3D Scene Flow Estimation with a Rigid Motion Prior

Christoph Vogel Konrad Schindler
Photogrammetry and Remote Sensing
ETH Zürich

Stefan Roth
Department of Computer Science
TU Darmstadt

Abstract

We present an approach to 3D scene flow estimation, which exploits that in realistic scenarios image motion is frequently dominated by observer motion and independent, but rigid object motion. We cast the dense estimation of both scene structure and 3D motion from sequences of two or more views as a single energy minimization problem. We show that agnostic smoothness priors, such as the popular total variation, are biased against motion discontinuities in viewing direction. Instead, we propose to regularize by encouraging local rigidity of the 3D scene. We derive a local rigidity constraint of the 3D scene flow and define a smoothness term that penalizes deviations from that constraint, thus favoring solutions that consist largely of rigidly moving parts. Our experiments show that the new rigid motion prior reduces the 3D flow error by 42% compared to standard TV regularization with the same data term.

1. Introduction

Unlike disparity estimation from stereo images or optical flow estimation from monocular videos, the estimation of *3D scene flow* has only recently gained attention. Probably the first to tackle dense scene flow estimation were Vedula *et al*. [19], who also coined the term. Given images from two (or more) different points in time, recorded with two (or more) synchronized cameras with known intrinsics as well as relative position and orientation, the goal of 3D scene flow is to *estimate both the 3D geometry and the 3D motion densely at every pixel*. Conceptually, it is thus the synthesis of two classical challenges, 3D shape reconstruction and dense motion estimation. Densely recovering both shape and motion of a dynamic scene has many important applications, such as human motion analysis, virtual and augmented reality, navigation and driver assistance, and more.

One of the challenges of 3D scene flow estimation is that it requires jointly solving the two-view (or multi-view) stereo problem at two different time steps, and the optical flow problem for each of the cameras. Here, we aim to estimate the 3D scene flow directly from the images without precomputing 2D optical flow [19] or disparity [22] in separate steps. From both stereo and optical flow, 3D scene flow estimation inherits the need for regularization: it is ill-posed because of depth ambiguity and the aperture problem, and thus requires prior knowledge. To date, most scene flow approaches have relied on relatively simple priors favoring smooth surfaces and motion fields [3, 12, 22].

Here, we first analyze the 3D scene flow problem geometrically, and show that smoothness terms from the 2D flow literature are systematically biased in the 3D case and should thus not be uncritically adopted. Specifically, we demonstrate their inherent tendency against motion discontinuities in viewing direction, which stems from the limited baseline in typical applications. When projecting the 3D flow onto the image plane and evaluating in 2D, as is commonly done, this does not become apparent. Only when evaluating the scene flow in 3D, which has been advocated only quite recently [3], this becomes clearer. The main goal of this paper is to address the issue and propose a more realistic prior model to improve scene flow estimation.

Rather than being completely agnostic about the scene, we assume that it is composed of independently, but *rigidly* moving 3D parts, which is approximately true for many scenes of practical interest. Starting from this assumption, we regularize the problem by encouraging a locally rigid 3D motion field. We develop a regularization framework that implicitly estimates rigid motion in local regions from depth and 3D flow, and penalizes deviations from that motion. A locally adaptive weighting scheme [14, 23] is used to handle motion discontinuities between independently moving parts. Such a local rigidity prior is rather different from global rigidity assumptions as they have been used in 2D flow estimation [16, 20], where image motion is assumed to predominantly arise from observer ego-motion.

Quantitative results on synthetic scenes demonstrate that the proposed rigidity prior avoids systematic biases of isotropic regularization, and leads to significantly more accurate motion fields, reducing the 3D flow error by 42% on average compared to standard total variation regularization. Results on real scenes show the applicability of the novel rigidity prior also for scenarios with articulated motion.

2011 IEEE International Conference on Computer Vision
978-1-4577-1102-2/11/$26.00

2. Related Work

The problem of 3D scene flow estimation was introduced by Vedula *et al*. [19], who defined it as the dense estimation of 3D geometry and motion vectors. This early work followed a two-step approach: First, 2D optical flow is estimated independently for each camera of a multi-camera array; after that, the 3D flow field is fitted to the 2D flow fields. Thus, optical flow computation and 3D modeling are decoupled. In particular, the constraint that flow fields in different images must be consistent reprojections of the same 3D motion is not used during optical flow estimation.

Only much later, Pons *et al*. [12] proposed an approach that estimated the 3D scene flow directly from the image data, rather than fitting it to intermediate 2D flow fields. Their method is a generalization of the level set method for multi-view stereo, to also include the motion field.

Wedel *et al*. [22] proposed to parametrize the scene flow completely in 2D image space. Stereo disparity is precomputed and kept fixed, thus depth and motion estimation are again decoupled. Given disparities, the scene flow constraints are employed to estimate the optical flow for one image, and the per-pixel disparity difference between the two time steps. As discussed below, the employed 2D smoothness term is problematic as it operates in image space: it favors smooth 2D projections, which is not the same as a smooth 3D motion field. In particular, gradients in viewing direction are projected away. Rabe *et al*. [13] subsequently extended this work with a Kalman filter modeling temporal smoothness over multiple frames.

In recent work, Valgaerts *et al*. [17] assume that only the camera intrinsics rather than the full calibration is known. They show that, even if the scene contains independent object motion and deformation, one can estimate both the scene flow and the relative orientation of the two cameras from the raw image data. The method alternates between estimating the orientation parameters and the scene flow.

The approach most similar to our work is by Basha *et al*. [3]. As done here, the scene flow is parametrized in 3D in terms of the depth w.r.t. a reference view as well as 3D flow vectors, and all unknowns are estimated jointly in one optimization framework. However, the scene flow is regularized using total variation, whereas we explicitly model the local rigidity of the scene. Furthermore, [3] concentrates on multi-camera setups with relatively large baselines, whereas we also evaluate the performance for the two-view case with narrow baselines, and thus weaker 3D constraints. Note that the latter case is important in practice: a key application are vehicle-mounted cameras [13], where the total baseline is limited by the physical dimensions of the vehicle.

Rigidity assumptions date back to early work by Adiv [1], who recovered 3D scene structure and ego-motion from 2D image motion. Our approach instead recovers depth and dense 3D scene flow directly from image sequences. Moreover, we do not strictly enforce rigidity in a set of segments, but rather penalize deviations from local rigidity. In this way our approach also differs from soft, but global rigidity priors that have been used for 2D flow estimation [16, 20]. Assuming a globally rigid scene with optical flow arising from ego-motion, they consequently penalize deviations of the flow from the epipolar lines. This works very well for scenarios with predominant observer motion, but cannot be expected to work as well for scenes with independently moving objects. Local rigidity has also been used for 3D motion capture with surfel [5] and mesh-based representations [7]. Here we focus on dense scene flow estimation without explicit surface representations in scenes with arbitrary depth and object discontinuities.

Our rigidity prior is also related to work of Nir *et al*. [11], who penalize deviations from local rigidity by over-parameterizing the flow field in terms of rigid motion parameters and penalizing parameter changes. While an attractive concept, penalizing changes in such an over-parametrization was found to be dependent on the choice of coordinate system [15]. Our approach avoids this issue by directly penalizing deviations from local rigidity without a reference coordinate system. Moreover, we extend the idea of local rigidity priors to the estimation of 3D scene flow.

3. Basic 3D Scene Flow Framework

Our goal is to compute the depth $d : \Omega \to (0, \infty)$ and the 3D motion field $\mathbf{w} = (w_x, w_y, w_z)^T : \Omega \to \mathbb{R}^3$, which we assume to be parametrized over the image domain $\Omega \subset \mathbb{R}^2$ of a reference camera. We, moreover, assume a setup of N cameras with known intrinsics and known relative position and orientation (*i.e*., a calibrated camera rig). Since we parametrize the scene with 3D entities, the images do not need to be rectified. Like most stereo and optical flow methods, we assume brightness constancy across time and viewpoint, and consequently penalize brightness differences using a data term E_D. This data term is combined with a spatial term E_S for regularization into an energy

$$E(d, \mathbf{w}) = E_D(d, \mathbf{w}) + \lambda E_S(d, \mathbf{w}), \tag{1}$$

where λ controls the amount of regularization. To enable efficient energy minimization we employ a relaxation similar to the one introduced in [24] for optical flow computation.

3.1. Setup and notation

In our approach the scene is parametrized over the image domain of a reference camera K_0. Additional views are denoted by $K_i, i = 1 \dots N$, where $K_i \in \mathbb{R}^{3\times4}$ specifies the respective projection matrix. We assume that the cameras capture the scene at two time steps $t \in \{0, 1\}$ yielding images I_i^t. The cameras project 3D scene points $\mathbf{P}^t$ at time t

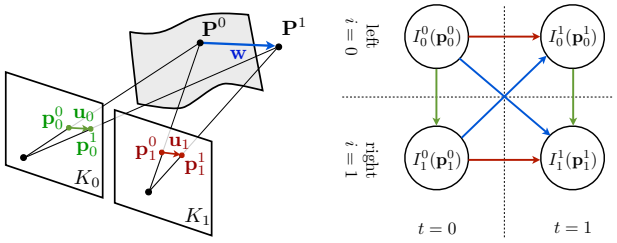

Figure 1. *(left)* The principle of scene flow computation: from image points $\mathbf{p}_0$ and $\mathbf{p}_1$ observed at two different times $t=0$ and $t=1$, estimate the 3D point $\mathbf{P}^0$ and its 3D motion vector $\mathbf{w}$; *(right)* Data terms in the two-view case: red – optical flow, green – stereo, blue – cross-couplings.

to homogeneous 2D image points $\mathbf{p}_i^t = (x_i^t, y_i^t, 1)^T$; note that subscripts denote the camera, superscripts the time. The surface points are parametrized by their pixel coordinates and depth in the reference camera at time $t=0$. Hence $\mathbf{P}^0 = \widetilde{K}_0^{-1}\mathbf{p}_0^0 / \|\widetilde{K}_0^{-1}\mathbf{p}_0^0\| \cdot d(\mathbf{p}_0^0)$, assuming *w.l.o.g.* that the camera center is the origin of the world system, such that $K_0 = [\widetilde{K}_0|0]$. In slight abuse of notation we sometimes abbreviate coordinates in the reference system as $\mathbf{x} = (x_0^0, y_0^0)$. At time $t=1$ the same point has been displaced to $\mathbf{P}^1 = \mathbf{P}^0 + \mathbf{w}(x_0^0, y_0^0)$. Per pixel, we thus need to determine four unknowns $d(\mathbf{x})$ and $\mathbf{w}(\mathbf{x})$. In the image plane of camera K_i the displaced point $\mathbf{P}^1$ is projected to $\mathbf{p}_i^1 = \mathbf{p}_i^0 + \mathbf{u}_i$. Here $\mathbf{u}_i = (u_i, v_i, 0)$ is the projection of the 3D scene flow and can be observed as the optical flow of camera i at point $\mathbf{p}_i^0$. The setup is summarized in Fig. 1.

3.2. Data term

Since the data term is not a main concern in this paper, we adopt a rather standard brightness constancy term, which penalizes brightness differences between image locations corresponding to the same scene point according to the reconstructed depth and motion. A difference to standard approaches, however, is that we exploit brightness constancy between all pairs of images (see Fig. 1, right). In real-world scenes brightness constancy only rarely holds exactly. To reduce the influence of outliers due to occlusions, shadows, specularities, *etc*. the penalty function thus should be robust. We use the function $\rho(x) = \sqrt{x^2 + \epsilon^2}$ [4], a differentiable variant of the L_1-norm.

Temporal correspondence in each view leads to $N+1$ brightness constancy terms for the optical flow:

$$D_i^{01} = \rho\left(I_i^1(\mathbf{p}_i^1) - I_i^0(\mathbf{p}_i^0)\right), \quad i = 0, \ldots, N \tag{2}$$

Geometric correspondence between the reference camera K_0 and one of the other views $K_i, i \neq 0$ in both time steps leads to two terms for the stereo correspondence:

$$D_{0i}^t = \rho\left(I_i^t(\mathbf{p}_i^t) - I_0^t(\mathbf{p}_0^t)\right), \quad t \in \{0, 1\} \tag{3}$$

For each camera we complement these terms with two "cross terms", which impose brightness constancy across views *and* time with the reference camera (Fig. 1, right):

$$D_{0i}^{01} = \rho\left(I_i^1(\mathbf{p}_i^1) - I_0^0(\mathbf{p}_0^0)\right) \tag{4a}$$
$$D_{i0}^{01} = \rho\left(I_0^1(\mathbf{p}_0^1) - I_i^0(\mathbf{p}_i^0)\right). \tag{4b}$$

Although for each image pair only three of the six terms are linearly independent, we use all six to increase robustness with respect to the influence of noise.

Adding the contributions of all terms leads to

$$E_D(d, \mathbf{w}) = \int_\Omega D_0^{01} + \sum_{i=1}^{N}\left(D_i^{01} + D_{0i}^{01} + D_{i0}^{01} + \sum_{t=0}^{1} D_{0i}^t\right) \mathrm{d}\mathbf{x}. \tag{5}$$

In practice we use a straightforward spatial discretization of E_D, which is amenable to standard gradient methods.

To increase robustness against occlusions, we remove those constraints from consideration, which according to the current estimate involve occluded pixels, as detected by straight-forward z-buffering. For instance, if a pixel $\mathbf{p}_1^0$ in camera K_1 is occluded, we suppress all three energy terms involving $I_1^0(\mathbf{p}_1^0)$.

4. Spatial Term

It was shown in [19], that the aperture problem is a property of the scene, therefore scene flow computation remains ill-posed for an arbitrary number of views. In our case the brightness constancy equation system Eqs. (2)–(4) possesses four unknowns per pixel and except for image noise, it only has rank 3 and constrains the set of admissible flow vectors to a 1D subspace. In addition, image gradients are only valid in a small neighborhood of a pixel and are especially susceptible to image noise. Hence some form of regularization is needed.

In general our spatial term consists of two parts, one dealing with the 3D surface and the second being responsible for preserving the regularity of the motion. Since we are dealing with scenes containing discontinuities, we assume the scene to be only piecewise smooth.

4.1. Standard total variation prior

A standard way to define the spatial term is to penalize strong gradients in the motion and depth field. In particular, many optical and scene flow algorithms advocate using the total variation or some relaxation:

$$E_S^{\mathrm{TV}}(d, \mathbf{w}) = \int_\Omega \rho(\nabla d) + \rho(\nabla w_x) + \rho(\nabla w_y) + \rho(\nabla w_z) \mathrm{d}\mathbf{x}. \tag{6}$$

Discussion: Total variation was found to work very well for 2D optical flow [*e.g.* 24]. However, in our experience it is not a good regularizer for 3D flow. To see why, consider the following: in a narrow-baseline setting the data term contributes very little information about the scene flow in z-direction (depth change) – large changes of w_z can be

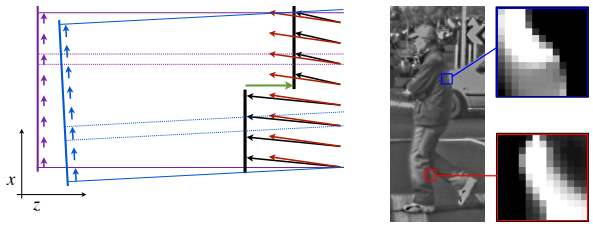

Figure 2. *(left)* For realistic baselines TV cannot correctly handle motion discontinuities in z-direction. Two cameras (blue and purple) observe a dynamic scene with two motions (black). Fitting an incorrect motion field (red) avoids the smoothness penalty at the motion boundary (green). The incorrect motion can be propagated far into both moving objects without incurring data penalty. *(right)* The weighting function η visualized for two patches. White pixels denote large and black pixels small weights.

compensated by very small changes of w_x and w_y to yield the same projected flow vectors in the images, and thus the same E_D (see Fig. 2, left). As a consequence, the TV regularizer has a built-in tendency against motion discontinuities in z-direction; taking $\nabla w_z \to 0$ will reduce the spatial term E_S with negligible effect on the data term. The effect cannot be countered by simply tuning the weight of ∇w_z, since its strength would need to depend on the local scene depth.

4.2. Rigidity prior

Many scenes of practical importance consist mostly of rigid objects. For such scenes, piecewise rigidity is expected to be a significantly better prior than isotropic smoothness. To that end, we define the spatial term as

$$E_S(d, \mathbf{w}) = E_S^{\text{TV}}(d) + \mu E_S^R(\mathbf{w}), \tag{7}$$

where we replaced the total variation term for the 3D flow with the rigidity prior

$$E_S^R(\mathbf{w}) = \int_\Omega \psi\big(v^R(\mathbf{x}; \mathbf{w})\big)\, \mathrm{d}\mathbf{x}. \tag{8}$$

Here, $v^R(\mathbf{x}; \mathbf{w})$ denotes the non-rigid motion residual of flow $\mathbf{w}$ at point $\mathbf{x}$ in the reference frame, and $\psi(\cdot)$ is a robust function to reduce the influence of outliers.

In contrast to rigid motion priors that have been used in optical flow, which assume a globally rigid scene [16, 20], we here propose to use a local rigidity constraint, namely that the motion of small neighborhoods in the scene can be described by a locally (rather than globally) rigid motion. To that end, let $\mathcal{C}(\mathbf{x})$ denote a local region centered at a point $\mathbf{x} \in \Omega$, and $\mathbf{r}(\cdot; \mathbf{w}|_{\mathcal{C}(\mathbf{x})})$ the rigid motion component of the 3D flow $\mathbf{w}$ in that region. We then define the non-rigid motion residual at $\mathbf{x}$ as the squared deviation from the rigid motion component, integrated over the region:

$$v^R(\mathbf{x}; \mathbf{w}) = \int_{\mathcal{C}(\mathbf{x})} \|\mathbf{r}(\mathbf{y}; \mathbf{w}|_{\mathcal{C}(\mathbf{x})}) - \mathbf{w}(\mathbf{y})\|_2^2 \eta(\mathbf{x}, \mathbf{y})\, \mathrm{d}\mathbf{y}. \tag{9}$$

Here, $\eta(\mathbf{x}, \mathbf{y})$ is a weighting function to allow spatially varying weights within $\mathcal{C}(\mathbf{x})$. The key difference to global rigidity priors is twofold: *(1)* The rigidity assumption is less global, since the rigid motion $\mathbf{r}(\cdot; \mathbf{w}|_{\mathcal{C}(\mathbf{x})})$ is assumed to be valid only for a small area, and moreover estimated from the flow itself – see below; *(2)* at the same time the formulation is also less local, since it aggregates motion residuals over larger neighborhoods in a way loosely related to non-local total variation [8][1]. Note that by defining the non-rigid motion residual over an entire region, the constraint is propagated more vigorously over larger distances, because neighboring regions strongly overlap. The weights $\eta(\mathbf{x}, \mathbf{y})$ as well as the robust function $\psi(\cdot)$ reduce this propagation at motion discontinuities to allow for a piecewise rigid 3D flow field. The challenge in employing this regularizer is that one needs to at the same time estimate the rigid motion $\mathbf{r}(\cdot; \mathbf{w}|_{\mathcal{C}(\mathbf{x})})$ and measure the deviation from that motion.

Discretization. To simplify the following treatment, we spatially discretize the rigidity prior from Eq. (8) as

$$E_S^R(\mathbf{w}) = \sum_{c \in C} \psi\left(v_{\text{dsc}}^R(c; \mathbf{w})\right), \tag{10}$$

where $c \in C$ denote the overlapping regions (overlapping $n \times n$ patches) in the reference frame. The non-rigid motion residual itself is discretized as

$$v_{\text{dsc}}^R(c; \mathbf{w}) = \left\|\mathbf{r}(c; \mathbf{w}_{(c)}) - \mathbf{w}_{(c)}\right\|_{N_c}^2, \tag{11}$$

where $\mathbf{w}_{(c)}$ is the concatenation of all flow vectors in c, and $\mathbf{r}(c; \mathbf{w}_{(c)})$ is the rigid motion component of the flow patch $\mathbf{w}_{(c)}$. The term $\|\mathbf{v}\|_{N_c} = \sqrt{\mathbf{v}^T N_c \mathbf{v}}$ denotes a Mahalanobis distance with the diagonal matrix N_c performing local weighting analogous to η above. In the following, we will show how for small motions we can express the rigid motion component as a projection onto the closest rigid motion subspace, such that $\mathbf{r}(c; \mathbf{w}_{(c)}) = A_c \mathbf{w}_{(c)}$. This leads to the final definition of our non-rigid motion residual as

$$v_{\text{dsc}}^R(c; \mathbf{w}) = \left\|A_c \mathbf{w}_{(c)} - \mathbf{w}_{(c)}\right\|_{N_c}^2. \tag{12}$$

The key property of this proposed rigidity prior is that it allows measuring deviations from locally rigid motion without any explicit representation of rigid motion (in contrast to [11]). In the following we derive the projection A_c.

Rigid motion subspace. It is well known [6] that a small rigid motion can be represented well by a translation $\mathbf{t}$ and a linear approximation of a rotation:

$$R = I + \sin\alpha [\mathbf{r}]_\times + (1 - \cos\alpha)(I - \mathbf{r}\mathbf{r}^T) \approx I + \alpha [\mathbf{r}]_\times. \tag{13}$$

[1] The most closely related non-local TV regularizer [8] is given as $\int_\Omega \psi\left(\int_\Omega \|\mathbf{w}(\mathbf{x}) - \mathbf{w}(\mathbf{y})\|_2^2 \eta(\mathbf{x}, \mathbf{y})\, \mathrm{d}\mathbf{y}\right) \mathrm{d}\mathbf{x}$.

Here $\mathbf{r}$ represents the rotation axis, $[\mathbf{r}]_\times$ is the cross-product matrix, α the rotation angle, and I the identity matrix. We can thus approximate Eq. (11) as

$$v_{\text{dsc}}^R = \sum_{\mathbf{x}\in c} \left\| \left([\mathbf{P}^0(\mathbf{x})]_\times | I\right) \begin{pmatrix} \alpha\mathbf{r} \\ \mathbf{t} \end{pmatrix} - \mathbf{w}(\mathbf{x}) \right\|^2 \eta(c,\mathbf{x}), \quad (14)$$

where $\mathbf{P}^0(\mathbf{x})$ is the 3D surface point at pixel $\mathbf{x}$, which is computed from the current depth estimate (see Sec. 3.1), and $\eta(c,\mathbf{x})$ is the local weight. By concatenating the matrices $([\mathbf{P}^0(\mathbf{x})]_\times | I) \in \mathbb{R}^{3\times 6}$ for all $\mathbf{x} \in c$ into $M_c \in \mathbb{R}^{3n^2\times 6}$, we can rewrite the non-rigid motion residual as

$$v_{\text{dsc}}^R(c;\mathbf{w}) = \left\| M_c \begin{pmatrix} \alpha\mathbf{r} \\ \mathbf{t} \end{pmatrix} - \mathbf{w}_{(c)} \right\|_{N_c}^2 . \quad (15)$$

By solving this weighted least squares problem, we obtain the projection onto the closest rigid motion subspace:[2]

$$\begin{pmatrix} \alpha\mathbf{r} \\ \mathbf{t} \end{pmatrix} = \left(M_c^T N_c M_c\right)^{-1} M_c^T N_c \mathbf{w}_{(c)} \quad (16)$$

$$A_c = M_c (M_c^T N_c M_c)^{-1} M_c^T N_c. \quad (17)$$

The matrix that needs to be inverted to construct the projection operator A_c is small, $(M_c^T N_c M_c) \in \mathbb{R}^{6\times 6}$, so that the construction process can be done efficiently.

Weights. The weighting matrix N_c plays a central role in the robustness of the rigid motion fitting procedure. In general, we can only expect points on the same surface, close to each other to undergo the same rigid motion. Therefore we use a spatially varying weight

$$\eta(c,\mathbf{x}) \propto e^{-1/\lambda_s(c,\mathbf{x})} . \quad (18)$$

Here, λ_s is a similarity measure that measures the likelihood that the surface point $\mathbf{P}^0$ corresponding to $\mathbf{x}$ belongs to the same surface as the center of region c. The sum of the weights in a patch is normalized to one. To define the similarity measure λ_s one could employ several different features of the flow and depth field [*c.f.* 14, 23]. Currently we consider the differences of the optical flow field and the disparity between the pixel $\mathbf{p}_0^0 = (\mathbf{x};1)$ and the patch center $\mathbf{c}_0^0$. The similarity $\lambda_{s,d}$ penalizes disparity differences:

$$\lambda_{s,d}(c,\mathbf{x}) = \frac{\gamma}{\max(\gamma, \|(\mathbf{c}_0^0 - \mathbf{c}_1^0) - (\mathbf{p}_0^0 - \mathbf{p}_1^0)\|)} . \quad (19)$$

The similarity $\lambda_{s,\mathbf{w}}$ for the optical flow is defined in the same way. Finally, λ_s is given by the product of both individual similarities. The threshold γ is set such that the weight stays within $[0,1]$. An example weighting is shown in Fig. 2. More sophisticated similarity measures, *e.g.* taking into account color differences, will remain future work.

[2]Note that the fit can be interpreted as a step of an alternating minimization procedure for the 3D scene flow and the rigid motion parameters.

Even though the weights increase the robustness of the rigid motion fitting procedure, errors and outliers may still occur. The robust function ψ therefore ensures that fits with a lower error have a higher smoothing effect on their neighbors than patches in which no sensible rigid motion could be found, for instance around motion discontinuities. We use the Lorentzian $\psi(s) = \log(1 + \frac{s}{2\sigma^2})$ in all experiments.

5. Experimental Results

We have performed numerous experiments on synthetic as well as real data in order to assess the performance of our new rigidity prior. To demonstrate the benefits of the proposed local rigidity prior, we compare to total variation regularization, while using the identical data term and optimization procedure.

Implementation details. To alleviate the negative influence of varying illumination on the data term, we preprocess the input images using structure-texture decomposition [2], which has been used successfully for 2D optical flow estimation – for details see [21].

Since image gradients are only valid in a small neighborhood, the energy minimization is embedded in a hierarchical coarse-to-fine scheme to better avoid local minima. We use a downsampling factor of 0.9 throughout our experiments. At each pyramid level we run 4 outer iterations of our optimization framework to minimize the energy functional from Eq. (1). The parameters are set to $\mu = 15$, $\sigma = 0.003$ and $\lambda = 1/150$. The regularization parameter λ is increased linearly with the current image size. The threshold γ is set to $0.85/512$ times the length of the image diagonal. The gradients of the warped images are computed using bicubic interpolation. For computing the non-rigid motion residual we use a neighborhood of 5×5 pixels and place an overlapping region at every fourth pixel. For the synthetic scenes the depth is initialized to a distant plane, whereas for the real images we initialize the depth with a dense stereo algorithm [9]. The similarities $\lambda_{s,d}$ and $\lambda_{s,\mathbf{w}}$ are computed from the current solution.

Error measures. Basha *et al.* [3] observed that the evaluation methodology used for optical flow and stereo should not be directly carried forward to the evaluation of scene flow algorithms. We agree that the deviation between 2D and 3D errors is an important issue – in particular, very different 3D flow fields can have almost identical 2D projections (see also Fig. 2). For completeness we measure both the 3D error of the recovered surface and motion field, and the errors of the projected 2D motion field in the image.

For the scene depth we follow [3] and report the *normalized root mean squared error* NRMS_d, where the normalization is w.r.t. the difference between the maximal and minimal point distances to the reference camera.

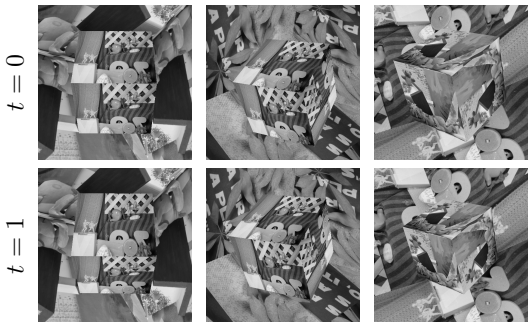

Figure 3. Three scenes from the *box* dataset (one per scenario). *(left) Rot* – pure rotation of the box, *(middle)* T_{xyz} – translations along all coordinate axes, *(right)* T_z – translation only in depth.

Similarly, for the 3D flow vectors we use the normalized end point error $\mathrm{NRMS}_\mathbf{w}$. However, other than [3] we prefer to normalize the flow vectors' endpoint errors with the diameter of the smallest sphere enclosing the motion field. The advantage of that definition is that vectors pointing in different directions are treated correctly, whereas the originally proposed normalization based on minimal and maximal magnitudes disregards differences in flow direction. Additionally, we also report the *average angular error* of the 3D flow vectors, $\mathrm{AAE}_\mathbf{w}$.

Finally, errors in the 2D projection of the motion field are quantified using common error metrics from the optical flow literature, namely the *average angular error* AAE and the *average end-point error* AEP.

5.1. Total variation vs. rigid prior

To evaluate the benefit of the rigid motion prior quantitatively, we have rendered 9 synthetic scenes with known ground truth. Each scene consists of two nested boxes that are textured with different images and undergo various types of rigid motion. The scenes are observed by two cameras. Fig. 3 shows a subset of the rendered scenes.

We note that with a TV motion prior our method becomes a re-implementation of [3] (up to minor differences), thus corresponds to the state of the art in 3D scene flow.

Our experiments clearly show the advantages of the proposed rigidity prior. It outperforms standard total variation in all cases, on different types of motion, especially in terms of 3D error – see Table 1.

When looking at the AEP, the gains in terms of projected 2D motion errors are relatively moderate, as predicted by our discussion above. In contrast to that the numbers for the AAE improve for two cases by roughly 25%, leading to the conjecture that noticeable differences in the projected flows mostly occur in areas with small or no motion.

Since both methods regularize depth using total variation, it is not surprising to see similar depth errors (NRMS_d). The true benefit of our approach becomes apparent when considering the error of the estimated 3D motion field: the rigid motion prior lowers the $\mathrm{NRMS}_\mathbf{w}$ by 8–53%. The angular error $\mathrm{AAE}_\mathbf{w}$ is lowered by 47% for the rotation examples, and even by 50–70% for the translation examples. Averaged over both 3D metrics and all scenarios, the rigid motion prior reduces the 3D motion estimation error by 42% (see Fig. 4 for a visual illustration.)

Table 1. Proposed local rigidity prior for the 3D motion field (*Rig*) *vs.* standard total variation (*TV*). Each error is averaged over three scene instances (per scenario); 2D errors are further averaged over all four projections.

Scene		3D Error $\mathrm{AAE}_\mathbf{w}$	3D Error $\mathrm{NRMS}_\mathbf{w}$	3D Error NRMS_d	2D Error AAE	2D Error AEP
Rot	Rig	**4.5°**	**7.3%**	11.7 %	**1.6°**	0.36
	TV	8.5°	9.8%	**11.6%**	1.7°	**0.35**
T_{xyz}	Rig	**2.5°**	**11.9%**	11.8 %	**1.5°**	**0.39**
	TV	8.6°	25.6 %	**11.7%**	2.3°	0.42
T_z	Rig	**3.9°**	**14.0%**	**9.9%**	**1.8°**	**0.35**
	TV	7.8°	15.3%	10.7 %	2.4°	0.37

Table 2 shows the evaluation results if errors in occlusion areas and around discontinuities are masked out, in the spirit of the Middlebury benchmark. As expected, all methods are inaccurate especially around discontinuities. However, the TV regularizer propagates the errors far into the smooth surfaces, hence masking occlusion and discontinuity areas brings only small improvements. On the contrary, the results for the rigid motion prior improve by a factor of 2–4, indicating that indeed the errors are mostly confined to the occlusion areas and discontinuities.

Qualitative Results. In Fig. 4 (left) we show the scene flow for one of the the T_z scenes – other cases are similar. While the rigidity prior recovers the correct motion pattern (aside from errors at the discontinuities), the result of TV regularization exhibits a clear systematic error. The flow difference of w_z in viewing direction is under-estimated, whereas the (w_x, w_y)-components show an incorrect expansion pattern in order to compensate for the missing part of the observed flow in the images. The T_{xyz} scenario exhibits a similar, even more irregular effect (see Fig. 4, right).

Table 2. As expected the error improves significantly if areas with occlusions (OC) and discontinuities (DC) are *omitted* from the evaluation. In the remaining areas, the rigid motion prior exhibits an even greater advantage over TV regularization.

Scene		$\mathrm{AAE}_\mathbf{w}$ [°] —	$\mathrm{AAE}_\mathbf{w}$ [°] OC	$\mathrm{AAE}_\mathbf{w}$ [°] OC&DC	$\mathrm{NRMS}_\mathbf{w}$ [%] —	$\mathrm{NRMS}_\mathbf{w}$ [%] OC	$\mathrm{NRMS}_\mathbf{w}$ [%] OC&DC
Rot	Rig	**4.5**	**4.1**	**3.3**	**7.3**	**6.7**	**4.5**
	TV	8.5	8.2	7.4	9.8	9.4	7.9
T_{xyz}	Rig	**2.5**	**2.1**	**1.5**	**11.9**	**10.3**	**6.4**
	TV	8.6	8.0	7.7	25.6	24.4	23.6
T_z	Rig	**3.9**	**2.5**	**1.2**	**14.0**	**10.6**	**5.0**
	TV	7.8	6.5	4.9	15.3	13.3	8.2

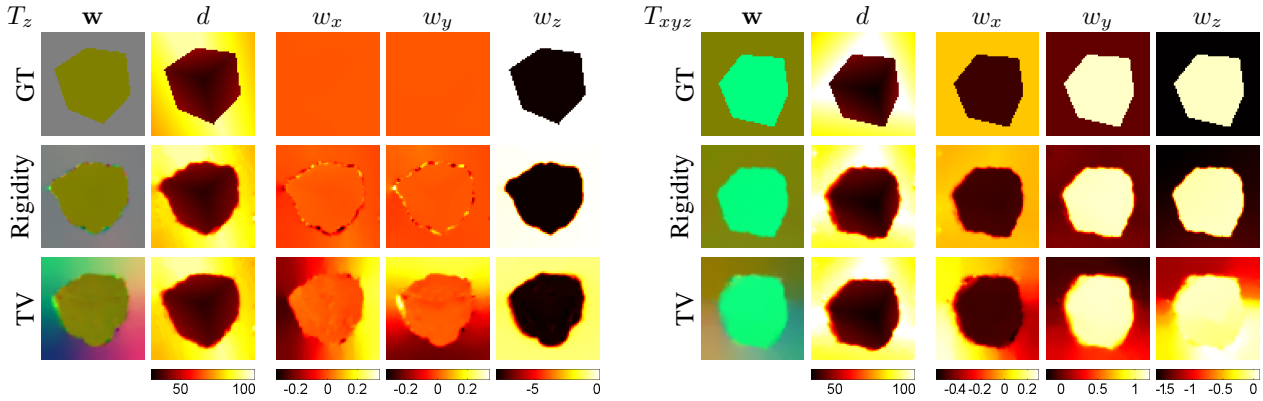

Figure 4. Results on two example scenes from scenarios T_z (left) and T_{xyz} (right). The plots show (from top to bottom) the ground truth, and the estimates with the rigidity and TV priors. *1st column:* 3D flow field normalized per coordinate (red: w_x, green: w_y, blue: w_z). *2nd column:* depth, *3rd-5th column:* individual components of the 3D flow as heatmaps. For T_z (left) one can clearly see the tendency of the TV regularizer to underestimate the motion difference of w_z in viewing direction and to compensate the resulting data error with wrong (x, y)-motion. The T_{xyz} scene (right) shows that although the error in the image plane is small (Table 1), the 3D motion is significantly misjudged by TV regularization, whereas the rigid motion prior manages to reconstruct the correct scene flow.

5.2. Comparison with 2D scene flow

To compare our algorithm with other scene flow algorithms [10, 17, 22] that have only been evaluated in 2D, we run it on the synthetic sphere sequence of [10], consisting of four images of two independently rotating hemispheres. Table 3 shows the errors of the 2D flow vectors and the disparity (3D flow errors are not available). Our method delivers comparable 2D errors. The tighter coupling between depth and flow in the rigidity prior leads to some artifacts at the extreme depth variations on the sphere's silhouette, which currently prevent it from achieving better results.

5.3. Real world data

To complement our quantitative experiments, we have tested the algorithm on several real world scenarios. Unfortunately, no real-world datasets with ground truth are available at present. We give three examples. The first and second scene were captured with a stereo rig, while the third was acquired by three cameras.

The first dataset (see Fig. 5, left) has three independently moving objects on a static background. The book and the box are rotated counter-clockwise, the toy cheetah is pushed to the left and slightly rotated clockwise. Note that although we use stereo for initialization, we did not rectify the images. As far as one can tell by visual inspection, both the shape and the objects' motions are recovered correctly.

In Fig. 5 (right) we show a reconstruction of a street scene from [18]. The images show two rigidly moving cars and a non-rigidly moving pedestrian, and have large textureless regions as well as complex occlusions. The frames were acquired from a moving car, such that flow in z-direction is observable everywhere except in the far distance. Our method is able to capture the non-rigid motion of the pedestrian, including the feet, and the motion of both cars (note, the left car is moving in the same direction as the cameras, hence the motion vectors are not well visible).

Finally, we show results for the *"Maria"* sequence from [3]. The scene shows a rotating face, and includes non-rigid motion of the hair, as well as large occlusion areas. The results displayed in (Fig. 6) visually appear to be correct.

Table 3. 2D errors for the *"sphere"* sequence [10].

	[17]	[10]	[22]	Rig
RMSE 2D Flow	0.63	0.69	0.77	0.75
RMSE Disparity	3.8	3.8	10.9	5.6

6. Conclusion

We have shown that standard smoothness priors from the 2D motion estimation literature lead to biases when applied to 3D scene flow estimation. To address this issue, we presented a method for regularizing 3D scene flow computation by penalizing deviations from the local rigidity of the motion and integrated it into an energy minimization framework. Our experiments on several different datasets demonstrated significant reductions of the 3D motion estimation error compared to standard total variation regularization, and showed the applicability to real-world scenes with articulated motion.

In future work we plan to improve the weighting scheme for the rigidity constraint, and to extend the method to sequences of more than two frames, for which the rigidity constraint is likely to be even more valuable.

References

[1] G. Adiv. Determining three-dimensional motion and structure from optical flow generated by several moving objects.

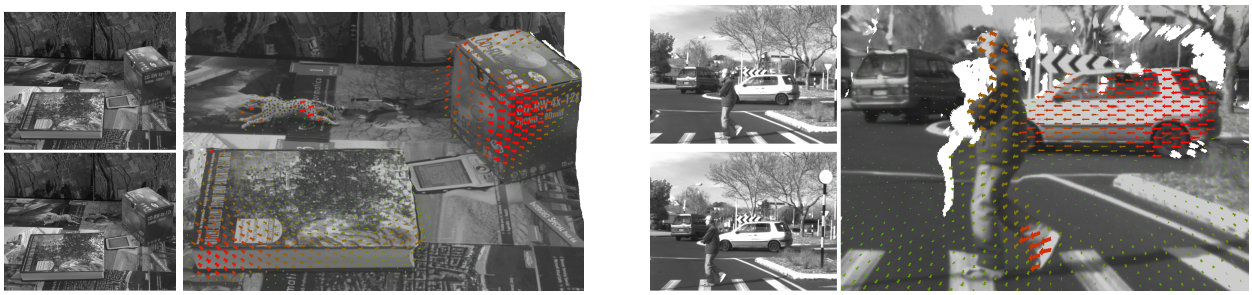

Figure 5. Results on two data sets: *(left)* "table" example; *(right)* "roundabout" scene. Each example: *(small)* one of the two image pairs; *(large)* reconstructed mesh with overlayed flow vectors (flow magnitude increases from green to red).

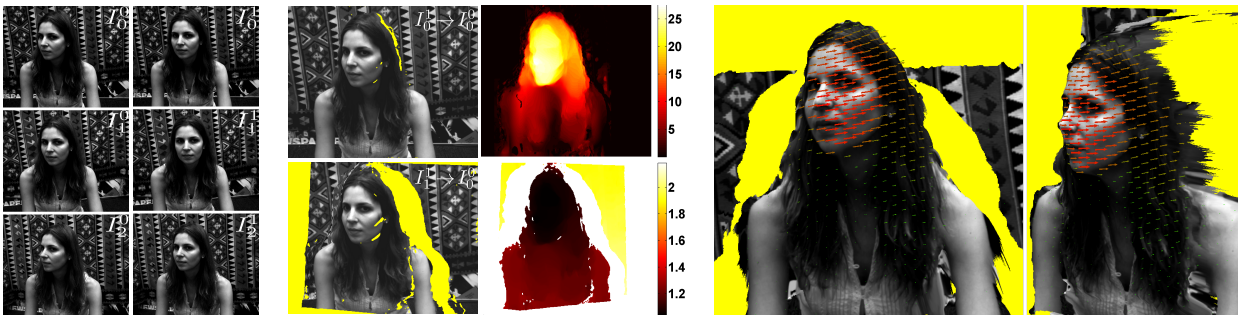

Figure 6. Results on the *"Maria"* data set. *(left)* Input images from three cameras at two time steps. The reference camera is shown on top. *(middle, clockwise from lower left)* Two warped images (yellow denotes occlusions), estimated flow magnitude [mm], and estimated scene depth [m]. *(right)* Reconstructed 3D surface and motion field.

PAMI, 7(4):384–401, 1985.

[2] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition – Modeling, algorithms, and parameter selection. *IJCV*, 67(1):111–136, 2006.

[3] T. Basha, Y. Moses, and N. Kiryati. Multiview scene flow estimation: A view centered variational approach. *CVPR'10*.

[4] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. *ECCV*, 2004.

[5] R. L. Carceroni and K. N. Kutulakos. Multi-view scene capture by surfel sampling: From video streams to non-rigid 3D motion, shape and reflectance. *IJCV*, 49:175–214, 2002.

[6] J. Q. Fang and T. S. Huang. Solving three-dimensional small-rotation motion equations: Uniqueness, algorithms, and numerical results. *CVGIP*, 26(2):183–206, 1984.

[7] Y. Furukawa and J. Ponce. Dense 3D motion capture from synchronized video streams. *CVPR*, 2008.

[8] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. *Multiscale Model. Simul.*, 7, 2008.

[9] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE TPAMI*, 30(2):328–341, 2008.

[10] F. Huguet and F. Devernay. A variational method for scene flow estimation from stereo sequences. *ICCV*, 2007.

[11] T. Nir, A. Bruckstein, and R. Kimmel. Over-parameterized variational optical flow. *IJCV*, 76(2):205–216, 2008.

[12] J.-P. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *IJCV*, 72(2):179–193, 2007.

[13] C. Rabe, T. Müller, A. Wedel, and U. Franke. Dense, robust, and accurate motion field estimation from stereo image sequences in real-time. *ECCV*, 2010.

[14] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. *CVPR*, 2010.

[15] W. Trobin, T. Pock, D. Cremers, and H. Bischof. An unbiased second-order prior for high-accuracy motion estimation. *DAGM*, 2008.

[16] L. Valgaerts, A. Bruhn, and J. Weickert. A variational model for the joint recovery of the fundamental matrix and the optical flow. *DAGM*, 2008.

[17] L. Valgaerts, A. Bruhn, H. Zimmer, J. Weickert, C. Stoll, and C. Theobalt. Joint estimation of motion, structure and geometry from stereo sequences. *ECCV*, 2010.

[18] T. Vaudrey, C. Rabe, R. Klette, and J. Milburn. Differences between stereo and motion behaviour on synthetic and real-world stereo sequences. *IVCNZ*, 2008.

[19] S. Vedula, S. Baker, R. Collins, T. Kanade, and P. Rander. Three-dimensional scene flow. *CVPR*, 1999.

[20] A. Wedel, D. Cremers, T. Pock, and H. Bischof. Structure- and motion-adaptive regularization for high accuracy optic flow. *ICCV*, 2009.

[21] A. Wedel, T. Pock, C. Zach, D. Cremers, and H. Bischof. An improved algorithm for TV-L1 optical flow. *Proc. of the Dagstuhl Motion Workshop*. Springer, 2008.

[22] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers. Efficient dense scene flow from sparse or dense stereo data. *ECCV*, 2008.

[23] M. Werlberger, T. Pock, and H. Bischof. Motion estimation with non-local total variation regularization. *CVPR*, 2010.

[24] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L1 optical flow. *DAGM*, 2007.