Boston U. Computer Science Tech. Report No. 2007-004, June 2007.
To appear in Computer Vision and Image Understanding

Multi-scale 3D Scene Flow from Binocular Stereo Sequences [1]

Rui Li and Stan Sclaroff

Computer Science Department, Boston University, Boston, MA 02215

Abstract

Scene flow methods estimate the three-dimensional motion field for points in the world, using multi-camera video data. Such methods combine multi-view reconstruction with motion estimation. This paper describes an alternative formulation for dense scene flow estimation that provides reliable results using only two cameras by fusing stereo and optical flow estimation into a single coherent framework. Internally, the proposed algorithm generates probability distributions for optical flow and disparity. Taking into account the uncertainty in the intermediate stages allows for more reliable estimation of the 3D scene flow than previous methods allow. To handle the aperture problems inherent in the estimation of optical flow and disparity, a multi-scale method along with a novel region-based technique is used within a regularized solution. This combined approach both preserves discontinuities and prevents over-regularization – two problems commonly associated with the basic multi-scale approaches. Experiments with synthetic and real test data demonstrate the strength of the proposed approach.

Key words: scene flow, stereo, disparity, optical flow, discontinuity, over-regularization

1 Introduction

There is increasing interest in methods that can estimate the motion of a 3D scene given video streams obtained via a multi-camera rig. The resulting estimates of 3D scene flow can be used in a wide variety of applications including robotics, autonomous navigation, automated model reconstruction and reverse engineering, human motion and gesture analysis, virtual reality and movie special effects, video compression and retrieval, etc.

Email addresses: lir@cs.bu.edu (Rui Li), sclaroff@cs.bu.edu (Stan Sclaroff).

[1] This research was funded in part by NSF grants CNS-0202067 and IIS-0208876, and ONR N00014-03-1-0108.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
JUN 2007		**00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Multi-scale 3D Scene Flow from Binocular Stereo Sequences	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Boston University,Computer Science Department,Boston,MA,02215	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Scene flow methods estimate the three-dimensional motion field for points in the world using multi-camera video data. Such methods combine multi-view reconstruction with motion estimation. This paper describes an alternative formulation for dense scene flow estimation that provides reliable results using only two cameras by fusing stereo and optical flow estimation into a single coherent framework. Internally, the proposed algorithm generates probability distributions for optical flow and disparity. Taking into account the uncertainty in the intermediate stages allows for more reliable estimation of the 3D scene flow than previous methods allow. To handle the aperture problems inherent in the estimation of optical flow and disparity, a multi-scale method along with a novel region-based technique is used within a regularized solution. This combined approach both preserves discontinuities and prevents over-regularization ? two problems commonly associated with the basic multi-scale approaches. Experiments with synthetic and real test data demonstrate the strength of the proposed approach.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**29**	

While the demonstrated applications of non-rigid 3D scene flow estimation are impressive, some aspects of the 3D motion estimation problem remain open. In particular, since the estimation of 3D motion relies on the information available from 2D image data, the estimation of 3D motion is generally susceptible to image noise. In order to obtain a reliable estimate of 3D motion, usually a large number of cameras are used in the stereo rig. There are also problems with estimation errors that arise in the regions of low contrast variation, or in areas of the surface which are visible to only a subset of the cameras. Ideally, one would prefer an estimation algorithm that can handle these problems in a principled way.

One of the main contributions of the paper is a method that can quantify and account for errors in non-rigid 3D scene flow estimates that arise due to image measurement errors. This is in direct contrast with past scene flow methods [51, 59, 60]; image measurement errors propagate through the 2D optical flow and disparity estimates, and ignoring this can lead to poor 3D scene flow estimates. Our method explicitly models the uncertainty of 2D optical flow and disparity estimation, and then systematically accounts for this uncertainty information in the estimation of 3D scene flow. Our experiments on data with known ground truth show that by incorporating the 2D uncertainties, the angular and the magnitude errors of the estimated 3D scene flow are at least one standard deviation smaller than those obtained with [51].

Another main contribution of the paper is a unified region-based, multi-scale algorithm for the estimation of optical flow and disparity. Compared to [43], the proposed multi-scale, region-based algorithm preserves the discontinuities, while at the same time, the optical flow and disparity within the same region are regularized through a parametric model fitting process. In contrast with [6], our region model fitting process makes direct use of the 2D uncertainty information and does not require setting parameters for a robust error norm. Moreover, our algorithm is capable of filling in the estimates of optical flow and disparity for regions that are only visible in one image by making use of adjacent regions that have valid estimates of optical flow and disparity. Using our formulation, we demonstrate improved accuracy of optical flow and disparity using standard data sets [49, 50].

In this paper, we focus on the challenging case of estimating dense 3D scene flow given only the minimal setup of two cameras in the stereo rig. Past approaches tend to require more cameras to gain a reasonable estimate of 3D scene flow; for instance, [51] reported experimental results in a rig of 51 cameras. In experiments, our algorithm outperforms [51] in the two-camera setup and produces qualitatively similar results as those of [59, 60] where three cameras are needed. We observe that it becomes possible to perform well in the challenging two-camera case if covariances in 2D flow and disparity estimation are explicitly propagated, and discontinuities are adequately modeled in a multi-scale, region-based framework. Finally, it should be emphasized that our algorithm can be used with a rig that includes more than two cameras without any modifications.

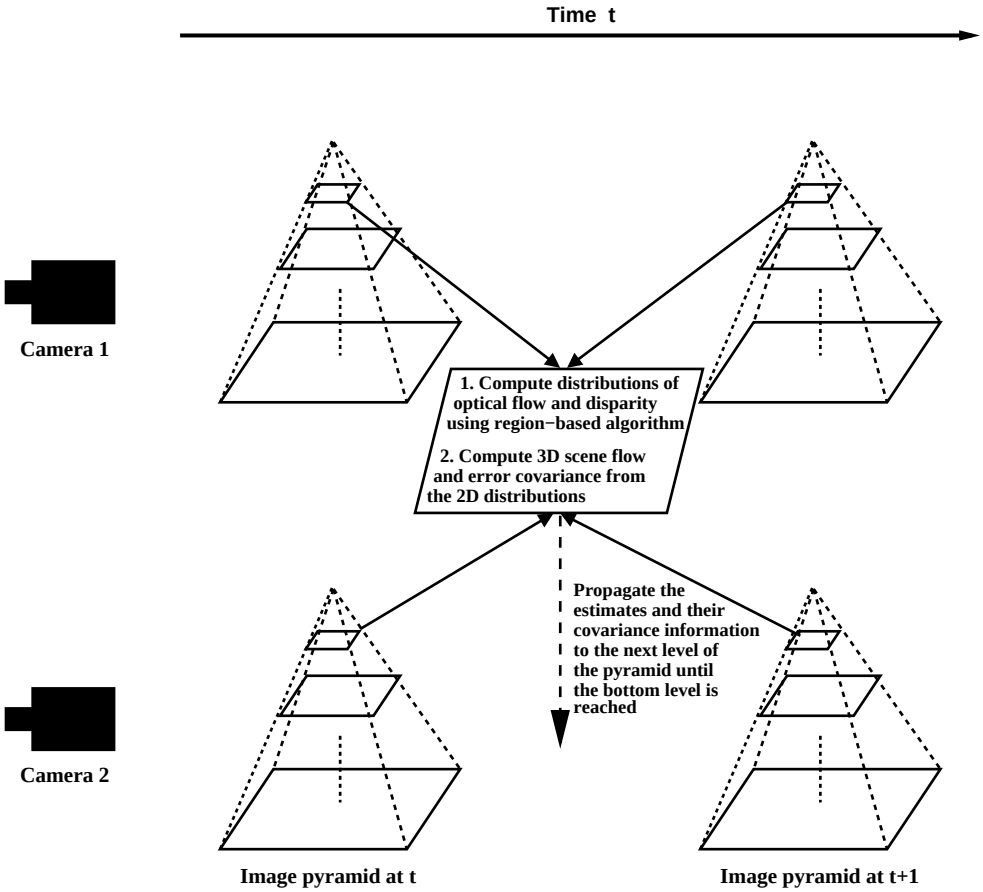

Fig. 1. *Overview*

The overview of the computational steps involved in estimating multi-scale 3D scene flow and its error covariance is shown in Fig. 1. The cameras are calibrated. The captured image streams are synchronized and rectified. At each image pyramid level, first we compute the distributions of optical flow and disparity, then we make use of regions from image segmentation to regularize the 2D displacements (*i.e.*, optical flow and disparity). Details of the region-based approach are described in Sections 3.1 and 3.2. The estimates of the 2D displacements and their associated error covariance are then used in a weighted least squares formulation to estimate 3D scene flow at this level. The 2D displacements and 3D scene flow are then propagated to the next pyramid level. The final estimates are obtained at the lowest pyramid level. The combined algorithm is described in Section 3.3. The integrated approach allows a principled way to propagate 2D information to 3D within a multi-scale framework.

2 Related Work

We broadly classify the related work into two categories. The first category includes methods for 3D motion estimation where the 3D point correspondence is explicitly recovered over time. The second category includes methods dynamic depth map estimation where there is no notion of 3D point correspondence.

2.1 3D Motion Estimation

There has been a fairly large amount of research done in the area of 3D motion estimation. We group the related work into four categories based on the setup and assumptions made.

2.1.1 Rigid Motion, Monocular Sequence

Structure-from-motion techniques [10, 47, 61] recover relative motion together with scene structure from a monocular image sequence. The scene is generally assumed to be rigid [47] or piecewise rigid [10, 61]; thus, only a restricted form of non-rigid motion can be analyzed via these techniques [3].

2.1.2 Non-rigid Motion, Monocular Sequence

By making use of *a priori* knowledge, or by directly modeling assumptions about the scene, techniques like [15, 31, 35, 46, 48, 56] can estimate non-rigid motion from a monocular image sequence.

In [31, 35], a deformable model is used and the 3D motion is recovered by estimating the parameters to deform a predefined model.

The older work of [48] assumes that the motion minimizes the deviation from rigid body motion. Recently, more researchers have discovered that in practice, many non-rigid objects deform with certain structure. Their shapes can be regarded as a rigid component plus a weighted combination of certain shape bases. In [15], the shape bases are first obtained by applying Principal Component Analysis on the shape points of a sparse mesh tracked by a stereo tracking algorithm; then this set of shape bases is used for online model-based monocular facial tracking. In [46], an Expectation-Maximization (EM) method is proposed to simultaneously estimate 3D shape and motion for each time frame. This method learns the parameters of a Gaussian, and robustly fills-in missing data points under the orthographic projection model. In [56], a set of constraints is proposed to eliminate the ambiguity when determining the shape bases. This method can provide a closed-form solution un-

der the weak-perspective projection model. Both [46, 56] are batch methods since tracked feature points from multiple video frames are required. In all three methods [15, 46, 56], only sparse 3D motion can be recovered. Furthermore, the assumption of a linear combination of shape bases may be insufficient to capture more complex and subtle shape variations.

2.1.3 Motion Stereo

With multiple cameras, stereo and 2D motion information can be combined to recover the 3D motion, *e.g.,* [25, 30, 41, 53, 57, 62]. Except for [25] and [30], nearly all techniques in this category assume rigidity of the scene and/or rigidity of the motion. For non-rigid tracking, [25] uses relaxation-based algorithms and [30] generalizes the deformable model-based approach of [31]. The first approach cannot provide dense 3D motion while the latter approach needs *a priori* knowledge of the scene, *i.e.,* the deformable model for the object to be tracked.

2.1.4 Non-rigid Motion, Multi-view

In an approach closely related to ours, Vedula, *et al.,* [51] introduce the concept of dense scene flow as the 3D counterpart of optical flow. They present a linear algorithm to compute scene flow from the optical flow fields of the video streams from multiple cameras. Given scene flow and initial 3D scene structure, dynamic scene structure can be recovered. Zhang *et al.,* [59] reformulate the scene flow estimation problem in terms of energy minimization; scene flow is computed by fitting an affine motion model to image blocks with global smoothness constraints. This algorithm is further improved in [60] so that discontinuities are preserved and occlusions are handled. However, in [60], the depth information in the occluded area is simply ignored although it has been computed; the 3D scene flow in this case is simply estimated from the multiple view optical flow constraints. In [36], Pons, *et.al.,* avoid the explicit computation of optical flow. They back-project the images onto the instantaneous 3D surface of the scene, and the depth recovery and 3D scene flow are modeled within an energy minimization framework. The costs used in the energy minimization are global and local statistical measures on the back-projected images. Occlusions are not handled in this approach. As is the case with other energy minimization approaches in this category [59, 60], some weights on the regularization terms must be determined beforehand [36].

2.2 Dynamic Depth Map

Approaches like [11, 58, 45] recover a dynamic depth map over time by making use of motion constraints, but do not output 3D scene flow. In [45], the scene is assumed to be piecewise planar. Motion constraints are used to predict the depth map of the

planar patch in the next time step. Patches are merged together through a greedy hypothesis testing algorithm that minimizes an image matching cost. Two other approaches in this category [11, 58] recover shape from dynamic scenes by finding correspondence in a 3D space time window. Both approaches are batch methods as multiple frames are needed for constructing the space time window; hence these techniques are not suitable for online algorithms. Furthermore, [11, 58] require active illumination to improve the accuracy in correspondence matching. All three systems only output dynamic depth maps; there is no point-to-point correspondence computed over time, hence there is no 3D scene flow being computed.

Our proposed method aims to compute dense 3D scene flow fields in a multiple camera setup by combining simultaneous 2D optical flow with stereo-depth information. No assumption of the scene or 3D motion is made. In particular, instead of trying to avoid or eliminate the uncertainty in the computation of 3D scene flow, we model, incorporate and propagate the uncertainty information in a principled and consistent way when we solve for the 3D scene flow. The first benefit of incorporating the uncertainty information is that we improve the accuracy of Vedula's [51] method. Another benefit of propagating the uncertainty information is that the propagated 3D uncertainty information is useful for applications like 3D tracking and 3D motion analysis. The proposed algorithm also marries a global method and a segmentation based local method in a multi-scale framework to preserve discontinuities and to fill in information when the areas of images are of low texture. Occlusions are also handled through the region merging process, where occluded regions may take on valid estimates from one of the adjacent regions. In experiments with synthetic data, we show that by incorporating the covariance information from optical flow and disparity, the estimation errors measured in terms of angle and magnitude of the estimated 3D scene flow are at least one standard deviation smaller than those obtained via [51]. In experiments with real data, the results we obtain with a two-camera setup are comparable to the results presented in [60] where three cameras are used.

3 Approach

Both optical flow and disparity estimation can be formulated as problems of finding corresponding points in two images. While optical flow estimation finds such correspondences in images taken at different times, disparity finds them in images captured by cameras in different views. Surveys of related techniques in [5, 39] show that although researchers have worked to estimate optical flow and disparity separately in the past, the techniques used to solve these two problems are similar. These techniques can be generally categorized as: phase-based, energy-based, feature-based and area-based methods. Phase-based methods [13, 21, 24, 54, 55] make use of Fourier phase information. Energy-based methods [1, 2, 19, 20, 28, 33, 34, 37, 40, 43] minimize a cost function plus a regularization term in a vari-

ational framework to solve for the 2D displacements (optical flow and disparity). Feature-based methods [4, 7, 16] match features (*e.g.* points, edges, curves, *etc.*) in two images. Area based methods [12, 14, 22, 23, 27] find optical flow/disparity by correlating image patches across images, *e.g.* correlation, mutual information, *etc.* Thus, closely-related methods have been developed for both optical flow and disparity computation.

To handle the discontinuities when estimating flow and disparity, there are methods that make use of color image segmentation, or a combination of flow/disparity and color, and local region model fitting when computing optical flow [6, 18, 32, 63] and stereo [45, 64]. None of these methods make use of uncertainty information for the flow/stereo estimates.

In this section, we first give a general formulation of the optical flow and disparity problem, followed by the description of a previous global approach by [43] that puts the optical flow estimation problem in a *maximum a posterior* (MAP) framework. The global approach suffers from over-regularization, especially in a multi-scale approach due to coarse-to-fine information propagation. To overcome this problem, we introduce a local motion model fitting method that makes use of image segmentation. The model fitting process involves the use of a weighted least squares method. Hence the output from the motion fitting process is just the linear transformation of the MAP estimates from [43]. The region-based approach is then generalized to address the problem of disparity estimation.

Let $I(x, y, \alpha)$ be the function of pixel position and time/view for the image signal ($\alpha = t$ for time, $\alpha = c$ for camera view). Let $\mathbf{v}$ be the 2D pixel displacement caused by change in time or camera view. Commonly, the goal is to find $\mathbf{v}$ such that

$$\nabla I \cdot \mathbf{v} + I_\alpha = 0. \tag{1}$$

In the following derivations, we use I to represent $I(x, y, \alpha)$ for simplicity. ∇I represents the 2D spatial gradient vector of the image and I_α represents the change in image caused either by time or view. The same equation has been used both in the context of optical flow [20] and stereo vision [29]. This problem is under-constrained as we are given a single linear equation for solving two unknowns ($|\mathbf{v}| = 2$). To get around this, some form of regularization is usually employed. The common formulation is to minimize:

$$E(\mathbf{v}) = \sum_{i \in \Omega} (\nabla I_i \cdot \mathbf{v} + I_{i\alpha})^2, \tag{2}$$

where i is the index of the pixel in a predefined neighborhood Ω. Minimizing Eq. 2 enforces smoothness of 2D pixel displacements in the neighborhood Ω. This smoothness constraint is often violated when there is a large displacement of the

pixels in the neighborhood Ω between the two images captured. To alleviate this problem, multi-resolution based approaches are widely adopted.

Simoncelli *et al.* proposed an approach that computes distributions of optical flow [43]. This approach computes the covariance of the flow estimates at each pixel based on the contrast properties over a local neighborhood at multiple scales. In this paper, we adapt this approach in an improved formulation. Variance estimates from [42] can provide us with the knowledge of the error in the coarser scale estimates so that we can make corrections during the estimation at finer scales. Our proposed approach takes care of the over-smoothing problem of [43] and still preserves the useful property of producing an estimate of the flow distribution at each pixel. Our approach is extended to estimate disparity distributions in stereo. Given the distributions of optical flow and disparity, we compute 3D scene flow via an integrated algorithm using weighted least squares, as described in Section 3.3. As will be seen in the experiments, utilizing flow and disparity distribution information in our formulation to 3D scene flow estimation yields superior results to [51] in the case of just two cameras. Compared to another region-based approach [60], the proposed algorithm is simple, has fewer system parameters to set, and yields good estimation results in the experiments.

3.1 Distributions of Optical Flow

Following [43], the uncertainty in optical flow computation is described through the use of a Gaussian noise model,

$$\nabla I \cdot (\mathbf{v} - \mathbf{n}_1) + I_t = n_2. \tag{3}$$

The image intensity signal is represented as a function I of position (denoted by image coordinates x and y) and time (denoted by t). The image gradient is $\nabla I = (I_x(x, y, t), I_y(x, y, t))^T$ and the temporal derivative of the image is I_t. The first random variable $\mathbf{n}_1 \sim \mathcal{N}(0, \mathbf{\Lambda}_1)$, describes the error resulting from the failure of the planarity assumption (i.e., $\mathbf{v}$ is constant in a small region). The second random variable, $n_2 \sim \mathcal{N}(0, \Lambda_2)$ describes the errors in the temporal derivative measurements.

Based on the constant brightness assumption and from Eq. 3, a MAP estimate of $\mathbf{v}$ can be derived. Let Λ_p be the prior distribution of $\mathbf{v}$, each optical flow vector (per pixel) is considered as a normal distribution with mean flow $\hat{\mathbf{v}}$ and covariance $\Lambda_{\mathbf{v}}$:

$$\Lambda_{\mathbf{v}} = \left[\sum_{i \in \Omega} \frac{g_i \mathbf{M}_i}{\sigma_1^2 \|\nabla I(x_i, y_i, t)\|^2 + \sigma_2^2} + \Lambda_p^{-1} \right]^{-1}, \tag{4}$$

$$\hat{\mathbf{v}} = -\Lambda_{\mathbf{v}} \cdot \sum_i \frac{g_i \mathbf{b}_i}{\sigma_1^2 \|\nabla I(x_i, y_i, t)\|^2 + \sigma_2^2}, \tag{5}$$

where

$$\mathbf{M} = \nabla I \nabla I^T = \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} I_x I_t \\ I_y I_t \end{pmatrix},$$

and g_i is the weight assigned to the neighboring pixel i, $\sigma_1^2 \mathcal{I} = \mathbf{\Lambda}_1$ ($\mathcal{I}$ is the identity matrix) and $\sigma_2^2 = \Lambda_2$. In practice, the neighborhood Ω often refers to a m by m window and a 2D Gaussian filter $\mathbf{g}$ of the same size is applied to get the weighted sums in Eq. 4 and Eq. 5 [43]. The inclusion of Λ_p makes Eq. 4 well-conditioned.

3.1.1 Coarse-to-Fine Estimation of Flow Distribution

The constant brightness assumption made in deriving Eq. 4 and 5 is often violated when relatively large motion (typically ≥ 2 pixels) occurs, to alleviate this problem and to propagate the uncertainty information at coarser scale levels (lower-resolution images) to finer scale levels (higher-resolution images), Simoncelli developed a filter-based coarse-to-fine algorithm [42]. Instead of the traditional application of Kalman filtering to propagate information over time, [42] propagates information across different scales. We only describe the basic solution here.

First, we define a *state evolution* equation for the estimated flow field $\hat{\mathbf{v}}$,

$$\hat{\mathbf{v}}^l = \mathbf{E}^{l-1} \hat{\mathbf{v}}^{l-1} + \mathbf{n}_0, \quad \mathbf{n}_0 \sim \mathcal{N}(0, \mathbf{\Lambda}_0), \tag{6}$$

where l is an index for scale such that larger values of l correspond to higher resolution level in the image pyramid. $\mathbf{E}$ is a linear interpolation operator used to extend the coarse scale flow fields to the finer scale flow fields, which is analogous to the state evolution matrix in the standard Kalman filtering framework. The random variable $\mathbf{n}_0$ represents the uncertainty of the prediction of the finer-scale flow fields from the coarser-scale flow fields; it is assumed to be point-wise independent, zero-mean and normally-distributed.

The measurement equation is defined based on Eq. 3:

$$-I_t^l = \nabla I^l \cdot \mathbf{v}^l + (n_2 + \nabla I^l \cdot \mathbf{n}_1). \tag{7}$$

Applying the standard Kalman filter framework (replacing the traditional Kalman filter time index t with scale index l), given Eq. 6 and Eq. 7, an optimal estimator

for $\mathbf{v}^l$ is derived from the estimate of the coarse scale $\hat{\mathbf{v}}^{l-1}$ and a set of fine scale derivative measurements:

$$
\begin{aligned}
\mathbf{\Lambda}'^l &= \mathbf{E}^{l-1}\mathbf{\Lambda}_{\mathbf{v}}^{l-1}(\mathbf{E}^{l-1})^T + \mathbf{\Lambda}_0 \ , \\
K^l &= \frac{\mathbf{\Lambda}'^l \nabla I^l}{(\nabla I^l)^T[\mathbf{\Lambda}'^l + \mathbf{\Lambda}_1]\nabla I^l + \Lambda_2} \ , \\
\nu^l &= -I_t^l - (\nabla I^l)^T \mathbf{E}^{l-1}\hat{\mathbf{v}}^{l-1} \ , \\
\hat{\mathbf{v}}^l &= \mathbf{E}^{l-1}\hat{\mathbf{v}}^{l-1} + K^l\nu^l \ , \\
\mathbf{\Lambda}_{\mathbf{v}}^l &= \mathbf{\Lambda}'^l - K^l(\nabla I^l)^T\mathbf{\Lambda}'^l \ .
\end{aligned} \tag{8}
$$

In the above equations, K^l is the *Kalman gain* and ν^l is the process innovation which is approximated as the temporal derivative of the warped images. The Kalman filter is directly related to recursive weighted least squares [44]. In our case, it is used to propagate covariance information from the previous spatial scale to the current spatial scale. A justification for this approximation and more details can be found in [42].

3.1.2 Region-based Parametric Model Fitting

Simoncelli's approach [43] tends to over-smooth the solution due to: (1) uniform window size for defining a neighborhood, and (2) level to level propagation of information.

One solution to the problem of uniform window size is to use window sizes that are adaptive to local image properties. Given that information propagation is actually the desirable property of a multi-scale approach, it is hard to address the over-smoothing problem caused by level-to-level information propagation. To solve this problem, we take inspiration from [6] by making use of a parametric model to fit flow within the regions obtained from image segmentation. In the motion estimation literature [6, 52], it is commonly assumed that motion of the pixels within the same region can be fitted to a parametric model. Following the conventions in [6, 52], we give a brief description of the motion model fitting process.

For each pixel, denoted by its coordinates, $\mathbf{x}_i = (x_i, y_i)$, within the same region, one of the following models is selected by the algorithm to fit flow vectors:

$$
\mathbf{F}(\mathbf{x}_i) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \mathbf{a} = [a_0 \ a_3]^T,
$$

$$
\mathbf{F}(\mathbf{x}_i) = \begin{bmatrix} 1 & x_i & y_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x_i & y_i \end{bmatrix}, \ \mathbf{a} = [a_0 \ a_1 \ a_2 \ a_3 \ a_4 \ a_5]^T,
$$

$$\mathbf{F}(\mathbf{x}_i) = \begin{bmatrix} 1 & x_i & y_i & x_i^2 & x_i y_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_i y_i & y_i^2 & 1 & x_i & y_i \end{bmatrix}, \ \mathbf{a} = [a_0 \ a_1 \ a_2 \ a_6 \ a_7 \ a_3 \ a_4 \ a_5]^T.$$

The two-parameter model corresponds to translation, the six-parameter model corresponds to affine motion, and the eight-parameter model corresponds to quadratic motion. The specifications of these models follow the convention of [6, 52].

Minimizing the following weighted least squares equation yields an estimate of the model parameters $\mathbf{a}_r$ for region r,

$$\hat{\mathbf{a}}_r = \arg\min_{\mathbf{a}_r} \sum_{i}^{r} (\mathbf{v} - \mathbf{F}(\mathbf{x}_i)\mathbf{a}_r)^T \Lambda_{\mathbf{v}}^{-1} (\mathbf{v} - \mathbf{F}(\mathbf{x}_i)\mathbf{a}_r). \tag{9}$$

Though this formulation is similar in spirit to that of [6], the robust error norm is not used as we have an uncertainty model for $\mathbf{v}$ from Eq. 4 and Eq. 5, which explicitly models the local image intensity information. Pixels in the region with reliable estimates of $\mathbf{v}$ naturally carry more weight in the fitting process. These pixels correspond to edge pixels or the regions with rich texture. Hence the fitting tends to be more robust. In this combined approach, the cost function of Eq. 9 is still convex and guaranteed to have an optimal solution given enough pixels within the region. Let $\hat{\mathbf{a}}$ be the optimal solution, the updated flow field $\hat{\mathbf{v}}'$ and corresponding covariance $\Lambda_{\mathbf{v}}'$ are computed as follows:

$$\begin{aligned} \hat{\mathbf{v}}' &= \mathbf{F}(\mathbf{x}_i)\hat{\mathbf{a}}, \\ \Lambda_{\mathbf{a}} &= (\mathbf{J}(\mathbf{x}_i)^T \Lambda_{\mathbf{v}}^{-1} \mathbf{J}(\mathbf{x}_i))^{-1}, \\ \Lambda_{\mathbf{v}}' &= \mathbf{F}(\mathbf{x}_i) \Lambda_{\mathbf{a}} \mathbf{F}(\mathbf{x}_i)^T, \end{aligned} \tag{10}$$

where $\mathbf{J}(\mathbf{x}_i)$ is the Jacobian of $\mathbf{F}(\mathbf{x}_i)$ evaluated at $\mathbf{x}_i$.

In the combined approach, first, image segmentation based on color/intensity information is performed at each resolution level of the image pyramid. In our implementation, segmentation is obtained via mean shift [9]. The order of the parametric model used for fitting follows the same set of rules as defined in [6]. Hence, the order of the parametric model is adaptive to the resolution level, region size and fitting residual error. A lower-order model is always preferred if a higher-order model fails to reduce the error residual. When the residual error of fitting an eight-parameter model is still high and the region size is large, the region is split by using mean shift [9] on the region flow field as color/intensity information alone is not enough. Model fitting is then performed on the newly split regions. This step can be recursive; the stopping criteria is either the region is small enough or the error residual is below a specified threshold. Figure 3.1.2 shows the process of computing optical flow and the resulting flow field on the Yosemite sequence [17]. In

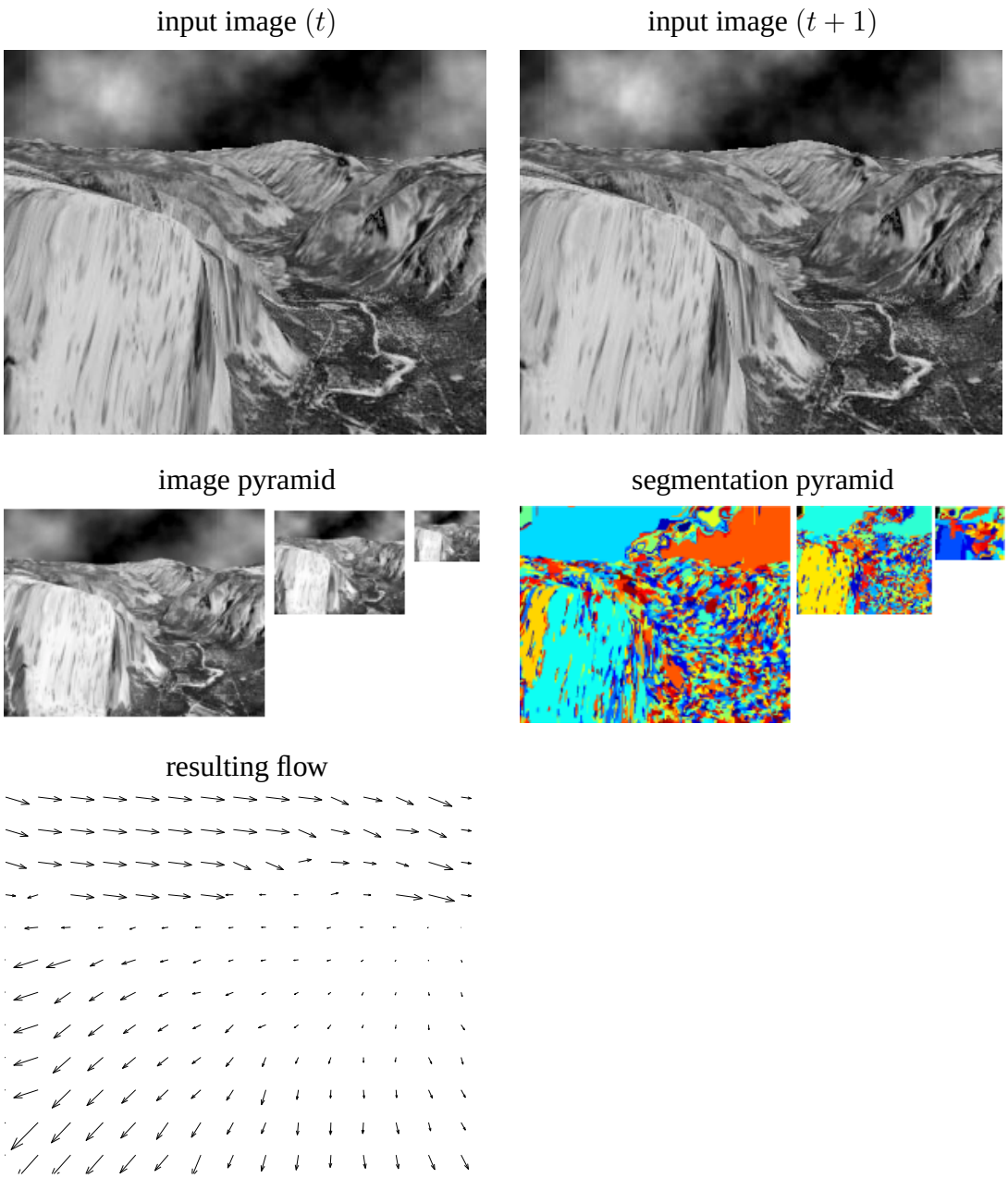

Fig. 2. *Example of flow computation for Yosemite sequence [17].*

Table 1, the angular error indicates that by using the combined approach, we are able to reduce the angular error in terms of mean and standard deviation compared to [6, 42]. Our method produced error close to the state of the art method [34], but it is pointed out later in [8] that the ground truth delivered with the Yosemite sequence has problems, making it difficult to assess the significance of slight difference of error reported for this sequence. Thus we consider the performance respectable, while also suitable for use in our unified framework for disparity and 3D scene flow estimation.

Method	Average Angular Error (AAE)	Standard Deviation of AAE
Simoncelli, *et al.* [42]	3.81°	7.09°
Black-Jepson [6]	2.29°	2.25°
Our Method	1.17°	2.08°
Brox,*et al.* [34]	0.99°	1.12°

Table 1
Evaluation results of the proposed approach on the Yosemite sequence. Flow computed at the cloudy sky area is not used for error computation.

3.2 Distributions of Disparity

Similar region model fitting algorithms [26, 45] have also been used for depth map computation so that sharp boundaries can be preserved. Hence with slight modification, the same integrated algorithm for optical flow computation can be used to compute disparity for input image pairs captured by a stereo rig. To do this, we just substitute the time index t and $t+1$ with the view index $\mathbb{L}$ and $\mathbb{R}$, where $\mathbb{L}$ refers to *left* view and $\mathbb{R}$ refers to the *right* view in a binocular stereo rig. Only horizontal displacement and corresponding variances are computed. Hence the estimation of disparity can be solved in the same way as optical flow. By using the method to estimate optical flow and disparity, we are able to combine them together in a consistent way for 3D scene flow estimation. Fig. 3 shows the process of computing disparity for the standard Teddy data set [50]. We have tested the performance of our algorithm using the evaluation tool and data set provided at `http://www.middlebury.edu/stereo`, our algorithm on average ranks among the top ten on the data set provided as shown in Table 2.

One of the evaluation parameters provided by the online evaluation system [50] is "E.T.", which stands for error threshold. It is the acceptable disparity error. If $E.T. = 1$, then the estimated disparity is considered correct if the difference between the ground truth disparity and the estimated disparity is within 1 pixel, otherwise, it is considered wrong. The evaluation system allows us to adjust the value from $0.5 - 2.0$ pixels. As shown in Table 2, our method ranks higher when the acceptable disparity threshold is larger. We believe this is due to the smoothing effect of the derivative filters used in our 2D displacement computation. The results obtained by our method on all four data sets are shown in Fig. 4.

A single consistent algorithm for computing optical flow and disparity is summarized in Algorithm 1. We use $\mathbf{v}$ in Algorithm 1 to represent optical flow and disparity as they are both 2D pixel displacements between two images I_1 and I_2.

The proposed unified algorithm presented in Alg. 1 is a two-stage algorithm. In the first stage, as with [42], the estimation of 2D displacements exploits the constant brightness assumption. Hence the proposed algorithm may have problems when estimating 2D correspondences for objects in the scene that have non-Lambertian

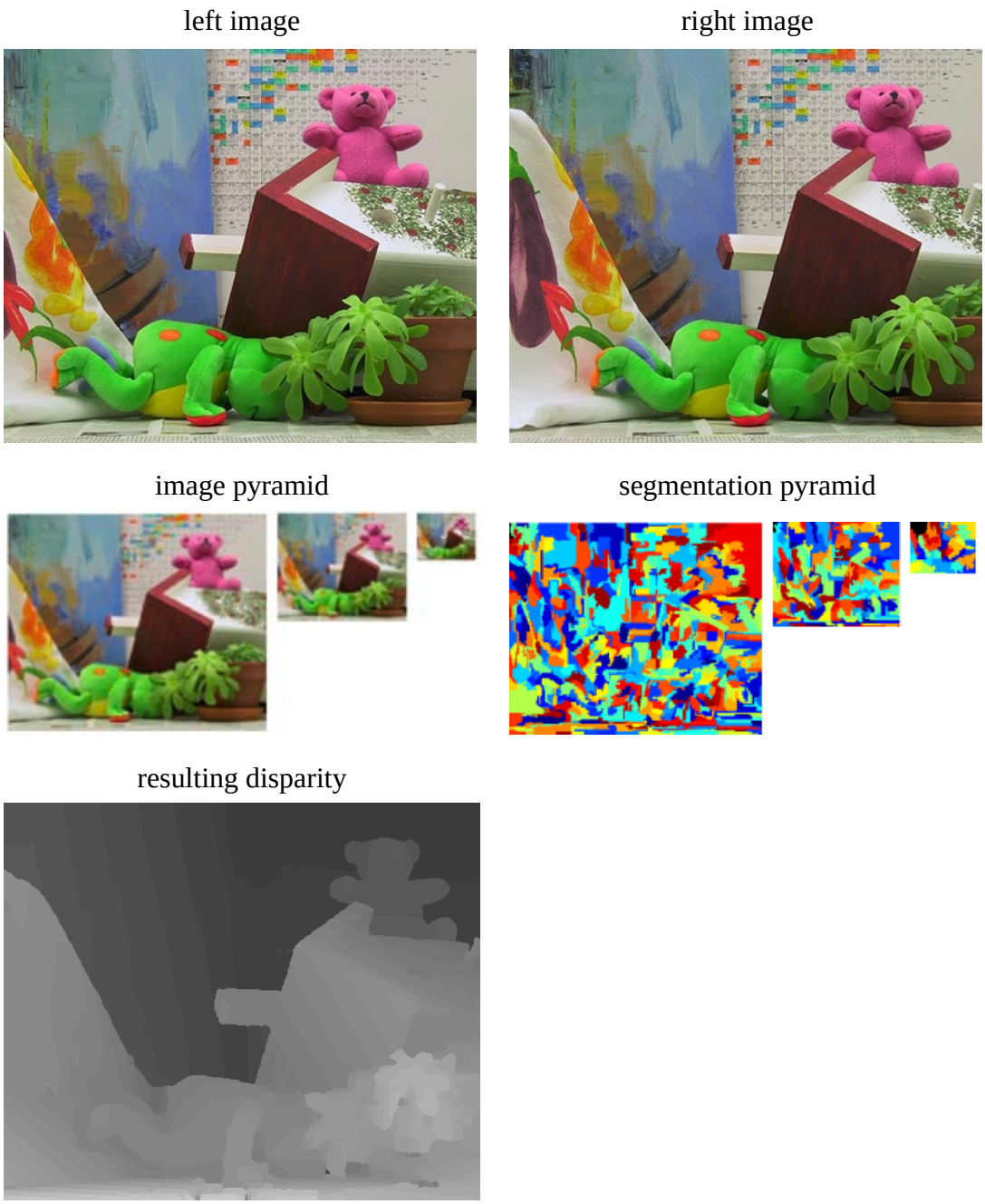

Fig. 3. *Example of disparity computation for Teddy data set [50].*

surface reflectance properties, or in the case of large baseline stereo. The second stage region model fitting process used in our algorithm helps to alleviate the large baseline problem. In experiments with real video data captured with relatively large baseline stereo cameras (the observed largest displacement is around 100 pixels between a stereo pair), the proposed algorithm is still able to produce reasonable estimates due to the region merging step where the information fill-in takes place.

In our experience, our algorithm gives a good estimate of optical flow and dispar-

E.T.	A.R.	Tsukuba			Venus			Teddy			Cones		
		nocc	all	disc	nocc	all	disc	nocc	all	disc	nocc	all	disc
0.5	10.3	21.5	22.5	23.7	5.81	6.12	9.73	16.6	21.7	33.7	11.3	17.5	18.8
0.75	9.4	21.0	21.9	21.3	0.45	0.68	3.11	10.2	14.9	24.3	5.79	12.0	12.7
1.0	7.8	1.64	2.65	8.75	0.13	0.30	1.86	7.59	11.7	19.3	4.74	10.7	10.8
1.50	5.8	1.02	1.83	5.40	0.13	0.30	1.83	4.48	7.12	13.3	3.76	9.21	9.25
2.0	5.4	0.69	1.40	3.74	0.12	0.27	1.72	3.15	4.91	10.0	3.24	8.37	8.32

Table 2
Evaluation results of the proposed approach on the stereo data set [50] using the evaluation measures of [38] as of August 28, 2006. The numbers represent the percentage of bad pixels (i.e., pixel whose absolute disparity error is greater than "E.T."). "E.T." stands for Error Threshold, which is the acceptable disparity error. "A.R." indicates the Average Ranking of our algorithm against 36 other algorithms listed in [50]. The errors reported in column "nocc" are the errors only evaluated in the non-occluded areas in the image; the errors reported in column "all" are the errors evaluated in the whole image excluding the border regions; and the errors reported in "disc" are the errors evaluated in the regions near depth discontinuities (including both neighborhoods of depth discontinuities and half-occluded regions).

Algorithm 1 Unified Algorithm for Computing Optical Flow and Disparity

construct image pyramids of L levels using the images I_1 and I_2.
initialize Λ_p and Λ_0 used in Eqs. 4 and 6.
for $l = 0$ to $L - 1$ of the pyramids built **do**
  if $l == 0$ **then**
    compute $\hat{\mathbf{v}}^l$, $\Lambda_{\mathbf{v}}^l$ [Eqs.4 and 5],
  else
    compute $\hat{\mathbf{v}}^l$, $\Lambda_{\mathbf{v}}^l$ [Eq.8].
  end if
  segment image at current level of the pyramid built from I_1, e.g., via mean shift [9].
  for each region from the segmentation **do**
    fit a parametric model to the $\hat{\mathbf{v}}^l$ of the current region according to the process described in [Section 3.1.2], the simplest translation model is always used first.
    compute $\hat{\mathbf{v}}'$, $\Lambda_{\mathbf{v}}'$ [Eq. 10] for pixels in that region.
  end for
  set $\hat{\mathbf{v}}^l = \hat{\mathbf{v}}'$ and $\Lambda_{\mathbf{v}}^l = \Lambda_{\mathbf{v}}'$.
end for

ity. However, one could argue that we should just use the most accurate algorithms available to solve optical flow and disparity, and then combine them together to solve for 3D scene flow. The key point here is that the inaccuracies in estimating optical flow fields and disparities are inevitable. It is often more desirable to explicitly model the inaccuracies of optical flow and disparity based image properties. Once we gain estimates of both the distributions of optical flow and disparity, we

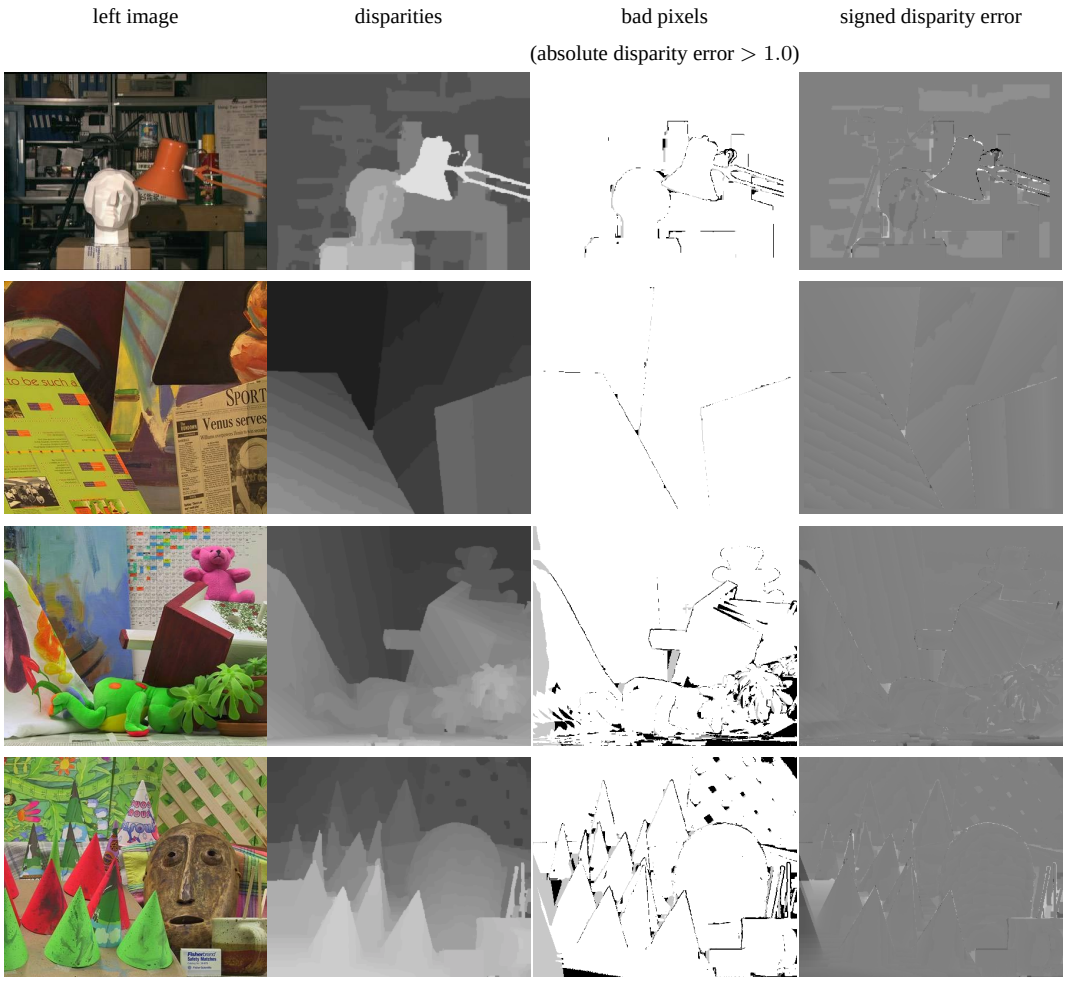

Fig. 4. *Results on all 4 data sets [50]. The first column shows the left image of the input stereo pair. The output disparity maps are displayed in the second column. The error threshold is* 1.0*, hence pixels have absolute disparity error* > 1.0 *pixels are labeled as bad pixels in the third column. The signed disparity error maps are displayed in the last column.*

can use and propagate this uncertainty information in the computation of the 3D scene flow. The end result of this principled way of modeling and propagating 2D uncertainty information is that we obtain better estimates of 3D scene flow together with the propagated error covariance information of the 3D estimates.

3.3 Computing 3D scene flow

Given the unified formulation for optical flow and disparity, we now formulate the computation of 3D scene flow. We will assume that the cameras are fully-calibrated and do not change in the experiments. Following [51], scene flow is defined as the 3D motion field of the points in the world, just as optical flow is the 2D motion field of the points in an image. Optical flow is simply the projection of the scene flow onto the image plane of a camera.

Given a 3D point $\mathbf{X} = (X, Y, Z)$, the 2D image of this point in view c is denoted as $\mathbf{x}_c = (x_c, y_c)$. The 2D components of $\mathbf{x}_c$ are

$$x_c = \frac{[\mathbf{P}_c]_1(X,Y,Z,1)^T}{[\mathbf{P}_c]_3(X,Y,Z,1)^T}, \quad y_c = \frac{[\mathbf{P}_c]_2(X,Y,Z,1)^T}{[\mathbf{P}_c]_3(X,Y,Z,1)^T}, \tag{11}$$

where $[\mathbf{P}_c]_j$ is the j^{th} row of the projection matrix $\mathbf{P}_c$. If the camera is not moving, then the 2D motion $\mathbf{v} = \frac{d\mathbf{x}_c}{dt}$ is uniquely determined by the following:

$$\frac{d\mathbf{x}_c}{dt} = \frac{\partial \mathbf{x}_c}{\partial \mathbf{X}} \frac{d\mathbf{X}}{dt}. \tag{12}$$

To solve for the scene flow $\mathbb{V} = \frac{d\mathbf{X}}{dt}$, two equations are needed. Hence at least two cameras are needed. The setup of the system of equations is simply

$$\mathbf{B}\mathbb{V} = \mathbf{U}, \tag{13}$$

where

$$\mathbf{B} = \begin{bmatrix} \frac{\partial x_{c_1}}{\partial X} & \frac{\partial x_{c_1}}{\partial Y} & \frac{\partial x_{c_1}}{\partial Z} \\ \frac{\partial y_{c_1}}{\partial X} & \frac{\partial y_{c_1}}{\partial Y} & \frac{\partial y_{c_1}}{\partial Z} \\ \cdot & \cdot & \cdot \\ & & \\ \cdot & \cdot & \cdot \\ \frac{\partial x_{c_N}}{\partial X} & \frac{\partial x_{c_N}}{\partial Y} & \frac{\partial x_{c_N}}{\partial Z} \\ \frac{\partial y_{c_N}}{\partial X} & \frac{\partial y_{c_N}}{\partial Y} & \frac{\partial y_{c_N}}{\partial Z} \end{bmatrix}, \ \mathbf{U} = \begin{bmatrix} \frac{\partial x_{c_1}}{\partial t} \\ \frac{\partial y_{c_1}}{\partial t} \\ \cdot \\ \\ \cdot \\ \frac{\partial x_{c_N}}{\partial t} \\ \frac{\partial y_{c_N}}{\partial t} \end{bmatrix}. \tag{14}$$

A singular value decomposition of $\mathbf{B}$ gives the solution that minimizes the sum of least squares of the error obtained by re-projecting the scene flow onto each of the optical flows.

3.4 Integrated Approach

As discussed in Section 2, it is known that the 2D image correspondence problem (across different views or across different time frames) is ill-posed. Hence it is difficult to estimate scene flow reliably from optical flow and disparity. One way to get around this is to use many cameras, as reported in [51], where 51 cameras were used to solve Eq. 13 reliably.

Rather than aiming to improve the accuracy by using more cameras, we propose to incorporate the covariances derived from the computation of optical flow and disparity. By taking the covariances from disparity and optical flow into account, the linear system of Eq. 13 tends to produce reasonable scene flow even when given only a small number of cameras. Furthermore, the estimated scene flow with covariances can be used for applications like probabilistic 3D tracking and 3D motion and structure analysis.

For a stereo pair, the 3D coordinate $\mathbf{X}$ is related to the disparity d and corresponding image coordinates $\mathbf{x}_{\mathbb{L}}$ and $\mathbf{x}_{\mathbb{R}}$ where $\mathbb{L}$ indicates left view and $\mathbb{R}$ indicates right view. Let T denote the baseline and f denote the focal length (both cameras are assumed to have the same focal length). The following equation defines the relationship between the 3D coordinates, 2D image coordinates in the left and right cameras, and the pixel disparity between left and right cameras.

$$X = \frac{T(x_{\mathbb{L}} + x_{\mathbb{R}})}{2d},\ Y = \frac{T(y_{\mathbb{L}} + y_{\mathbb{R}})}{2d},\ Z = \frac{fT}{d}. \tag{15}$$

Hence we solve Eq. 14 for scene flow, $\mathbb{V}$ by:

$$\hat{\mathbb{V}} = \arg\min_{\mathbb{V}} (\mathbf{B}\mathbb{V} - \mathbf{U})^T \mathbf{W}^{-1} (\mathbf{B}\mathbb{V} - \mathbf{U}). \tag{16}$$

In the binocular setup, $\mathbf{W}$ is derived from the 2D covariances from the disparity d where $\Lambda_d = diag(\sigma_d^2, \sigma_d^2)$, and the 2D flow field $\mathbf{v}$ where $\Lambda_{\mathbf{v}} = diag(\sigma_{\mathbf{v}_x}^2, \sigma_{\mathbf{v}_y}^2)$. Assuming independence of the estimates of optical flow and disparity, then

$$\mathbf{W} = \Lambda_d \Lambda_{\mathbf{v}}. \tag{17}$$

By covariance propagation, the error covariance of scene flow $\mathbb{V}$ is:

$$\Lambda_{\mathbb{V}} = (\mathbf{B}^T \mathbf{W}^{-1} \mathbf{B})^{-1} = \mathbf{B}^{-1} \mathbf{W} \mathbf{B}^{-T}. \tag{18}$$

Algorithm 2 describes the single integrated method for computing optical flow, disparity and 3D scene flow. To compute the scene flow for two consecutive frames in the stereo video streams, we use $I(\mathbb{L})$ to denote the left video stream and $I(\mathbb{R})$ to denote the right video stream. First we build image pyramids of height L for $I(t, \mathbb{L}),\ I(t+1, \mathbb{L}),\ I(t, \mathbb{R})$ and $I(t+1, \mathbb{R})$. Pyramid images are indexed by l, where $l = 0$ is the index for image at the lowest resolution level and $l = L - 1$ is the index for image at the highest resolution level. Hence $I^l(t, \mathbb{L})$ refers to the pyramid image at level l and the image that is used to construct the pyramid is captured by left camera at time t. The optical flow fields computed at each level of the pyramid for the binocular views are denoted as $\mathbf{v}^l(\mathbb{L})$ and $\mathbf{v}^l(\mathbb{R})$. Disparity is denoted as d^l.

Algorithm 2 Algorithm for computing 3D scene flow

construct image pyramids of L levels using the images $I(t,\mathbb{L})$, $I(t+1,\mathbb{L})$, $I(t,\mathbb{R})$ and $I(t+1,\mathbb{R})$.
initialize Λ_p and Λ_0 used in Eqs. 3 and 5.
for $l=0$ to $L-1$ **do**
  if $l==0$ **then**
    compute$\Lambda^l_{\mathbf{v}(\mathbb{L})}$, $\mathbf{v}(\hat{\mathbb{L}})^l$, $\Lambda^l_{\mathbf{v}(\mathbb{R})}$, $\mathbf{v}(\hat{\mathbb{R}})^l$, Λ^l_d and $\hat{d}^l$ using Eqs.4 and 5,
  else
    compute $\Lambda^l_{\mathbf{v}(\mathbb{L})}$, $\mathbf{v}(\hat{\mathbb{L}})^l$, $\Lambda^l_{\mathbf{v}(\mathbb{R})}$, $\mathbf{v}(\hat{\mathbb{R}})^l$, Λ^l_d and $\hat{d}^l$ using Eq.8,
  end if
  segment $I^l(t,\mathbb{L})$ and $I^l(t,\mathbb{R})$ via mean shift[9],
  for each region of $I^l(t,\mathbb{L})$ from the segmentation **do**
    fit a parametric model to the flow computed between $I^l(t,\mathbb{L})$ and $I^l(t+1,\mathbb{L})$ and disparity of $I^l(t,\mathbb{L})$ and $I^l(t,\mathbb{R})$ of the pixels in the region [Section 4.1.2],
    compute $\hat{\mathbf{v}}'(\mathbb{L})$, $\Lambda'_{\mathbf{v}}(\mathbb{L})$, $\hat{d}'$ and Λ'_d [Eq. 10] for every pixel in that region,
  end for
  for each region of $I^l(t,\mathbb{R})$ from the segmentation **do**
    fit a parametric model to the flow computed between $I^l(t,\mathbb{R})$ and $I^l(t+1,\mathbb{R})$ of the pixels in the region [Section 4.1.2],
    compute $\hat{\mathbf{v}}'(\mathbb{R})$, $\Lambda'_{\mathbf{v}}(\mathbb{R})'$, $\hat{d}'$ and Λ'_d [Eq. 10] for pixels in that region,
  end for
  set $\hat{\mathbf{v}}^l(\mathbb{L})=\hat{\mathbf{v}}'(\mathbb{L})$, $\hat{\mathbf{v}}^l(\mathbb{R})=\hat{\mathbf{v}}'(\mathbb{R})$. $\Lambda^l_{\mathbf{v}(\mathbb{L})}=\Lambda_{\mathbf{v}'(\mathbb{L})}$, $\Lambda^l_{\mathbf{v}(\mathbb{R})}=\Lambda_{\mathbf{v}'(\mathbb{R})}$, $\hat{d}^l=\hat{d}'$ and $\Lambda^l_d=\Lambda'_d$,
  if $l==0$ **then**
    solve $\hat{\mathbb{V}}^l$ [Eq. 16],
  else
    solve $\hat{\mathbb{V}}^l$ [Eq. 16], using $\hat{\mathbb{V}}^{l-1}$ as the initial estimate,
  end if
end for

4 Experiments

Two sets of experiments are conducted to demonstrate the effectiveness of the weighted least squares method and the performance of the algorithm.

4.1 Synthetic 3D Data

To show the effectiveness of the weighted least squares method, 3600 3D points on a planar surface with known 3D scene flow, 2D optical flow and disparity are generated. The motion of the points on the surface follow a deforming Gaussian surface. Hence, each point moves in slightly different direction with different magnitude,

which corresponds to a non-rigid motion. Gaussian noise with different variances is added to the 2D optical flow and disparity. The magnitudes of the noise variances range from 2% to 10% of the average displacements (i.e., 2D optical flow and disparity). Three methods are tested with and without propagating the noise estimated while computing 2D optical flow and disparity. Accuracy of the computed 3D scene flow is measured using the average angular error and average magnitude between computed 3D scene flow and known 3D motion. The mean and standard deviation of the angular and magnitude error of the estimated 3D scene flow are reported based on the average of 10 runs of the experiments at each noise level.

Method 1: Eq. 13 without incorporating covariance [51].

Method 2: Eq. 16 where only the covariance of 2D optical flow is used.

Method 3: Eq. 16 where both the covariance of 2D optical flow and the variance disparity are used.

Fig. 5 shows the mean and standard deviation of angular and magnitude error and Fig. 6 shows the results of sample frames based on the recovered 3D scene flow. The insight we gain from these experimental results is that taking noise into consideration yields more reliable 3D scene flow estimates. When estimating 3D scene flow with real image data, the computations of optical flow and disparity are always inaccurate due to the camera noise and the image properties, *e.g.*, image regions with no texture or repetitive texture, image regions with low contrast and motion blur when we capture image sequences, *etc*. Hence, even when equipped with the most accurate optical flow and disparity algorithms, the 2D image quantities still cannot always be evaluated accurately. However, by carefully choosing the right algorithms to account for these errors and taking them into consideration when estimating 3D scene flow, we can improve the accuracy of estimates. The importance and effectiveness of Algorithm 2 are demonstrated with real video sequences in the next experiment.

4.2 Real Videos

To evaluate the algorithm in practical applications, experiments have been conducted with videos of real scene sequences. In all the experiments conducted, $\sigma_1 = 0.008$ (σ_1 is related to the error from the failure of the assumption that the displacements are constant in a small region), $\sigma_2 = 0.0$ (σ_2 is related to error when computing temporal derivatives) and $\sigma_0 = 0.10$ (σ_0 is diagonal entry in the covariance matrix during information propagation) and initial $\sigma_p = 0.5$ (σ_p represents the prior distribution information of the displacements). Five-level image pyramids are used in all the test cases. These parameters are determined during experiments. The results from three different video sequences are presented in Figs. 7 and 8.

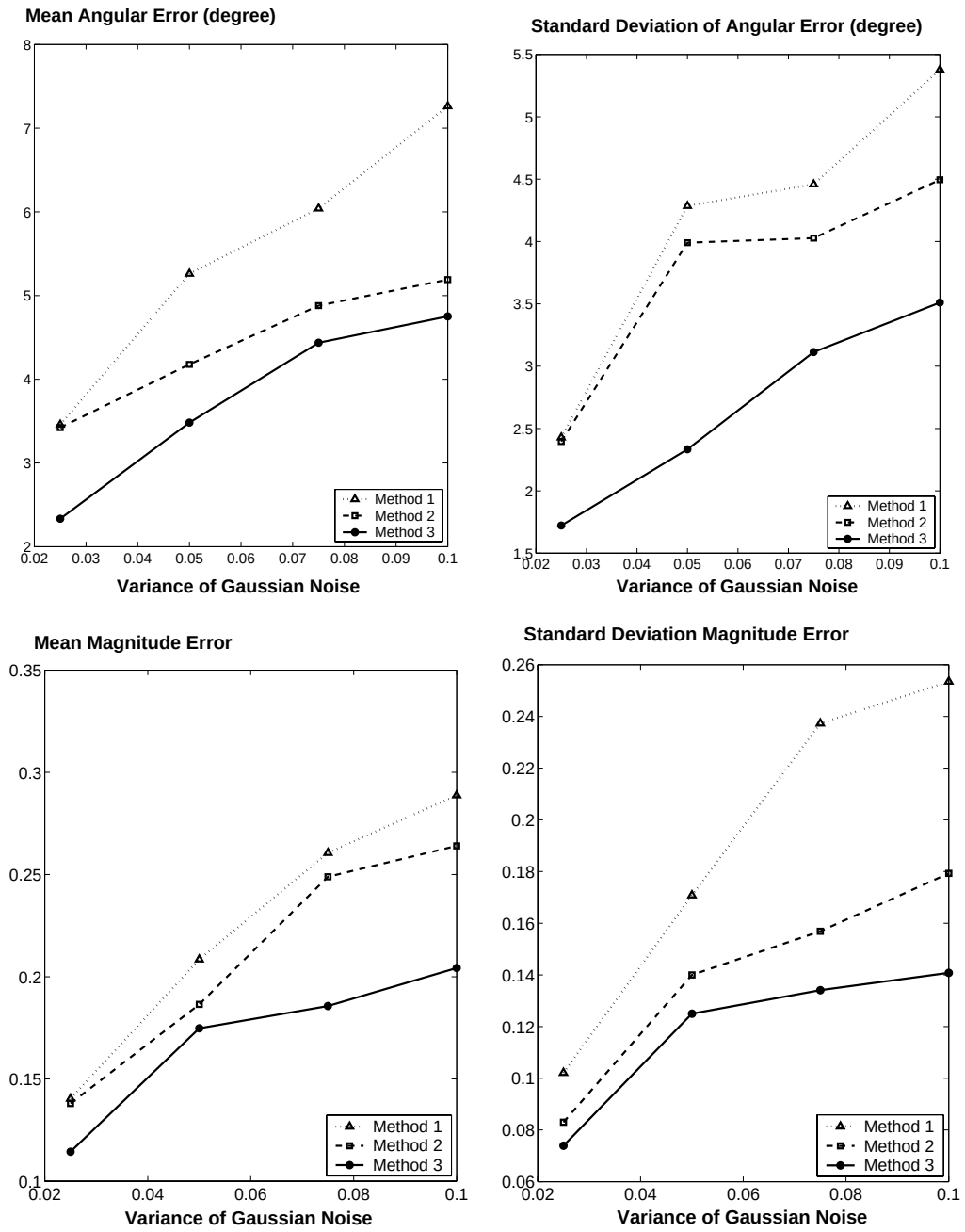

Fig. 5. *Angular error (first row) and magnitude error (second row) of synthetic data with added Gaussian noise.*

The sequences were captured with Videre MEGA-D system: a binocular stereo camera connected with Matrox capture card through fire wire cable. The frame rate of the stereo sequence is around 30 frames/sec with resolution of 320×240. The scene flow algorithm is implemented Matlab and C++. Experiments were conducted on an AMD Athlon MP $2100+$ machine. Dense scene flow is computed for each frame in about two minutes per frame. The acquired sequences are rectified and the calibration information is given. The binocular video sequences are

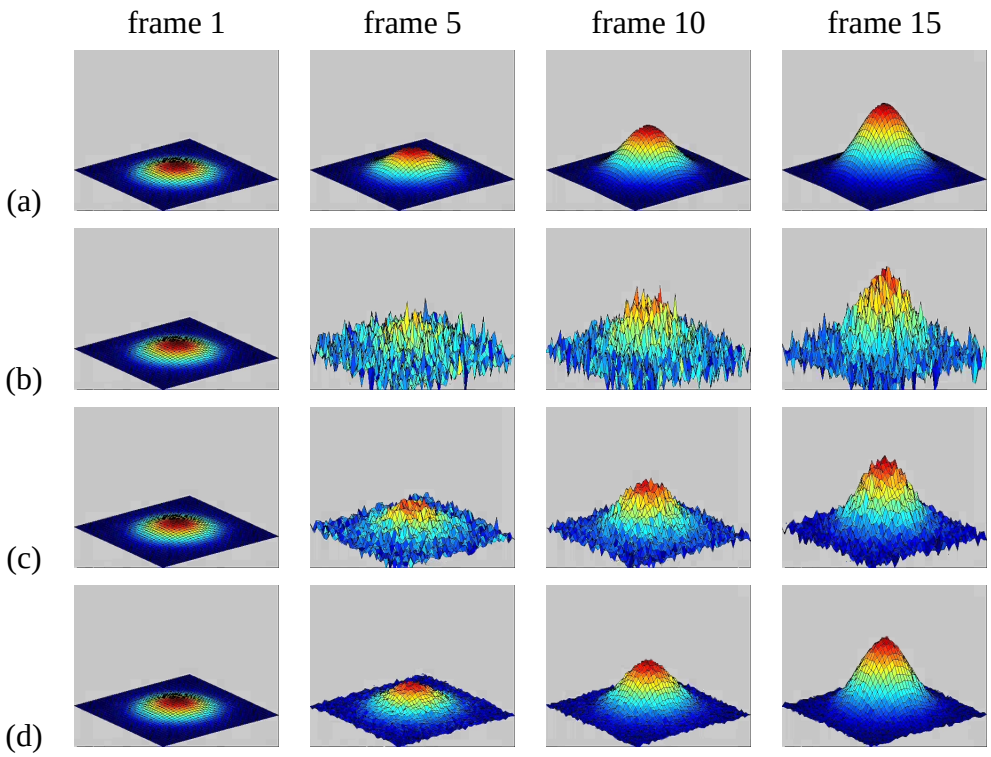

Fig. 6. *Results of sample frames from the synthetic data (noise level = 5%). (a) surface deformation estimated by the groundtruth 3D scene flow; (b) surface deformation estimated by Method 1 [51]; (c) surface deformation estimated by Method 2; (d) surface deformation estimated by Method 3.*

acquired in an uncontrolled illuminated environment as seen in Fig. 7; hence the estimates of optical flow and disparity tend to be noisy.

In Fig. 7, the observable motions in the first sequence are the backward movement of right hand and the forward movement of left hand. The second row of Fig. 8 shows the 2D projection of the estimated 3D flows in the left and right views, the Z velocities and the variances. From the result, we can see that the 3D movements of the left and right hands have been described reliably. In the second sequence, both hands are moving forward. The projections and Z motion of the 3D flow demonstrate similar reliability.

To show the ability to recover non-rigid motion, we captured a sequence with a deforming sponge (by pushing the top of the sponge) with painted texture patterns. Only the upper part of sponge has obvious Z motion (towards the camera) and the rest of the sponge only moves slightly towards the table. This is shown clearly in projected flows and the recovered Z motion. The variances of the Z velocities are shown in the last column of images. The variances give information about the reliability of the estimates. Darker areas indicate lower variance and brighter areas represent higher variance. From Fig. 7, one can see that the variance is tied to the 2D image properties, *e.g.*, local image contrast and texture information. This

observation again verifies the promise of our proposed method. One thing to note is that there is a bit of shadow movement being captured by the algorithm. Exploiting strong geometric cues (*e.g.*, the desk is flat) would help remove this artifact, but this is not the focus of the proposed algorithm.

In the sequences we captured, the largest displacement between the stereo pairs is around 100 pixels and there are occluded areas present in all the captured sequences. However, we still can produce reasonable estimates in those areas. This is due to the region merging step where the information fill-in takes place.

Comparisons with the approach of [51] are shown in Fig. 8. The inputs to Eq. 13 are obtained using Algorithm 1, but the covariances are not used. Without ground truth data, it is difficult to quantitatively evaluate the results. Here we use the error image of the warped image of $I(t)$ and the target image $I(t+1)$. If the projected flow fields are accurate, we should have smaller warped image error. In all three test sequences, the proposed method produced much smaller warped image errors compared to those of [51]. Qualitatively, our proposed method produced much cleaner projected flow and Z-motion in all three test sequences. Similar results have been shown in [60] with three cameras while we only used a binocular stereo rig.

5 Discussion and Conclusions

A multi-scale integrated algorithm for 3D scene flow computation was proposed. A region-based probabilistic algorithm was introduced to compute the distributions of optical flow and disparity. Covariances and variances from the probabilistic multi-scale framework for optical flow and disparity computation are combined to estimate 3D scene flow. Occlusions due to large displacement are handled through a region merging process that allows occluded regions to take on valid estimates from adjacent regions. Experiments with synthetic and real data demonstrate much better performance with just two cameras compared to [51]. This superior performance is obtained via taking care of the uncertainty from 2D computation and using region information to regularize the 2D estimates. At the same time, the proposed method computes covariances of the estimated 3D scene flow. The covariances are propagated from the 2D image data and hence provide a measure of how reliable the estimated scene flow is. We expect that the covariances should provide a good initialization for related 3D tracking algorithms. Our proposed approach is general and can be used in a multi-camera setup (E.g. [51, 59, 60]) and should enable improved estimation of 3D scene flow. One way to extend our approach in a multi-camera setup is to choose a reference camera along the lines of [59, 60].

We are currently investigating how to incorporate the proposed algorithm in tracking applications such as vision-based human-computer interfaces. Other interesting applications include analyzing and annotating events in stereo video through anal-

ysis of 3D scene flow. Future work includes extending our formulation to exploit available prior information, e.g., the shape information of the object, to eliminate the inaccuracy like what is shown in Fig. 7, where the shadow affects the estimates.

References

[1] L. Alvarez, R. Deriche, J. Weickert, and J. Sánchez. Dense disparity map estimation respecting image discontinuities: A PDE and scale-space based approach. *International Journal of Visual Communication and Image Representation*, 13:3–21, 2001.

[2] L. Alvarez, J. Weickert, and J. Sánchez. Reliable estimation of dense optical flow fields with large displacements. *IJCV*, 39(1):41–56, 2000.

[3] S. Avidan and A. Shashua. Non-rigid parallax for 3D linear motion. In *CVPR*, pages 62 – 66, 1998.

[4] N. Ayache and F. Lustman. Fast and reliable passive trinocular stereovision. In *Proc. ICCV*, pages 422–427, 1987.

[5] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *IJCV*, 12(1):43–77, 1995.

[6] M. Black and A. Jepson. Estimating optical flow in segmented images using variable-order parametric models with local deformations. *PAMI*, 18(10):972 – 986, 1996.

[7] A. Brint and M. Brady. Stereo matching of curves. *Image and Vision Computing*, 8(1):50–56, 1990.

[8] T. Brox, A. Bruhn, and J. Wickert. Variational motion segmentation with level sets. In *ECCV*, pages 471–483, 2006.

[9] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619, 2002.

[10] J. P. Costeira and T. Kanade. A multibody factorization method for independently moving objects. *IJCV*, 29(3):159 – 179, 1998.

[11] J. Davis, R. Ramamoorthi, and S. Rusinkiewicz. Spacetime stereo: A unifying framework for depth from triangulation. In *CVPR*, 2003.

[12] F. Devernay and O. Faugeras. Computing differential properties of 3D shapes from stereoscopic image without 3D models. In *CVPR*, pages 208–213, 1994.

[13] D. Fleet and A. Jepson. Computation of component image velocity from local phase information. *IJCV*, 5:77–104, 1990.

[14] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6(1):35–49, 1993.

[15] S. B. Gokturk, J-Y. Bouguet, and R. Grzeszczuk. A data-drive model for monocular face tracking. In *Proc. ICCV*, pages 701–708, 2001.

[16] W. Grimson. Computational experiments with a feature based stereo algorithm. *PAMI*, 7(1):17–34, 1985.

[17] D. J. Heeger. Model for the extraction of image flow. *J. Opt. Soc. Am. A*, 4(8):1455–1471, 1987.

[18] B. Heisele, U. Krebel, and W. Ritter. Tracking non-rigid, moving objects based on color cluster flow. In *CVPR*, pages 253–257, 1997.

[19] F. Heitz, P. Pérez, and P. Bouthemy. Recovering local surface structure through local phase different methods. *CVGIP: Image Understanding*, 59(1):125–134, 1994.

[20] B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185 – 203, 1981.

[21] M. Jenkin and A. Jepson. Recovering local surface structure through local phase different methods. *CVGIP: Image Understanding*, 59:72–93, 1994.

[22] M. Jenkin, A. Jepson, and J. Tsotsos. Techniques for disparity measurement. *CVGIP: Image Understanding*, 53(1):14–30, 1991.

[23] D. Kalivas and A. Sawchuk. A region matching motion estimation algorithm. *CVGIP: Image Understanding*, 54(2):275–288, 1991.

[24] K. Langley, T. Atherton, R. Wilson, and M. Lacombe. Vertical and horizontal disparities from phase. *Image and Vision Computing*, 9(4):296–302, 1991.

[25] W.-H. Liao, S. J. Aggrawal, and J. K. Aggrawal. The reconstruction of dynamic 3D structure of biological objects using stereo microscope images. *Machine Vision and Applications*, 9:166 – 178, 1997.

[26] M. Lin and C. Tomasi. Surfaces with occlusions from layered stereo. *PAMI*, 26(8):710–717, 2004.

[27] J. Little, H. Butlhoff, and T. Poggio. Parallel optical flow using local voting. In *Proc. ICCV*, pages 454–459, 1988.

[28] B. Lucas. *Gneralized image matching by the method of differences*. PhD thesis, Carnegie Mellon University, 1984.

[29] B. D. Lucas and T. Kanade. An interative image registration technique with an application to stereo vision. In *IJCAI*, pages 674 – 679, 1981.

[30] S. Malassiotis and M. G. Strintzis. Model-based joint motion and structure estimation from stereo images. *CVIU*, 65(1):79 – 94, 1997.

[31] D. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis. *PAMI*, 15(6):580 – 591, 1993.

[32] D. Mukherjee, Y. Deng, and S. Mitra. Region based video coder using edge flow segmentation and hierarchical affine region matching. In *SPIE, Visual communications and Image processing*, pages 338–349, 1998.

[33] H. Nagel. Extending the oriented smoothness constraint into the temporal domain and the estimation of derivatives of optic flow. In *ECCV*, pages 139–148, 1990.

[34] N. Papenberg, A. Bruhn, T. Brox, S. Didas, and J. Weickert. Highly accurate optic flow computation with theoretically justified warping. *IJCV*, 67(2):141–158, April 2006.

[35] A. P. Pentland and B. Horowitz. Recovery of nonrigid motion and structure. *PAMI*, 13(7):730 – 742, 1991.

[36] J. Pons, R. Keriven, O. Faugeras, and G. Hermosillo. Variational stereovision and 3d scene flow estimation with statistical similarity measures. In *Proc. ICCV*, 2003.

[37] L. Robert and R. Deriche. Dense depth map reconstruction: A minimization

and regularization approach which preserves discontinuities. In *ECCV*, pages 439–451, 1996.

[38] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame correspondence. *IJCV*, 47:7 – 42, 2002.

[39] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithm. *IJCV*, 47(1/2/3):7–42, 2002.

[40] J. Shah. A nonlinear diffusion model for discontinuous disparity and half-occulusions in stereo. In *CVPR*, pages 34–40, 1993.

[41] Y. Q. Shi, C. Q. Shu, and J. N. Pan. Unified optical flow field approach to motion analysis from a sequence of stereo images. *Pattern Recognition*, 27(12):1577 – 1590, 1994.

[42] E. P. Simoncelli. Bayesian multi-scale differential optical flow. In B. Jahne, H. Haussecker, and P. Geissler, editors, *Handbook of Computer Vision and Applications*, chapter 14, pages 397 – 422. Academic Press, 1999.

[43] E. P. Simoncelli, E. H. Adelson, and D. J. Heeger. Probability distributions of optical flow. In *CVPR*, 1991.

[44] Gilbert Strang. *Introduction to Applied Mathematics*. Wellesley-Cambridge Press, 1986.

[45] H. Tao, H. S. Sawhney, and R. Kumar. Dynamic depth recovery from multiple synchronized video sequences. In *CVPR*, pages 118–124, 2001.

[46] L. Torresani, A. Hertzman, and C. Bregler. Learning non-rigid 3D shape from 2D motion. In *Proc. of NIPS*, pages 1555–1562., 2003.

[47] S. Ullman. *The interpretation of Visual Motion*. MIT Press, 1979.

[48] S. Ullman. Maximizing the rigidity: The incremental recovery of 3-D shape and nonrigid motion. *Perception*, 13:730 – 742, 1984.

[49] `http://www.cs.brown.edu/people/black/Sequences/yosemite.tar.gz`.

[50] `http://www.middlebury.edu/stereo`.

[51] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. In *Proc. ICCV*, volume 2, pages 722 – 729, 1999.

[52] J. Wang and E. H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 3(5):625 – 638, 1994.

[53] A. M. Waxman and J. H. Duncan. Binocular image flows: Steps toward stereo-motion fusion. *PAMI*, 8(6):715 – 729, 1986.

[54] J. Weng. A theory of image matching. In *Proc. ICCV*, pages 200–209, 1990.

[55] J. Wiklund, C. Westelius, and H. Knutsson. Hierachical phase based disparity estimation. In *Proc. ICIP*, pages 128–131, 1992.

[56] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. In *ECCV*, pages 233–246, 2004.

[57] G. S. Young and R. Chellappa. 3-D motion estimation using a sequence of noisy stereo images: Models, estimation, and uniqueness. *PAMI*, 12(8):735 – 759, 1999.

[58] L. Zhang, B. Curless, and S. M. Seitz. Spacetime stereo: Shape recovery for dynamic scenes. In *CVPR*, 2003.

[59] Y. Zhang and C. Kambhamettu. Integrated 3D scene flow and structure re-

covery from multiview image sequences. In *CVPR*, 2000.

[60] Y. Zhang and C. Kambhamettu. On 3D scene flow and structure estimation. In *CVPR*, 2001.

[61] Z. Zhang and O. Faugeras. *3D Dynamic Scene Analysis*. Springer-Verlag, 1992.

[62] Z. Zhang and O. Faugeras. Estimation of displacements from two 3-D frames obtained from stereo. *PAMI*, 14(12):1141 – 1156, 1992.

[63] C. Zitnick, N. Jojic, and S. Kang. Consistent segmentation for optical flow estimation. In *Proc. ICCV*, pages 1308–1315, 2005.

[64] C. Zitnick, S. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *ACM Transactions on Graphics*, 23(3):600–608, 2004.

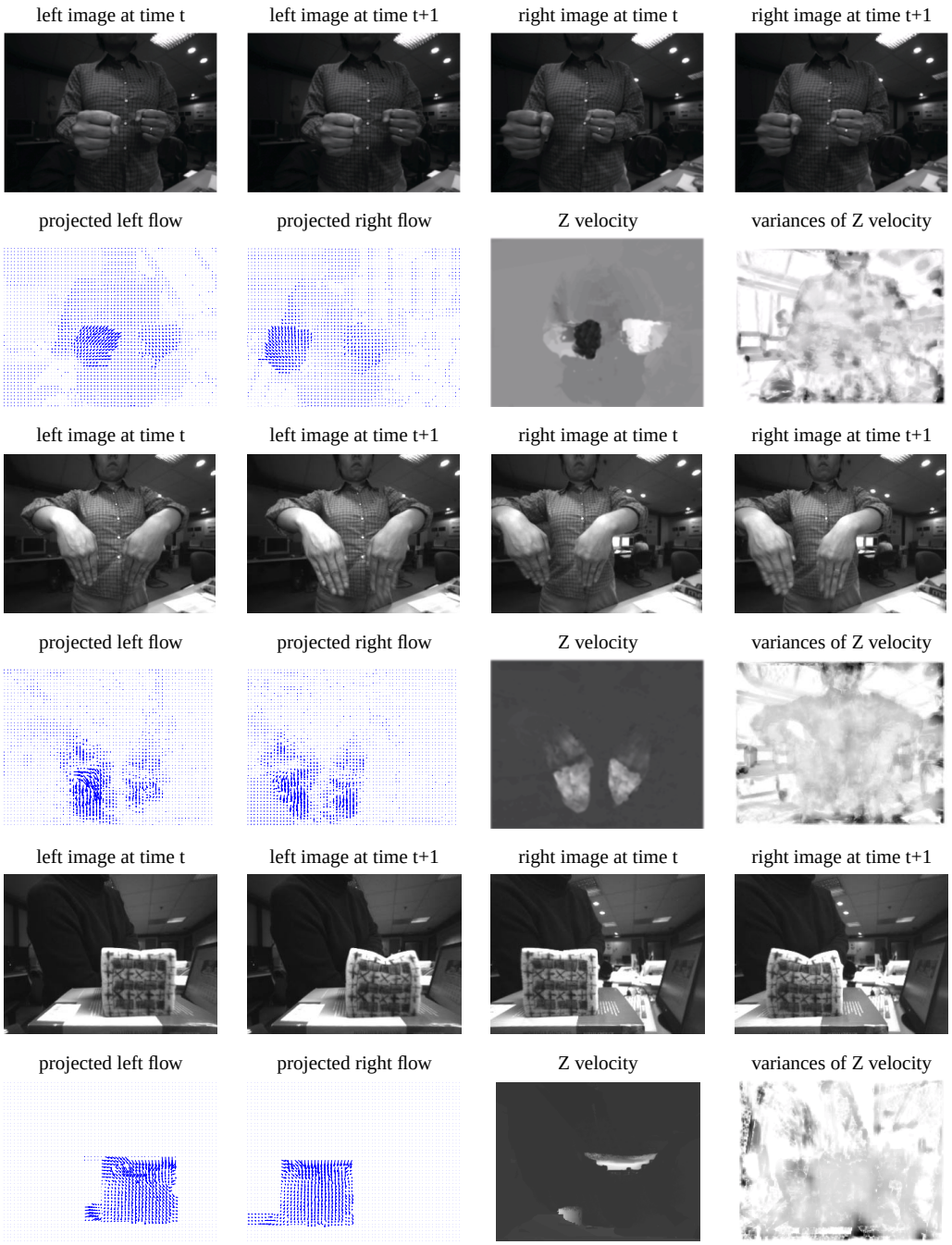

Fig. 7. *Experimental results with three real sequences. The motions presented in the first sequence are one fist moving forward and the other backward. The second sequence shows the motion of both hands moving forward. A deforming sponge (by pushing the top) is shown in the third sequence. In the rows that demonstrate the estimation results, the first two columns are the projections of estimated 3D scene flow in left and right view, the third column is the Z velocity intensity image, the darker area represents the hand moving away from the camera, the brighter area indicates the hand moving towards the camera, the last column shows the variances of the Z velocity where the darker areas represent the places where the estimates for Z velocity are more reliable and the brighter areas represent the places where the Z velocity estimates are less reliable.*

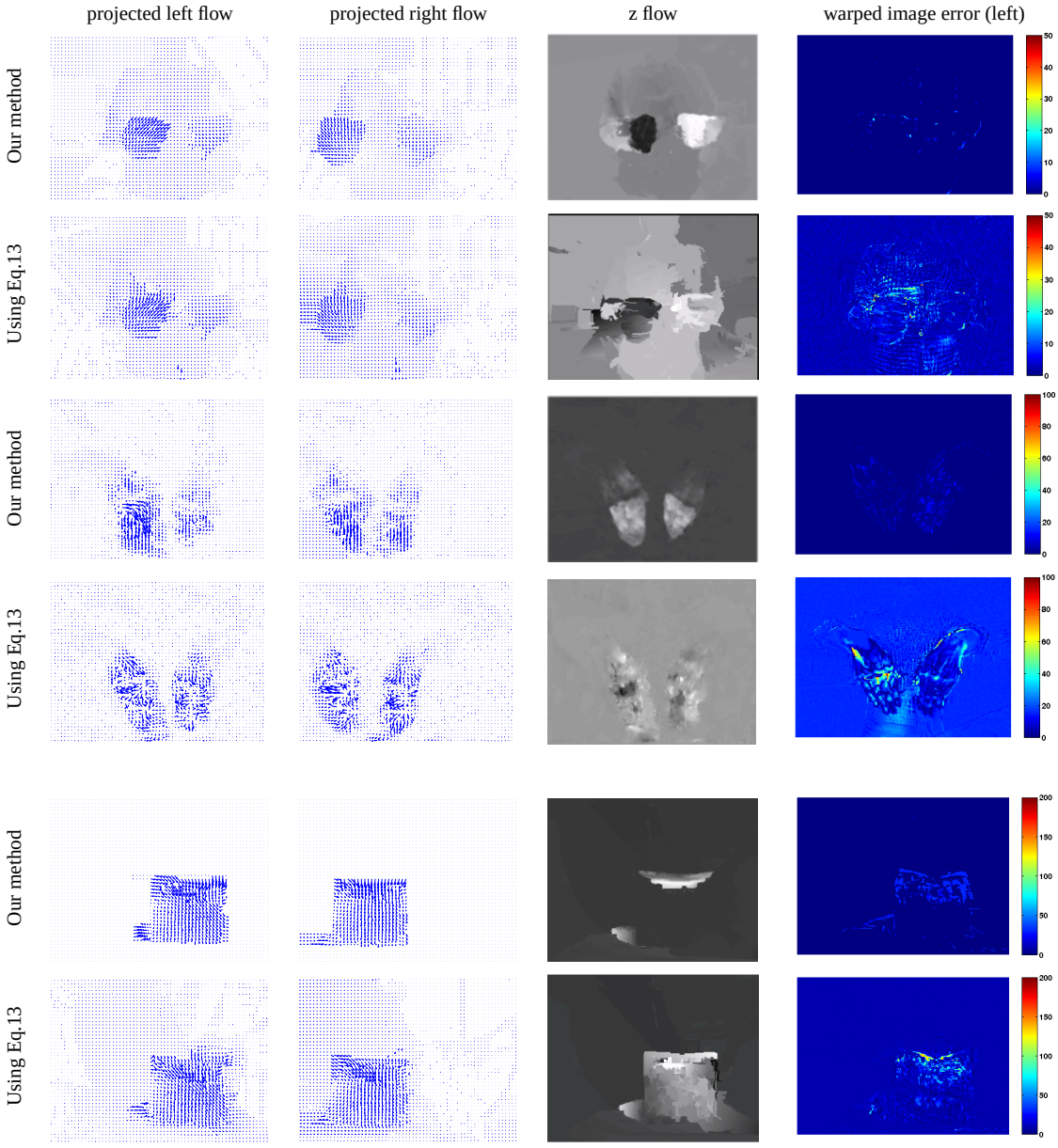

Fig. 8. *Comparison with the estimation results using Eq. 13 with the three video sequences. The first, third and fifth rows show the results of using Algorithm 2 while the second, fourth and sixth rows show the results using Eq. 13, which is equivalent to [51].*