Depth and Scene Flow from a Single Moving Camera

Neil Birkbeck Dana Cobzaş Martin Jägersand
University of Alberta
Edmonton, Alberta, Canada
{birkbeck,dana,jag}@cs.ualberta.ca

Abstract

We show that it is possible to reconstruct scene flow and depth from a single moving camera whose motion is known. To do so, we assume that the local scene motion can be approximated by a constant velocity in a small temporal window. This assumption makes it possible to unambiguously reconstruct scene flow and depth using as few as 3 frames from the sequence. We propose a variational approach to directly solve for structure and flow, and we demonstrate the results on challenging real-world data with both rigid and non-rigid motion. The experiments illustrate that the inclusion of flow in the case of non-rigid motion allows us to reconstruct a better geometry than if motion was simply ignored.

1. Introduction

Reconstructing scene flow, the 3D motion of surfaces in a scene, has been studied in several contexts in computer vision (e.g., [16, 21, 15]). In many cases, the scene flow is reconstructed in two phases: first geometry is reconstructed and then 2D optic flow in multiple images is used to recover the 3D scene flow. Both components of this reconstruction require a multi-camera setup to provide synchronized and calibrated video streams. In this work, we relax these multi-camera constraints, and address the problem of recovering scene flow and scene structure from a monocular camera sequence with known camera motion.

A great deal of progress has been made on reconstructing camera motion and scene structure of static scenes from monocular video streams, but many natural objects are non-rigid. Scene flow is a useful tool for analyzing such non-rigid motion, and it can be used for motion analysis, re-rendering in between time steps [15], and for the prediction of geometry[12]. Furthermore, the addition of temporal information (even in the case of non-rigid motion) is known to improve geometric reconstructions[17, 21, 5, 20]. Unfortunately, these scene flow methods are limited to situations where reconstruction of geometry at each time frame is possible(i.e., all surfaces must be viewed by multiple cameras).

Although most existing scene flow methods rely on several synchronized, calibrated cameras[10, 16], there are some implementations specific to a rectified binocular setting [8, 18]. In the monocular case, there has been some work on recovering the structure of deforming objects, but in each case assumptions must be made. For example, some common assumptions are that the deformations are a linear function of a set of basis shapes [2], follow a simple Gaussian model [13], or maximize rigidity[14]. All of these methods reconstruct only sparse geometry and deformations (although some do also recover orthographic camera parameters). In contrast, we make the assumption of constant velocity (over a short period of time), and use a variational approach to recover dense geometry and flow.

We illustrate that a simple assumption on the flow, namely that the flow has a constant velocity over a small sequence of images, allows the reconstruction of both depth and the scene flow. Dense reconstructions will only be possible on sequences that also have sufficient camera motion and loosely obey the above assumption. However, as we show, under good circumstances, if correspondences are known it is straightforward to reconstruct both flow and geometry (Section 3.1). The brightness constancy constraint can be augmented with our constant velocity assumption, which we integrate into a variational algorithm that recovers both the flow and geometry directly from images without requiring a-priori correspondences (Section 3.2). In contrast to existing variational approaches, our representation, although image-based, does not require rectified stereo pairs and can easily integrate information from several views. To summarize, our contributions are the following:

- We derive a linear algorithm to recover depth and 3D flow of points undergoing constant velocity over a short temporal window.
- We formulate a variational energy (based on optic flow [3] and existing scene flow methods [8]) to reconstruct the depth and flow. Our image-based representation of depth and flow easily extends to more than two images.

- We demonstrate the approach on challenging monocular sequences exhibiting non-rigid motion.

2. Related Work

As mentioned earlier, many applications of scene flow utilize multi-camera studios to first reconstruct either depth maps or voxel-based models. Constraints between differential motion of the surface and its differential motion in the images links the intra-camera optic flow to the 3D surface flow[16]. This decoupled reconstruction can be improved by simultaneously estimating the scene flow with the structure (e.g., using clustered regions with parametric flow[21], or by propagating covariances from depth and optic-flow estimation into the 3D flow estimates [9]). Although it is possible to derive similar differential constraints, which in our case would take into account the motion of the camera, we instead derive a geometric constraint on the flow.

The variational formulation, popular for both optic-flow and depth reconstruction, has also been used to recover scene flow. Pons *et al*. formulate an energy based on image warping that is implemented using level sets [10]. Such a scene-based representation could be an alternative to the image-based representation used in this work. In contrast, our formulation is similar to the image-based representations that operate with images from rectified stereo pairs [8, 18]. In these works, structure is represented as disparity maps (one for each time step) and flow is represented as *u*, *v* offsets[8]. Data terms in the energy functional establish pairwise constraints between pairs of the 4 images (e.g., left and right at time t and $t+1$). Again, these methods rely on multiple cameras to obtain the structure (at each time frame); we utilize a similar data energy as these formulations and incorporate our geometric constraints directly into the flow and structure recovery.

Carceroni & Kutulakos used a surfel representation to reconstruct both scene structure and flow [5]. Each surfel was a small oriented patch with a Phong reflectance model, an affine motion model, and a bump map. Static surfels were estimated using an approach similar to voxel-carving; joint estimation of all surfel parameters (including motion parameters) between time frames was used to improve this reconstruction. Extensions of this patch-based approach to longer sequences has been proposed through a multi-view patch tracking method based on image registration [6]. If the orientation and position of a patch is known in the first frame, such patch-based methods could be applied to monocular sequences [4], but this is a chicken and egg problem of how to reconstruct the initial pose of a moving patch from a moving monocular camera.

There has also been some work on recovering the deformations (or flow) of an object from a single view when a template geometry is known (e.g., [11]). In our case, the structure and its appearance are unknown, so these techniques are not applicable. Assumptions about maximizing rigidity have also been used to incrementally improve an approximate sparse geometry undergoing non-rigid motion[14]. Similarly, non-rigid structure and motion techniques[13, 2] are capable of recovering the structure and deformation model (as well as orthographic camera motion), but such techniques rely on correspondences attained from an external process over several frames. Unlike these methods, our formulation assumes known camera motion and incorporates the recovery of correspondences directly into the reconstruction.

Exploiting flow has been shown to improve reconstructions, for example, the 6D shape and motion carving of Vedula et al. [17], which performs a voxel-carving in a 6D space that links voxels between time steps, illustrated that using both time frames simultaneously improves the geometric reconstruction. Similar connections between the use of temporal information (e.g., joint recovery of motion and flow) and the improved shape estimate have been demonstrated in other scene flow reconstruction algorithms [5, 21]. Again, although we use a monocular sequence, our primary objective is to exploit surface motion to reconstruct a better geometry; in other words, the flow is a by-product used to aid reconstruction.

Regarding the constant velocity constraint, Avidan and Shashua show that points undergoing linear motion can be recovered with as few as five views from a monocular sequence from a moving camera[1]. Our constraint is similar, although we assume constant velocity, which is a subset of linear motion, allowing a reconstruction with as few as three views. Similarly, some have considered restricting motion to a scaled piecewise affine model over a larger time window[21]. As in our case, the constraint over a time window allows more accurate estimation of the parameters (e.g., in their case the affine model parameters, and in our case the velocity). The constant velocity assumption over a temporal window has been alluded to in earlier scene flow works (e.g., [5]). In fact all multi-camera methods that compute scene flow using image sets from two adjacent time frames are essentially assuming constant velocity for the two frames, but this assumption has not been exploited to extract structure and flow for more than two consecutive images from a monocular sequence as it is in our work.

3. Monocular Scene Flow

Assume we are given a time sequence of F images I_i with calibration $\mathbf{P}_i = \mathbf{K}_i[\mathbf{R}_i\mathbf{t}_i]$ taken at times t_i. The objective is to reconstruct a dense geometry and 3D velocity field at each time frame, j, using a small temporal neighborhood of images, $N(j)$, with $|N(j)| >= 2$. The situation is illustrated in Figure 1. We first discuss the reconstruction from a single reference frame assuming known correspondences of points with constant velocity (Section 3.1). These

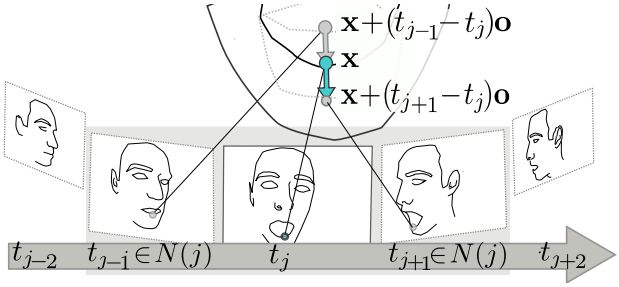

Figure 1. Reconstruction in reference view j using neighboring images, $N(j)$; constant velocity is more likely to hold over a small temporal window.

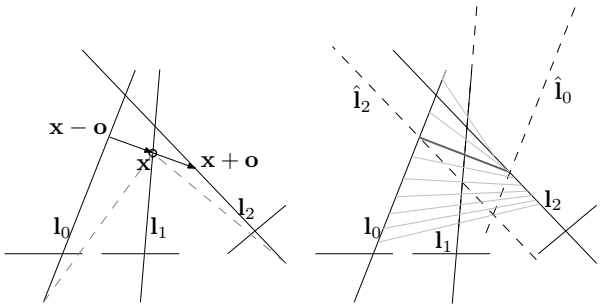

Figure 2. Three view setup with ambiguous reconstruction. Geometrically, the solutions can be obtained by rotating l_0 and l_2 about l_1 (giving $\hat{l}_0$ and $\hat{l}_1$) for each rotation center (e.g., depth) on l_1.

constraints are used to derive a variational energy for reconstructing geometry and flow directly from images (Section 3.2). Finally, we develop constraints to ensure longer sequences will be temporally consistent (Section 3.3).

3.1. Known Correspondences

Given correspondences, $\mathbf{a}_{ki} = [a_{ki}, b_{ki}]^T$, of a 3D point k in the images, the objective is to reconstruct the shape, $\mathbf{x}_k = [x_k, y_k, z_k, 1]^T$, and velocity, $\mathbf{o}_k = [u_k, v_k, w_k, 0]^T$, of the point in a reference time frame.

In a two view case where camera motion is known and the object is slowly moving (or almost rigid), most of the variation between the two images will be due to depth variation along the known epipolar geometry. Two dimensional flow components orthogonal to the epipolar lines are surely due to 3D flow, but 3D flow can also cause variation along the epipolar line. In such cases it is impossible to separate depth from flow; even when depth is known, the flow is ambiguous. Constraints on the offset (e.g., the minimal such offset, or the offset is parallel to the image plane) give a unique solution, but neither of these are plausible constraints.

An unconstrained three frame setup leads to a similar ill-posed configuration. However, with the extra frame it is possible to derive more physically plausible constraints. Assume that the point has a constant velocity through time $t_i \in \{0, 1, 2\}$ and is observed by a moving camera, then it is often possible to reconstruct a unique solution. Unfortunately, some degenerate camera configurations exist: no camera motion implies infinitely many solutions, and coplanar backprojected rays give a one parameter family of solutions (see Figure 2 for a 2D top down example). Note, in the latter degenerate case, unlike the two-view example, knowing the depth allows the recovery of correct 3D flow. Reconstructing the structure and velocity at t_1 gives three non-linear constraints on the 3D point $\mathbf{x}_k$ and its offset:

$$\Pi(\mathbf{P}_0(\mathbf{x}_k + (t_0 - t_1)\mathbf{o}_k)) = \Pi(\mathbf{P}_0(\mathbf{x}_k - \mathbf{o}_k)) = \mathbf{a}_{j0} \quad (1)$$

$$\Pi(\mathbf{P}_1(\mathbf{x}_k)) = \mathbf{a}_{j1} \quad (2)$$

$$\Pi(\mathbf{P}_2(\mathbf{x}_k + (t_2 - t_1)\mathbf{o}_k) = \Pi(\mathbf{P}_2(\mathbf{x}_k + \mathbf{o})) = \mathbf{a}_{j2} \quad (3)$$

where Π is the projective division operator.

Much like triangulation in multi-view stereo [7], this non-linear problem can be transformed into a linear one:

$$[\mathbf{P}_i]_1(\mathbf{x}_k + t_{i1}\mathbf{o}_k) - [\mathbf{P}_i]_3(\mathbf{x}_k + t_{i1}\mathbf{o})a_{ji} = 0 \quad (4)$$

$$[\mathbf{P}_i]_2(\mathbf{x}_k + t_{i1}\mathbf{o}_k) - [\mathbf{P}_i]_3(\mathbf{x}_k + t_{i1}\mathbf{o})b_{ji} = 0 \quad (5)$$

Where $t_{i1} = t_i - t_1$ (e.g., $t_{01} = -1$, $t_{11} = 0$, and $t_{21} = 1$), and $[\mathbf{P}_j]_m$ is row m of matrix $\mathbf{P}_j$, for a total of 6 equations in the 6 unknowns. The system of equations is rank deficient whenever the backprojected rays from the correspondences lie in the same plane.

The above argument suggests that $|N(j)| = 2$ is sufficient for reconstruction of geometry and flow, but this minimal view formulation is sensitive to noise. Non-degenerate configurations give a unique solution that satisfies the non-linear system exactly. An obvious way to circumvent noise is to use more image frames (e.g., increase $|N(j)|$). Unfortunately, given that all the frames are taken at fixed time intervals, using more frames from the sequence means that constant velocity must hold over a longer time period, which may become less likely in a real setting. An alternative to using extra image frames is to regularize the reconstruction; we investigate this approach in our variational formulation.

3.2. Reconstruction from Images

The previous discussion demonstrated that it is possible to simultaneously reconstruct both scene flow and depth from a single moving camera with known motion. The algorithms required correspondences between points be known in advance. Instead of obtaining correspondences from some procedure that is unaware of the constant velocity assumption (e.g., through optic flow), in this section, we derive a direct estimation process using a variational formulation that encodes the constant velocity assumption.

When the correspondences are unknown, we restate our objective as the recovery of a dense per-pixel mapping of

structure and flow at a reference frame (e.g., frame j). The shape in this frame can be represented by a per-pixel map, the disparity (inverse depth) map. The 3D point corresponding to a 2D point $\mathbf{x} = [x, y]^T$ with disparity d is

$$\wp(\mathbf{P}_j; \mathbf{x}, d) = \mathbf{R}_j^T(\mathbf{K}_j^{-1}[x/d, y/d, 1/d]^T - \mathbf{t}_j)$$

The constant velocity assumption can now be stated in conjunction with the brightness constancy constraint (e.g., the constraint typically used in optic flow), which gives the *constant velocity and constant brightness constraint*:

$$\sum_{i \in N(j)} |I_i(\Pi(\mathbf{P}_i(\wp(\mathbf{P}_j; \mathbf{x}, d) + (t_i - t_j)\mathbf{o})) - I_j(\mathbf{x})|^2 \quad (6)$$

which states the intensities of the flowed point in nearby temporally adjacent frames, $N(j)$, should minimize the difference to the brightness/color in frame j. We use the notation, $\underset{j \to i}{W}$, as the shorthand representation for the backprojection of the 2D point, which is then displaced, and projected into image i:

$$\begin{aligned} \underset{j \to i}{W}(\mathbf{x}, d, \mathbf{o}) = \underset{j \to i}{W}(\mathbf{x}, d, u, v, w) = \left[\frac{\hat{x}}{\hat{z}}, \frac{\hat{y}}{\hat{z}}\right]^T = \\ \Pi(\mathbf{P}_i(\wp(\mathbf{P}_j; \mathbf{x}, d) + (t_i - t_j)\mathbf{o})) = \\ \Pi(\mathbf{K}_i\mathbf{t}_i + \mathbf{K}_i\mathbf{R}_i(\wp(\mathbf{P}_j; \mathbf{x}, d) + t_{ij}\,[u\ v\ w]^T)) \end{aligned}$$

The dense structure and flow, $\mathbf{d} = [d, u, v, w]^T$, for a reference frame j can the be recovered as the minimum of the following energy functional:

$$\begin{aligned} E_j(\mathbf{d}) = \sum_{i \in N(j)} \int \Psi_d(|I_i(\underset{j \to i}{W}(\mathbf{x}, d, u, v, w)) - I_j(\mathbf{x})|^2) d\mathbf{x} \\ + \alpha \int \Psi_s(|\nabla d|^2) d\mathbf{x} \\ + \beta \int \Psi_u(|\nabla u|^2 + |\nabla v|^2 + |\nabla w|^2) d\mathbf{x} \end{aligned} \quad (7)$$

where, $\Psi_*(x^2)$ are robust functions. This functional is an extension of the variational optic flow methods to disparity and scene flow [3]. Similar to these methods, we use $\Psi_d(x^2) = \Psi_s(x^2) = \Psi_u(x^2) = \sqrt{x^2 + \epsilon^2}$ for some small ϵ. The first term, the data term, measures the consistency using the *constant velocity and brightness constraint*; the other terms regularize the disparity and flow separately.

The minimum of Eq. 7 is an image-based representation of structure and flow for a single reference frame j that utilizes the constant velocity assumption over a temporal window $N(j)$. The temporal window should contain a subset of images where this assumption is likely to hold.

Defining $|\nabla \mathbf{o}|^2 = |\nabla u|^2 + |\nabla v|^2 + |\nabla w|^2$, the Euler-Lagrange equations of Eq.7 are

$$\sum_{i \in N(j)} \Psi'_d(I_{ij}^2) I_{ij} I_{id} - \alpha \nabla \cdot (\Psi'_s(|\nabla d|^2) \nabla d) = 0 \quad (8)$$

$$\sum_{i \in N(j)} \Psi'_d(I_{ij}^2) I_{ij} I_{iu} - \beta \nabla \cdot (\Psi'_u(|\nabla \mathbf{o}|^2) \nabla u) = 0 \quad (9)$$

$$\sum_{i \in N(j)} \Psi'_d(I_{ij}^2) I_{ij} I_{iv} - \beta \nabla \cdot (\Psi'_u(|\nabla \mathbf{o}|^2) \nabla v) = 0 \quad (10)$$

$$\sum_{i \in N(j)} \Psi'_d(I_{ij}^2) I_{ij} I_{iw} - \beta \nabla \cdot (\Psi'_u(|\nabla \mathbf{o}|^2) \nabla w) = 0 \quad (11)$$

Where

$$\begin{aligned} I_{ij} &= I_i(\underset{j \to i}{W}(\mathbf{x}, d, u, v, w)) - I_j(\mathbf{x}) \\ I_{id} &= \frac{\partial}{\partial d} I_i(\underset{j \to i}{W}(\mathbf{x}, d, \mathbf{o})) = \nabla I_i \frac{\partial}{\partial d} \underset{j \to i}{W}(\mathbf{x}, d, \mathbf{o}) \\ \frac{\partial}{\partial d} \underset{j \to i}{W} &= \begin{bmatrix} \frac{\hat{z}\frac{\partial \hat{x}}{\partial d} - \hat{x}\frac{\partial \hat{z}}{\partial d}}{\hat{z}^2} \\ \frac{\hat{z}\frac{\partial \hat{y}}{\partial d} - \hat{y}\frac{\partial \hat{z}}{\partial d}}{\hat{z}^2} \end{bmatrix} \\ \frac{\partial \mathbf{p}}{\partial d} &= \mathbf{K}_i\mathbf{R}_i\mathbf{R}_j^T\mathbf{K}_j^{-1}\,[1\ 1\ 1]^T\,\frac{1}{d} \\ \frac{\partial \mathbf{p}}{\partial u} &= \mathbf{K}_i\mathbf{R}_i\,[(t_i - t_j)\ 0\ 0]^T \end{aligned}$$

with $\mathbf{p} = [\hat{x}, \hat{y}, \hat{z}]^T$. $\frac{\partial \mathbf{p}}{\partial v}$ and $\frac{\partial \mathbf{p}}{\partial w}$ are similar to $\frac{\partial \mathbf{p}}{\partial u}$. The abbreviations, I_{iu}, I_{iv}, I_{iw}, are defined similar to I_{id}.

We solve the Euler-Lagrange equations similar to the optic flow counterpart[3]. A fixed point iteration is defined and the data term is linearized (see the Appendix). This linearized equation is solved for several iterations over several scales of an image pyramid (we perform several linearizations at each scale) with a multi-grid method that uses a point coupled Gauss-Seidel method as the basic solver for pre- and post-smoothing. The dimensions of levels of our image pyramid differ by a factor of 2 (see Algorithm 1).

Initialization: One approach to compute a suitable initialization would be to compute optic flow between the reference image and its neighbors and then use techniques in Section 3.1 to extract an initial disparity and flow. However, we found it sufficient to obtain a coarse initial disparity from stereo on lower resolution images, which implicity assumes zero flow. The lower resolution images help account for some of the flow. Disparity is first estimated in a discrete framework (e.g., similar to Zhang *et al.* [19]). These disparities are refined using a variational disparity estimation (e.g., Eq. 7 with no flow components). As the coupled flow and disparity refinement is initialized on a low resolution, we found it sufficient to initialize the 3D flow to zero.

3.3. Longer temporal constraints

It is unlikely that the constant velocity assumption will hold over a long range of frames, so it is desirable to have a

Algorithm 1: disp_flow(I_j, $N(j)$)

Data: $\mathbf{d}_0 = \{d_0, v_0, u_0, w_0\}$ (e.g., d_0 from stereo, u_0, v_0, w_0 zero). Reference image I_j and neighbors $N(j)$, α, β

Result: The disparity and flow for I_j, $\mathbf{d} = \{d, u, v, w\}$

$\mathbf{d} = \mathbf{d}_0$ *// Solve on image pyramid;*

for $l \leftarrow l_{max}$ **to** 0 **do**

  $\mathbf{d}$ = resample_solution_for_level(l, $\mathbf{d}$);

  for $h \leftarrow 1$ **to** n_{lin} **do**

    // Solve linearized problem using multi-grid;

    $\mathbf{d}$ = solve_Ej($\mathbf{d}, \alpha, \beta$);

    // update $\mathbf{d}$ *in place, e.g., k = k + 1;*

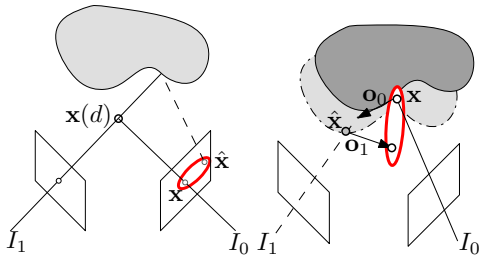

Figure 3. Left: a simple disparity map consistency constraint where the disparity for point $\mathbf{x}$ is penalized by the distance to the point $\hat{\mathbf{x}}$. Right: a flow constraint penalizing the distance in 3D.

formulation that will allow for changes in velocity. A possible image-based method is to solve the variational problem for each image using a close set of neighboring images and then to enforce constraints between these independent solutions.

First, consider how a depth consistency constraint could be encoded in the variational approach. If the problem is to only recover depth from each view (e.g., flow is regularized to be zero), it is straightforward to incorporate a constraint that links the shape in each image. Motivated by the geometric consistency used by Zhang et al.[19], which uses neighboring disparity maps to penalize the matching score, we can define an image-based penalizer on disparity.

For example, given the geometry of the cameras, a disparity d for a point $\mathbf{x}$ in image 0 will map the point to some location in image 1. The disparity in image 1 is used to map the point back to image 0, giving a point $\hat{\mathbf{x}}$. Inconsistent disparity maps can be penalized by minimizing $|\hat{\mathbf{x}} - \mathbf{x}|^2$. The situation is illustrated in Figure 3.

Similar constraints can be defined when each image contains independent estimates of both structure and flow (see Fig. 3). Using image j as a reference, the constraints can be incorporated into the variational problem as the following:

$$C_{ji}(\mathbf{d}) = \int\int \kappa \Psi_d(|h_i(\underset{j \rightarrow i}{W}(\mathbf{x}, \mathbf{d})) - \wp_j(\mathbf{x}, \mathbf{d}))|^2) \mathbf{dx} \tag{12}$$

Algorithm 2: disp_flow_many($\{I_i\}$)

Result: $\{\mathbf{d}_j\}$, disparity and flow for all images.

// Solve for each view independently;

for $j \leftarrow 1$ **to** F **do**

  d = basic_stereo(j, $N(j)$);

  $\mathbf{d}_j = \{d, \mathbf{0}, \mathbf{0}, \mathbf{0}\}$;

  $\mathbf{d}_j$ = disp_flow(j, $N(j)$, $\mathbf{d}_j$);

// Solve again with constraints;

for $l \leftarrow l_{max}$ **to** 0 **do**

  for $j \leftarrow 1$ **to** F **do**

    $\mathbf{d}_j$ = resample_solution_for_level(l, $\mathbf{d}_j$);

    for $h \leftarrow 1$ **to** n_{iters} **do**

      $\mathbf{D} = \{\mathbf{d}_k | k \in N(j)\}$;

      // Linearize and minimize Eq. 13 w.r.t $\mathbf{d}_j$*;*

      $\mathbf{d}_j$ = solve_cons(j, $N(j)$, $\mathbf{d}_j$, $\mathbf{D}$);

where $\wp_j(.,.)$ is the backprojection function for image j. The function h_i is the backproject-and-offset function and uses the current estimates of structure and flow from image i. Using this formulation, the structure and flow for all images can be extracted in a manner that ensures consistency and allows deviations from constant velocity.

The combined problem of estimating the disparity and flow from all frames with constraints can be thought of as finding the minimum to the following expression:

$$E_{total}(\{\mathbf{d}_i\}) = \sum_{j=1}^{F} (E_j(\mathbf{d}_j) + \sum_{i \in N(j)} C_{ji}(\mathbf{d}_j)) \tag{13}$$

We obtain an approximate solution to this equation by first solving for the disparity and flow at each image independently (e.g., minimize E_j), and then in a second pass introduce the constraints, C_{ji} and solve again for E_j holding all the other disparity and flow constant (Algorithm 2).

4. Experiments

We first consider a synthetic example where the motion obeys our constant velocity assumption. The setup consists of three views roughly 4 units from a planar object that deforms into a bumpy surface with some shifting (Fig. 4). The object is 2 units wide and has flows of 10% of this size.

To test the sensitivity of our estimate to perturbations in camera baseline, we have run our algorithm on varying proportions of the full camera baseline (which is roughly 1.3 and 1.9 units for left and right cameras respectively).

Figure 5 shows the distance to ground truth for the initial disparity estimation and for the refined disparity and flow estimations (these are computed as average difference in depth for the disparity values and average vector difference for the flows). The figure shows that our method outperforms the initial disparity, which is ignorant of the flow.

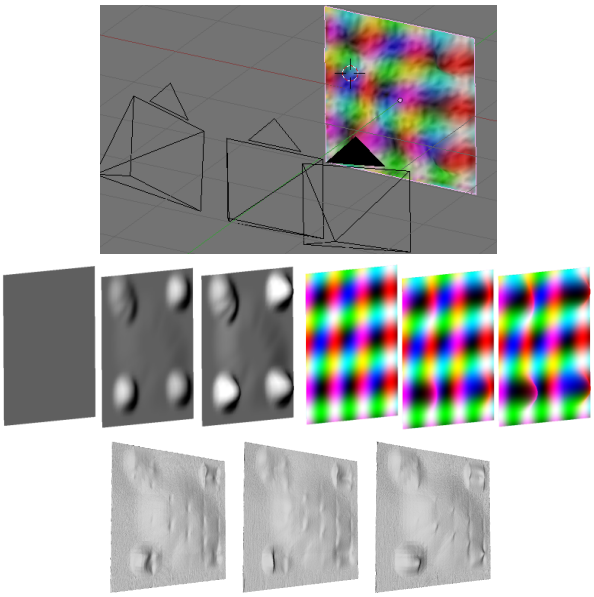

Figure 4. Top: the moving camera setup (full baseline depicted here). Middle: shaded and textured (side-views) of the deforming object. Bottom: the initial disparity estimate (e.g, zero flow), which averages the deformations.

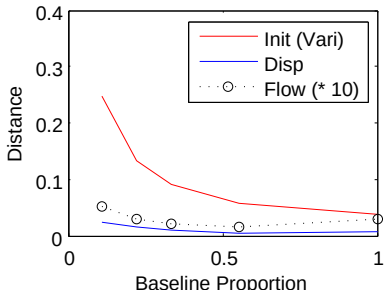

Figure 5. Distance to ground truth for initial disparity estimate (Init) and recovered disparity (Disp) and flow on synthetic plane sequence.

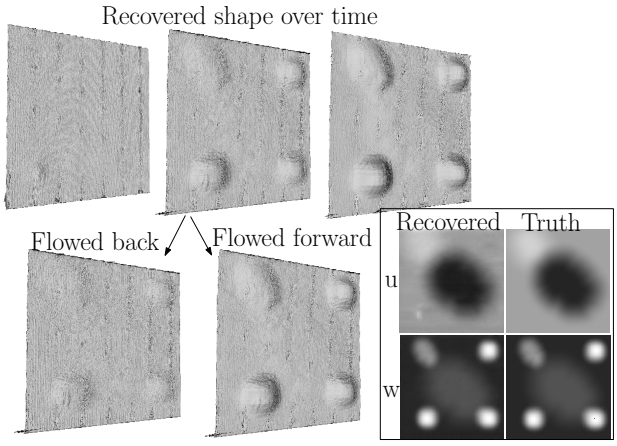

Figure 6. Top: the recovered geometry for each time step (and the flowed geometry from time 1) are similar to the ground truth (Fig. 4). Bottom right: recovered u and w components beside the ground truth ($t = 1$, $\alpha = 0.83$).

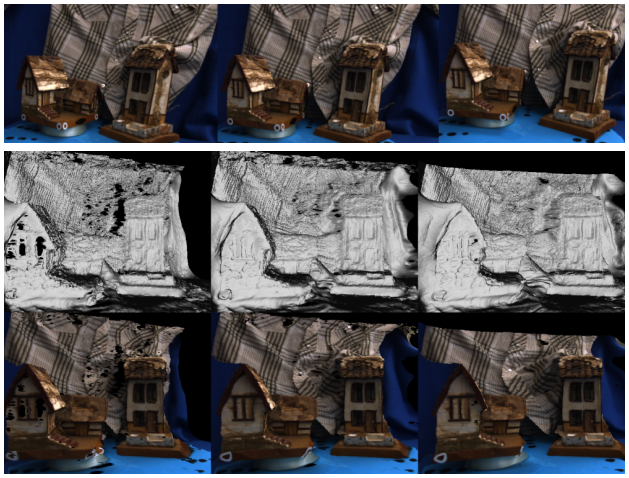

Figure 7. Top: cropped input views. Bottom: this novel viewpoint illustrates that the geometry was accurately reconstructed for each of the time sequences (notice the rightward motion of the right house and the rotation of the left house).

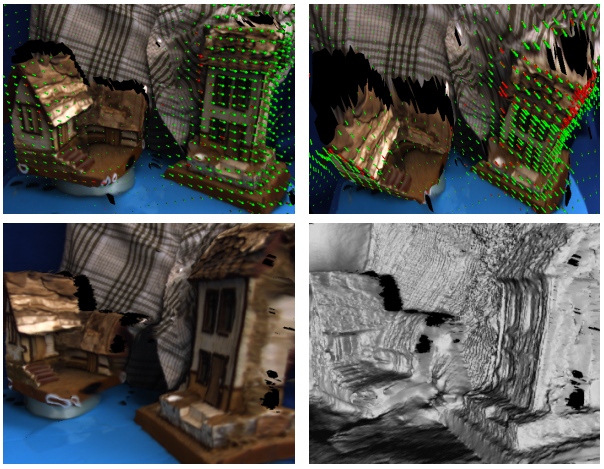

Figure 8. Top: 3D views of the overlaid flow (red is in front of object, green is behind). Bottom: novel viewpoint of the geometry.

Also, as expected the estimation degrades with too small baseline, and the reconstructed flow degrades with too large of a baseline (possibly due to increased occlusions).

Figure 6 illustrates our results for the full baseline case. Notice that this geometry looks like the recovered (and ground truth) geometry in the neighboring frames. This similarity is evident in the image representations of the u and w components compared to the ground truth. The average depth error is 0.007 ($\sim 0.35\%$ of the object size) and average flow error of 0.0029 ($\sim 1.5\%$ of the flow size).

4.1. Real sequences

The first sequence has three views of two houses: the left house is rotating counter clockwise (from top), the right house translates to the right and is draped with a shirt that

Sequence	F	size	α	β	κ	l_{max}	n_{iters}
Two house	3	800x600	8	4.8	0.02	4	10
Mouth	3	800x600	6.25	5.63	0.18	4	20

Table 1. Information and parameters for the real sequences

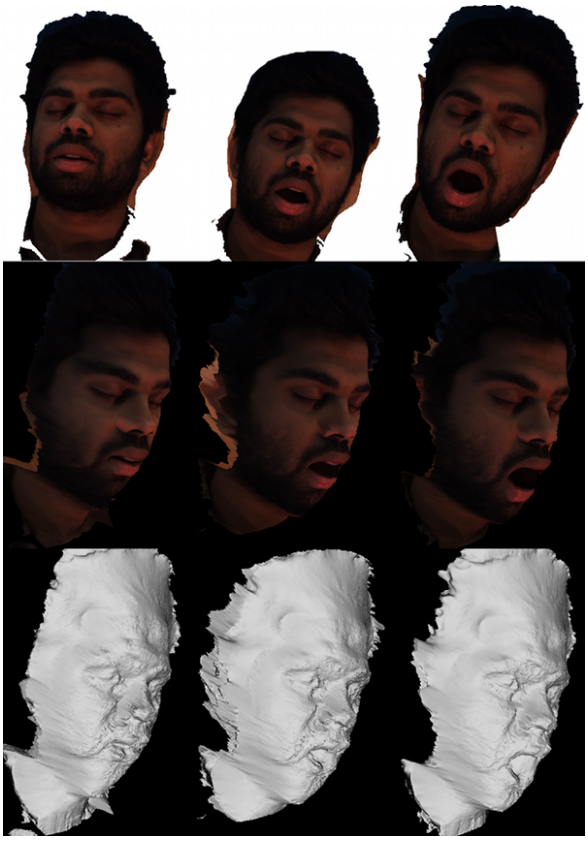

Figure 9. Top: input views. Middle & Bottom: textured and shaded results from a novel viewpoint.

moves non-rigidly, and the background is stationary (parameters are in Table 1). Camera motion is extracted relative to the background using a calibration pattern. Figure 7 shows the input views and the reconstructed geometry at each time from a novel viewpoint; Figure 8 shows the 3D flow and another viewpoint. The geometry is consistent through the 3 frames, and the flow (Fig. 8) represents the rigid house motion, the non-rigid cloth motion, and the stationary ground.

The second real data set was three views of a person opening his mouth. Camera motion was again extracted with a calibration pattern and is relative to the subjects skull. Figure 9 shows the input views and the resulting geometric reconstructions. In this case there is some motion in the z-coordinate not present in the sequence. Notice that the motion of the camera is significant enough to make this sequence challenging for strict optic flow; the addition of depth and flow allows successful recovery. See Figure 10 for the images warped to time frame 1.

Figure 10. Images 0 (left) and 2 (right) warped to image 1 (middle) using the recovered depth and flow.

5. Conclusions

We have proposed a variational method to estimate scene flow and geometry from a moving monocular camera with known motion. Assuming a constant velocity over a small temporal window in such circumstances allows for reconstruction of both geometry and structure; this assumption is embedded in our estimation.

Our approach does have some limitations. The primary limitation is that the monocular camera motion must be known. We intend to use this technique to recover the dense deformations from a multi-view human capture studio where there is little overlap in views. Motion of the human's articulated links (extracted through multi-camera tracking) would give the relative motion of the camera, from which our method could be applied. Although we have addressed the problem primarily for the monocular case, we expect that aspects of our approach could be adapted to the multi-view case where input views have little overlapping regions; this integration could take advantage of other cues, such as the bounds of the visual hull or stereo constraints in the regions with sufficient overlap.

Another limitation is the flow should be well approximated by constant velocity over a short time frame. A worst case scenario is when the surface motion reverses directions. Clearly, surface motion in an infinitely small time window can be well approximated as constant velocity. But we rely on the motion of the camera relative to the object in order to reconstruct the depth, so there is a trade-off between being able to reconstruct depth (e.g., enough baseline) and the approximation of constant velocity holding.

The current representation is completely image-based. This representation has made it necessary to link reconstructions over longer sequences (e.g., using constraints on the flow). Our current constraints (and energy) do not take into account occlusions. Perhaps a scene-based representation (e.g., [10]) would be more natural to link constraints through longer sequences.

A. Linearization of the Euler-Lagrange

Our solution to Eqs 8-11 follows the derivation for optic flow[3]. First, a fixed point iteration is defined:

$$\sum_{i\in N(j)} \Psi'_d((I_{ij}^{k+1})^2) I_{ij}^{k+1} I_{id}^k - \alpha\nabla\cdot(\Psi'_s(|\nabla d^{k+1}|^2)\nabla d^{k+1}) = 0$$

$$\sum_{i\in N(j)} \Psi'_d((I_{ij}^{k+1})^2) I_{ij}^{k+1} I_{iu}^k - \beta\nabla\cdot(\Psi'_u(|\nabla \mathbf{o}^{k+1}|^2)\nabla u^{k+1}) = 0$$

$$|\nabla \mathbf{o}^{k+1}|^2 := |\nabla u^{k+1}|^2 + |\nabla v^{k+1}|^2 + |\nabla w^{k+1}|^2$$

The equations for v and w are similar to u.

Letting $d^{k+1} = d^k + \delta d^k$, $u^{k+1} = u^k + \delta u^k$, $v^{k+1} = v^k + \delta v^k$, and, $w^{k+1} = w^k + \delta w^k$, the data term can be linearized as

$$\begin{aligned} I_{ij}^{k+1} =& I_i(\underset{j\to i}{W}(\mathbf{d}^{k+1})) - I_j \\ \approx& I_i(\underset{j\to i}{W}(\mathbf{d}^k)) - I_j + \\ & I_{id}\delta d^k + I_{iu}\delta u^k + I_{iv}\delta v^k + I_{iw}\delta w^k \\ =& I_{ij}^k + I_{id}\delta d^k + I_{iu}\delta u^k + I_{iv}\delta v^k + I_{iw}\delta w^k \end{aligned}$$

Letting

$$\nabla I_i^k = [I_{id}^k\ I_{iu}^k\ I_{iv}^k\ I_{iw}^k\ I_{ij}^k]^T,$$

and

$$\delta\mathbf{d}^k = [\delta d^k\ \delta u^k\ \delta v^k\ \delta w^k\ 1]^T,$$

we have $I_{ij}^{k+1} \approx (\nabla I_i^k)^T \delta\mathbf{d}^k$. Defining, $S_i^k = \nabla I_i^k (\nabla I_i^k)^T$, the Euler-Lagrange equations become:

$$\sum_{i\in N(j)} \Psi'_d(\cdot)[S_i^k]_1\delta\mathbf{d}^k - \alpha\nabla\cdot(\Psi'^k_s(\cdot)\nabla d^{k+1}) = 0$$

$$\sum_{i\in N(j)} \Psi'_d(\cdot)[S_i^k]_2\delta\mathbf{d}^k - \beta\nabla\cdot(\Psi'^k_u(\cdot)\nabla u^{k+1}) = 0$$

$$\sum_{i\in N(j)} \Psi'_d(\cdot)[S_i^k]_3\delta\mathbf{d}^k - \beta\nabla\cdot(\Psi'^k_u(\cdot)\nabla v^{k+1}) = 0$$

$$\sum_{i\in N(j)} \Psi'_d(\cdot)[S_i^k]_4\delta\mathbf{d}^k - \beta\nabla\cdot(\Psi'^k_u(\cdot)\nabla w^{k+1}) = 0$$

$$\Psi'^k_d(\cdot) := \Psi'_d((\delta\mathbf{d}^k)^T S_i^k \delta\mathbf{d}^k)$$

$$\Psi'^k_s(\cdot) := \Psi'_s(|\nabla(d^k + \delta d^k)|^2)$$

$$\Psi'^k_u(\cdot) := \Psi'_u(|\nabla u^{k+1}|^2 + |\nabla v^{k+1}|^2 + |\nabla w^{k+1}|^2)$$

where $[S_i^k]_n$ is the $n-$th row of S_i^k. After discretization the above system is solved with multi-grid methods[3].

References

[1] S. Avidan and A. Shashua. Trajectory triangulation: 3d reconstruction of moving points from a monocular image sequence. *IEEE Trans. PAMI*, 22(4):348–357, 2000. 2

[2] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, volume 2, pages 690–696 vol.2, 2000. 1, 2

[3] A. Bruhn and J. Weickert. Towards ultimate motion estimation: Combining highest accuracy with real-time performance. In *ICCV '05*, pages 749–755, 2005. 1, 4, 8

[4] J. M. Buenaposada and L. Baumela. Real-time tracking and estimation of plane pose. In *ICPR (2)*, pages 697–700, 2002. 2

[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. *Int. J. Comput. Vision*, 49(2-3):175–214, 2002. 1, 2

[6] F. Devernay, D. Mateus, and M. Guilbert. Multi-camera scene flow by tracking 3-d points and surfels. *CVPR*, 2:2203–2212, 2006. 2

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 3

[8] F. Huguet and F. Devernay. A variational method for scene flow estimation from stereo sequences. In *ICCV*, pages 1–7. IEEE, 2007. 1, 2

[9] R. Li and S. Sclaroff. Multi-scale 3d scene flow from binocular stereo sequences. *CVIU*, 110(1):75–90, 2008. 2

[10] J.-P. Pons, R. Keriven, and O. Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *CVPR*, pages 822–827, 2005. 1, 2, 7

[11] M. Salzmann, F. Moreno-Noguer, V. Lepetit, and P. Fua. Closed-form solution to non-rigid 3d surface registration. In *ECCV (4)*, pages 581–594, 2008. 2

[12] N. C. T. M. A. Smith, D. Redmill and D. Bull. Time varying volumetric scene reconstruction using scene flow. In *BMVC '07*, 2007. 1

[13] L. Torresani, A. Hertzmann, and C. Bregler. Learning non-rigid 3d shape from 2d motion. In *NIPS*, 2003. 1, 2

[14] S. Ullman. Maximizing rigidity: the incremental recovery of 3-d structure from rigid and nonrigid motion. *Perception*, 13(3):255–274, 1984. 1, 2

[15] S. Vedula, S. Baker, and T. Kanade. Image-based spatio-temporal modeling and view interpolation of dynamic events. *ACM TOG*, 24(2):240 – 261, April 2005. 1

[16] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(3):475–480, 2005. 1, 2

[17] S. Vedula, S. Baker, S. Seitz, and T. Kanade. Shape and motion carving in 6d. In *CVPR '00*, June 2000. 1, 2

[18] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers. Efficient dense scene flow from sparse or dense stereo data. In *ECCV '08*, pages 739–751, Berlin, Heidelberg, 2008. Springer-Verlag. 1, 2

[19] G. Zhang, J. Jia, T.-T. Wong, and H. Bao. Recovering consistent video depth maps via bundle optimization. In *CVPR '08*, pages 1–8, 2008. 4, 5

[20] L. Zhang, B. Curless, and S. M. Seitz. Spacetime stereo: Shape recovery for dynamic scenes. In *CVPR*, pages 367–374, June 2003. 1

[21] Y. Zhang and C. Kambhamettu. Integrated 3d scene flow and structure recovery from multiview image sequences. *CVPR*, 2:2674, 2000. 1, 2