconstructed by our approach, while the other parts of the scene and especially the background motion are accurately estimated. Yet compared to previous methods the error in the vicinity of the eraser is considerably reduced.

Further failures can occur from strongly non-rigid motions, such as elasto-plastic deformations. Here, the deformations will be approximated by piecewise-rigid transformations, leading to incorrect results.

E. Conclusion

The implicit motion segmentation presented in this work is capable of stabilizing variational scene flow estimation, resulting in piecewise-rigid flows. In contrast to other methods that rely on segmentation, our approach utilizes the clustering properties of the L_0 regularization and thus performs the clustering of motions implicitly.

REFERENCES

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels. Technical report, Ecole Polytechnique Fédéral de Lausssanne (EPFL), 2010.

[2] L. Bar, T. F. Chan, G. Chung, M. Jung, N. Kiryati, N. Sochen, and L. A. Vese. Mumford and shah model and its applications to image segmentation and image restoration. *Handbook of mathematical methods in imaging*, pages 1–52, 2014.

[3] P. J. Besl and N. D. McKay. Method for registration of 3-D shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.

[4] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK ; New York, 2004.

[5] 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*, pages 25–36. Springer, 2004.

[6] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theor.*, 52(2):489–509, Feb. 2006.

[7] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40(1):120–145, 2011.

[8] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992.

[9] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics Automation Magazine*, 13(2):99–110, June 2006.

[10] S. Ghuffar, N. Brosch, N. Pfeifer, and M. Gelautz. Motion estimation and segmentation in depth and intensity videos. *Integrated Computer-Aided Engineering*, 21(3):203–218, 2014.

[11] V. Golyanik, K. Kim, R. Maier, M. Nießner, D. Stricker, and J. Kautz. Multiframe scene flow with piecewise rigid motion. In *Proc. Int. Conf. 3D Vision (3DV)*, pages 273–281, 2017.

[12] E. Herbst, X. Ren, and D. Fox. RGB-D flow: Dense 3-D motion estimation using color and depth. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 2276–2282. IEEE, 2013.

[13] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger. Volumedeform: Real-time volumetric non-rigid reconstruction. In *Proc. Europ. Conf. Computer Vision*, pages 362–379. Springer, 2016.

[14] M. Jaimez, C. Kerl, J. Gonzalez-Jimenez, and D. Cremers. Fast odometry and scene flow from RGB-D cameras based on geometric clustering. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 3992–3999. IEEE, 2017.

[15] M. Jaimez, M. Souiai, J. Gonzalez-Jimenez, and D. Cremers. A primal-dual framework for real-time dense RGB-D scene flow. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 98–104. IEEE, 2015.

[16] T. Möllenhoff, E. Laude, M. Möller, J. Lellmann, and D. Cremers. Sublabel-accurate relaxation of nonconvex energies. *CoRR*, abs/1512.01383, 2015.

[17] T. Möllenhoff, E. Strekalovskiy, M. Moeller, and D. Cremers. The Primal-Dual Hybrid Gradient Method for Semiconvex Splittings. *SIAM Journal on Imaging Sciences*, 8(2):827–857, Jan. 2015.

[18] J. Quiroga, T. Brox, F. Devernay, and J. Crowley. Dense semi-rigid scene flow estimation from RGBD images. In *Proc. Europ. Conf. Computer Vision*, pages 567–582. Springer, 2014.

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.

[19] J. Quiroga, F. Devernay, and J. Crowley. Local/global scene flow estimation. In *Proc. Int. IEEE Conf. Image Processing (ICIP)*, pages 3850–3854. IEEE, 2013.
[20] O. Rodriguez. Des lois geometriques qui regissent les desplacements dun systeme solide dans lespace et de la variation des coordonnees provenant de deplacements consideres independamment des causes qui peuvent les produire. *Journal de Mathématiques Pures et Appliquées*, 5:380–440, 1840.
[21] M. Rünz and L. Agapito. Co-fusion: Real-time segmentation, tracking and fusion of multiple objects. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 4471–4478, May 2017.
[22] M. Slavcheva, M. Baust, D. Cremers, and S. Ilic. Killingfusion: Non-rigid 3d reconstruction without correspondences. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 3(4):7, 2017.
[23] J. Solomon, M. Ben-Chen, A. Butscher, and L. Guibas. As-killing-as-possible vector fields for planar deformation. *Computer Graphics Forum*, 30(5):1543–1552, 2011.
[24] O. Sorkine and M. Alexa. As-rigid-as-possible surface modeling. In *Proc. Symp. Geometry processing*, volume 4, pages 109–116, 2007.
[25] M. Storath, A. Weinmann, J. Frikel, and M. Unser. Joint image reconstruction and segmentation using the Potts model. *Inverse Problems*, 31(2):025003, Feb. 2015.
[26] E. Strekalovskiy and D. Cremers. Real-time minimization of the piecewise smooth mumford-shah functional. In D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 127–141. Springer International Publishing, 2014.
[27] J. Stueckler and S. Behnke. Efficient dense 3D rigid-body motion segmentation in RGB-D video. In *Proc. British Machine Vision Conference*, 2013.
[28] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.
[29] D. Sun, E. B. Sudderth, and H. Pfister. Layered RGBD scene flow estimation. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 548–556, 2015.
[30] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. In *Proc. IEEE Int. Conf. on Computer Vision*, volume 2, pages 722–729, 1999.
[31] Y. Wang, W. Yin, and J. Zeng. Global convergence of admm in nonconvex nonsmooth optimization. *Journal of Scientific Computing*, 78(1):29–63, 2019.
[32] Z. Yan and X. Xiang. Scene flow estimation: A survey. *CoRR*, abs/1612.02590, 2016.
[33] 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. *Computer Graphics Forum*, 37(2):625–652, 2018.

Authorized licensed use limited to: UB Siegen. Downloaded on October 13,2020 at 14:01:50 UTC from IEEE Xplore. Restrictions apply.