A Comparison of Scene Flow Estimation Paradigms

Iraklis Tsekourakis[1] and Philippos Mordohai[1]

Stevens Institute of Technology
Hoboken NJ 07030, USA

Abstract. This paper presents a comparison between two core paradigms for computing scene flow from multi-view videos of dynamic scenes. In both approaches, shape and motion estimation are decoupled, in accordance to a large segment of the relevant literature. The first approach is faster and considers only one optical flow field and the depth difference between pixels in consecutive frames to generate a dense scene flow estimate. The second approach is more robust to outliers by considering multiple optical flow fields to generate scene flow. Our goal is to compare the isolated fundamental scene flow estimation methods, without using any post-processing, or optimization. We assess the accuracy of the two methods performing two tests: an optical flow prediction, and a future image prediction, both on a novel view. This is the first quantitative evaluation of scene flow estimation on real imagery of dynamic scenes, in absence of ground truth data.

1 Introduction

Scene flow is the 3D motion field that generates the optical flow when projected onto the image plane of a camera. It is arguably the last frontier that remains unexplored in 3D computer vision, even though the problem was formulated a relatively long time ago [1, 2] and there are exciting applications, such as free-viewpoint video, motion capture, augmented reality and autonomous driving/driver assistance. The primary reasons for this are the inherent difficulty of the problem and the lack of data with ground truth that would aid the development of algorithms.

For over a decade since the original publication that defined the field [1], the relevant literature consisted of a sparse set of papers [3–9] that seemed well-positioned to spark a breakthrough, which has not come yet. It took a breakthrough in sensing technology with the advent of consumer depth cameras (RGB-D cameras, such as the Microsoft Kinect) to enable reliable scene flow estimation from RGB-D monocular sequences [10–14]. RGB-D cameras, however, suffer from their own limitations, namely their inability to operate outdoors due to sunlight interference and their short range. Moreover, monocular inputs hinder free-viewpoint video applications since texture information is not available for even the slightest viewpoint changes, with a few exceptions [12, 13]. Collet et al. [15] present outstanding results on free-viewpoint video using a dense set of RGB and IR sensors. It is desirable, therefore, to achieve high-quality scene flow estimation from two or more passive cameras, since such a configuration would not suffer from these limitations. Our inputs are multi-view videos acquired by synchronized, calibrated, stationary cameras [16, 17]. The synchronized and calibrated requirements for the cameras can be relaxed, but this is currently out of scope.

In this paper, we focus on point-wise methods that estimate scene flow in a decoupled fashion, by alternating between depth and motion estimation. This choice is not necessarily an endorsement of this type of scene flow estimation. Other strategies have different strengths and weaknesses: joint estimation of both shape and motion leads to higher consistency at a significant computational cost; 3D shape tracking does not allow the shape to be modified based on temporal constraints; patch-based estimation allows more complex reasoning and regularization, but may impose too much rigidity. All these approaches have attractive features, but we believe that our analysis should start with the fundamental case of estimating the depth and scene flow of each pixel.

A pixel with depth and scene flow estimates can be linked to corresponding pixels in different views at the same time instant via depth, to the corresponding pixel in the same view at the next time instant via optical flow, derived from scene flow, and to corresponding pixels in different views at the next time instant via depth and scene flow. We evaluate two viewpoint-based paradigms for estimating scene flow from depth maps and optical flow fields:

- a more common approach, which we term Optical Flow and Depth Difference (***OF+D***), in which scene flow is decomposed in two components: one parallel and one orthogonal to the image plane,
- an approach, we term Multiple Optical Flows (***MOF***), which combines multiple optical flows at each reconstructed 3D point to derive its scene flow.

Throughout this effort, we ensure that the two paradigms are tested on identical inputs under identical conditions to the extent possible.

Besides the difficulty of the problem itself, the second challenge to our research is the lack of suitable data with ground truth for quantitative evaluations. This is due to the lack of a convenient technique for generating ground truth motion fields, in 2D or 3D [18, 19]. This has forced most authors to evaluate their methods on synthetic or static data. To overcome this obstacle, we perform two types of quantitative evaluation using frames from a sequestered camera in lieu of ground truth. In the first evaluation, we attempt to predict the optical flow of the validation camera by projecting the estimated scene flow. In the second evaluation, we attempt to predict the RGB image at time $t+1$ given RGB images and estimated scene flow for the other cameras at time t. See Fig. 1. According to all criteria, *MOF* is superior to *OF+D*. Moreover, the difference between its image predictions and predictions using the actual data of the validation camera is small.

In summary, the contributions of this work are:

- the *MOF* algorithm, which is a robust extension of the work of Vedula et al. [1],
- a comprehensive comparison of two viewpoint-based scene flow estimation paradigms, and
- novel criteria for evaluating scene flow estimation in the absence of ground truth scene flow.

The conclusions that can be drawn from our experiments are: (i) it is important to not neglect optical flows from additional views besides the reference view, if they are available, since using them can lead to significant noise reduction; and (ii) novel view

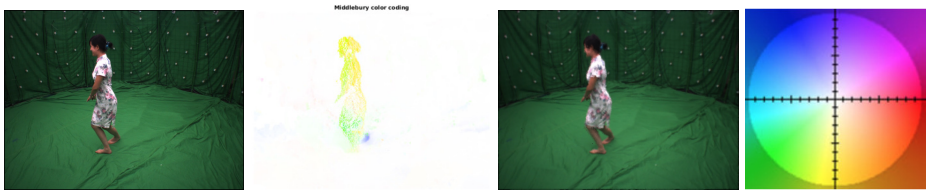

Fig. 1. Left to right: Input image from a sequestered camera [17] at time instant t (left), visualization of predicted flow using *MOF*, and prediction of RGB at time *t+1*, and Middlebury color coding [18].

prediction using the *MOF* algorithm is very similar in quality to predicted frames using data from the validation camera itself. This is an indication that scene flow has been estimated well.

2 Related Work

In this section, we review the scene flow literature. We categorize prior work according to whether the shape and motion estimation is joint or decoupled, whether they operate on pixels or surfaces, and whether they consider the topology of the shape fixed throughout the sequence. We include methods operating on RGB-D inputs, but not methods that segment N rigid bodies, e.g., [20].

We begin with joint shape and motion estimation. In the paper that defined the term, Vedula et al. [1] formulate and analyze three algorithms depending on the degree to which scene geometry is known. In a separate paper, Vedula et al. [21] extend space carving [22] to the 6D space of all shapes and flows, resulting in a tight approximation to shape and scene flow. Other joint estimation approaches that consider optical flow fields or temporal derivatives in multiple images have represented spatio-temporal shape using subdivision surfaces [4], surfels [3], level sets [23], watertight meshes [24], probabilistic occupancy-motion grids [25], and spatiotemporal orientation distributions [26].

Viewpoint-based methods for joint scene flow estimation include variational methods [5, 7, 27–29]. Typically, the shape is initialized by stereo matching and then shape and motion are jointly estimated resulting in convergence to the nearest local minimum of an appropriate energy function. Other viewpoint-based methods explore the high dimensional search spaces using Markov Random Fields [30], winner-take-all or dynamic programming [6], or by growing correspondence seeds [31].

An alternative approach is to segment the reference image in a set of patches and estimate their 3D motion from frame to frame [32–34]. Unlike points that only allow a 3D displacement estimation, patches allow the rotational motion estimation and also provide a basis for regularization. A more recent approach by Vogel et al. [9] models occlusion and relationships among neighboring patches.

To reduce computational complexity, authors decouple shape and motion estimation [8, 35–37]. Wedel et al. [8] take into account stereo image pairs from two consecutive times and compute both depth and 3D motion associated with each point in the image.

This is equivalent to our *OF+D* paradigm. On the other hand, Li et al. [38] extract a watertight mesh from point clouds reconstructed by variational stereo and address scene flow as volumetric deformation using [1] to estimate scene flow at each vertex.

If the topology of the shape can be recovered in the first frame and remains fixed throughout the sequence, methods that can track the shape over time have been proposed. The primitives to be tracked can be meshes [15, 39–41], patches [42, 43], volumetric elements [44, 45] or sparse features followed by motion propagation to the rest of the mesh [46–51].

Recently, the emergence of consumer depth cameras has lead to the development of decoupled depth and motion estimation methods leveraging the accurate depth provided by these sensors. Due to the availability of a single viewpoint, most of these methods fall under the *OF+D* paradigm [10, 11, 52–54] Methods that operate on patches [55], layers [14], meshes [12], local twist motions [56] and 6D volumetric motion fields [13] have also been reported, but in all cases a single flow field is available.

None of these publications includes quantitative evaluation on videos of dynamic scenes. Some show results on synthetic [5] or static scenes [57]. The KITTI benchmark [58], used by [9], depicts static scenery. The exception is the work of Sizintsev and Wildes [59, 26] who present quantitative evaluation of scene flow using a motorized stage to generate ground truth. Ground truth is acquired using structured light sensors on stop motion sequences. While this study is unprecedented and valuable, the experimental setup is not ideal since the fiducial markers that are placed on each independently moving surface to aid motion estimation also aid the algorithms being evaluated.

3 Problem Statement

The objective of this work is to compare two methods that isolate the fundamental 3D scene flow estimation of dynamic scenes captured by multiple, calibrated, stationary, synchronized cameras without any form of post-processing or optimization. Throughout we use the term *view* to indicate a viewpoint, or a specific camera, and *frame* to denote an image taken at a specific time t. We implement two different methods. They both need 2D optical flows and depth estimates as inputs. In order to compute the depths, we generate the cost volume using plane-sweeping stereo [60] and we extract the final depth estimates using SGM [61] on the cost volume. We compute optical flows for all cameras using the software of Sun et al. [62].

The first method is *OF+D*. It is a straightforward implementation of decoupled 3D shape and motion estimation. Given an optical flow estimate for a pixel **u**, we can approximate the scene flow that corresponds to the 3D point that pixel **u** projects to, as the summation of the optical flow vector with the depth difference between pixel **u** at time t and pixel **u'** at time $t + 1$. **u'** is the location of pixel **u** according to the optical flow. The details are presented in Section 4. Second, we propose the *MOF* method that estimates the 3D scene flow using multiple 2D optical flows and depth estimates. An optical flow estimate provides two linear constraints on the three unknowns in the scene flow (V_x, V_y, V_z of the 3D motion). If we have two or more views of the same part of the scene that are not coplanar, we can recover the scene flow. Assuming we have more than two cameras, we apply MSAC [63] to remove outlying optical flows. We then use

the method of Vedula et al [1, 2] as a solver on the selected candidate optical flows in order to extract the scene flow. Implementation details are given in Section 5.

Due to lack of publicly available multi-view datasets with scene flow ground truth, we evaluate the accuracy of the estimated scene flows for both methods first by comparing the projected scene flow on a novel view, that was excluded by all steps of the scene flow estimation, with the optical flow estimated using the data of the novel view. Second, we measure the quality of predicted frames on the novel view at time $t + 1$ generated from frames at time t using the projected scene flow. While novel view synthesis as an evaluation metric is forgiving in textureless areas, it has been shown to be an effective evaluation strategy in general [64–66] and specifically for free viewpoint video [67].

Inputs The depth maps used as inputs for both methods, *OF+D* and *MOF*, for the *ballet* and *breakdance* datasets were provided by the creators of the videos [16]. For *Cheongsam* and *Redskirt* [17], depths are computed in a multi-view configuration (See Section 7 for details on the datasets used). The depth estimation combines the plane-sweeping algorithm with Semi-Global matching (SGM) optimization. In plane sweeping stereo, we define a family of planes parallel to the image plane of the *reference view*. For each pixel, depth hypotheses are formed by intersecting the corresponding ray with the set of planes. We then define a square window centered at that pixel in the reference view and warp it to the *target views* using the homographies from the reference view to the target views through the current plane. We compute the normalized cross-correlation (NCC) between the window on the reference view and each warped window on the target views, and store the average as the likelihood of assigning to the pixel the depth corresponding to the current plane. Target images in which the matching window falls out of bounds are excluded. The likelihood volume is converted to a cost volume by negating the NCC scores. SGM is used for extracting a depth map that approximately optimizes a global two-dimensional energy function by combining 1D minimization problems in 8 or 16 directions. We use eight paths for dynamic programming and 256 discretized depths per pixel. We use the rSGM implementation provided by Spangenberg et al. [68]. The second input used for the methods of this work are the optical flow estimates. They are computed using the software of Sun et al. [62], which is a modern implementation of the Horn and Schunck model [69].

4 Scene Flow estimation using Optical Flow and Depth Difference (OF+D)

In this section, we describe multi-view scene flow estimation, using the optical flow for a single camera, and the depth difference between two consecutive frames for the same camera. The 2D optical flow is the projection of the scene flow onto the images. Respectively, the back-projected 2D optical flow onto the 3D space at the depth of the 3D point, represents the scene flow that is parallel to the image. What is missing to complete the scene flow vector is the change of depth, which is perpendicular to the image plane.

We use as inputs the optical flow OF_t, and the depths D_t and D_{t+1} of the current and the next frame for the reference camera. For each pixel *(u,v)*, we project the 2D optical flow vector onto the reconstructed 3D point **x** at the depth $D_t(u,v)$. The resulting vector $SF_p(\boldsymbol{x})$ represents the scene flow component parallel to the image. In order to compute the change of depth, we first need to estimate the depth of the same point in the second frame. *(u,v)* at time $t+1$ will be located at $(u + OF_t(u,v,1), v + OF_t(u,v,2))$. Given that the target image location does not have integer pixel coordinates, we estimate the depth by applying bilinear interpolation. The vector $SF_o(\boldsymbol{x})$, which is orthogonal to the camera plane has norm equal to the difference of the depths:

$$\begin{aligned} |SF_o(\mathbf{x})| =& D_{t+1}(u + OF_t(u,v,1), v + OF_t(u,v,2)) \\ & - D_t(u,v) \end{aligned} \tag{1}$$

Finally, the scene flow estimate for the 3D point **x** that corresponds to the initial pixel *(u,v)* at time *t* is computed as:

$$SF(\mathbf{x}) = SF_p(\mathbf{x}) + SF_o(\mathbf{x}) \tag{2}$$

5 Scene Flow estimation using multiple Optical Flows (MOF)

This section describes the second scene flow estimation paradigm. The inputs to *MOF* are the optical flows of the reference and the neighboring cameras on the left and on the right of the reference camera and the depths of the reference cameras. These are computed as described in 3. Depending on the configuration of each of the multi-view datasets tested during this project, we use optical flows for 1 or 2 cameras on each side of the reference camera. The main idea is that since we have a number of 2D optical flows from different viewpoints, we can leverage them in order to estimate scene flow accurately.

Based on the work of Vedula et al [1, 2], the relationship of a 3D point $\mathbf{x} = (x,y,z)^T$ and the 2D image coordinates $\mathbf{u_i} = (u_i, v_i)^T$ of its projection in a camera C_i is described by the following formulas:

$$u_i = \frac{[\mathbf{P}_i]_1 (x,y,z,1)^T}{[\mathbf{P}_i]_3 (x,y,z,1)^T}, \tag{3}$$

$$v_i = \frac{[\mathbf{P}_i]_2 (x,y,z,1)^T}{[\mathbf{P}_i]_3 (x,y,z,1)^T}, \tag{4}$$

where $[\mathbf{P}_i]$ is the *j*th row of the 3 x 4 projection matrix $\mathbf{P}_i$ of camera C_i. If we know the 3D geometry, at time t, the differential relationship between **x** and $\mathbf{u_i}$ is represented by a 2 x 3 Jacobian $\frac{\eth \mathbf{u_i}}{\eth \mathbf{x}}$. This Jacobian describes the relationship between a small change in the point on the surface and its image in camera i. Now, if $\mathbf{x} = \mathbf{x}(t)$ is the 3D path of a point in the world, its scene flow is $\frac{d\mathbf{x}}{dt}$. The image of this point in camera C_i is $\mathbf{u_i} = \mathbf{u_i}(t)$. The relationship between the optical flow and the scene flow is given by:

$$\frac{d\mathbf{u}_i}{dt} = \frac{\eth \mathbf{u_i}}{\mathbf{x}} \frac{d\mathbf{x}}{dt} \tag{5}$$

This equation shows how optical flow can be computed if the scene flow is known. On the other hand, an optical flow estimate provides two linear constraints on the three unknowns in the scene flow. Thus, if we have two or more non-parallel cameras viewing the same part of the scene, the scene flow can be recovered. We use all the available optical flows in order to get more accurate scene flow estimates. The method of Vedula et al. gives the solution that minimizes the sum of least squares of the error obtained by reprojecting the scene flow onto each of the optical flows.

Given $N >= 2$ cameras observing the same surface of the scene, we can solve for $\frac{d\mathbf{x}}{dt}$ by setting up the system of equations $\mathbf{B}\frac{d\mathbf{x}}{dt} = \mathbf{U}$, where

$$B = \begin{bmatrix} \frac{\eth u_1}{\eth x} & \frac{\eth u_1}{\eth y} & \frac{\eth u_1}{\eth z} \\ \frac{\eth v_1}{\eth x} & \frac{\eth v_1}{\eth y} & \frac{\eth v_1}{\eth z} \\ . & . & . \\ . & . & . \\ \frac{\eth u_N}{\eth x} & \frac{\eth u_N}{\eth y} & \frac{\eth u_N}{\eth z} \\ \frac{\eth v_N}{\eth x} & \frac{\eth v_N}{\eth y} & \frac{\eth v_N}{\eth z} \end{bmatrix}, U = \begin{bmatrix} \frac{\eth u_1}{\eth t} \\ \frac{\eth v_1}{\eth t} \\ . \\ . \\ \frac{\eth u_N}{\eth t} \\ \frac{\eth v_N}{\eth t} \end{bmatrix}. \tag{6}$$

However, in our problem there are outliers that lead to high errors in the resulting scene flow. To prevent these outliers from corrupting the least squares estimate, we apply the m-estimator sample consensus (MSAC) algorithm [63] to find the set of inlying optical flows that minimize the maximum likelihood error, starting from minimal samples of two optical flows. The selected inliers are then used in the least squares method (Eq. 6) in order to extract the scene flow. The capability of MSAC to discriminate between strong and weak inliers, compared to RANSAC that does not discriminate, leads to significant improvement in the solutions.

6 Evaluation Methodology

In the absence of ground truth, we perform two tests to evaluate the different scene flow estimation techniques. As a first test, given depth, we project the estimated 3D scene flow onto a novel view and compute the errors of the projected optical flow compared to the optical flow estimated using the data of the novel view. For the second test, we use the color prediction error in a novel view. The novel view prediction error was initially proposed by Szeliski [64] and later by other authors [19, 65–67]. In all cases, we use a completely separate *validation camera* for evaluation and completely exclude its frames from optical flow and depth estimation. We always choose an *extrapolating view* with regards to each reference camera and its neighboring cameras used for the computation of scene flow, for validation so that errors are more pronounced in it. According to Szeliski [64], synthesizing extrapolating views is more challenging due to increased sensitivity to depth and other errors.

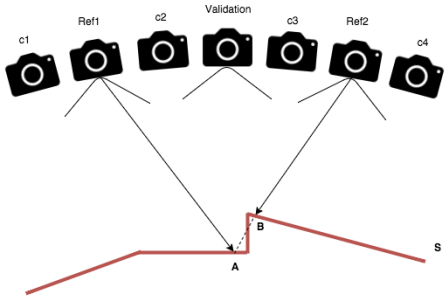

Fig. 2. An example of the multi-view setup used. S is the observed surface. For both sides of the validation camera we use a reference camera with two supporting cameras for the optical flow and depth estimation. Ref1 has c1 and c2 as supporting cameras, and Ref2 has c3 and c4. The importance of having this symmetric setup is illustrated by the fact that occlusions in one reference camera can be recovered by the symmetric reference camera. For example, point A is occluded in *Ref2*, but the correct flow at A in the validation camera can be projected from *Ref1*.

Yet, if we compute the scene flow of a reference camera on the one side of the validation camera, this creates an asymmetry in the noise of the projected optical flow on the opposite side of the validation camera. Thus, we compute the scene flow based on a second reference camera on the other side of the validation camera leading to more accurate and symmetric results. The configuration used can be seen in Fig. 2. It can be easily seen that occlusions could create noise in the estimation of the scene flow if only one reference camera is used. For example, in Fig. 2, $Ref1$ can see point A, that is what the validation camera sees as well, while A is occluded in $Ref2$. $Ref2$ sees point B instead. Figure 3 shows the importance of symmetry using data from the ballet sequences. All optical flows of the figure are shown in the validation camera. The left and right subfigures show the projected scene flows estimated for reference cameras on the left and right of the validation camera respectively. It is obvious that there are white parts (missing scene flow estimates) due to occlusions on each side of the dancer. Using both reference cameras lead to more accurate and dense scene flow estimates.

Fig. 3. Symmetric reference cameras on ballet data. Left to right: projected scene flow estimated from a reference camera to the left of validation camera, optical flow estimated for the validation camera using the software of Sun et al. [62], projected scene flow estimated from a reference to the right of validation camera, and Middlebury color coding [18].

"Ground truth" optical flow: In order to have a comparison for both tests, we estimate the optical flow of the novel view using the data of this "excluded" validation camera. We name this "ground truth" optical flow or *OF*.

6.1 Novel view optical flow prediction test

We project the 3D scene flow onto the validation camera in order to predict the optical flow for this excluded view. The resulting 2D optical flow is estimated without the use of the validation view data. We compare the projected optical flow from the *MOF* and *OF+D* estimated scene flows with the "ground truth" optical flow. The hypothesis here is that if the projected optical flow is similar to the "ground truth" optical flow, then the scene flow estimates are accurate and consistent between different views. When we project the scene flow to the validation view, it falls at non-integer pixel locations. In order to overcome this issue, we apply a bilateral filter, based on the implementation of Yoon and Kweon [70], to get a dense optical flow projection.

For each integer pixel location in the validation view, we find the k nearest 3D points that project close to the pixel. We then apply the bilateral filter, as the weighted summation of the neighbors based on color similarity and proximity, to compute optical flow at integer pixel coordinates. Let p and q be the target pixel and the neighbor pixel to be weighted respectively. We do not use data (colors) from the novel view in any step of this process. In order to represent the color of the target pixel, we find the 3D points that project close to the target pixel, and we use the color information of the nearest neighbor. Now p has been assigned a color and Δc_{pq} and Δg_{pq} represent the color difference and the spatial distance between pixels p and q. $f(\Delta c_{pq}, \Delta g_{pq})$ represents the strength of grouping by similarity and proximity. The color difference is computed as the Euclidean distance between two colors in RGB space.

$$\Delta c_{pq} = \sqrt{(R_p - R_q)^2 + (G_p - G_q)^2 + (B_p - B_q)^2} \tag{7}$$

The strength of grouping by color similarity is defined using the Laplacian kernel as

$$f_s = exp(-\frac{\Delta c_{pq}}{\gamma_c}) \tag{8}$$

where γ_c is determined empirically. Correspondingly, the strength of grouping by proximity is defined using the Laplacian kernel as

$$f_p = exp(-\frac{\Delta g_{pq}}{\gamma_p}) \tag{9}$$

and γ_p is also determined empirically. According to (8) and (9), the weights can be written as

$$w(p,q) = k \cdot exp(-\frac{\Delta c_{pq}}{\gamma_c} - \frac{\Delta g_{pq}}{\gamma_p}) \tag{10}$$

where *k* is a normalizing factor. Applying the bilateral filter described by the equations (7-10) we obtain dense optical flow. In Section 7, we present quantitative results using the two of the most commonly used measures of flow accuracy, namely the endpoint (pixel distance) and angular errors, as defined in [18].

6.2 Novel view future image prediction test

For the second test, given a current frame of the validation camera, we use the projected optical flow, densified as above, to predict the RGB values of the image in the next frame. We need to apply a filter again to predict RGB values at integer pixel locations of the target frame at time $t+1$. We once again find the nearest neighbors for each pixel and we average the optical flow of the neighbors with weights based on proximity. Equation (10) this time is modified to use proximity only as

$$w(p,q) = k \cdot exp(-\frac{\Delta g_{pq}}{\gamma_p}) \tag{11}$$

Following this process, and given a current frame of the validation camera, we get a dense RGB prediction of the next frame. We then use the Manhattan distance in RGB over all pixels between the predicted image and the actual image of the next frame. Results are shown in Section 7.

7 Experimental Results

In this section, we present and evaluate the estimated scene flows computed by both the techniques described in this paper. We use four different multi-view video sequences made publicly available by their authors in widely different configurations. Cheongsam [17] is captured in a dome of diameter equal to 4.2 m by twenty cameras in a ring around the scene. It has 20 cameras and every video is 30 frames long, but one of them had to be dropped due to missing frames. Redskirt [17] is captured in the same dome, but the videos are 20 frames long. The ballet and breakdance data [16] are acquired by eight cameras forming a 30° arc, thus with much narrower baselines. The depth range in this scene is 7.6 m. Depth maps are provided for the ballet and breakdance videos.

All experiments are performed using constant parameters for all parts of the methods tested, except the number of neighboring views in the computation of the scene flow using multiple candidate optical flows (Section 5). We use two neighboring views on each side of the reference camera for the ballet and breakdancer datasets, and one neighboring view on each side for the Cheongsam and Redskirt datasets, due to the wide angle between the neighboring views. The NCC window size for the plane-sweeping algorithm is 5×5 and 256 fronto-parallel planes with subpixel spacing are used for all datasets. For SGM, we use the rSGM implementation [68] with 8 paths, P_1 = 11, α = 0.5, γ = 35 and $P_{2,min}$ = 17. In the bilateral filter, we use γ_c = 7 and γ_p = 4.

For the Cheongsam and Redskirt data we create cost volumes for every frame using the plane-sweeping algorithm. Using these cost volumes and SGM, we extract the depth maps that were used as inputs to both scene flow estimation methods. The optical flows

	Endpoint		Angular	
	OF+D	MOF	OF+D	MOF
Cheongsam	8.61	0.64	29.79	5.73
Redskirt	4.95	0.75	25.21	7.45
ballet	4.42	3.22	33.80	26.93
breakdance	4.11	1.89	42.40	37.24

Table 1. Average Endpoint and Angular errors of OF+D and MOF estimated scene flows, projected (resulting in optical flows) on novel views that were excluded from all estimation steps. In lieu of ground truth data, the errors are estimated based on the "ground truth" optical flows computed using the data of the novel views (*OF*). The average is taken over all pixels of all frames of a single evaluation camera. Angular errors are displayed in degrees.

	"GT" OF	OF+D	MOF
Cheongsam	4.57	7.23	4.95
Redskirt	5.69	7.81	6.56
ballet	6.67	8.37	7.56
breakdance	8.30	9.88	9.13

Table 2. Average RGB L1 distance of next frame prediction on novel views compared to actual images. The average is taken over all pixels of all frames of a single evaluation camera. The column labeled *GT OF* is the prediction using the "ground truth" optical flow.

between consecutive frames needed as input to the methods described in this paper are estimated using the software of Sun et al. [62]. Then, we compute scene flow estimates for every frame in the videos tested using both methods as described in Sections 4 and 5. We evaluate quantitatively the scene flows by projecting to 2D optical flows on a novel view according to Section 6.

Tables 1 and 2 summarize the accuracy of all methods tested. Table 1 presents the errors in optical flow prediction, while Table 2 shows the results on novel view synthesis as described in Section 6. These errors are averaged over all frames for all datasets. It has to be noted here, that the angular errors between a flow vector (u, v) and the "ground truth" flow (u_{GT}, v_{GT}) were computed as defined in [18] according to the following formula

$$AE = cos^{-1}\left(\frac{1.0 + u \times u_{GT} + v \times v_{GT}}{\sqrt{1.0 + u^2 + v^2}\sqrt{1.0 + u_{GT}^2 + v_{GT}^2}}\right) \tag{12}$$

This metric aims to provide a *relative* measure of performance that avoids the "divide by zero" problem for zero flows. Errors in large flows are penalized less in AE than errors in small flows. Accuracy, in general is worse on the ballet and breakdance videos which have a lower frame rate than Cheongsam and Redksirt making motion estimation harder. The hypothesis for both tests, is that if accuracy using projected scene flows onto a novel view, whose data were not used to estimate the scene flows, is not degraded significantly compared to the one using the "ground truth" optical flows, then the scene flow estimation is considered successful. *MOF* presents significantly better and more

Fig. 4. Novel view next frame predition for cheongsam, redskirt, ballet and breakdance data. First column: Four instances of current frames of the validation camera. Second column: Next frames. Third column: Next frame prediction based on "ground truth" optical flow estimation *OF*. Fourth column: Next frame prediction based on estimated SF using *OF+D*. Fifth column: Next frame prediction based on estimated SF using *MOF*. Data from the novel views were not used for *OF+D* and *MOF* methods. Corresponding quantitative results shown in Table 2.

robust performance than the *OF+D* method, and is always quite close to the *"illegal"* prediction using the "ground truth" optical flow.

Figure 4 shows instances of a current frame, the next frame, the estimated next frame using the scene flow computed by the two methods, and an estimation of the next frame using the "ground truth" optical flow computed of the software of Sun et al. for all datasets tested.

8 Conclusion

We have compared the two most prevalent core computations for scene flow estimation in multi-view videos. *OF+D*, which is also common in RGB-D scene flow estimation, may fail because it relies on a single optical flow field. *MOF* is more robust because it uses multiple measurements. Depending on data-specific factors, the average improvement of *MOF* compared to *OF+D* can be as high as 35%, as in the Cheongsam sequence. The increase of the error compared to the *OF* that actually uses the data from the novel view can be as low as 10%, as in the breakdance sequence, but it never gets higher than 15%. The data-specific factors include the mutli-view configuration of the cameras such as the angle between the neighboring cameras, and the baseline, as well as the frequency content of the images, and the speed of the objects in the scene.

The evaluation methodology using view and optical flow field synthesis proposed in this work, enables quantitative analysis of scene flow in cases where there are no ground truth data available. We claim that this analysis is more informative than tests on synthetic or static (stereo) data, e.g. from [57].

Acknowledgments This research has been supported in part by the National Science Foundation award #1217797 and #1527294.

References

1. Vedula, S., Baker, S., Rander, P., Collins, R.T., Kanade, T.: Three-dimensional scene flow. In: ICCV. (1999) 722–729
2. Vedula, S., Baker, S., Rander, P., Collins, R.T., Kanade, T.: Three-dimensional scene flow. PAMI **27**(3) (2005) 475–480
3. Carceroni, R.L., Kutulakos, K.N.: Multi-view scene capture by surfel sampling: From video streams to non-rigid 3D motion, shape and reflectance. IJCV **49**(2-3) (2002) 175–214
4. Neumann, J., Aloimonos, Y.: Spatio-temporal stereo using multi-resolution subdivision surfaces. IJCV **47**(1-3) (2002) 181–193
5. Huguet, F., Devernay, F.: A variational method for scene flow estimation from stereo sequences. In: ICCV. (2007)
6. Gong, M.: Real-time joint disparity and disparity flow estimation on programmable graphics hardware. CVIU **113**(1) (2009) 90 – 100
7. Basha, T., Moses, Y., Kiryati, N.: Multi-view scene flow estimation: A view centered variational approach. In: CVPR. (2010)
8. Wedel, A., Brox, T., Vaudrey, T., Rabe, C., Franke, U., Cremers, D.: Stereoscopic scene flow computation for 3d motion understanding. IJCV **95** (2011) 29–51

9. Vogel, C., Schindler, K., Roth, S.: 3D scene flow estimation with a piecewise rigid scene model. IJCV **115**(1) (2015) 1–28
10. Herbst, E., Ren, X., Fox, D.: RGB-D flow: Dense 3-D motion estimation using color and depth. In: IEEE International Conference on Robotics and Automation (ICRA). (2013) 2276–2282
11. Jaimez, M., Souiai, M., Stuckler, J., Gonzalez-Jimenez, J., Cremers, D.: Motion cooperation: Smooth piece-wise rigid scene flow from rgb-d images. In: 2015 International Conference on 3D Vision (3DV). (2015) 64–72
12. Dou, M., Taylor, J., Fuchs, H., Fitzgibbon, A., Izadi, S.: 3d scanning deformable objects with a single rgbd sensor. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 493–501
13. Newcombe, R.A., Fox, D., Seitz, S.M.: Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 343–352
14. Sun, D., Sudderth, E.B., Pfister, H.: Layered rgbd scene flow estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 548–556
15. Collet, A., Chuang, M., Sweeney, P., Gillett, D., Evseev, D., Calabrese, D., Hoppe, H., Kirk, A., Sullivan, S.: High-quality streamable free-viewpoint video. ACM Transactions on Graphics (TOG) **34**(4) (2015) 69
16. Zitnick, C.L., Kang, S.B., Uyttendaele, M., Winder, S., Szeliski, R.S.: High-quality video view interpolation using a layered representation. ACM Trans. on Graphics **23**(3) (2004) 600–608
17. Liu, Y., Dai, Q., Xu, W.: A point cloud based multi-view stereo algorithm for free-viewpoint video. IEEE Trans. on Visualization and Computer Graphics **16**(3) (2010) 407–41
18. Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M., Szeliski, R.: A database and evaluation methodology for optical flow. IJCV **92**(1) (2011) 1–31
19. Mordohai, P.: On the evaluation of scene flow estimation. In: Unsolved Problems in Optical Flow and Stereo Estimation workshop. (2012)
20. Menze, M., Geiger, A.: Object scene flow for autonomous vehicles. In: CVPR. (2015) 3061–3070
21. Vedula, S., Baker, S., Seitz, S.M., Kanade, T.: Shape and motion carving in 6D. In: CVPR. (2000) 592–598
22. Kutulakos, K.N., Seitz, S.M.: A theory of shape by space carving. IJCV **38**(3) (2000) 199–218
23. Pons, J.P., Keriven, R., Faugeras, O.D.: Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. IJCV **72**(2) (2007) 179–193
24. Kwatra, V., Mordohai, P., Kumar Penta, S., Narain, R., Carlson, M., Pollefeys, M., Lin, M.: Fluid in video: Augmenting real video with simulated fluids. Computer Graphics Forum **27**(2) (2008) 487–496
25. Guan, L., Franco, J.S., Boyer, E., Pollefeys, M.: Probabilistic 3D occupancy flow with latent silhouette cues. In: CVPR. (2010)
26. Sizintsev, M., Wildes, R.: Spacetime stereo and 3d flow via binocular spatiotemporal orientation analysis. PAMI **36**(11) (2014) 2241–2254
27. Liu, F., Philomin, V.: Disparity estimation in stereo sequences using scene flow. In: British Machine Vision Conference. (2009)
28. Valgaerts, L., Bruhn, A., Zimmer, H., Weickert, J., Stoll, C., Theobalt, C.: Joint estimation of motion, structure and geometry from stereo sequences. In: ECCV. (2010)
29. Vogel, C., Schindler, K., Roth, S.: 3d scene flow estimation with a rigid motion prior. In: ICCV. (2011)
30. Isard, M., MacCormick, J.P.: Dense motion and disparity estimation via loopy belief propagation. In: Asian Conf. on Computer Vision. (2006) II:32–41

31. Cech, J., Sanchez-Riera, J., Horaud, R.: Scene flow estimation by growing correspondence seeds. In: CVPR. (2011)
32. Li, R., Sclaroff, S.: Multi-scale 3D scene flow from binocular stereo sequences. CVIU **110**(1) (2008) 75–90
33. Tao, H., Sawhney, H.S., Kumar, R.: Dynamic depth recovery from multiple synchronized video streams. In: CVPR. (2001) 118–124
34. Zhang, Y., Kambhamettu, C.: On 3-d scene flow and structure recovery from multiview image sequences. PAMI **33**(4) (2003) 592–606
35. Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., Cremers, D.: Efficient dense scene flow from sparse or dense stereo data. In: ECCV. (2008) I: 739–751
36. Rabe, C., Müller, T., Wedel, A., Franke, U.: Dense, robust, and accurate motion field estimation from stereo image sequences in real-time. In: ECCV. (2010) IV: 582–595
37. Müller, T., Rannacher, J., Rabe, C., Franke, U.: Feature- and depth-supported modified total variation optical flow for 3d motion field estimation in real scenes. In: CVPR. (2011)
38. Li, K., Dai, Q., Xu, W.: Markerless shape and motion capture from multiview video sequences. IEEE Trans. on Circuits and Systems for Video Technology **21**(3) (2011) 320–334
39. Furukawa, Y., Ponce, J.: Dense 3D motion capture from synchronized video streams. In: CVPR. (2008)
40. Furukawa, Y., Ponce, J.: Dense 3D motion capture for human faces. In: CVPR. (2009)
41. Courchay, J., Pons, J.P., Monasse, P., Keriven, R.: Dense and accurate spatio-temporal multi-view stereovision. In: Asian Conf. on Computer Vision. (2009) II: 11–22
42. Cagniart, C., Boyer, E., Ilic, S.: Free-form mesh tracking: a patch-based approach. In: CVPR. (2010)
43. Popham, T., Bhalerao, A., Wilson, R.: Multi-frame scene-flow estimation using a patch model and smooth motion prior. In: BMVC Workshop. (2010)
44. Allain, B., Franco, J.S., Boyer, E.: An efficient volumetric framework for shape tracking. In: CVPR. (2015)
45. Huang, C.H., Allain, B., Franco, J.S., Navab, N., Ilic, S., Boyer, E.: Volumetric 3d tracking by detection. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2016)
46. Starck, J., Hilton, A.: Correspondence labelling for wide-timeframe free-form surface matching. In: ICCV. (2007)
47. Ahmed, N., Theobalt, C., Rossl, C., Thrun, S., Seidel, H.P.: Dense correspondence finding for parametrization-free animation reconstruction from video. In: CVPR. (2008)
48. Varanasi, K., Zaharescu, A., Boyer, E., Horaud, R.: Temporal surface tracking using mesh evolution. In: ECCV. (2008) II: 30–43
49. Zeng, Y., Wang, C., Wang, Y., Gu, X., Samaras, D., Paragios, N.: Dense non-rigid surface registration using high-order graph matching. In: CVPR. (2010) 382–389
50. Budd, C., Huang, P., Hilton, A.: Hierarchical shape matching for temporally consistent 3d video. In: 3DIMPVT. (2011) 172–179
51. Huang, P., Hilton, A., Budd, C.: Global temporal registration of multiple non-rigid surface sequences. In: CVPR. (2011)
52. Letouzey, A., Petit, B., Boyer, E., Team, M.: Scene flow from depth and color images. In: BMVC. (2011)
53. Ferstl, D., Reinbacher, C., Riegler, G., Rüther, M., Bischof, H.: aTGV-SF: Dense variational scene flow through projective warping and higher order regularization. In: 3DV. (2014)
54. Hadfield, S., Bowden, R.: Scene particles: Unregularized particle-based scene flow estimation. PAMI **36**(3) (2014) 564–576
55. Hornacek, M., Fitzgibbon, A., Rother, C.: Sphereflow: 6 dof scene flow from rgb-d pairs. In: CVPR. (2014)

56. Quiroga, J., Brox, T., Devernay, F., Crowley, J.: Dense semi-rigid scene flow estimation from rgbd images. In: ECCV. (2014) 567–582
57. Scharstein, D., Szeliski, R.S.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV **47**(1-3) (2002) 7–42
58. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The International Journal of Robotics Research **32**(11) (2013) 1231–1237
59. Sizintsev, M., Wildes, R.: Spatiotemporal stereo and scene flow via stequel matching. PAMI **34**(6) (2012) 1206–1219
60. Gallup, D., Frahm, J.M., Mordohai, P., Yang, Q., Pollefeys, M.: Real-time plane-sweeping stereo with multiple sweeping directions. In: CVPR. (2007)
61. Hirschmüller, H.: Stereo processing by semiglobal matching and mutual information. PAMI **30**(2) (2008) 328–341
62. Sun, D., Roth, S., Black., M.: Secrets of optical flow estimation and their principles. In: CVPR. (2010)
63. Torr, P.H., Zisserman, A.: Mlesac: A new robust estimator with application to estimating image geometry. CVIU **78**(1) (2000) 138–156
64. Szeliski, R.: Prediction error as a quality metric for motion and stereo. In: ICCV. (1999) 781–788
65. Flynn, J., Neulander, I., Philbin, J., Snavely, N.: Deepstereo: Learning to predict new views from the world's imagery. In: CVPR. (2016)
66. Waechter, M., Beljan, M., Fuhrmann, S., Moehrle, N., Kopf, J., Goesele, M.: Virtual rephotography: Novel view prediction error for 3d reconstruction. arXiv preprint arXiv:1601.06950 (2016)
67. Kilner, J., Starck, J., Guillemaut, J.Y., Hilton, A.: Objective quality assessment in free-viewpoint video production. Signal Processing: Image Communication **24**(1-2) (2009) 3 – 16
68. Spangenberg, R., Langner, T., Adfeldt, S., Rojas, R.: Large scale semi-global matching on the cpu. In: IEEE Intelligent Vehicles Symposium. (2014) 195–201
69. Horn, B.K.P., Schunck, B.G.: Determining optical flow. Artifical Intelligence **17**(1-3) (1981) 185–203
70. Yoon, K.J., Kweon, I.S.: Adaptive support-weight approach for correspondence search. PAMI **28**(4) (2006) 650–656